Architecture de surveillance-commande pour les systèmes à événements discrets complexes

Éric Zamaï

N° ordre : 2760 - Année 1997

Thèse

Présentée devant **L'Université Paul Sabatier (Sciences)**
en vue de l'obtention du TITRE de
DOCTEUR de l'Université Paul Sabatier de TOULOUSE.
spécialité : **Informatique Industrielle**

par
Eric ZAMAI
Maître ès Sciences Electronique, Electrotechnique et Automatique

ARCHITECTURE DE SURVEILLANCE-COMMANDE POUR LES SYSTEMES A EVENEMENTS DISCRETS COMPLEXES

Soutenue le 30 Septembre 1997 devant la commission d'examen :

M.	COURVOISIER	*Professeur des Universités*	*- Président*
E.	CRAYE	*Professeur des Universités*	*- Rapporteur*
E.	NIEL	*Habilité à diriger les recherches*	*- Rapporteur*
D.	NOYES	*Professeur des Universités*	*- Examinateur*
R.	VALETTE	*Directeur de recherche au CNRS*	*- Examinateur*
M.	COMBACAU	*Maître de conférences*	*- Directeur de thèse*

Rapport LAAS N° 97361
Thèse préparée au Laboratoire d'Analyse et d'Architecture
des Systèmes du CNRS.

7, avenue du Colonel Roche
31 077 Toulouse Cedex 4.

Avant-Propos

Le travail présenté dans ce mémoire a été effectué au Laboratoire d'Analyse et d'Architecture des Systèmes (L.A.A.S) du C.N.R.S. A ce titre, je tiens à remercier Messieurs Alain COSTES et Jean Claude LAPRIE, responsables successifs du L.A.A.S, de m'avoir accueilli dans leur laboratoire et de m'avoir ainsi permis d'effectuer cette recherche.

Je remercie également Monsieur Jean Claude HENNET, Directeur de Recherche au C.N.R.S pour son accueil dans le groupe D.C.S.P. (Décision et Conduite dans les Systèmes de Production).

J'exprime toute ma gratitude à Monsieur Marc COURVOISIER, Professeur à l'Université Paul Sabatier de Toulouse et Directeur de l'Institut National des Sciences Appliquées de Toulouse, d'avoir accepté de présider le jury de soutenance.

Je tiens tout particulièrement à apporter le témoignage écrit de ma profonde reconnaissance à mon directeur de thèse, Monsieur Michel COMBACAU, Maître de Conférences à l'Université Paul Sabatier, pour son soutien, sa confiance, sa disponibilité, ses critiques et sa sympathie qui m'ont été nécessaires au cours de ce long travail. Les travaux présentés dans ce mémoire ont largement bénéficié de ses conseils judicieux. Pour tout cela, je te dis Merci.

Je suis très reconnaissant à Messieurs Etienne CRAYE, Professeur des Universités à l'Ecole Centrale de Lille et Eric NIEL, Habilité à diriger les recherches, Maître de Conférences à L'Institut National des Sciences Appliquées de Lyon, pour avoir accepté d'étudier mes travaux et d'en être les rapporteurs ainsi que pour l'intérêt et l'attention qu'ils ont accordés à cette étude.

Je tiens également à exprimer ma gratitude à Messieurs Daniel NOYES, Professeur des Universités à l'Ecole Nationale d'Ingénieurs de Tarbes et Robert VALETTE, Directeur de Recherche au C.N.R.S, pour leurs remarques et leurs conseils, contribuant ainsi à parfaire la version finale de ce mémoire et enfin pour avoir accepté de prendre part au jury.

Je souhaiterais aussi remercier Messieurs Daniel NOYES et François SOLER, Assistant Ingénieur, pour l'accueil qu'ils m'ont réservé lors de ma visite à l'Ecole Nationale d'Ingénieurs de Tarbes.

Je ne saurais oublier tous les membres du groupe D.C.S.P. pour la bonne ambiance dans laquelle j'ai passé ces trois années de recherche. Je réserve une pensée toute particulière pour mes collègues de travail : Audine CHAILLET-SUBIAS pour les nombreuses et fructueuses discussions que nous avons eues, et pour le temps consacré à la lecture

du manuscrit, Adel BENZINA et Nabil BEN-KHALIFA, pour leur disponibilité, leur humour et leur amitié et enfin à tous mes compagnons de thèse : Laurent, Jean-Marc, Jean-Luc, Isabelle, Jérôme, ... pour la sympathie et l'amitié que j'ai trouvées en eux.

Enfin, j'exprime mes remerciements à tout le personnel technique et administratif du L.A.A.S, et en particulier envers Madame Eliane DUFOUR pour son efficacité et sa gentillesse et Messieurs Christian BERTY, Daniel DAURAT et Jean CATALA pour leur rigueur et leur disponibilité.

Je dédie ce paragraphe à tous ceux qui m'ont aidé et supporté dans cette épreuve. Je pense en particulier à Marie qui a su me comprendre, me réconforter et me donner l'énergie nécessaire pour achever ce travail, à mes parents dont l'amour et le dévouement sont exemplaires, à mes proches parents et amis avec qui j'ai passé de si bons moments. Que ce mémoire soit un témoignage de mon affection.

⋆ ⋆ ⋆

Table des Matières

Partie I Cadre de l'étude 3

Chapitre 1 La commande dans le domaine manufacturier 5

1.1 Introduction 5

1.2 Les systèmes flexibles de production manufacturière 5

1.2.1 Flexibilité physique de l'atelier 6

1.2.2 Flexibilité décisionnelle du système de gestion 6

1.3 Synthèse des Systèmes Automatiques 8

1.3.1 Présentation des Systèmes Automatiques 8

1.3.2 Hiérarchisation de la commande 10

1.4 Les Systèmes à Événements Discrets 11

1.4.1 La commande des S.E.D. 11

1.4.2 La modélisation de la commande 12

1.5 Conclusion 12

Chapitre 2 Problématique de la surveillance temps réel 15

2.1 Introduction 15

2.2 Caractérisation de la surveillance temps réel 15

2.3 Les fonctions de la surveillance 16

2.4 Intégration ou séparation de la surveillance 16

2.5 Répartition de la surveillance 17

2.6 Commande-Surveillance et modèle du procédé 18

2.6.1 Approche développée par le CRAN/LACN (Nancy) 19

2.6.2 Approche développée par le LAIL (Lille) 19

2.6.3 Approche développée par le LAMIH (Valenciennes) 21

2.6.4 Approche développée à l'Université de Carnegie Mellon 22

2.6.5 Approche développée par le LAI (lyon) 23
2.6.6 Approche développée par le LAAS (Toulouse) 24
2.7 Conclusion 25

Chapitre 3 Étude critique **27**
3.1 Introduction 27
3.2 Concept d'activités 27
3.3 Point de vue commande 28
3.3.1 Élaboration hors ligne des modèles 29
3.3.1.1 Modèle de Référence 29
3.3.1.2 Modèle de commande 29
3.3.2 Exploitation en ligne des modèles 30
3.3.2.1 Communication commande-référence 30
3.3.2.2 Communication inter-niveaux 30
3.4 La surveillance 30
3.4.1 Élaboration hors ligne 30
3.4.1.1 La boucle de surveillance 30
3.4.1.2 Système d'information 32
3.4.2 Fonctionnement de la surveillance 32
3.5 Avantages de l'approche 33
3.5.1 Commande 33
3.5.2 Surveillance 33
3.6 Limitations de l'approche 34
3.7 Conclusion 34

Partie II Une approche de Surveillance-Commande **37**

Chapitre 1 Généralités **39**
1.1 Introduction 39
1.2 Réceptivité 39
1.3 Traitements de surveillance-commande 41
1.3.1 Les états d'un système de surveillance-commande 41
1.3.2 Concept d'activité étendue 43

1.3.3 Un modèle de référence pour la surveillance-commande 44
1.3.4 Prise en compte de la politique de surveillance de l'entreprise . . . 45
1.4 Un outil de spécification : SADT 46
1.5 Les Réseaux de Petri à Objets comme outil de modélisation 47
1.6 Conclusion 48

Chapitre 2 Spécification d'un nœud de Surveillance-Commande 49
2.1 Introduction 49
2.2 Spécification 49
2.2.1 Contexte général 49
2.2.2 Flot d'informations et fonctionnalités pour un nœud 51
2.2.2.1 Données en entrée 51
2.2.2.2 Données de contrôle 51
2.2.2.3 Données en sortie 51
2.2.2.4 Supports 51
2.2.2.5 Les fonctionnalités principales 52
2.2.3 Diagramme A0 : nœud de surveillance-commande 53
2.2.3.1 Fonctionnalités à réaliser 53
2.2.3.2 Interfaçage des fonctionnalités Superviser et Surveiller-Commander 55
2.2.4 Diagramme A01 : Superviser 55
2.2.4.1 Fonctionnalités à réaliser 55
2.2.4.2 Interfaçage des fonctionnalités de Superviser 55
2.2.5 Diagramme A02 : Surveiller-Commander 57
2.2.5.1 Fonctionnalités à réaliser 57
2.2.5.2 Interfaçage des fonctionnalités de Surveiller-Commander 57
2.3 Conclusion 59

Chapitre 3 Définition d'un nœud de surveillance-commande 61
3.1 Introduction 61
3.2 Élaboration du modèle de référence pour la surveillance-commande . . . 62
3.2.1 Activités de surveillance-commande 63
3.2.1.1 Liste exhaustive des activités de surveillance-commande 63
3.2.1.2 Critères d'admissibilité 64

3.2.1.3 Activités utilisables 66
3.2.2 Les processus 72
3.2.3 Les changements d'états 72
3.2.3.1 Couplage 73
3.2.3.2 Découplage 74
3.2.3.3 Les processus élémentaires admissibles 75
3.2.4 Le modèle de référence pour la surveillance-commande 75
3.2.4.1 Le modèle obtenu 75
3.3 Elaboration du modèle de la stratégie de surveillance-commande 77
3.3.1 Intégration de la politique de l'entreprise 78
3.3.2 Intégration des besoins de transformation du produit 80
3.3.3 Respect des contraintes imposées dans le modèle de référence pour la surveillance-commande 80
3.4 Application de la surveillance-commande 81
3.4.1 Acquérir les informations 83
3.4.2 Orientation de l'information 83
3.4.3 Représenter l'évolution du traitement de surveillance-commande . 87
3.5 Mécanismes de communication 88
3.6 Conclusion 90

Partie III Exemple d'application 91

Chapitre 1 Présentation de l'atelier et du système de Surveillance-Commande 93

1.1 Introduction 93
1.2 Caractéristiques techniques 93
1.2.1 Machines Outils 93
1.2.2 Commandes Numériques 94
1.3 Hiérarchisation de la surveillance-commande 95
1.4 Modèles de référence pour la surveillance-commande 97
1.5 Modèles de la stratégie 98
1.5.1 Module Coordination 99
1.5.2 Module Tour 101
1.5.3 Autres modules 104
1.6 Conclusion 104

Chapitre 2 Etude du fonctionnement de la surveillance-commande **107**

2.1 Introduction 107
2.2 Présentation du scénario 1 107
2.3 Fonctionnement 109
2.4 Présentation du scénario 2 114
2.5 Fonctionnement 115
2.5.1 Module tour 115
2.5.2 Module coordination 117
2.5.3 Module poste opérateur 118
2.5.4 Module coordination 119
2.6 Conclusion 120

Conclusion Générale **121**

Bibliographie **125**

Annexes **133**

Introduction

Depuis ces vingt dernières années, dans le domaine de la productique, le souci principal des utilisateurs s'est porté sur une automatisation à outrance des procédés industriels complexes tels que les ateliers flexibles de production manufacturière. Dans un contexte économique devenu austère, cette automatisation tend à remplacer l'homme dans toutes ses tâches. Les concepteurs de ces systèmes automatisés se sont tout d'abord penchés sur l'élaboration de systèmes de commande des procédés. La conséquence la plus visible en est une amélioration importante des performances et des gains de l'entreprise. Cependant, même si ces systèmes ont été longtemps considérés comme **la** solution idéale aux problèmes posés par les industriels, il n'en demeure pas moins que les industriels sont incapables de réagir lorsqu'une situation inconnue se présente à eux. Pour cette raison, les besoins d'automatiser la surveillance, jusqu'alors domaine réservé à l'homme, se sont fait ressentir. La plupart des approches qui traitent de la surveillance s'accordent à la définir en fonction de cinq éléments majeurs : la *détection* des symptômes de l'anomalie, le *diagnostic* du symptôme pour en trouver les causes, l'élaboration d'une solution permettant le retour en fonctionnement normal (*décision*), la *reprise* pour appliquer cette solution et enfin l'*urgence* dédiée à l'application des traitements "brutaux", rapides et pré-définis sur le procédé lorsque la situation remet en cause la sécurité de l'opérateur ou la structure du procédé. Le point commun aux diverses fonctions de la surveillance est leur besoin important d'informations concernant l'état du procédé. Or, il n'est pas toujours possible d'acquérir a posteriori toutes ces informations par le biais des capteurs placés sur le procédé. L'intégration d'un **modèle du procédé** dont l'état est en permanence remis à jour par les évolutions provoquées par la commande et les signaux émis par les capteurs s'est révélée nécessaire.

De nombreuses solutions, intégrant tous ces éléments, existent mais elles ne répondent pas toujours aux besoins de la surveillance et aux besoins réels de l'entreprise en proposant une surveillance rigide et figée.

Le travail que nous présentons dans ce document propose d'apporter sa contribution au domaine de la surveillance et de la commande en temps réel des systèmes à événements discrets complexes. Partant du constat qu'une des limitations principales de la surveillance s'explique par l'exploitation des informations qui transitent par le système de commande, nous ne présentons plus la surveillance comme un palliatif à la commande mais plutôt la commande comme un cas simple et fréquent de la gestion en temps réel du système de production. Le recours à un superviseur de surveillance-commande autorise

cette démarche.

Pour apporter une solution au problème lié à la rigidité des traitements de surveillance (détection, diagnostic, décision et reprise) généralement imposée, nous proposons à l'utilisateur un modèle de référence de la surveillance-commande représentant l'ensemble des activités de surveillance et des contraintes qui les lient. Ce modèle lui permet d'élaborer son propre modèle de la "stratégie" de surveillance adapté au procédé considéré. Ce modèle de surveillance tient compte de la politique de surveillance de l'entreprise et des contraintes liées aux produits à fabriquer. Le respect des contraintes imposées dans le modèle de référence garantit à son utilisateur la cohérence des traitements qu'il a élaborés pour son entreprise.

La définition d'un **module générique de surveillance-commande** permet de mettre en œuvre les principes que nous venons d'énoncer.

Ce mémoire est organisé en trois parties dont les thèmes sont donnés ci-après.

La première partie présente de manière générale la problématique de la commande et de la surveillance dans le domaine manufacturier. Les besoins requis par le système de commande sont tout d'abord étudiés au travers de notions comme la flexibilité, la structuration ou la modélisation de la commande. Ensuite, une étude de la surveillance temps réel est réalisée en s'appuyant sur les travaux réalisés dans différentes équipes de recherche appartenant à des laboratoires comme le LAIL (Lille), le CRAN (Nancy), le LAI (Lyon), l'université de Carnegie Mellon, le LAMIH (Valenciennes) ou encore le LAAS (Toulouse). Toutes ces approches intègrent les différents éléments requis par la surveillance, à savoir un modèle du procédé et les fonctions de surveillance. Cette partie s'achève sur une étude critique de l'approche développée au LAAS.

La partie II est entièrement consacrée à la présentation de notre approche de la surveillance-commande. Une étude des pré-requis de la surveillance y est d'abord décrite. Partant de cette étude, une spécification du système de surveillance-commande est réalisée pour dégager les fonctionnalités à intégrer au système de surveillance-commande. Cette spécification est supportée par le formalisme SADT. L'élaboration du système de surveillance-commande fait suite à cette spécification. Elle conduit à la réalisation d'un module générique de surveillance-commande.

Dans la partie III nous développons un exemple d'application sur un atelier réel. La cellule flexible considérée est celle de l'Ecole Nationale d'Ingénieur de Tarbes. Les avantages de notre approche tels que la prise en compte de la politique de surveillance de l'utilisateur et des contraintes issues du produit à transformer pour construire le modèle de surveillance ou la gestion des informations qui transitent par un module de surveillance-commande y sont exposés. L'illustration de notre approche sur cet exemple met en évidence son adéquation à des procédés non entièrement automatisés dans lesquels l'opérateur humain joue un rôle important tant au niveau de la commande que de la surveillance.

Partie I

Cadre de l'étude

Chapitre 1

La commande dans le domaine manufacturier

1.1 Introduction

Ce chapitre est consacré à la description générale d'une structure de commande hiérarchisée des systèmes complexes de production manufacturière. Nous nous intéressons en particulier à la commande des ateliers flexibles de petites et moyennes séries. Le système physique considéré est constitué de machines polyvalentes sur lesquelles passe une grande diversité de produits.

Dans le premier paragraphe, nous présentons ces systèmes en considérant l'aspect flexibilité. L'étude qui en découle s'articule sur deux entités essentielles : l'**atelier** et le **système de gestion** associé. Nous abordons ensuite la description des systèmes de production automatisés. L'analyse qui en est faite est orientée temps réel et introduit un aspect organisationnel de la commande des systèmes de production. L'adéquation d'une structure hiérarchique et modulaire à la commande du procédé y est, en effet, montrée. Dans le dernier paragraphe, nous décrivons la classe de systèmes complexes sur lesquels nous avons développé notre approche : les systèmes à événements discrets complexes (S.E.D. complexes). Au travers de leur présentation, nous abordons les principes généraux de la commande et de sa modélisation.

1.2 Les systèmes flexibles de production manufacturière

De nos jours, dans le domaine manufacturier, il est évident que toute entreprise doit disposer d'un outil de production performant. Ces performances sont caractérisées par l'augmentation de la rentabilité et de la souplesse de production, c'est à dire par la capacité d'adaptation qui rend un système de production flexible [Roche, 1988].

L'intégration de tels critères de flexibilité de production doit être réalisée à deux niveaux différents et indissociables du système de production [Abdallah, 1996] [Marty, 1994] : d'une part l'atelier qui permet de concrétiser ou de réaliser matériellement la demande formulée initialement par le client, d'autre part le système de gestion qui prend en compte cette demande initiale (carnet de commande, cahier des charges, etc.) pour amener l'atelier à la réaliser.

1.2.1 Flexibilité physique de l'atelier

Pour qu'un atelier soit flexible, il faut qu'il soit capable de traiter la diversité des produits qui lui sont donnés à transformer. En d'autres termes, les éléments qui le composent doivent être polyvalents. Parmi ceux-ci, nous trouverons tout d'abord les centres d'usinage constitués d'une seule machine à commande numérique alimentée en pièces et en outils : tours, fraiseuses, pointeuses. Ensuite, un regroupement n'excédant guère plus de trois ou quatre machines et des zones de stockage tampon autour d'un système de transport permet de caractériser une cellule flexible souvent appelée cellule d'usinage. Le système de transport alors utilisé est ici central aux cellules flexibles élémentaires. Il est souvent constitué de "chariots filoguidés" suivant des parcours pré-définis en forme d'anneaux, de convoyeurs, ou de "chariots radioguidés". Enfin, en regroupant ces cellules flexibles, nous obtenons, au niveau le plus élevé, l'atelier flexible [Tawegoum, 1995]. Chacune des cellules flexibles le constituant est desservie par un système de transport (cf. figure 1.1) plus conséquent comme les "chariots navettes" ou encore les "convoyeurs en anneau".

Une organisation rationnelle de ces divers éléments induit une grande capacité d'adaptation du système physique aux variations de demande auxquelles il est soumis.

Néanmoins, comme nous l'avons déjà souligné plus haut, la seule flexibilité de l'atelier ne permet pas de garantir la flexibilité de l'ensemble du système de production. Le système de gestion associé doit être également en mesure de s'adapter non seulement aux consignes du client mais également à la souplesse offerte par l'atelier. Nous parlerons alors de flexibilité décisionnelle.

1.2.2 Flexibilité décisionnelle du système de gestion

Au sein de l'équipe Décision et Conduite dans les Système de Production du Laboratoire d'Analyse et d'Architecture des Systèmes, une structure décisionnelle à cinq niveaux est considérée comme étant l'approche optimale pour intégrer ces contraintes. Nous considérons ainsi, comme le montre la figure 1.2, les niveaux **Planification**, **Ordonnancement Prévisionnel**, **Ordonnancement Temps Réel**, **Coordination** et **Commande locale** [Ahmed *et al.*, 1996]. Les trois derniers niveaux présentent un caractère temps réel. Le rôle de ces différents niveaux peut être défini de la façon suivante.

La **Planification** [Hetreux, 1996] consiste à définir un plan de fabrication en fonction du carnet de commande, du cahier des charges fourni par le client et des moyens

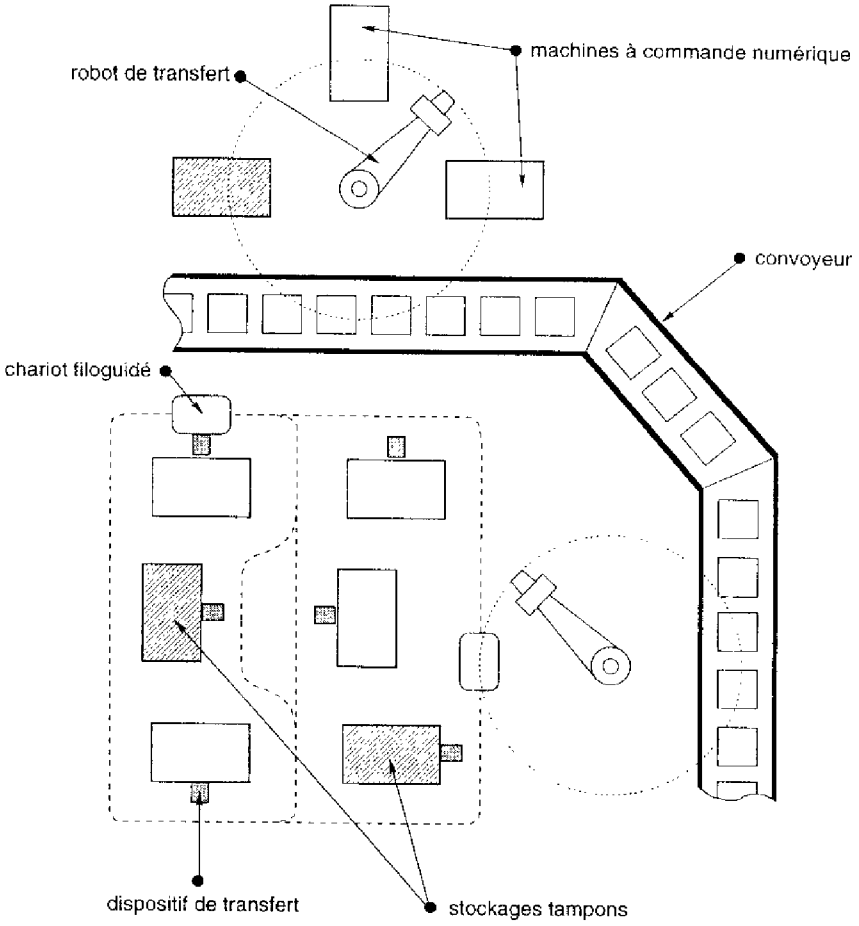


FIG. 1.1 – *exemple d'un atelier flexible*

de production. Ce plan fait apparaître des valeurs comme la quantité, la qualité, les délais de production mais également une pré-allocation des ressources.

L'**Ordonnancement Prévisionnel** [Lopez, 1991] permet ensuite de déterminer un ensemble d'ordres partiels de passage des produits à transformer sur les diverses ressources de l'atelier. Bien sûr, cette allocation tient compte de la pré-allocation des ressources déjà effectuée au niveau planification, des délais, mais également des contraintes de capacité et de disponibilité des ressources, des contraintes de séquencement d'opérations et enfin de critères à optimiser tels que le temps de production, les coûts de matières premières, etc.

L'**Ordonnancement Temps-Réel** [Billaut, 1993] souvent appelé *pilotage* assure la cohérence entre les décisions prévisionnelles (ordonnancement prévisionnel et planification) et les contraintes temps réel issues du comportement réel de l'atelier. Ce niveau est souvent qualifié de "charnière" [Combacau, 1991] puisqu'il permet de gérer au mieux les degrés de liberté non encore explicités par les niveaux supérieurs en fonction de l'état réel des ressources de l'atelier.

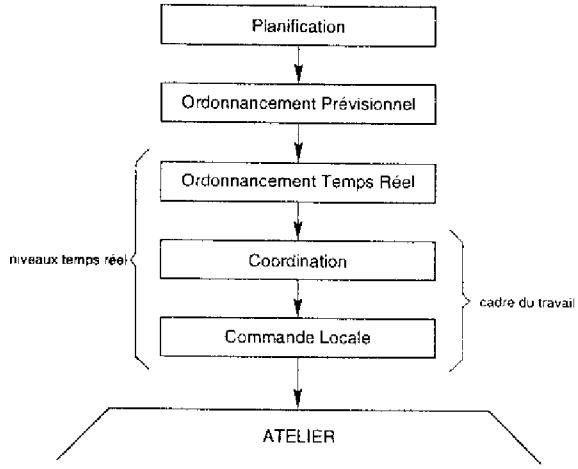


FIG. 1.2 – *structure décisionnelle à cinq niveaux pour un atelier flexible*

La **Coordination** [Bako, 1990] permet de gérer de manière cohérente les interactions entre les différentes ressources de l'atelier en fonction des contraintes telles que les ressources partagées, séquencements obligatoires, synchronisations diverses ou parallélisme, etc.

La **Commande Locale** est le niveau le plus bas de notre structure décisionnelle. Elle correspond à la jonction entre les capteurs/actionneurs du procédé et le système de commande chargé de mettre en œuvre les décisions prises au niveau coordination. De par sa situation, la commande locale est un élément essentiel pour la surveillance puisque c'est à ce niveau seulement que les signaux émis par le procédé sont pris en compte par le biais des capteurs.

La flexibilité décisionnelle est donc liée à la flexibilité de chacun de ces cinq niveaux. Le pouvoir d'adaptation de chacun d'entre eux permet de prendre des décisions qui sont des "compromis entre la conservation d'une marge de manœuvre par le niveau considéré et la délégation de cette marge de manœuvre aux niveaux inférieurs" [Combacau, 1991].

Dans le cadre de notre travail, nous nous sommes particulièrement intéressés aux deux derniers niveaux de cette structure décisionnelle, à savoir la coordination et la commande locale (niveaux temps réel). Pour cette raison, nous allons les détailler au travers d'une présentation générale des systèmes automatiques de production.

1.3 Synthèse des Systèmes Automatiques

1.3.1 Présentation des Systèmes Automatiques

Lorsqu'on parle de Systèmes Automatiques (SA) de production, une façon classique de les présenter consiste à décrire une boîte capable de transformer des produits en

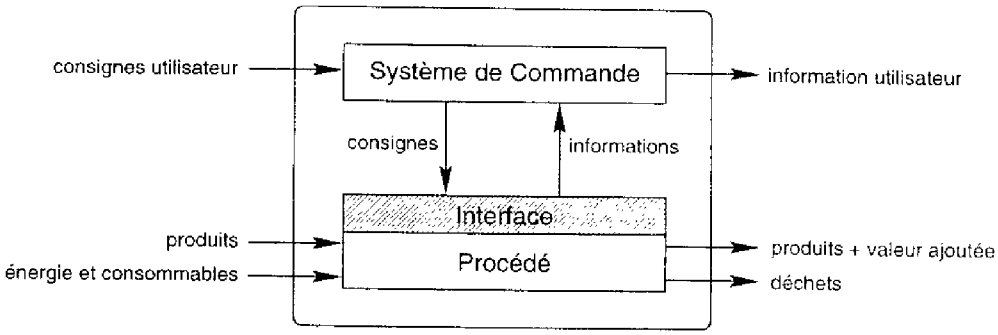


FIG. 1.3 – *synthèse d'un SA*

d'autres produits augmentés d'une valeur ajoutée. Cette valeur ajoutée est calculée sur la base de critères tels que le poids, la concentration, la dimension, la forme, etc. Bien sûr, cette transformation est obtenue selon les consignes imposées par le client. L'analyse d'une telle transformation nous conduit naturellement à détailler le contenu de cette boîte. La figure 1.3 nous en donne d'ailleurs une représentation. Un système automatique de production est constitué de trois blocs principaux. Premièrement, le procédé, également appelé partie opérative [Cruette, 1991], se compose d'un ensemble d'entités de transformation physiques, spatiales, etc., des produits (usinage, assemblage, transport, etc.). Deuxièmement, le système de commande, encore appelé partie commande, est constitué d'entités de gestion structurées en plusieurs niveaux décisionnels (cf. paragraphe précédent). Il est chargé des décisions concernant les actions à entreprendre sur le procédé pour transformer les produits selon les consignes imposées par l'utilisateur. A ces fins, le système de commande émet vers le procédé des consignes de commande qui doivent être exécutées. En retour, le procédé agit sur les produits et atteste des actions effectuées par l'envoi d'informations (cf. figure 1.3).

Bien sûr, cet échange d'informations serait impossible sans tenir compte d'un troisième et dernier bloc, l'interface. Celle-ci met en forme les informations qui transitent du système de commande vers le procédé (consignes de commande) ou du procédé vers le système de commande (informations) au moyen de deux classes de composants :

- les *actionneurs* chargés de convertir les consignes de commande en actions effectives sur le procédé,
- les *capteurs* considérés comme des systèmes de mesure pour la commande. Leur type est très différent selon les quantités qu'ils mesurent (capteurs de proximité, capteurs analogiques de mesure de courant électrique, capteurs d'effort, etc.). Un équilibre entre le coût lié à l'implantation de ces capteurs et l'efficacité du système de production considéré doit être respecté. Pour cette raison, le choix des grandeurs à prendre en compte est un problème crucial.

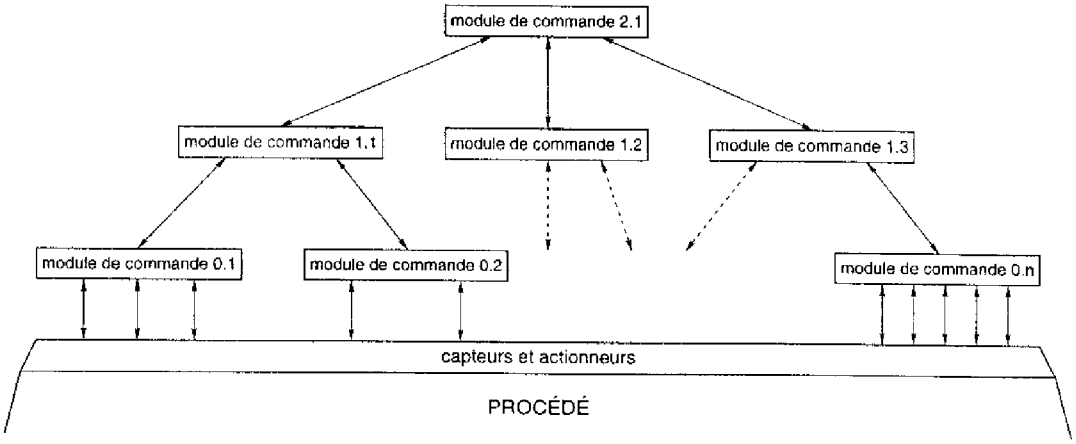


FIG. 1.4 – *structure hiérarchique et modulaire de la commande temps réel*

1.3.2 Hiérarchisation de la commande

La structure décisionnelle à cinq niveaux que nous avons décrite précédemment ne peut être en réalité complètement figée. En effet, la taille et la complexité des applications étant souvent différentes et conséquentes, une "sous-structuration" des niveaux coordination et commande locale est souvent nécessaire. Comme le souligne d'ailleurs [Jones, 1989] ou [O'Grady *et al.*, 1994], une démarche classique pour faire face à la complexité de certaines applications consiste à décomposer le système de commande global en plusieurs autres niveaux de complexité moindre. Il s'agit alors de *hiérarchiser* le système de commande. Ensuite, comme le préconise [Verlinde, 1989], pour améliorer cette décomposition, chacun des niveaux obtenus peut également être décomposé horizontalement en plusieurs modules indépendants. La structure obtenue est alors qualifiée de **structure hiérarchique et modulaire de la commande temps réel** [Sahraoui, 1987] [Combacau, 1991] [Parayre, 1992]. Bien entendu, le niveau le plus bas d'une telle structure hiérarchique correspond au niveau de commande locale.

D'un point de vue du fonctionnement, dans chaque niveau de cette hiérarchie (du plus haut niveau à celui de la commande locale) un module se charge d'affiner les ordres qu'il reçoit du niveau supérieur. Ces affinements successifs consistent à sélectionner et à organiser les services offerts par le niveau immédiatement inférieur en tenant compte à la fois des contraintes locales du niveau d'abstraction considéré (séquencements ou synchronisations obligatoires, ressources physiques ou abstraites partagées, état courant du sous-système commandé, etc.) mais également des contraintes véhiculées par la requête (délais à respecter, caractérisation technique du produit, localisation temporelle, etc.).

En fonctionnement normal, lorsqu'un service est rendu par un module de la structure hiérarchique, il émet vers le niveau supérieur un compte rendu d'exécution attestant de la bonne exécution du service rendu.

Une structure hiérarchisée et modulaire offre donc des perspectives avantageuses en terme de conception pour faire face à la complexité de la commande des systèmes de production manufacturière. Notons également que par rapport à une architecture de

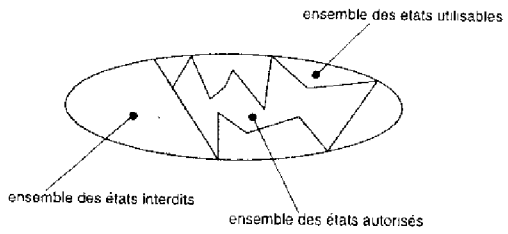


FIG. 1.5 – *les états d'un système de commande*

contrôle hétérarchique, une approche hiérarchisée et modulaire permet de résoudre les problèmes de conflits entre plusieurs centres décisionnels. En effet, puisqu'un module de niveau inférieur ne peut être en relation qu'avec un seul module de niveau supérieur, tous les modules de niveau inférieur sont coordonnés. Les décisions contradictoires sont alors exclues.

1.4 Les Systèmes à Événements Discrets

Dans le cadre de notre travail, nous nous sommes limités à l'étude des Systèmes à Événements Discrets (S.E.D.). Nous considérons que la connaissance exacte des variations des grandeurs continues miroirs de l'état du procédé n'est pas utile. Par exemple, dans le cas d'un robot manipulateur, il n'est pas utile de suivre de manière continue l'évolution de la base du robot. Seules certaines de ces valeurs ou positions atteintes sont suffisantes [Andreu, 1996]. Ces valeurs sont les variables d'état du système. Elles varient brutalement à des instants déterminés. De plus, à un instant t donné, la connaissance des valeurs de ces variables permet de calculer leur valeur future à l'instant suivant $t+1$. Ces valeurs caractérisent ainsi l'état présent [Valette *et al.*, 1988] du système commandé.

1.4.1 La commande des S.E.D.

Commander un S.E.D. consiste à amener le procédé d'un état particulier à un autre en passant par un certain nombre d'états intermédiaires observables (système à événements discrets). Atteindre chacun de ces états revient à satisfaire "pas à pas" la consigne envoyée vers le procédé. Notons que le mode de communication entre le système de commande et le procédé est qualifié d'*Appel/Réponse* [Zamaï, 1996] [Chaillet *et al.*, 1997]. La partie commande envoie en effet une requête (*Appel*) vers le procédé. La réalisation de cette requête correspond à une évolution du procédé vers un état prévu par la commande. Lorsque le procédé a atteint cet état, un compte rendu (*Réponse*), envoyé vers la partie commande, atteste de la bonne exécution du service demandé. Pour que la partie commande puisse prendre en compte de tels comptes rendus, elle doit forcément posséder une image des états de ce procédé. La connaissance de ces états lui permet ainsi de prévoir l'état suivant. Une description détaillée de ces états peut être trouvée dans [Combacau, 1991]. Elle met en évidence trois classes d'états (cf. figure 1.5) : les états

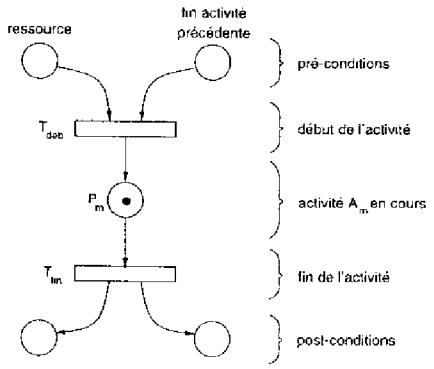


FIG. 1.6 – *modélisation par réseaux de Petri d'une activité de commande*

interdits (transgression de contraintes structurelles, de dépassement d'une limite de capacité, accès simultanés à une ressource partagée, etc.), utilisables (utilisation nominale du procédé) et autorisés (états parcourus volontairement en imposant une commande particulière).

Spécifier une commande revient alors à décrire une séquence particulière permettant de parcourir certains états dits autorisés. Nous rejoignons en cela les concepts de base de la commande des S.E.D., à savoir, les concepts d'événements (cause d'un changement d'état), d'activités (action déclenchée sur un événement et caractérisant un changement d'état) et de processus (suite séquentielle d'activités).

1.4.2 La modélisation de la commande

Plusieurs outils de modélisation des S.E.D. sont envisageables (machines à états, Statecharts, Réseaux de Petri, Grafcet, etc. [Bucci *et al.*, 1995] [Ayache, 1995]). A titre d'exemple, une modélisation de la commande des S.E.D. par Réseaux de Petri nous est donnée dans la figure 1.6. Elle permet de modéliser aisément et de manière claire diverses contraintes comme les exclusions mutuelles, séquencements obligatoires, limite de capacité, pré-conditions diverses (ressources nécessaires ou autres), etc. De plus, le marquage du réseau de Petri nous renseigne sur l'état courant du système de commande. Par exemple, dans la figure 1.6, la place P_m nous montre que l'activité A_m est en cours d'exécution. Nous remarquerons que l'exécution de cette activité revient finalement à associer une *ressource* (machine, cellule, etc.) gérée par la *commande* pour transformer un *produit*. Le jeton contenu dans la place P_m est un n-uplet des éléments ressource et produit : $A_m =< ressource, produit >$. Dans cette représentation, la fonction commande est implicite.

1.5 Conclusion

Nous avons présenté dans ce chapitre la problématique générale de la conduite des ateliers flexibles de production manufacturière. Nous avons montré l'importance de la

prise en compte à plusieurs niveaux de la flexibilité (flexibilité de l'atelier et flexibilité décisionnelle) pour faire face d'une part aux contraintes de production (fluctuations de l'offre et de la demande) mais également à d'éventuelles indisponibilités de ressources matérielles. D'un point de vue commande, l'organisation en une structure hiérarchique et modulaire a été mise en évidence pour faire face à la complexité des systèmes à commander. Dans le contexte des systèmes à événements discrets, la commande consiste, "pas à pas", à satisfaire à une consigne de haut niveau. Le type de communication alors préconisé dans de telles approches se limite à l'**Appel/Réponse**. L'appel correspond à l'envoi d'une requête vers le procédé. La réponse, issue du procédé, atteste de la "bonne ou mauvaise" exécution de cette requête. En fonctionnement normal, ces réponses sont toujours positives, le service demandé est correctement réalisé. En revanche, dès qu'une réponse ne correspond pas à celle attendue, cela signifie que le procédé n'obéit plus aux lois fixées par la commande. Le procédé n'est plus dans un état considéré normal par rapport à la commande. Réagir pour corriger la dérive constatée, c'est à dire en annuler les conséquences et les causes, c'est aborder le problème de la surveillance en temps réel, sujet du chapitre suivant.

Chapitre 2

Problématique de la surveillance temps réel

2.1 Introduction

Jusqu'à présent, nous avons présenté le système de commande dans un contexte exempt de défaillances : tout compte rendu remontant du procédé traduit une fin normale d'exécution d'une activité de commande. Toutefois, lorsqu'une application réelle est considérée, le fonctionnement normal n'est pas toujours assuré. Partant de ce constat, lorsqu'une requête de commande est envoyée vers le procédé, rien ne garantit que le compte rendu reçu soit celui attendu. La déviation caractérise la défaillance. Deux raisons peuvent être à l'origine de ce dysfonctionnement : déficience d'un élément du procédé ou du système de commande. Dans notre approche, nous ferons l'hypothèse que le système de commande est exempt de défaillances. Ainsi, et en replaçant ce problème dans le contexte hiérarchisé que nous avons défini plus haut, une défaillance correspond à la remise en cause par le niveau inférieur de la requête émise par le niveau supérieur.

Lorsque ces défaillances sont prises en compte, nous évoluons dans le domaine de la surveillance temps réel.

2.2 Caractérisation de la surveillance temps réel

Avant de définir la surveillance d'un point de vue général, nous devons préalablement définir ce que nous appelons le temps réel. Selon [GRP, 1988],

> "peut être qualifié de temps réel au sens logique, toute application mettant en œuvre un système informatique dont le fonctionnement est assujetti à l'évolution dynamique de l'état réel d'un environnement qui lui est connecté, et dont il doit contrôler le comportement. C'est donc l'asynchronisme de l'évolution par rapport au contrôle automatisé qui est la source des difficultés du temps réel".

Le système de surveillance doit donc agir sur le procédé dans des délais compatibles avec sa dynamique. Pour cela, le système de surveillance doit posséder un ensemble d'éléments capables de répondre aux besoins des différentes étapes du traitement de défaillance.

2.3 Les fonctions de la surveillance

Le premier rôle à remplir par un système de surveillance consiste à *détecter* toute déviation de comportement du système commandé par rapport à celui prévu par la commande. Cette première fonctionnalité est assurée par la fonction **détection** [Holoway, 1990] [Combacau, 1991] [Cruette, 1991] [Toguyeni, 1992]. Nous verrons plus loin qu'il existe plusieurs façons d'exploiter cette fonctionnalité selon les approches considérées.

Après avoir détecté une défaillance, il s'agit, la plupart du temps, d'identifier la cause ou l'origine de la défaillance. Ce rôle est attribué à la fonction **diagnostic** [Combacau, 1991] [Chaillet, 1995] [Hammami *et al.*, 1995]. Bien sûr, dans un tel contexte (défaillance), toutes les hypothèses de fonctionnement normal valables pour la commande sont remises en cause.

Quel que soit le résultat fourni par la fonction diagnostic (cause de la défaillance trouvée ou non), il s'agit de remettre le système surveillé dans un fonctionnement normal par rapport à la commande ou au procédé. Les fonctions **décision** [Combacau, 1991] et **reprise** [de Bonneval *et al.*, 1992] [Mabrouk *et al.*, 1996] assureront ce rôle important. Elles devront respectivement élaborer une solution de reprise pour ensuite l'appliquer.

En permanence, il faut recueillir l'ensemble des informations émanant du procédé afin de maintenir une image aussi fidèle que possible du système commandé au sein du système de surveillance. Ceci est assuré par une fonction **suivi** [Tawegoum, 1995].

Enfin, dans le cas où la défaillance détectée signifie la transgression d'une contrainte structurelle (choc entre deux chariots par exemple) ou la mise en danger de l'opérateur, il faut immédiatement réagir en appliquant des procédures pré-définies et prioritaires. Le rôle est dévolu à la fonction **urgence** [Combacau, 1991] [de Bonneval, 1993].

Muni de cet ensemble de fonctions, le système de surveillance peut réagir à toute évolution observable du procédé [Tawegoum *et al.*, 1994]. Son efficacité dépend bien sûr de l'organisation des fonctions que nous venons de présenter rapidement.

2.4 Intégration ou séparation de la surveillance

Jusqu'à présent, nous avons mené cette présentation des systèmes de commande et de surveillance en respectant l'ordre chronologique d'apparition des problèmes : automatisation de la commande, donc du fonctionnement normal, puis prise en compte des problèmes de surveillance qui en découlent. Dans ce type d'approche, la surveillance est présentée comme un palliatif à la commande, nous en verrons plus loin les conséquences.

A ce titre, ce palliatif peut être envisagé de trois manières différentes : en considérant une intégration de la surveillance au système de commande, en la séparant de la commande ou enfin, à mi chemin entre la séparation et l'intégration, en adoptant une approche mixte de la surveillance à la commande.

La première approche, intégration de la surveillance à la commande, consiste à traiter les comportements normaux et anormaux par rapport à la commande, de la même façon, sans véritable distinction. Cela suppose bien entendu une connaissance accrue du système à gérer quel que soit le type de fonctionnement. Dans ce cas, toutes les évolutions doivent être prévues à l'avance. A chaque évolution normale ou anormale un traitement spécifique est associé. Ce dernier est élaboré hors ligne. L'efficacité d'une telle approche est donc fortement liée à l'exhaustivité du recensement des évolutions et ne garantit jamais une couverture totale des situations de défaillance.

La deuxième approche, séparation de la surveillance et de la commande, présente un avantage certain quant à la possibilité d'allouer des outils différents pour la commande et pour la surveillance. La commande a, dans ce cas, la charge du fonctionnement normal, tandis que la surveillance gère le fonctionnement anormal. En revanche, la séparation de la surveillance et de la commande entraîne inévitablement des conflits d'actions sur le procédé. En situation de défaillance, la surveillance peut imposer des évolutions spécifiques du procédé. Toutefois, ces évolutions, considérées normales pour la surveillance, ne le seront sûrement pas pour la commande.

Une solution à ce problème consiste en une approche mixte (détection et reprise intégrées à la commande, séparation des autres fonctions). Dans ce cas, nul besoin de prévoir à l'avance l'ensemble des évolutions anormales. Seules les évolutions normales sont à prendre en compte. En effet, sont considérées anormales les évolutions qui ne peuvent être traitées par la commande. Notons que dans ce cas, l'activité de commande A_m présentée dans la figure 1.6 du paragraphe 1.4.2 est étendue puisque qu'elle intègre implicitement la fonction détection. Les autres fonctions de la surveillance restent, quant à elles, séparées de la commande [Combacau, 1991].

Nous avons montré dans les paragraphes précédents que pour faire face à la complexité des applications mises en jeu dans le domaine manufacturier, le recours à une structure hiérarchisée de commande est nécessaire. La prise en compte de défaillances du procédé nous a ensuite amené à considérer une approche mixte de la surveillance. Abordons maintenant le problème lié à l'intégration de la surveillance dans la structure hiérarchique de commande.

2.5 Répartition de la surveillance

Les principes de fonctionnement de la commande dans un contexte hiérarchisé imposent que seul le module ayant émis une requête vers le niveau inférieur est apte à la modifier si celle-ci est remise en cause au niveau inférieur :

> "lorsqu'un niveau se voit confier une tâche à exécuter (il reçoit une requête), il possède toutes les connaissances nécessaires à son exécution mais il n'a pas,

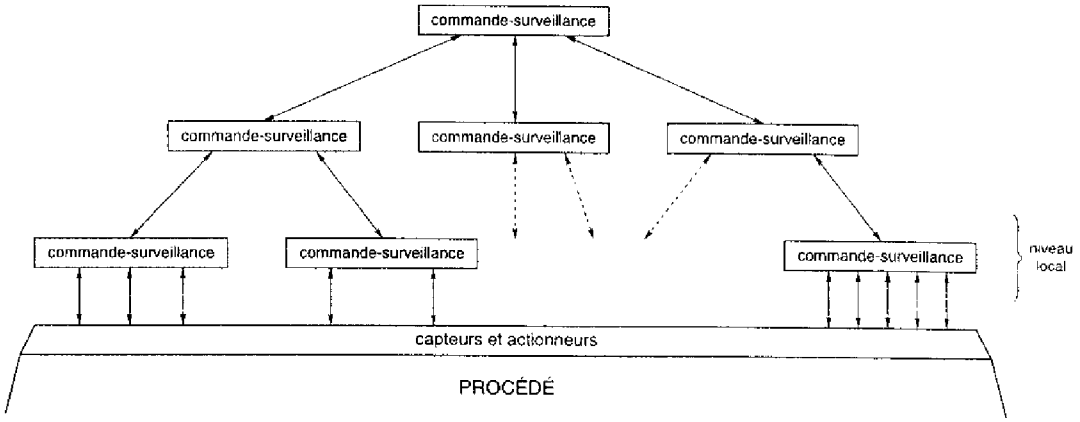


FIG. 2.1 – *structure hiérarchique et modulaire de commande surveillance*

en général, la connaissance de la raison pour laquelle cette requête lui a été envoyée" [Combacau, 1991].

Cela implique donc que pour modifier la requête émise, c'est à dire reconsidérer la commande en cours, le système de surveillance prenne le relai. Pour cette raison, la répartition du système de surveillance dans chacun des modules de la hiérarchie de commande (cf. figure 2.1) est souhaitable.

Notons que la détection de la défaillance émanant du procédé ne peut être faite qu'au niveau local. Si les conséquences de la défaillance interdisent l'exécution de la requête émise par le niveau supérieur, le traitement de défaillance doit être propagé dans la hiérarchie. Cette propagation prend fin lorsqu'un niveau a la connaissance nécessaire pour traiter le problème. A ce niveau, le système de surveillance **confine** les conséquences de la défaillance : le niveau directement supérieur n'est pas informé de celle-ci.

En conséquence, chaque niveau doit être en mesure d'assurer son propre traitement de défaillance. Sachant que les principales informations requises par ces traitements [de Bonneval *et al.*, 1991] [Chaillet *et al.*, 1994] [Hammami *et al.*, 1996] concernent l'état courant du procédé, nous pouvons admettre que la structure hiérarchique de commande et de surveillance doit posséder un **modèle** de ce procédé dans chacun de ces modules. Nous présentons dans le paragraphe suivant les différentes utilisations de ce modèle.

2.6 Commande-Surveillance et modèle du procédé

La majorité des travaux portant sur la surveillance des défaillances du procédé a mis en évidence la nécessité d'intégrer au système de commande-surveillance un modèle du procédé dont l'état est en permanence mis à jour par les évolutions provoquées par la commande et par les signaux émis par les capteurs. Nous y retrouvons beaucoup d'informations accumulées pendant le fonctionnement normal qui sont exploitables après l'occurrence d'une défaillance.

L'étude qui est présentée dans ce paragraphe est extraite de [Zamaï, 1994]. Elle porte sur l'utilisation de ce modèle du procédé à des fins de surveillance. Elle est bien entendu restreinte aux S.E.D.

2.6.1 Approche développée par le CRAN/LACN (Nancy)

Cette approche est fondée sur l'utilisation d'un modèle du comportement normal de chaque Élément de la Partie Opérative (E.P.O.) [Lhoste, 1991]. Chacun de ces modèles est utilisé en tant que filtre de commande. Ces filtres assurent que les requêtes issues de la partie commande sont cohérentes vis à vis de l'état courant modélisant les E.P.O.

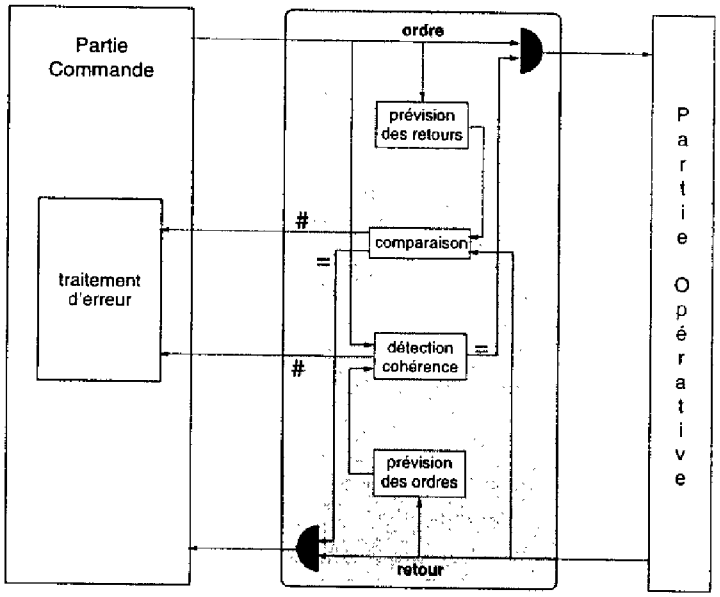


FIG. 2.2 – *utilisation du modèle en filtre : émulation du comportement*

Ces filtres permettent également de comparer les comptes rendus émanant de la partie opérative avec ceux prévus et calculés par le filtre (cf. figure 2.2). Une différence entre ces deux valeurs traduit une défaillance de la partie opérative. La fonction détection est séparée de la commande et autorise la poursuite du traitement d'erreur par une phase de diagnostic [Lhoste, 1991].

2.6.2 Approche développée par le LAIL (Lille)

Dans cette approche, le modèle du procédé est utilisé en filtre, positionné entre la partie commande et la partie opérative (procédé) au sein d'un module de surveillance (cf. figure 2.3).

Le modèle du procédé est utilisé par le filtre de commande dont le rôle est de s'assurer de la validité des consignes à envoyer vers le procédé selon son état courant (modèle). Ceci est rendu possible grâce à une remise à jour permanente du modèle. Nous noterons que

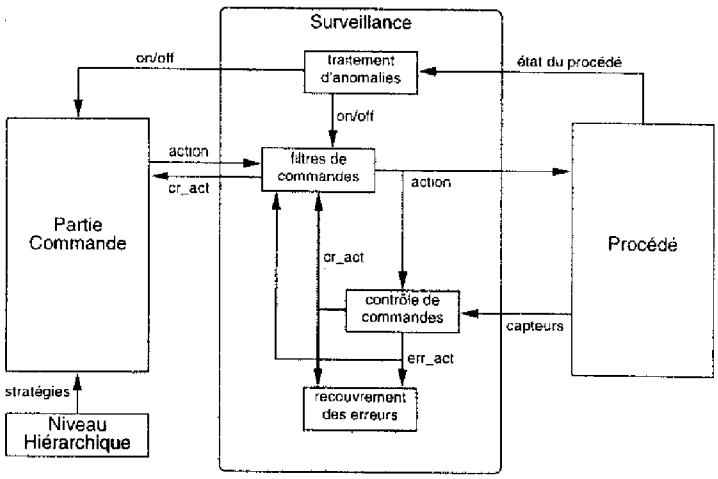


FIG. 2.3 – *utilisation du modèle en filtre de commande*

ce dernier est basé sur une représentation du comportement des différents composants du système physique.

Le bloc de contrôle commande (cf. figure 2.3) a un rôle dual de celui du filtre de commande. Il agit en effet sur les comptes rendus émanant du procédé et non sur les consignes. Un contrôle de ces informations (filtre de valeur de capteurs [Cruette, 1991]) par simulation du comportement, permet alors de vérifier la réalisation des services demandés au procédé.

Lorsqu'une erreur de capteur est décelée (par le contrôle de commande) le bloc de recouvrement des erreurs est activé afin de replacer la partie opérative dans un "état initialement souhaité par le système de commande" [Khattabi, 1993]. Enfin, un bloc de

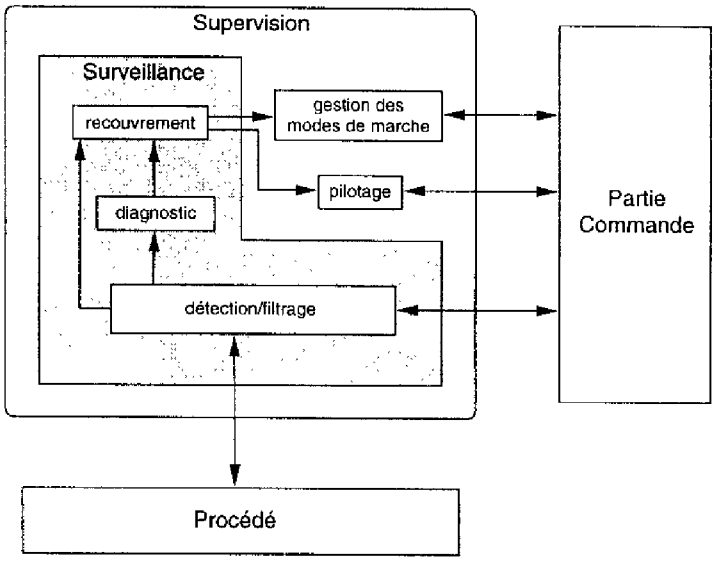


FIG. 2.4 - *vue partielle du modèle fonctionnel de la supervision*

traitement d'anomalies est chargé des défaillances matérielles inattendues. Son rôle est double : détecter ce type d'anomalies puis gérer la mise hors service du composant à l'origine de la défaillance.

Plus récemment [Toguyeni *et al.*, 1996], cette approche a été étendue par l'adjonction d'un système de supervision de la surveillance, du pilotage et de la gestion des modes de marche (cf. figure 2.4). La surveillance est toujours basée sur le principe du filtre pour détecter d'éventuelles évolutions anormales du procédé (détection). Ensuite, selon la gravité de la défaillance (étape de classification [Toguyeni, 1992]) un recouvrement d'erreur est envisagé (conséquences graves de la défaillance) ou un diagnostic est lancé. Le pilotage est dédié quant à lui à la résolution des indéterminismes de la partie contrôle (validité des commandes). Enfin, la gestion des modes de marche permet la conduite du procédé en adéquation avec le mode dans lequel la production se trouve.

2.6.3 Approche développée par le LAMIH (Valenciennes)

L'approche qui est développée dans ce laboratoire est basée sur le Modèle de l'Exploitation des Systèmes Automatisés de Production (MESAP) [Parayre, 1992]. Un exemple de définition fonctionnelle par le MESAP d'une cellule flexible est donné dans la figure

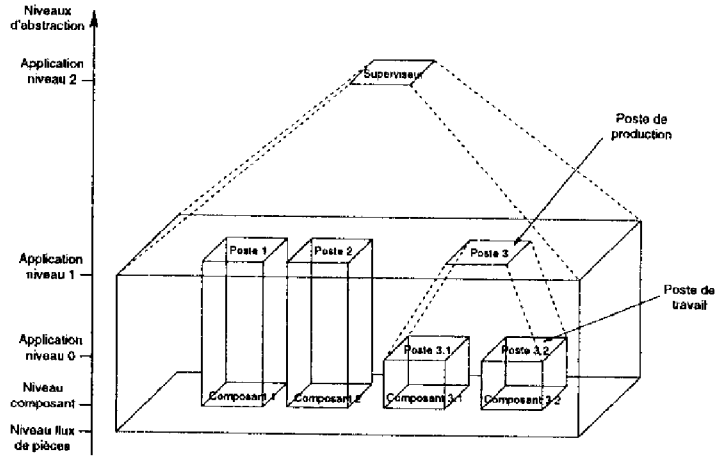


FIG. 2.5 – *exemple de modélisation d'une cellule basée sur le MESAP*

2.5. Le but principal recherché dans [Mabrouk, 1996] consiste à compléter le MESAP en décrivant un nouveau modèle pour aider l'opérateur dans les phases de reconfiguration du système. Le concept de reconfiguration qui est proposé est défini comme un ensemble de modifications (scénarii de reconfiguration) du comportement dynamique et de modes de marche des entités du Système Automatisé de Production (SAP).

La reconfiguration, étape nécessaire après l'occurrence d'une défaillance, est décomposée en deux étapes distinctes. D'abord, à partir du mode de marche courant du SAP, il s'agit de déterminer les scénarii qui permettent de rejoindre le mode de marche à partir duquel la reconfiguration est possible. Une fois ce mode atteint, la phase de reconfiguration du système de commande est lancée. Elle est basée sur l'utilisation d'un outil

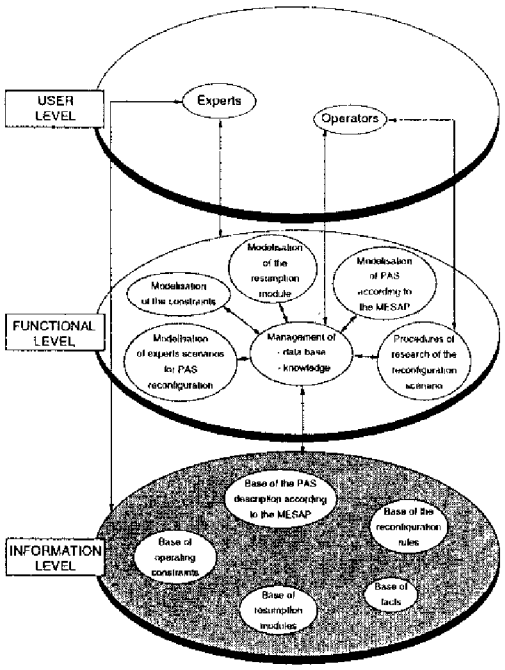


FIG. 2.6 – *structure générale de l'outil d'aide à la reconfiguration*

d'aide à la reconfiguration [Mabrouk *et al.*, 1996]. La structure générale de cet outil est présentée dans la figure 2.6.

2.6.4 Approche développée à l'Université de Carnegie Mellon

Les travaux de Holloway et Krogh [Holoway, 1990] [Holloway, 1991] placent le modèle du procédé en tant qu'émulateur des évolutions normales de la partie opérative. Son rôle est en effet de calculer des fenêtres temporelles d'occurrence des comptes rendus émis par le procédé quand celui-ci est soumis à une commande particulière. Pour une consigne donnée (cf. figure 2.7), un bloc de comparaison permet de vérifier si un compte rendu émis par le procédé arrive bien à la date prévue par le modèle. Ce bloc assure également la mise à jour du modèle du comportement (corrections) de manière à suivre les éventuelles dérives des caractéristiques du procédé (en particulier le vieillissement du matériel). Notons que lorsqu'une défaillance est détectée, le système de comparaison effectue un pré-diagnostic en situant précisément le contexte de la détection : compte rendu concerné, commande en cours et type de symptôme. Contrairement aux approches présentées précédemment, seules les défaillances du procédé sont ici détectées. De ce fait, si les consignes envoyées au procédé et à l'émulateur ne sont pas cohérentes vis à vis de l'état courant du procédé, rien ne garantit que l'émulateur (modèle du comportement normal du procédé) aura le même comportement que le procédé réel. Si ce n'est pas le cas, une erreur de commande sera interprétée comme une défaillance du procédé. Le modèle du procédé est alors désynchronisé de la partie opérative.

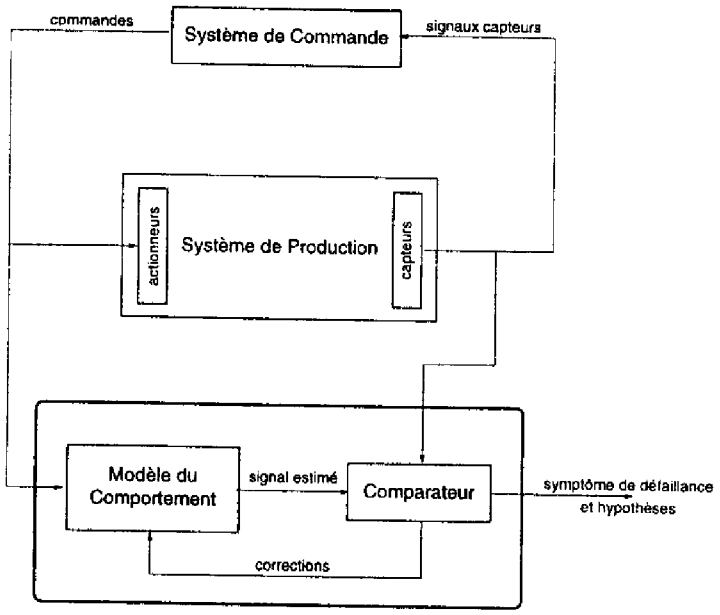


FIG. 2.7 – *utilisation du modèle en émulateur*

2.6.5 Approche développée par le LAI (lyon)

Dans le cadre de la surveillance, cette approche intègre au système de commande par retour d'état, représenté dans la figure 2.8, un système de détection et de diagnostic (cf. figure 2.9). Ce système requiert un modèle du procédé, appelé modèle de référence, qui représente son comportement sans faute [Rezg, 1996]. Ce modèle transforme alors une commande en un vecteur de marquage. La détection est fondée sur la comparaison

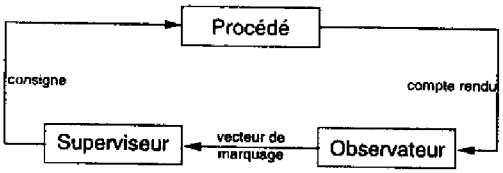


FIG. 2.8 – *retour d'état par utilisation d'un observateur*

du vecteur de marquage construit à partir des comptes rendus émis par le procédé et de celui émis par le modèle du procédé. Le système de diagnostic se charge ensuite de déterminer et de localiser la faute selon le *processus de danger* [Niel *et al.*, 1994].

Cette approche met en œuvre, pour une défaillance détectée et pré-diagnostiquée, une fonction de compensation (appelée également superviseur) chargée du type d'actions à entreprendre pour assurer la poursuite de l'exécution du service engagé tout en évitant de passer par les états interdits (états à partir desquels certaines lois de fonctionnement correct risquent d'être violées). Ceci est réalisé en contrôlant l'accessibilité à certains marquages, malgré la faute [Rezg, 1996].

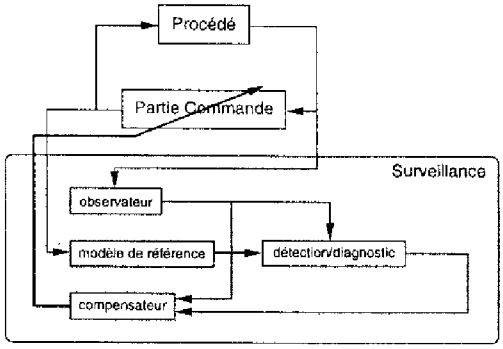


FIG. 2.9 – *utilisation du modèle en émulateur*

2.6.6 Approche développée par le LAAS (Toulouse)

Le modèle du procédé est ici utilisé comme un modèle de référence par la commande et contient un modèle des comportements normaux du procédé. D'un point de vue surveillance, la détection est ici intégrée à la commande. Les fonctions diagnostic, reprise et urgence sont séparées de la commande [Combacau, 1990]. Avant chaque émission de requêtes de commande vers le procédé, le modèle de référence est consulté par la partie commande (cf. figure 2.10) pour vérifier si toutes les conditions sont réunies à l'envoi de

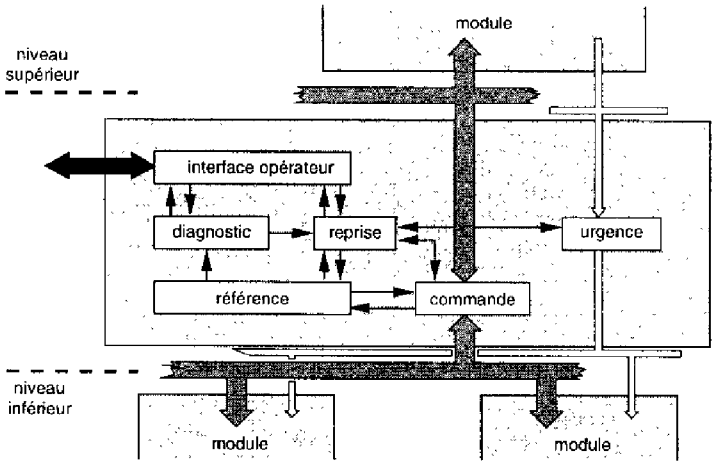


FIG. 2.10 – *utilisation du modèle en référence*

la requête. Si ce n'est pas le cas, une erreur de commande est détectée. Sinon, la requête est émise. La partie commande se met en attente du compte rendu d'exécution de cette requête. Si ce compte rendu émis par le procédé arrive pendant la fenêtre temporelle qui lui est allouée par l'ordonnancement temps réel, la partie commande et le modèle du procédé évoluent simultanément. Le modèle du procédé est alors actualisé. Dans le cas contraire, le compte rendu reçu est rejeté par la partie commande (détection). Toutefois, s'il traduit une évolution possible dans le modèle de référence, ce modèle évolue indépendamment de la commande. Ceci permet de garantir une représentation exacte

de l'état courant du procédé. La prise en compte d'une telle défaillance se poursuit par un traitement de surveillance figé et commun à toutes les approches de surveillance. Ce traitement fait intervenir successivement les fonctions de diagnostic et de reprise [Combacau, 1991] [de Bonneval *et al.*, 1991]. Toutefois, si la défaillance est caractérisée critique par la détection, une procédure d'urgence est alors lancée.

2.7 Conclusion

Au travers de ce chapitre, nous avons fait un tour d'horizon des problèmes posés par la prise en compte en temps réel des défaillances du procédé. Progressivement, nous avons présenté les besoins liés à l'intégration de la surveillance au système de commande. Ces besoins se sont exprimés en terme de fonctionnalités, comme la détection, le diagnostic, la décision, la reprise, le suivi ou encore l'urgence, mais également d'un point de vue structurel. Dans ce dernier cas, une approche mixte préconisant d'une part l'intégration de la détection et de la reprise à la commande et d'autre part, la séparation des autres fonctions de la surveillance a été mise en avant. Rapporté dans la structure hiérarchique de commande définie au chapitre 1, une répartition de la surveillance dans les différents modules de la hiérarchie de commande s'est alors avérée nécessaire. Une étude de différentes approches traitant de la surveillance en temps réel a été ensuite réalisée. Elle montre le rôle essentiel joué par un modèle du procédé, à la fois pour la commande mais également pour la surveillance. Il ressort également de cette étude que le modèle du procédé n'est pas le seul point commun à toutes ces approches. Le type fonctionnement de la surveillance qu'elles imposent est en effet toujours limité à la détection d'une défaillance, le diagnostic de celle-ci et enfin une phase de recouvrement permettant de ramener le procédé dans un état considéré normal pour la commande. L'étude détaillée d'une de ces approches est réalisée dans le chapitre suivant.

Chapitre 3

Étude critique

3.1 Introduction

L'approche de surveillance que nous avons élaborée et que nous présentons dans la seconde partie de ce mémoire est fondée sur celle développée au Laboratoire d'Analyse et d'Architecture de Systèmes depuis quelques années [Combacau, 1991] [de Bonneval, 1993] [Chaillet, 1995]. Pour cette raison, nous la détaillons dans ce chapitre dans le but d'en extraire les avantages à conserver et les lacunes à pallier.

Ce chapitre s'articule sur cinq paragraphes principaux. Dans un premier temps, nous faisons un rappel du concept d'activité. Nous enchaînons ensuite sur la présentation du système de commande puis sur celui de surveillance. Nous terminons par l'analyse des avantages et limitations de l'approche.

3.2 Concept d'activités

Dans cette approche, le système de commande est basé sur l'utilisation conjointe de deux modèles (cf. figure 3.1), le modèle de commande et le modèle de référence (modèle du procédé). Ces deux modèles s'appuient sur le concept d'activité et utilisent les réseaux de Petri à Objets (cf. 2.6) quel que soit le niveau de la structure hiérarchique de commande surveillance considéré (niveaux coordination ou commande locale).

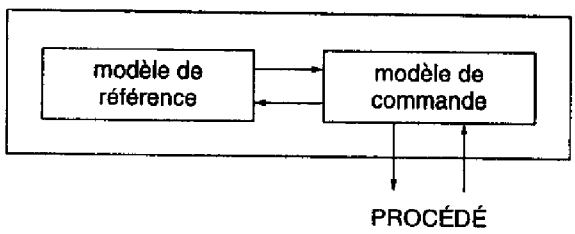


FIG. 3.1 – *structure de commande à deux modèles*

Comme nous l'avons vu à la fin du premier chapitre de cette partie, une activité de commande est caractérisée par une association spécifique de diverses entités dans le but de faire subir un changement d'état à un produit considéré. Ces entités sont généralement des ressources, des produits et de manière implicite la fonction commande chargée de l'exécution des séquences opératoires. Elles correspondent à une utilisation particulière du procédé pour transformer le produit. Ces éléments sont modélisés au moyen de jetons qui sont des n-uplets d'objets.

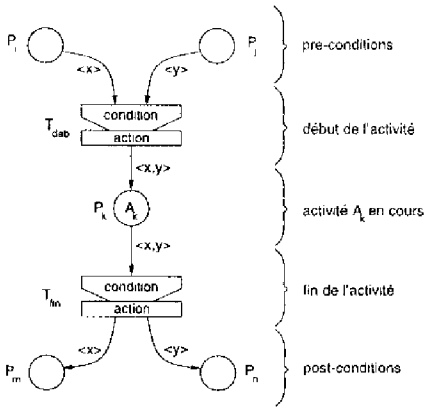


FIG. 3.2 – *activité de commande modélisée par RdPO*

Comme le montre la figure 3.2, une activité de commande est délimitée par deux événements représentés par deux transitions t_{deb} et t_{fin} du réseau de Petri. Chacune de ces transitions conditionne la validité de l'événement par des tests correspondant à des conditions de tir. Ces tests concernent les places d'entrées (disponibilité des ressources par exemple) de t_{deb} et t_{fin} ainsi que les interprétations qui y sont associées.

Lorsqu'une activité de commande est terminée, la transition t_{fin} est sensibilisée. Son tir entraîne le marquage des places P_m et P_n en général appelées post-conditions.

3.3 Point de vue commande

Cette approche se distingue en utilisant deux modèles différents pour la commande. Dans chacun de ces deux modèles des contraintes distinctes sont représentées :

1. Dans le modèle de commande sont représentées toutes les contraintes liées à la transformation du produit. Il s'agit donc de contraintes opérationnelles.

2. Dans le modèle de référence sont représentées toutes les contraintes liées au procédé (exclusions mutuelles, séquencements obligatoires, ressources nécessaires à l'exécution d'une ou plusieurs activités de commande, limite de capacité, etc.). Il s'agit donc de contraintes structurelles et matérielles.

Dans ce paragraphe, nous allons tout d'abord nous intéresser à la phase d'élaboration (hors ligne) de ces deux modèles puis à leur exploitation (en ligne).

3.3.1 Élaboration hors ligne des modèles

3.3.1.1 Modèle de Référence

La démarche d'élaboration du modèle de référence est qualifiée d'ascendante. Elle consiste, en partant du niveau le plus bas (éléments du procédé) jusqu'au niveau le plus haut (vision globale de l'atelier), à modéliser les services offerts par chacun de ces niveaux. De cette façon, pour un module à un niveau donné, toutes les activités (et les contraintes qui les lient) réalisables par le sous-système commandé par le module considéré sont représentées. Il est important de noter que cette modélisation ne tient aucun compte d'une commande particulière. Cela garantit un degré de couverture important des services offerts par le sous-système commandé.

De l'utilisation d'un tel modèle du procédé découlent au moins trois avantages : premièrement, toutes les contraintes liées à la structure physique du procédé sont rejetées dans le modèle de référence; le réseau de commande est donc allégé. Deuxièmement, le modèle de référence est une source d'information considérable pour les fonctions de la surveillance. Enfin, lors de l'étape d'élaboration du modèle de commande (hors ligne), le modèle de référence fournit au concepteur une aide précieuse. Il propose en effet les activités de commande offertes au niveau d'abstraction considéré et les contraintes qui lient ces activités.

3.3.1.2 Modèle de commande

L'élaboration du modèle de commande consiste à imposer une utilisation particulière de certains services offerts par le sous-système commandé. Il s'agit d'imposer :

- l'ordre d'exécution de certaines activités proposées dans le modèle de référence,
- les dates limites de ces activités,
- l'affectation des ressources physiques ou abstraites chargées de les exécuter,
- et enfin, le contenu des messages échangés entre les modèles de référence et de commande ainsi qu'entre la commande et les niveaux adjacents.

La démarche est ici descendante. Successivement, au travers des différents niveaux d'abstraction de la structure hiérarchique, une requête de commande de haut niveau est décomposée en une séquence de requêtes plus élémentaires. Un modèle de commande est donc implanté dans chaque module concerné par la décomposition de la requête.

Les contraintes déjà exprimées dans le modèle de référence n'ont pas à être redécrites dans le modèle de commande. En revanche, les activités de commande y sont dupliquées. Leur organisation dans le modèle de commande définit la séquence de commande capable de satisfaire la requête émise par le niveau supérieur.

3.3.2 Exploitation en ligne des modèles

3.3.2.1 Communication commande-référence

Puisque les contraintes structurelles sont rejetées dans le modèle de référence, il va de soi que lors de l'exécution d'une activité de commande elles doivent être vérifiées. Une activité de commande ne peut donc être lancée que si les contraintes imposées par les transitions du réseau de commande et du réseau de référence sont validées. Cette consultation du réseau de référence par la commande revient en fait à fusionner les transitions qui encadrent l'activité de commande à exécuter. Le détail de cette fusion peut être trouvé dans [Combacau, 1991].

3.3.2.2 Communication inter-niveaux

Comme la plupart des approches que nous avons pu recenser [Khattabi, 1993] [Rezg, 1996] [Holoway, 1990] [Combacau, 1991], celle-ci utilise une communication entre le système de commande-surveillance et le procédé purement orientée commande. Le mécanisme employé est un appel de procédure à distance. Lorsqu'une requête de commande est émise vers le niveau inférieur, un compte rendu d'exécution du service demandé est attendu. Bien entendu, le format du compte rendu est préalablement fixé de sorte qu'il puisse être associé à la requête correspondante [Combacau, 1991].

Cependant, nous verrons plus loin que la simplicité d'un tel mécanisme de communication dans un contexte de fonctionnement normal devient un désavantage indéniable pendant le traitement de défaillance.

3.4 La surveillance

D'un point de vue surveillance, cette approche a pris en compte, dès le début [Sahraoui, 1987] [Combacau, 1991], toutes les fonctions de surveillance requises (détection, diagnostic, décision/reprise, urgence). L'étude progressive de certaines fonctions [de Bonneval, 1993] [Chaillet, 1995] a mis en évidence quelques limitations que l'on retrouve dans toutes les approches citées au chapitre précédent. Comme pour la commande, nous allons nous attacher ici à décrire les différentes phases d'élaboration et d'exploitation de la surveillance.

3.4.1 Élaboration hors ligne

3.4.1.1 La boucle de surveillance

Comme nous l'avons vu au chapitre 2 de cette partie, le traitement de base d'une défaillance consiste à enchaîner successivement les fonctions de surveillance telles que la détection, le diagnostic et la reprise. Comme le souligne à juste titre El Khattabi

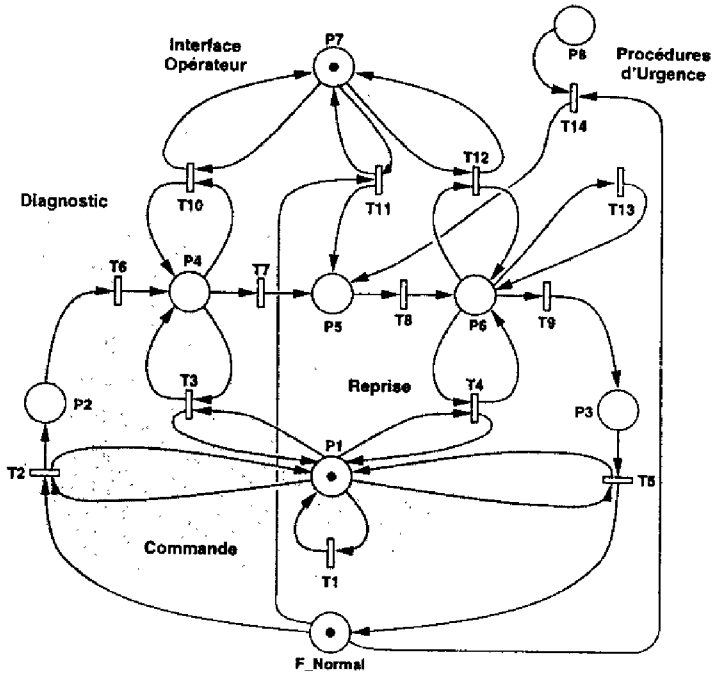


FIG. 3.3 - *traitement global de surveillance*

dans ses travaux [Khattabi, 1993], ce traitement est "purement séquentiel". La figure 3.3 nous en donne une représentation sous la forme d'un réseau de Petri issu des travaux [Combacau, 1991].

Lorsqu'une défaillance est détectée (T2) pendant la commande (la détection est intégrée à la commande), la détection génère alors un symptôme de défaillance (P2). Le marquage de cette place remplit les conditions nécessaires au déclenchement du diagnostic (T6). Le diagnostic peut alors, si besoin est, consulter les informations contenues dans les modèles de commande (T3) ou faire appel à un opérateur (T10). Cette fonction se termine (T7) par la génération d'une conclusion de diagnostic (P5) constituant une information nécessaire au déclenchement de la reprise (T8).

Durant son exécution (P6), la reprise peut accéder aux informations réparties dans les modèles de commande (T4) et à celles détenues par l'opérateur (T12) afin d'établir une solution de retour en fonctionnement normal (F_normal). Le début de la phase d'élaboration effective de la reprise débute dès le tir de la transition T9. Le tir de T5 représente alors la modification des modèles de commande en vue de la reprise. La commande assure ensuite l'application de la séquence de reprise (reprise intégrée à la commande) comme si c'était une séquence normale de commande (T1). La détection est à nouveau active.

3.4.1.2 Système d'information

Les travaux [Chaillet *et al.*, 1993b] ont montré la carence en information, en particulier au niveau des liens entre les machines, du modèle de référence pour "un système de surveillance en temps réel, global et efficace" [Chaillet, 1995]. Ce manque d'information est évident lorsque l'on considère le traitement d'une propagation de défaillance d'une activité de commande à une autre. Il s'agit alors de mettre en œuvre un mécanisme de propagation descendant (dans la structure hiérarchique) pour retrouver quelle activité de commande est à l'origine de la défaillance détectée au cours d'une autre activité de commande. Dans ce but, un système d'information, contenant des informations relatives à la structure du procédé (entités physiques, liens entre ces entités, etc...), est mis en place en parallèle à la structure hiérarchique de commande surveillance (cf. figure 3.4). Un centre de gestion assure les différentes communications entre la structure hiérar-

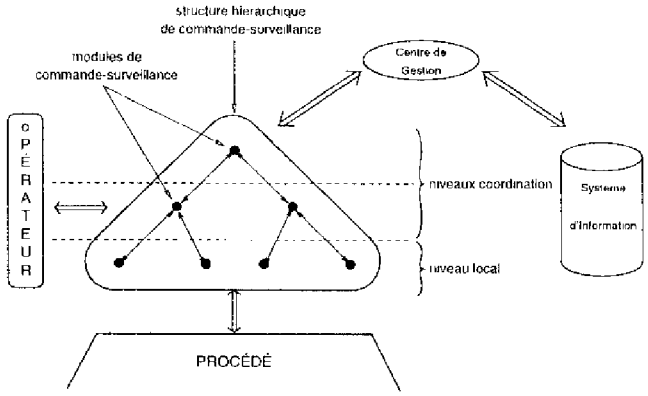


FIG. 3.4 – *architecture globale de l'approche LAAS*

chique et le système d'information en proposant divers services comme la "Recherche", la "Vérification", la "Mise à jour" ou la "Récupération". Une utilisation contrôlée de ces services permet de déclencher des diagnostics détaillés dans les modules concernés [Chaillet, 1995].

3.4.2 Fonctionnement de la surveillance

Dès qu'une défaillance est détectée, un symptôme de défaillance est envoyé au diagnostic. Le déclenchement de cette fonction active un raisonnement à base de règles permettant d'élaborer des hypothèses de diagnostic. Si aucune hypothèse n'est générée, le traitement de défaillance est propagé vers le niveau supérieur dans lequel un traitement similaire est réalisé. Dans le cas contraire, les données profondes incluses dans le système d'information sont utilisées pour préciser les hypothèses faites. Cette étape de recherche correspond à un affinement de diagnostic. Elle est entièrement détaillée dans [Chaillet, 1995]. Elle peut cependant se résumer de la façon suivante. La recherche réalisée consiste à sélectionner un module dit candidat. Ce module candidat est défini comme étant une entité du système d'information ayant un rapport avec la défaillance

détectée. Lorsqu'un tel module est trouvé, le centre de gestion renvoie un compte rendu à la fonction reprise à l'initiative de la recherche. Ce compte rendu contient plusieurs données, en particulier le numéro d'identification du module correspondant au candidat. De cette façon, la fonction reprise peut lancer un traitement de diagnostic dans ce module. Lorsque ce diagnostic se termine, il transmet ses résultats à la fonction reprise du module.

3.5 Avantages de l'approche

L'approche à modèle de référence induit un certain nombre d'avantages, pour le système de commande et pour le système de surveillance.

3.5.1 Commande

Hors ligne, le modèle de référence représente un support important pour l'aide à l'élaboration des séquences de commande. En ligne, et en particulier en fonctionnement normal, le système de commande est bien adapté aux systèmes complexes par structure hiérarchique de commande. Le rôle de chaque modèle est clairement identifié : le modèle de commande est consacré à l'aspect "opérationnel" de la fabrication alors que le modèle de référence est chargé d'autoriser l'exécution des activités de commande en fonction de contraintes liées à la structure du procédé.

3.5.2 Surveillance

D'un point de vue surveillance, cette approche propose une détection à deux niveaux :

1. une détection orientée commande :
   - détection des erreurs de commande par le biais du modèle de référence,
   - détection des évolutions anormales du procédé par rapport à celles attendues par la commande.
2. une détection orientée procédé lorsque son comportement n'étant déjà pas conforme à celui prévu par la commande ne représente aucune évolution possible dans le modèle de référence. Il y a donc transgression d'une ou plusieurs contraintes structurelles.

Quel que soit le symptôme de défaillance détecté, toutes les fonctions de surveillance bénéficient d'importantes sources d'information : le modèle de référence, le modèle de commande et le système d'information.

3.6 Limitations de l'approche

Comme la plupart des approches de commande surveillance [Khattabi, 1993] [Combacau, 1991] [Niel *et al.*, 1994], etc., la communication entre deux niveaux décisionnels, basée sur le mécanisme d'appel de procédure à distance [Zamaï, 1996], est exclusivement conçue pour les besoins de la commande et non ceux de la surveillance. Ceci engendre une limitation importante illustrée par l'exemple suivant. Supposons qu'un module de commande surveillance de niveau local décide, après détection et diagnostic d'une défaillance, de lancer une séquence de reprise dont la durée risque de dépasser celle imposée par le niveau de coordination. Deux messages successifs doivent être pris en compte par le niveau coordination :

1. un message d'erreur envoyé dès le début de la séquence de reprise et signalant que la requête ne peut être exécutée,
2. un message signalant que la séquence de reprise lancée au niveau local est terminée. Le module de niveau local est donc revenu en fonctionnement normal.

Le premier message est correctement interprété comme un symptôme de défaillance. En revanche, le second message ne peut être pris en compte par un simple mécanisme d'appel de procédure à distance. Ce protocole ne peut en aucun cas satisfaire aux besoins de la surveillance.

Le mécanisme de communication n'est pas la seule limitation de cette approche. En effet, nous avons pu noter qu'une hypothèse très restrictive a été faite sur les traitements de surveillance proposés. La boucle de surveillance définie est trop rigide. Elle n'autorise pas, par exemple, le lancement d'un diagnostic détaillé sans passer par une séquence de reprise, ou bien ne prévoit pas la possibilité de détecter une défaillance puis de lancer une procédure d'urgence appropriée tout en diagnostiquant la défaillance pour ensuite envisager une solution de reprise, etc... Tous ces exemples montrent que chacune des fonctions de commande surveillance peut être un point d'entrée dans la boucle de surveillance vers laquelle les messages tels que "requête de diagnostic détaillé", "compte rendu d'exécution", "fin de reprise locale", etc... doivent être orientés.

3.7 Conclusion

Ce chapitre a présenté une synthèse des solutions préconisées dans l'approche développée au Laboratoire d'Analyse et d'Architecture de Systèmes. Un module générique offrant les différentes fonctions de commande et de surveillance y est proposé. Le système de commande utilise deux modèles, le modèle du procédé (référence) et le modèle de commande. Ces deux modèles sont basés sur le concept d'activité. Les fonctions de diagnostic et de reprise sont séparées de la commande. La détection est quant à elle intégrée à la commande. Elle est basée sur la comparaison du comportement réel du procédé

(signaux émis par les capteurs) et du comportement prévu spécifié dans le modèle de commande.

Les avantages d'une telle approche sont multiples. Le modèle du procédé permet une élaboration plus simple des modèles de commande. Ensuite, en phase d'exploitation, l'efficacité du système de commande est considérablement accrue grâce à la répartition des contraintes dans les deux modèles. Seules les contraintes opérationnelles sont en effet représentées dans le modèle de commande.

D'un point de vue surveillance, les avantages sont également nombreux : détection à deux niveaux, fonctions nécessaires aux besoins requis par la surveillance, accès à une base de données distribuée dans chaque module (modèles de référence et de commande) et au système d'information séparé de la structure hiérarchique.

Cependant, cette approche présente des limitations suffisamment importantes pour remettre en cause les principes de fonctionnement du système de surveillance : mécanisme de communication inter-niveaux orienté commande, rigidité des traitements de surveillance, un seul point d'entrée dans la boucle classique de surveillance (détection).

Tenant compte de ces avantages et limitations, nous allons maintenant présenter dans la partie II notre contribution : la prise en compte des besoins de la surveillance pour l'élaboration d'une approche de commande-surveillance.

Partie II

Une approche de Surveillance-Commande

Chapitre 1

Généralités

1.1 Introduction

L'approche que nous présentons ici se singularise en abordant la commande comme un cas simple et fréquent de la surveillance. Pour cette raison, nous avons appelé notre approche, approche de surveillance-commande pour la distinguer des approches de commande-surveillance dans lesquelles la surveillance est greffée au système de commande.

Notre premier travail consiste à briser la structure complète du nœud (module) de commande-surveillance développé dans l'approche LAAS. Nous avons conservé toutefois la plupart des fonctions de surveillance-commande existantes ainsi que le modèle de référence pour la commande et le système d'information dont l'utilité n'est pas à redémontrer ici. Seule la fonction reprise [de Bonneval, 1993] a été décomposée en la fonction décision et la fonction reprise. Cette distinction a été faite de manière à bien séparer la phase d'élaboration d'une solution (décision) de sa mise en œuvre effective (reprise) pour corriger les effets d'une défaillance. Une fonction "suivi" a été ensuite implantée pour satisfaire aux besoins de la surveillance en terme de mise à jour des modèles du procédé (modèle de référence pour la commande et système d'information).

Dès lors, doté de cet ensemble de fonctions, il s'agit de restructurer le nœud de surveillance-commande de manière à accroître les performances de la surveillance, c'est à dire augmenter la réceptivité du nœud aux informations externes et définir les traitements de surveillance-commande à appliquer en toutes circonstances.

1.2 Réceptivité

Pour être réactif, le nœud de surveillance-commande doit être réceptif aux événements externes. Ces événements sont en fait toutes les informations qui transitent par le nœud de surveillance-commande pendant les phases de production. En préliminaire à la conception d'un nœud satisfaisant au critère de réceptivité, nous avons réalisé une étude exhaustive mettant en évidence l'existence de différentes classes d'informations.

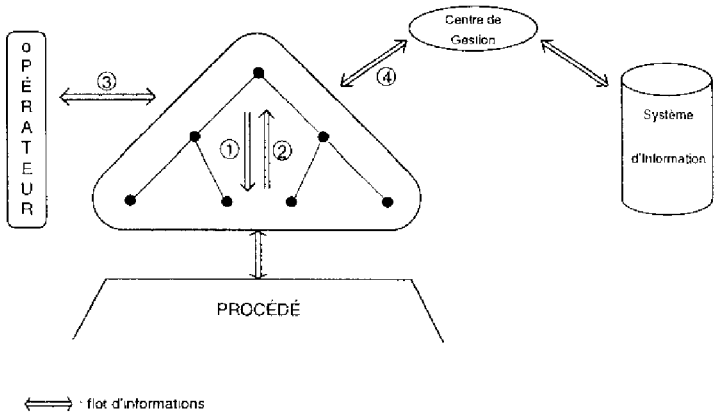


FIG. 1.1 – *les flots d'informations dans le système de surveillance-commande*

Durant le fonctionnement normal (exempt de défaillances) ou le traitement de défaillances, un ensemble d'informations est échangé entre les nœuds des niveaux adjacents de la hiérarchie, entre les nœuds et l'opérateur et enfin, entre les nœuds et le centre de gestion (cf. figure 1.1). Nous avons classé ces informations en quatre catégories de flots.

flot 1 : ce sont les requêtes de commande, de diagnostic [Chaillet, 1995], de reprise ou d'urgence imposées par le niveau supérieur,

flot 2 : flot d'informations représentant l'ensemble des comptes rendus émis par le sous système contrôlé. Il s'agit en particulier des comptes rendus d'exécution de service de commande, des fins de reprise locales, des comptes rendus dûs à des évolutions non contrôlées (intervention en mode manuel d'un opérateur sur le procédé), etc,

flot 3 : flot transversal matérialisant un échange d'informations entre l'opérateur et les nœuds de surveillance-commande. Cet échange est nécessaire lorsque le système de surveillance intégré au nœud de surveillance-commande n'est pas en mesure de prendre la "bonne" décision ou bien lorsqu'il lui manque des informations pour prendre cette décision,

flot 4 : flot de données mis en évidence lorsque le nœud de surveillance-commande doit consulter le système d'information via le centre de gestion [Chaillet *et al.*, 1993a]. Ceci se produit par exemple lorsqu'un diagnostic détaillé est lancé dans les niveaux inférieurs. En effet, les informations contenues dans le système d'information permettront de trouver le nœud "cible" à l'origine du traitement de la défaillance (cf. [Chaillet, 1994] et §3.4.1.2).

Il existe donc quatre classes de données [Chaillet-Subias *et al.*, 1997] face auxquelles un nœud de surveillance-commande doit réagir en choisissant un traitement approprié. Par exemple, suite à l'occurrence d'une défaillance et selon sa gravité, le nœud devra soit lancer un traitement classique de défaillance (diagnostic, décision, reprise), soit l'ignorer et continuer à produire, c'est à dire satisfaire la requête de commande issue du niveau supérieur, soit encore lancer une procédure d'urgence, etc.

Dans certaines situations, ce choix peut être remis en cause et modifié selon l'occurrence d'autres perturbations. Reprenons par exemple le cas du traitement classique de défaillance (diagnostic, décision, reprise). En supposant que le procédé continu à évoluer durant un processus de diagnostic ou de décision, des informations traduisant cette évolution doivent être considérées.

En effet, si ce nœud de surveillance-commande ne prenait pas en compte cette évolution, il n'aurait plus la connaissance de l'état réel du procédé; il serait alors illusoire de prétendre décider d'une quelconque forme de reprise et de l'appliquer. Plus grave encore est le cas d'une information traduisant la transgression d'une contrainte structurelle. Le déclenchement d'une procédure d'urgence doit interrompre le traitement de surveillance et la commande en cours. La prise en compte de cette information est donc nécessaire quel que soit le traitement engagé dans le nœud de surveillance-commande [Zamaï *et al.*, 1997].

En conséquence, une action lancée dans un nœud de surveillance-commande ne doit pas entraver la prise en compte d'informations. Pour cela, des mécanismes de communication inter-niveaux doivent être définis. Ils seront exposés dans la suite de ce mémoire.

1.3 Traitements de surveillance-commande

Comme nous venons de l'entrevoir, une fois l'information prise en compte, il s'agit d'y associer le traitement adéquat de surveillance-commande. Cela consiste à enchaîner correctement les fonctions de surveillance-commande. Pour cela, il nous faut non seulement garantir l'intégrité de ces traitements mais aussi leur cohérence (par exemple, lancer une décision avant un diagnostic n'est pas cohérent). Nous allons nous attacher, dans ce qui suit, à décrire comment modéliser ces traitements.

1.3.1 Les états d'un système de surveillance-commande

La gestion d'un système de production manufacturière ne peut être abordée sans considérer les défaillances du procédé, nous venons de le voir. Ces défaillances ont pour origine des déficiences du matériel (casse d'outils, usure, etc.), de mauvaises utilisations du procédé (commandes erronées) ou encore des défaillances propres au système de commande. La commande d'un tel système n'est donc pas suffisante pour garantir la transformation du produit correspondant aux désirs de l'utilisateur. Pour cette raison, un système de surveillance est requis. Il doit permettre de prendre en compte toutes ces défaillances et bien évidemment d'apporter une solution de manière à satisfaire le but fixé par l'utilisateur. Cela revient à imposer un comportement particulier au procédé à la fois en fonctionnement normal (en l'absence de défaillances) mais également en fonctionnement anormal (en présence d'une ou plusieurs défaillances). Ce comportement consistera donc à amener le procédé d'un état dans un autre en respectant un ensemble de contraintes. Ces contraintes sont de quatre types :

C1. celles découlant de l'utilisation du procédé (résistance des matériaux, ressources

partagées, plan de fabrication, etc.),

C2. celles correspondant aux besoins de fabrication (qualité requise pour le produit à fabriquer, temps de fabrication, etc.),

C3. celles imposées par la surveillance (exclusions mutuelles entre les fonctions de surveillance comme le diagnostic et la décision, par exemple),

C4. celles découlant des besoins de l'entreprise en terme de surveillance (tolérance aux fautes [Conan, 1996], qualité de production, etc.) c'est à dire de la "politique" de surveillance.

Dans le cadre de notre travail, nous nous intéresserons plus particulièrement aux deux dernières classes de contraintes. La caractérisation des contraintes liées à la commande du procédé est déjà entièrement réalisée dans [Combacau, 1991]. Dans ces travaux, nous l'avons vu précédemment, la nécessité de faire coopérer un modèle de commande (contraintes C2) et le modèle du procédé correspondant (contraintes C1) est clairement affichée.

Nous proposons de reprendre un schéma similaire pour les modèles de la surveillance, l'un représentant les contraintes de type C4 et l'autre celles de type C3, ce dernier modèle décrivant l'ensemble des états accessibles du système de surveillance et de ses règles de fonctionnement.

A priori, le système de surveillance-commande peut se trouver dans un état formé par l'utilisation d'une des sept fonctions de surveillance-commande ou d'un sous-ensemble de ces fonctions. Cela correspond en fait à 2^7 (7 fonctions de surveillance-commande utilisées ou non utilisées) combinaisons ou **états observables** pour le système de surveillance-commande.

Par exemple, la combinaison diagnostic, commande et détection, peut-être associée à un état du système de surveillance-commande dans lequel un diagnostic est lancé suite à une défaillance détectée en cours de commande du procédé.

En revanche, la combinaison mettant en jeu le diagnostic, la décision et la commande ne présente aucun intérêt et peut même être considérée comme néfaste vis à vis de la cohérence en terme de surveillance. En effet, il est incohérent de prendre une décision pour amener une solution de recouvrement de défaillance tant que le diagnostic n'a pas trouvé la cause de la défaillance préalablement détectée.

Il en est de même pour des combinaisons comme commande, détection et urgence puisque détecter pendant l'exécution d'une procédure d'urgence ne peut qu'aboutir au déclenchement d'alarmes en cascade. L'exécution de la procédure d'urgence entraînant des évolutions sûrement contradictoires avec celles prévues par la commande.

Ainsi, partant d'un ensemble d'**états observables** (2^7) que nous avons partiellement identifiés et qui seront entièrement définis dans le chapitre III, nous décrivons un nouvel ensemble, plus réduit, en prenant en considération des contraintes de surveillance. Nous appellerons ce sous-ensemble, ensemble des états utilisables.

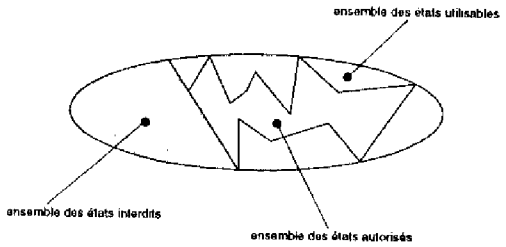


FIG. 1.2 – *les états d'un système de surveillance-commande*

L'utilisateur devra exploiter ce sous-ensemble obtenu, de manière à le réduire encore, prenant en considération les contraintes de type C4 liées au fonctionnement de surveillance-commande désiré. Il s'agira alors de décrire le modèle de la stratégie de surveillance. Nous le verrons plus loin, les contraintes qui y sont représentées découlent en particulier du produit à fabriquer et de la politique de surveillance-commande préconisée par l'entreprise. Ces contraintes permettent ainsi de décrire l'ensemble des états autorisés par l'utilisateur. La figure 1.2, offre une vision globale des états d'un système de surveillance-commande en y montrant les trois catégories d'états :

- **les états interdits** représentant le sous-ensemble des états dans lesquels la cohérence du système de surveillance-commande n'est plus assurée,
- **les états utilisables** qui sont les états appartenant à l'ensemble des **états observables** diminué de l'ensemble des états interdits,
- **les états autorisés** qui caractérisent les états appartenant à la stratégie de surveillance-commande prévue par l'utilisateur.

Remarque : il est tout à fait normal de représenter ici les états du système de surveillance-commande (figure 1.2) de la même manière que ceux du système de commande (cf. figure 1.5 §1.4.1). Que ce soit pour la surveillance ou pour la commande, les définitions des états observables, interdits, utilisables ou autorisés restent les mêmes.

1.3.2 Concept d'activité étendue

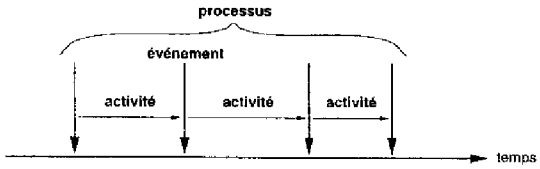
Le concept d'état présenté ne permet pas de modéliser de manière suffisamment détaillée le comportement du système de surveillance-commande. En effet, les aspects séquencements, contraintes, conditions de changement d'états n'y sont pas clairement représentés. Pour cette raison, nous nous sommes dirigés vers les concepts d'activité, événement et processus couramment employés dans les domaines de la simulation [Benzakour, 1985] ou de la commande [Combacau, 1991] des systèmes à événements discrets.

D'après ce que nous avons défini dans la première partie I de ce mémoire, exécuter une activité pour la commande revient à associer des ressources, des produits, la commande

et la détection (première intégration de la surveillance). Toutefois, la fonction détection n'est pas la seule fonction de surveillance à devoir être introduite dans l'association ou activité (nous rejoignons ici la notion de combinaison déjà vue pour le concept d'état). En effet, lorsqu'un diagnostic est lancé, il est appliqué à une ressource pour une commande particulière. Le diagnostic intervient donc lui aussi dans l'association. On peut facilement montrer [Zamaï *et al.*, 1998] qu'il en est de même pour d'autres fonctions de surveillance comme la décision, la reprise, l'urgence ou encore le suivi.

En conséquence, pour prendre en compte les besoins de la surveillance (ensemble des états utilisables), nous devons étendre le concept d'activité à toutes les autres fonctions de la surveillance. Nous proposons dans ce sens une représentation, plus complète, du concept d'activité. **Une activité représente un état particulier du système de surveillance-commande** obtenu en **associant** les différents éléments, ressource, produit, et une ou plusieurs des fonctions de surveillance-commande (le diagnostic, la décision, le suivi, l'urgence, la reprise, la détection et la commande). Par exemple, l'activité de production optimale sera représentée par l'association <*ressource, produit, détection, suivi, commande*>. Chacun de ces éléments appartenant ou non à l'association, le nombre d'activités observables est égal à : $\sum_{i=0}^{9} C_9^i = 2^9$ (9 étant le nombre des éléments pouvant apparaître dans une activité).

Le changement d'un ou plusieurs de ces éléments (ajout ou suppression) dans l'association se traduira par une évolution de l'activité considérée vers une autre tout à fait différente. Par itération, nous retrouvons ici la définition d'un processus [Feuvrier, 1971] [Valette, 1995], c'est à dire une suite séquentielle d'activités disjointes. Ces changements d'activités sont tous provoqués par des événements. Ces événements sont eux mêmes provoqués par les fins de ces activités de surveillance-commande. La figure 1.3 nous donne



activité = <combinaison de i éléments pris parmi 9>
éléments = {ressource, produit, commande, détection, suivi, diagnostic, décision, reprise, urgence}

FIG. 1.3 – *concept d'activité étendue*

une représentation assez générale des concepts rappelés précédemment.

1.3.3 Un modèle de référence pour la surveillance-commande

Dans la plupart des travaux qui traitent de la surveillance [Rezg, 1996] [Khattabi, 1993] [Holoway, 1990] [Combacau, 1991], la surveillance est basée sur un ensemble restreint (sous-ensemble des états utilisables) de règles telles que : "dès détection d'une défaillance du procédé, enchaîner un diagnostic, puis une décision et enfin une reprise permettra de résoudre le problème" ou encore "si transgression d'une contrainte structurelle, lancer la procédure d'urgence associée à ce type de transgression". Le nombre de ces règles est

toutefois trop faible pour répondre aux problèmes de surveillance rencontrés (cf. partie I chapitre 3.6). En effet, que faire sur réception d'un compte rendu signalant qu'un nœud du niveau inférieur sera indisponible pour une durée indéterminée? Quelle suite donner à la réception d'un retour en fonctionnement optimal de ce même module? Que faire d'une requête d'affinement de diagnostic? N'y a t-il pas un intérêt à lancer un processus de diagnostic durant l'exécution d'une procédure d'urgence et envisager de futures solutions de reprise? La fin de l'exécution de cette procédure d'urgence doit-elle être attendue pour lancer ce diagnostic? C'est à toutes ces questions qu'un système de surveillance doit apporter des réponses.

En fait, tel le modèle du procédé (modèle de référence) qui offre pour la commande des activités utilisables et les contraintes qui les lient, nous proposons un modèle identique, d'un point de vue conceptuel, pour la surveillance-commande qui exprimera les règles élémentaires que doit respecter un système de surveillance-commande. Nous appellerons ce modèle **modèle de référence pour la surveillance-commande.**

Ce modèle de référence représentera donc l'ensemble des règles qui régissent la façon dont les fonctions de surveillance-commande sont gérées. Ceci se rapproche fortement des "politiques obligatoires" de la sûreté de fonctionnement [J. Laffont, 1997] qui imposent des règles incontournables permettant de garantir l'intégrité et la cohérence du système de surveillance. En fait, modéliser ces règles consiste à décrire le "fonctionnement normal" de la surveillance-commande. Parmi ces règles, nous pouvons citer :

1. les exclusions mutuelles : par exemple, il ne faut pas détecter pendant l'exécution d'une procédure d'urgence puisque les traitements appliqués sur le procédé vont généralement à l'encontre d'une utilisation normale de ce procédé. Les informations émises pendant ces phases par les capteurs risquant d'être incohérentes,

2. les séquencements obligatoires : un diagnostic doit toujours être exécuté avant de prendre une décision (par exemple). Sauf dans le cas d'une procédure d'urgence, il est impossible d'envisager une solution de reprise sans connaître les causes de la défaillance,

3. les limites de capacité : le système de diagnostic ne peut traiter plus de x défaillances simultanément.

Disposant maintenant du **concept d'activité étendu aux besoins de la surveillance**, nous nous appuierons sur ce formalisme pour décrire le modèle de référence de la surveillance-commande.

1.3.4 Prise en compte de la politique de surveillance de l'entreprise

Comme pour le modèle de commande, le modèle de surveillance-commande (séquence opérationnelle d'activités de surveillance-commande) sera différent selon l'entreprise et la commande à appliquer au produit. Par exemple, la détection d'une défaillance sur une

chaîne de fabrication de boulons n'entrainera pas forcément un traitement complet de surveillance visant à récupérer le produit. Dans ce cas, la politique de surveillance peut être simple: continuer à produire quand même jusqu'à dix détections de défaillances. Au delà, il faudra envisager un traitement complet de défaillance (détection, diagnostic, décision et reprise). En revanche, une telle tolérance sera inacceptable pour la fabrication d'un élément de sécurité d'un avion, d'une voiture ou bien pour la conduite d'une centrale nucléaire. Toute détection de défaillance pourra entraîner soit le déclenchement d'une procédure d'urgence soit l'abandon de la commande en cours.

Nous n'imposons donc plus au concepteur une surveillance préalablement conçue, rigide, mais au contraire nous lui donnons la possibilité de concevoir sa propre surveillance (ensemble des états autorisés), selon ses besoins, ceux de l'entreprise, ceux découlant du produit à fabriquer (la commande donc) et enfin ceux imposés dans le modèle de référence de la surveillance-commande.

Les règles de surveillance telles que nous les avons vues au travers de quelques exemples ne peuvent être intégrées aux lois de surveillance-commande décrites dans le modèle de référence pour la surveillance-commande: elles vont dépendre de l'entreprise et du type de fabrication. Ce type de règles sera intégré hors ligne par l'utilisateur lors de la conception du modèle de la stratégie de surveillance-commande.

1.4 Un outil de spécification : SADT

La résolution d'un problème, quel qu'il soit, passe par une phase préliminaire d'analyse des besoins permettant de définir un cahier des charges précis de la solution envisagée. Notre approche n'a pas échappé à ce principe. Le choix de la méthode d'analyse et de l'outil de représentation du cahier des charges doit répondre à un impératif essentiel : permettre de lever toute ambiguïté de spécification.

Parmi les méthodes les plus couramment utilisées (SADT, SART, MERISE, etc...) [Chapurlat, 1996] [Huvenoit, 1994], notre choix s'est porté sur la méthode SADT [Ross, 1977]. Elle offre un véritable "langage pour communiquer" [Technologie, 1989] car elle est universellement connue.

Cette méthode de spécification fonctionnelle permet de bien mettre en évidence les interactions et les activités (au sens SADT) des différentes fonctionnalités du système étudié, et ce, de manière exhaustive. L'analyse préconisée par cette méthode est de type descendante, hiérarchique et modulaire.

La description du système que l'on désire étudier adopte comme point de départ le contexte global le plus abstrait possible du système. De ce contexte, la méthode SADT préconise de décomposer le système en activités de complexité moindre dans le but d'obtenir, à chaque phase de décomposition, des paliers que nous sommes en mesure de maîtriser (cf. figure 1.4).

Cette démarche utilise un formalisme graphique simple, les actigrammes, qui facilite la compréhension (cf. figure 1.4). Un actigramme décrit une activité modélisée par une

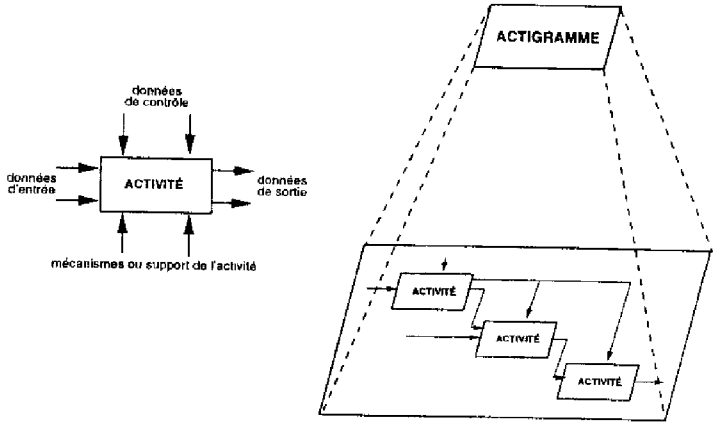


FIG. 1.4 – *actigrammes et décomposition des activités*

boîte. Cette boîte reçoit un flot de données rentrant, les données d'entrée et produit, un flot de données sortant, les données de sortie. Les données d'entrée sont modifiées sous l'effet de directives de contrôle (flèches arrivant sur le dessus de la boîte) en fonction desquelles se déroule l'activité (déclenchement, inhibition, paramétrage). Cette boîte sera également amenée à utiliser les mécanismes ou supports proposés par les flèches situées en dessous de la boîte.

Le cahier des charges établi, l'élaboration d'un modèle des activités de surveillance-commande doit s'appuyer sur un outil possédant un pouvoir d'expression adapté. Dans notre cas, l'outil doit exprimer aisément les concepts d'activités, de processus d'état et les contraintes qui s'y reportent.

1.5 Les Réseaux de Petri à Objets comme outil de modélisation

Nous n'allons pas ici redémontrer pourquoi nous avons choisi les Réseaux de Petri à Objets (RdPO) en tant qu'outil de modélisation. En effet, nous trouverons une étude comparative des méthodes courantes de modélisation dans [Bucci *et al.*, 1995], [Marty, 1994] et [Combacau, 1991], la dernière [Combacau, 1991] mettant en évidence l'adéquation des RdPO à nos attentes :

1. prise en compte de contraintes logiques telles que l'exclusion mutuelle, partage de ressource et séquencements obligatoires,
2. possibilité de modéliser des politiques décisionnelles locales,
3. modélisation facile des mécanismes de communication,
4. possibilité d'analyse formelle de la structure de contrôle sous-jacente garantissant l'absence de blocage dans le modèle (réseau vivant, réseau ré-initialisable, réseau borné, etc.).

La définition formelle du modèle réseaux de Petri à Objets peut être trouvée dans [Sibertin-Blanc, 1988] ou [Valette, 1995] et ne sera pas rappelée.

1.6 Conclusion

Jusqu'à présent, nous avons présenté de manière très générale les concepts de base que doit intégrer une approche de surveillance-commande. Ainsi, partant du concept d'activité, étendu aux besoins de la surveillance, nous avons montré l'utilité de décrire un modèle de référence des traitements de surveillance-commande. Ce modèle devra ensuite être utilisé par le concepteur pour l'aider dans sa phase de conception hors ligne de la stratégie de surveillance-commande. Il a de plus la possibilité d'intégrer d'autres contraintes que celles imposées dans le modèle de référence : des contraintes issues de la politique de surveillance pratiquée dans son entreprise et des contraintes liées au produit à fabriquer. Enfin, nous avons montré qu'il est indispensable de doter notre système de surveillance-commande d'un mécanisme de communication inter-niveaux efficace.

Nous avons ensuite présenté une méthode d'analyse des besoins (SADT) qui permet de représenter l'expression de ce que nous cherchons à appréhender et l'outil de modélisation adapté. Il s'agit donc maintenant d'appliquer cette méthode pour spécifier entièrement notre approche de surveillance-commande au niveau d'un nœud. C'est ce que nous proposons de faire dans la suite de ce mémoire.

Chapitre 2

Spécification d'un nœud de Surveillance-Commande

2.1 Introduction

"Spécifier un problème, c'est définir ce que le système doit faire et non comment il doit le faire" [Technologie, 1989]. C'est dans cet esprit que nous allons analyser ici les besoins d'un nœud de surveillance-commande en terme d'activités (au sens SADT) et de données, c'est à dire d'un point de vue statique. Cela nous permettra de dégager clairement quelles sont les fonctionnalités que doit intégrer ce nœud de surveillance-commande.

Remarque: dans le but d'éviter toute confusion nous emploierons tout au long de cette spécification le terme "fonctionnalités" en lieu et place du terme "activité" réservé à SADT. Ainsi, le terme activité conserve le sens que nous lui avons donné au §1.3.2.

2.2 Spécification

Une application stricte de la méthode SADT conduit à une description extrêmement énumérative de l'ensemble des fonctionnalités et des flots d'entrée/sortie qui leur sont associés. Cette description complète est fournie en annexe page 135 du document. Nous ne reprenons dans ce chapitre que les diagrammes SADT et la description des particularités de la spécification liées à notre approche. La présentation qui suit est donc volontairement limitée et incomplète.

2.2.1 Contexte général

Le rôle principal d'un nœud de surveillance-commande est de prendre des décisions cohérentes vis-à-vis d'événements externes. Ces décisions devront satisfaire à trois types

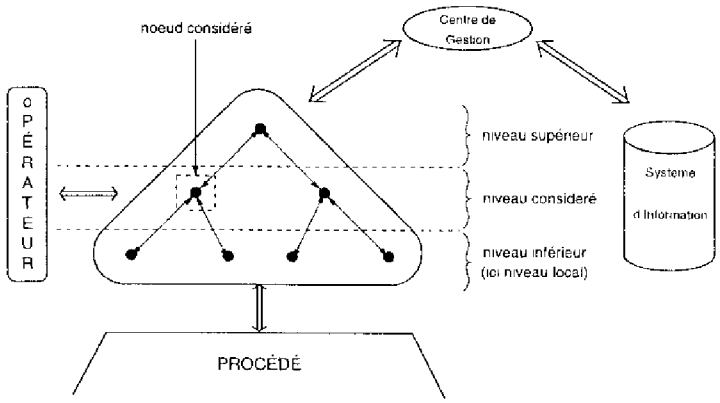


FIG. 2.1 - *vue globale de l'architecture de surveillance-commande*

de contraintes, celles imposées par les règles élémentaires de surveillance-commande, celles imposées par l'entreprise et enfin celles liées au produit à fabriquer. Ces événements externes qui déclencheront ces prises de décision correspondent aux occurrences d'informations émanant de l'environnement dans lequel est placé le nœud, c'est à dire des nœuds des niveaux inférieurs et supérieurs, de l'opérateur humain et du centre de gestion. L'environnement d'un nœud de surveillance-commande est rappelé dans la figure 2.1; le diagramme A-0 de la spécification est donné quant à lui sur la figure 2.2.

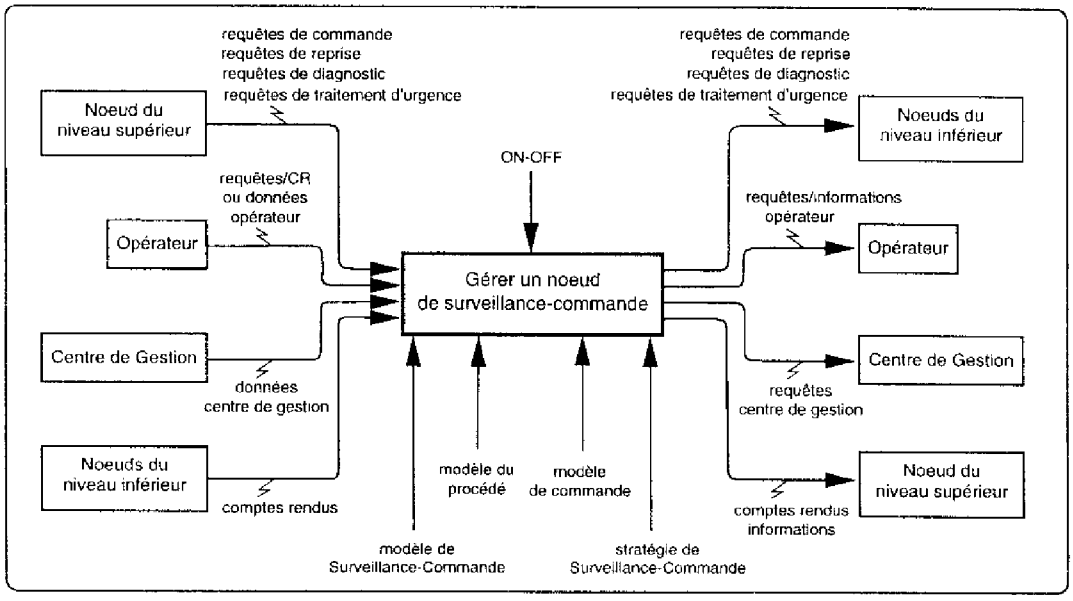


FIG. 2.2 - *diagramme A-0*

2.2.2 Flot d'informations et fonctionnalités pour un nœud

2.2.2.1 Données en entrée

En sus des classiques requêtes de commande, diagnostic et reprise, des comptes rendus qui leur sont associés et des données issues du système d'information, un nœud doit pouvoir prendre en compte des *requêtes opérateur*. Elles correspondent à l'intervention d'un opérateur au sein du système de surveillance-commande. Par ce biais, l'opérateur peut déclencher n'importe quel processus ou bien se substituer à tout ou partie du système. C'est le cas par exemple lorsque l'opérateur décelle une défaillance inobservable par le système de surveillance-commande qui ne dispose pas d'un nombre suffisant de capteurs.

2.2.2.2 Données de contrôle

Le système de surveillance-commande d'un nœud de la structure hiérarchique doit être bien entendu activé ou désactivé. Un flot de contrôle "ON-OFF" est prévu à cet effet (cf. figure 2.2).

2.2.2.3 Données en sortie

Les données produites par un nœud sont classiques : il s'agit de requêtes issues de processus d'affinement des requêtes reçues par le module et des requêtes en direction du centre de gestion ou de l'opérateur.

En revanche, les **comptes rendus** d'activité qui apparaissent sur le diagramme recouvrent bien plus que les comptes rendus d'exécution des séquences de commande. Il peut en effet s'agir des comptes rendus de diagnostic détaillé, de comptes rendus d'exécution de reprise locale ou d'**informations** émises en direction du niveau supérieur pour signaler un état particulier (durée d'indisponibilité par exemple).

2.2.2.4 Supports

- *modèle de commande*, il s'agit de la séquence opératoire conçue pour réaliser le service de commande demandé par le niveau supérieur. Cette spécification a été élaborée en prenant en compte les possibilités offertes par le procédé (contraintes modélisées dans le modèle du procédé), le produit à transformer et enfin la "politique" de production de l'entreprise.
- *modèle du procédé* : ce support a la même vocation que le précédent mais il est plutôt dédié à la commande. Néanmoins, son rôle sera également très important lors d'un traitement de défaillance car il représente l'état réel dans lequel se trouve le procédé. Ceci permettra par exemple, connaissant cet état, de pouvoir appliquer des séquences de reprise.

- *modèle de Surveillance-Commande*: il représente l'ensemble des activités de surveillance-commande offertes à l'utilisateur ainsi que les contraintes qui les lient (séquencements obligatoires, exclusions mutuelles, etc.). Ce modèle décrit l'ensemble des "états utilisables" §1.3.1.

- *stratégie de surveillance-commande*: décrit la séquence d'activités de surveillance-commande respectant les contraintes imposées par la fabrication du produit, les contraintes de surveillance décrites dans le modèle de surveillance-commande et enfin la politique de l'entreprise. Ceci correspond aux "états autorisés" §1.3.1.

2.2.2.5 Les fonctionnalités principales

1. **acquérir** les informations arrivant sur le nœud de surveillance-commande. Cette fonctionnalité garantit la réceptivité du nœud de surveillance-commande.

2. **orienter** les informations reçues vers les fonctions de surveillance-commande capables de les traiter, vers les niveaux adjacents lorsqu'il s'agit de résultats dédiés aux autres nœuds de la hiérarchie, vers l'opérateur ou le centre de gestion. Par exemple, une requête de commande devra être orientée vers la fonction commande, un compte rendu d'exécution vers la fonction détection et la fonction commande, une requête de diagnostic détaillé vers la fonction diagnostic, une fin d'activité de commande vers le niveau supérieur,

3. **représenter l'état courant du traitement de surveillance-commande.** Comme nous l'avons spécifié plus haut dans le chapitre 1, le traitement d'une information se résume assez rarement à l'exécution d'une seule fonction de surveillance-commande. En fait, un traitement de surveillance-commande correspond à un enchaînement et une collaboration de plusieurs fonctions de surveillance-commande.

4. **commander** le sous-système, c'est assurer la réalisation du service de commande demandé par le niveau supérieur. Cela revient à exécuter des séquences d'activités de commande correspondant à l'affinement du service demandé.

5. **détecter** les évolutions anormales du sous-système contrôlé par rapport aux évolutions prévues. Cette détection se fait toujours par rapport à un ensemble d'événements attendus par le système de commande.

6. **diagnostiquer** les défaillances du procédé pour identifier la défaillance dont les symptômes ont été reconnus. Aucune contrainte de temps n'est fixée pour que cette fonction propose une explication au problème constaté. Cela sous entend bien évidemment que le procédé peut évoluer durant cette recherche.

7. **décider** ce qu'il est envisageable de faire pour tendre vers un retour en fonctionnement normal. Dans ce but, cette fonctionnalité sera amenée soit à établir un point de reprise et la séquence permettant d'y accéder, soit de propager le traitement de défaillance vers les niveaux supérieurs si elle n'a pas la connaissance suffisante pour justement établir la séquence de reprise.

8. **reprendre**, c'est appliquer ce qu'il a été envisagé de faire par la fonctionnalité "décider" pour revenir vers un fonctionnement normal ou bien satisfaire une requête de reprise envoyée par le niveau supérieur. C'est précisément le cas lorsqu'il y a eu propagation de traitement de défaillance.

9. **dialoguer** avec l'opérateur, de manière à permettre aux fonctionnalités "diagnostiquer" et "décider" de demander de l'aide à l'opérateur. Cette aide est justifiée par un manque d'information du système de surveillance-commande (état de l'outil par exemple).

10. **suivre** les évolutions du sous-système contrôlé. En fait, il s'agit d'absorber toutes les informations provenant du sous-système de manière à recaler les modèles (modèle de référence et système d'information) sur l'état réel observé. Cela permettra par exemple d'envisager et d'appliquer des séquences de reprise connaissant l'état réel dans lequel se trouve le sous-système contrôlé (point de départ pour l'élaboration d'une séquence de reprise).

11. **appliquer des traitements d'urgence** quand nécessaire et rapidement. Cela suppose bien sûr que toutes les procédures d'urgences ont été pré-définies par l'utilisateur et associées aux symptômes de défaillance autorisant leur déclenchement.

12. **échanger** des informations avec la base de données du système d'information. Ces échanges vont permettre de satisfaire aux besoins exprimés dans les "requêtes centre de gestion".

2.2.3 Diagramme A0 : nœud de surveillance-commande

2.2.3.1 Fonctionnalités à réaliser

Au premier niveau d'abstraction, nous avons choisi de séparer la gestion des informations de leur traitements. De cette façon, nous avons distingué deux grandes fonctionnalités dans lesquelles nous avons réparti l'ensemble des 12 fonctions préalablement énumérées :

- **A01 : Superviser** les informations qui correspond au regroupement de *acquérir (1)* l'information, l'*orienter (2)* vers la fonction estimée apte à "comprendre" cette information et enfin *déclencher le traitement de surveillance-commande (3)* défini et établi par le concepteur. Cette fonctionnalité est donc globalement dédiée à la mise en œuvre de la stratégie de surveillance-commande conçue par l'utilisateur selon les consignes et informations reçues des "quatre directions" (niveau supérieur, niveau inférieur, opérateur et centre de gestion).

- **A02 : Surveiller-Commander** ; cette fonctionnalité regroupe les fonctionnalités de surveillance-commande (commander, détecter, diagnostiquer, décider, reprendre, dialoguer, suivre, appliquer des traitements d'urgence et échanger) qui vont être gérées par la fonctionnalité "Superviser". Ces différentes fonctionnalités

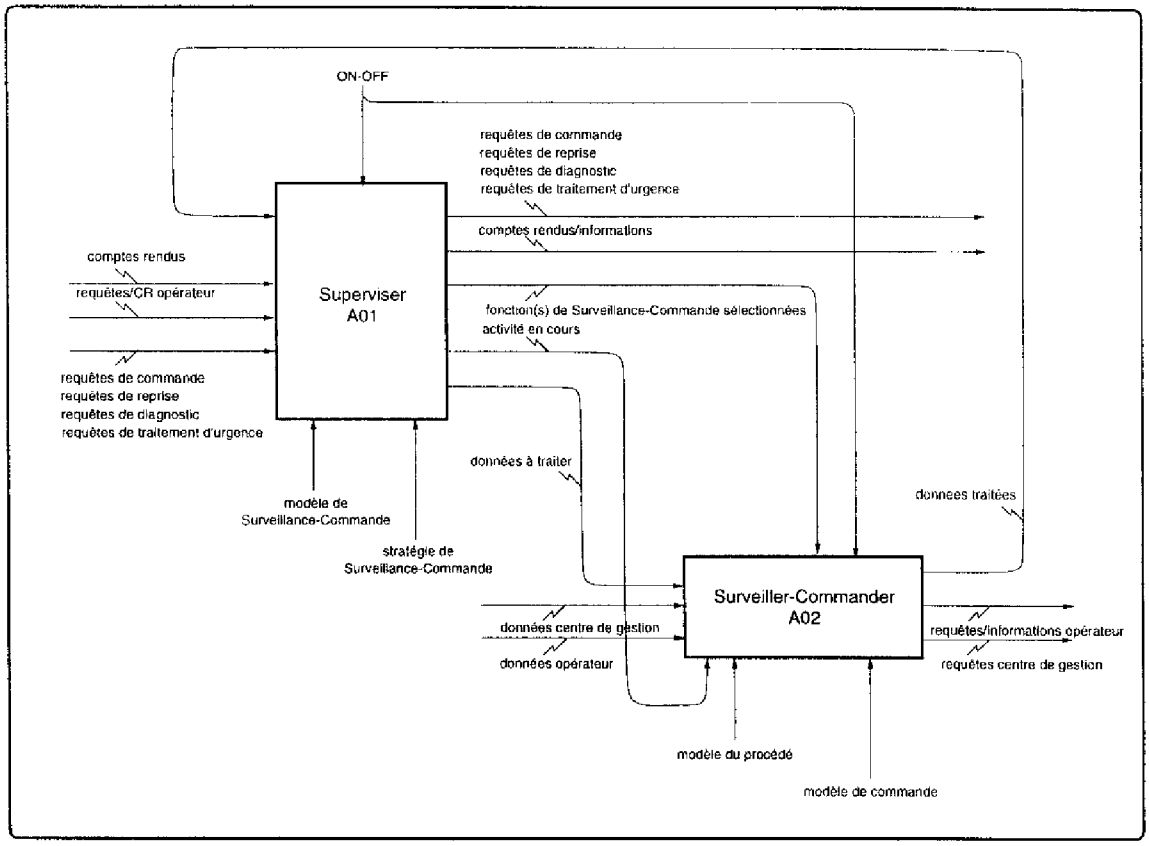


FIG. 2.3 – *diagramme A0*

seront sollicitées au gré des activités courantes du processus de surveillance-commande fixées par la stratégie de surveillance-commande.

Chacune de ces deux fonctionnalités présente un fort degré d'abstraction. Une décomposition plus détaillée s'avère donc nécessaire.

2.2.3.2 Interfaçage des fonctionnalités Superviser et Surveiller-Commander

Nous ne reprenons pas la liste des flots déjà décrits pour le niveau A-0. Les seules données internes sont :

- les **données à traiter** qui sont prises en compte par le superviseur et orientées vers les fonctions de surveillance-commande capable de les traiter,
- la **fonction(s) de Surveillance-Commande sélectionnée(s)** pour traiter ces données
- les **données traitées** par les fonctions de surveillance-commande sélectionnées.

2.2.4 Diagramme A01 : Superviser

2.2.4.1 Fonctionnalités à réaliser

Nous avons déjà indiqué que la fonctionnalité A01 est une abstraction des fonctionnalités *Acquérir (1), Orienter (2) et Représenter (3).* Ces trois fonctionnalités sont nécessaires et suffisantes pour réaliser la gestion des informations transitant par le nœud de surveillance-commande considéré.

2.2.4.2 Interfaçage des fonctionnalités de Superviser

- les **données reçues** correspondent à une des données d'entrée de la fonctionnalité "Acquérir",
- dès qu'une donnée d'entrée est prise en compte, la fonctionnalité "Acquérir" envoie la donnée **choix destination** de manière à activer la fonctionnalité "Orienter" qui se chargera d'orienter les "données reçues" vers la fonctionnalité correspondante.
- l'**activité de surveillance-commande en cours d'exécution** est nécessaire à la fonctionnalité "Orienter". En effet, si nous nous référons au §1.3.2 page 43 un processus est constitué d'une suite logique d'activités. Chacune de ces activités est composée d'une association d'éléments, en particulier les fonctions de surveillance-commande (commande, diagnostic, détection, etc.).

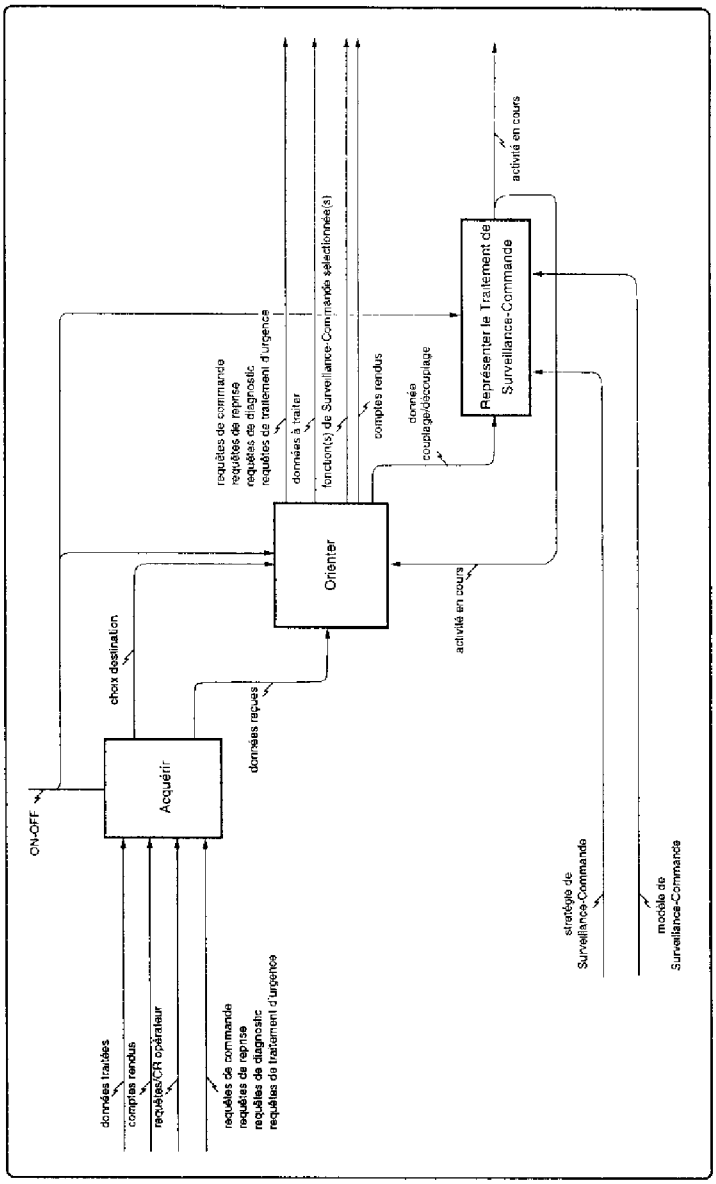


FIG. 2.4 – *diagramme A01*

- la seule donnée d'entrée requise par la fonctionnalité "Représenter le Traitement de Surveillance-Commande" est la **donnée couplage/découplage**. Cette donnée correspond aux événements évoqués précédemment. Ces événements provoquent des changements d'états ou plutôt des changements d'activités. Le processus ainsi décrit est celui modélisé dans la "stratégie de commande".

2.2.5 Diagramme A02 : Surveiller-Commander

2.2.5.1 Fonctionnalités à réaliser

La fonctionnalité A02 est une abstraction des fonctionnalités *Détecter (5), Commander (4), Diagnostiquer (6), Décider (7), Reprendre (8), Appliquer (11)* un Traitement d'urgence, *Suivre (10), Dialoguer (9)* avec l'opérateur et *Échanger (12)* des informations avec la base de données contenues dans le système d'information. Nous avons décidé ici de répartir les deux dernières fonctionnalités (Dialoguer et Échanger) dans les 7 autres fonctionnalités et ce, de la manière suivante :

- **A021 : Détecter** = 5.
- **A022 : Commander** = 4, 12. La fonctionnalité 12 est intégrée à *Commander (4)* de manière à autoriser la mise à jour des données contenues dans les modèles du procédé (modèle de référence et système d'information) en fonctionnement normal c'est à dire en l'absence de défaillance.

  Remarque : pour les mêmes raisons, toutes les autres fonctionnalités doivent être liées à 12.
- **A023 : Diagnostiquer** = 6, 9, 12. Les fonctionnalités 9 et 12 sont requises par la fonctionnalité "Diagnostiquer" pour lui permettre de faire appel à l'opérateur et d'interagir avec le système d'information (recherche de l'activité responsable de la défaillance, récupération de données pertinentes, mise à jour de la base de données, vérification d'une conclusion de diagnostic et enfin détermination du chemin dans la hiérarchie permettant d'accéder au nœud cible responsable de la défaillance [Chaillet, 1995]).
- **A024 : Décider** = 7, 9, 12.
- **A025 : Reprendre** = 8, 12.
- **A026 : Appliquer Traitement d'Urgence** = 11, 12.
- **A027 : Suivre** = 10, 12.

2.2.5.2 Interfaçage des fonctionnalités de Surveiller-Commander

- **détecter sélect.**. La fonctionnalité "Détecter" a été sélectionnée pour caractériser les "données à traiter",

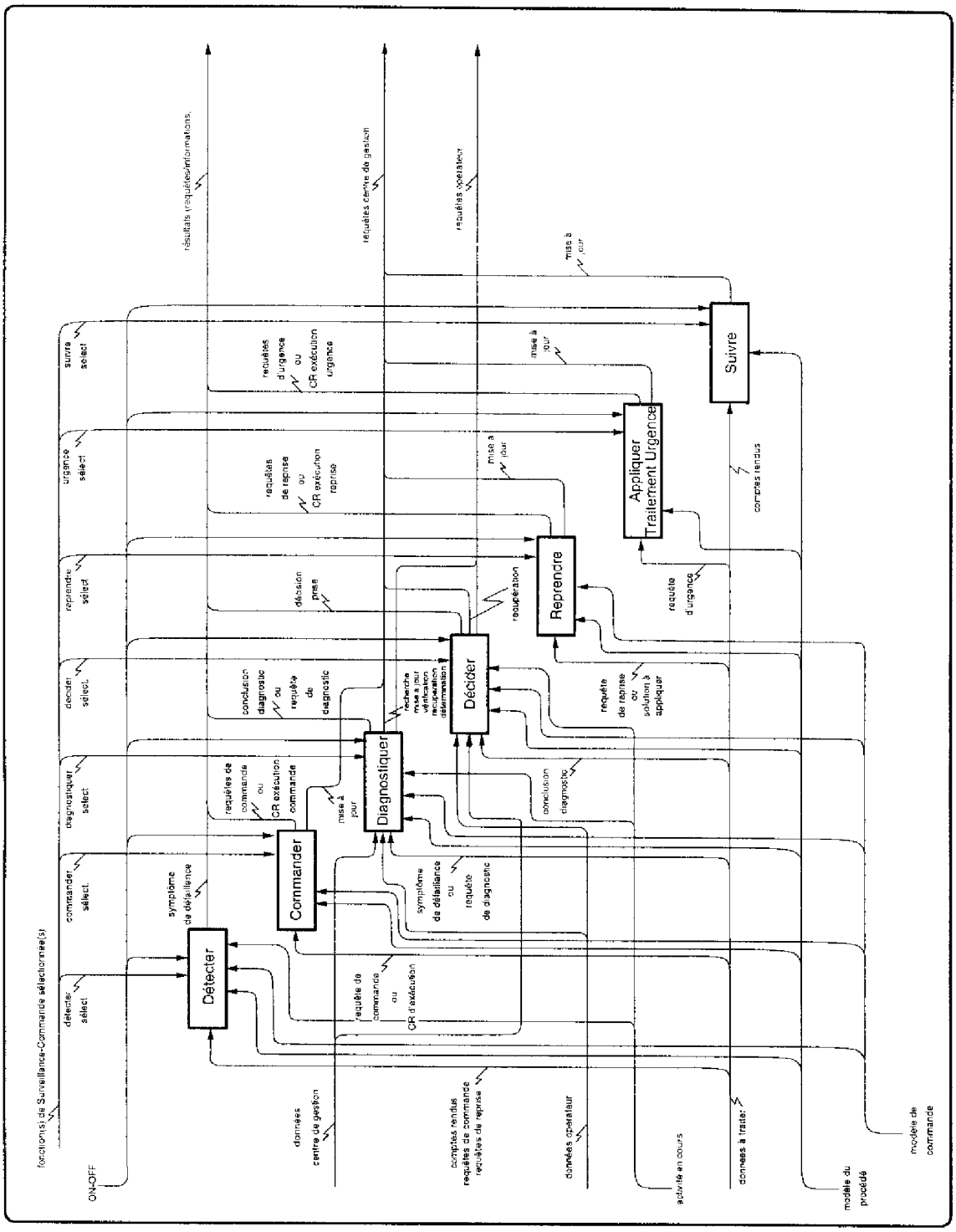


FIG. 2.5 – *diagramme A02*

- "Détecter" consiste à déceler toute évolution anormale par rapport à ce qui est prévu. Il est donc nécessaire de disposer d'une représentation de ce que l'on qualifie de "normal". Pour cette raison, la fonctionnalité "Détecter" disposera de deux modèles du fonctionnement normal et d'une représentation de l'état courant du système de surveillance-commande.

- **requêtes de diagnostic** : la fonctionnalité "Diagnostiquer" doit pouvoir assurer du diagnostic sur demande d'un niveau supérieur. Ce besoin a été mis en évidence dans les travaux de [Chaillet, 1995] pour préciser des résultats de diagnostic de haut niveau.

- **requêtes de reprise** correspond à l'affinement d'un service de reprise ou des **C.R. d'exécution de reprise** (fin normale, anormale),

2.3 Conclusion

Dans ce chapitre, une analyse volontairement limitée de la fonction "Gérer un nœud de surveillance-commande" a été présentée pour mettre en évidence les caractéristiques essentielles que doit intégrer un nœud de surveillance-commande. Cette phase de spécification a été supportée par méthode SADT. Nous avons dégagé les principales fonctionnalités caractérisant un nœud de surveillance-commande ainsi que les différentes données mises en jeu. La nécessité de disposer d'un "Superviseur" capable de gérer l'ensemble de ces données a été clairement montrée. Ce "Superviseur" a pour rôle principal de mettre en œuvre la stratégie de surveillance-commande fournie par l'utilisateur, ceci dans le but de produire.

Notre approche préconise la prise en compte de l'aspect surveillance, au même titre que la commande et ce, dès la phase de spécification. Ceci nous amène à présent à nous engager dans la dernière phase de notre étude, la conception d'un nœud de surveillance-commande.

Chapitre 3

Définition d'un nœud de surveillance-commande

3.1 Introduction

Le chapitre précédent nous a permis de dresser un cahier des charges des besoins requis par l'élaboration d'un système de surveillance-commande. Nous allons donc naturellement aborder ici la phase de conception de ce système. Cette conception s'attachera principalement à élaborer les différentes fonctionnalités que nous avons mises à jour sans toutefois aller jusqu'à concevoir les différentes fonctions de surveillance-commande. Cette étape [Chaillet, 1995], [Combacau, 1991], [Hammami *et al.*, 1996] sort en effet largement du cadre de notre travail.

La solution que nous proposons (cf. figure 3.1) consiste à décrire au sein de chaque nœud de surveillance-commande quatre blocs distincts : le bloc **modèle de référence pour la surveillance-commande** représentant tout ce qu'il est possible de faire ("ce que l'on peut faire") d'un point de vue surveillance et commande, le bloc **stratégie de surveillance-commande** imposant une utilisation particulière de certaines activités proposées dans le modèle de référence ("ce que l'on veut faire"), le bloc d'**acquisition et d'orientation** de l'information et enfin de l'**ensemble des fonctions de surveillance-commande** chargées de traiter les informations qui leur sont envoyées. L'ensemble formé par les trois premiers blocs forme le **superviseur.**

Ce chapitre va donc s'organiser de la façon suivante : premièrement nous allons élaborer le modèle de référence pour la surveillance-commande, deuxièmement nous proposerons un guide méthodologique de conception des stratégies de surveillance-commande (appelées également modèles de surveillance-commande), puis nous décrirons les principaux mécanismes qui animent le superviseur. Enfin, une description des interactions entre les blocs du module de surveillance-commande sera réalisée.

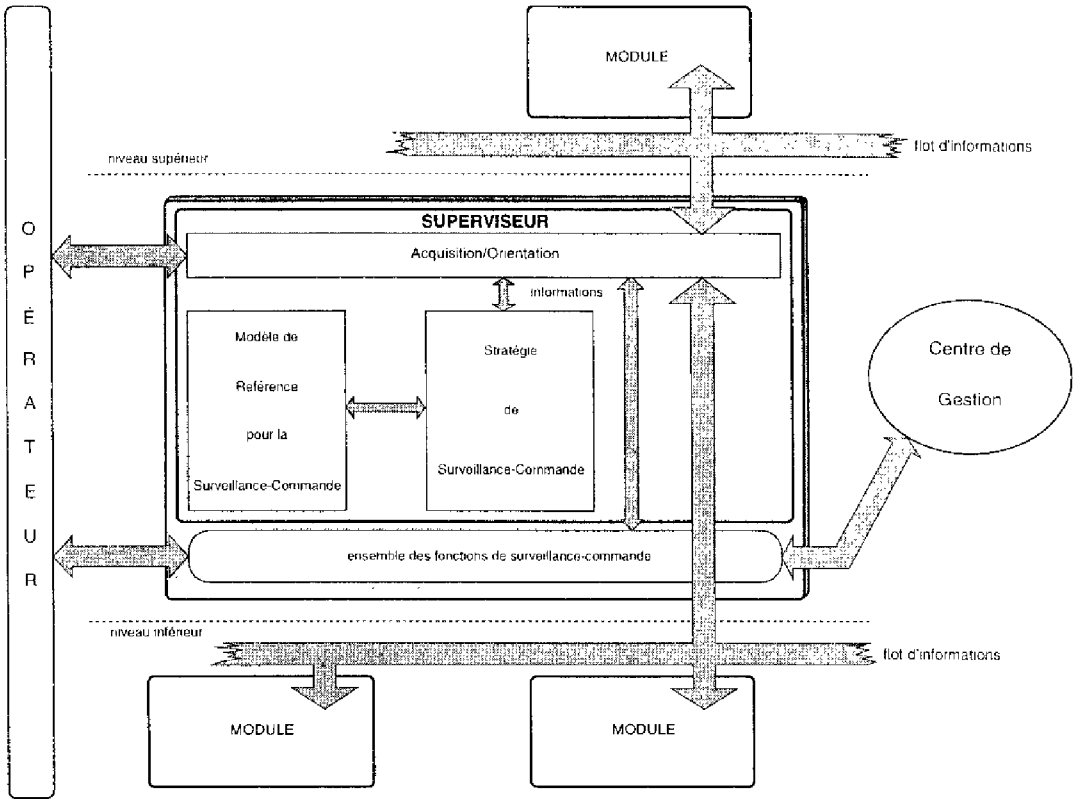


FIG. 3.1 – *structure d'un nœud (module) de surveillance-commande*

3.2 Élaboration du modèle de référence pour la surveillance-commande

Comme nous l'avons vu dans le chapitre 1 de cette partie, le modèle de référence que nous proposons pour la surveillance-commande représente la gestion des fonctions de surveillance-commande de la même manière que le modèle de référence pour la commande [Combacau, 1991] gère les ressources physiques du système commandé.

Ce modèle de référence pour la surveillance-commande contient un réseau de Petri à Objets modélisant l'ensemble des états et des évolutions utilisables de la surveillance-commande. Ces possibilités d'évolution sont les mêmes quel que soit le niveau considéré de la structure hiérarchique puisque les règles de surveillance-commande sont définies valides quel que soit ce niveau (à n'importe quel niveau de la structure de surveillance-commande, le système ne devra jamais exécuter en même temps et sur la même ressource physique une séquence de reprise et une séquence de commande). En revanche, ces possibilités d'évolution seront utilisées différemment selon le niveau considéré : c'est le cas lors de l'élaboration de la stratégie de surveillance-commande (cf. §3.3 page 77).

Ce modèle représente donc, sous forme d'activités (cf. §1.3.2 page 43), l'ensemble des comportements définis normaux vis-à-vis de la surveillance. Il exprime en cela les

contraintes liant ces activités. Les objets (jetons) qui circulent à l'intérieur de ce réseau représentent la dynamique de surveillance-commande, c'est à dire des instances de la structure de données que l'on manipule (associations des éléments tels que les ressources, les produits et les fonctions de surveillance-commande).

La conception de ce modèle de référence va suivre la démarche suivante : premièrement nous allons définir l'ensemble des activités qui doivent être intégrées au système de surveillance-commande, deuxièmement nous mettrons en évidence les événements qui provoquent les évolutions entre ces activités, troisièmement nous décrirons les processus de surveillance-commande et enfin nous présenterons le modèle de référence obtenu.

3.2.1 Activités de surveillance-commande

L'extension du concept d'activité aux besoins de la surveillance nous a conduit à intégrer les fonctions de surveillance aux associations définissant l'activité (cf. §1.3.2 page 43). Ces activités sont des n-uplets constitués d'une combinaison de i (i variant de 0 à 9) éléments pris parmi 9 : **ressources** (machines/outils, robots, zones de stockage, etc.), **produits** (matière première, composants, etc.), fonction **commande**, et toutes les autres fonctions de surveillance (**détection, diagnostic, décision, reprise, suivi et urgence**). Bien sûr, ces fonctions ne sont pas toujours actives (associées) en même temps et c'est pour cette raison que nous obtenons un grand nombre d'activités de surveillance-commande. En revanche, notons dès à présent que l'existence d'une activité est liée à l'existence de l'élément ressource (physique ou abstraite). Une précision doit être apportée sur l'activité 0 de la table de vérité de la figure 3.2 qui correspond à notre choix de représentation d'une ressource non utilisable, hors tension par exemple.

3.2.1.1 Liste exhaustive des activités de surveillance-commande

Le but de ce paragraphe est de construire l'ensemble des états (activités) observables de surveillance-commande tel que nous l'avons défini dans le paragraphe §1.3.2 page 43. Pour obtenir l'ensemble exhaustif de ces activités, une solution simple consiste à utiliser une table de vérité à 9 entrées, chacune de ces entrées étant un des 9 éléments préalablement cités. Une vue partielle de cette table (21 premières lignes) est montrée dans la figure 3.2. La table complète est constituée de 512 lignes (2^9) ou activités. Les 0 et les 1 contenus dans les cases de cette table déterminent si l'élément correspondant (en haut de la colonne) appartient (1) ou n'appartient pas (0) à l'activité. Une activité est ainsi déterminée à chaque ligne de cette table. Pour alléger l'écriture, les activités seront souvent référencées par leur numéro. Ces numéros correspondent à la conversion en décimal du nombre binaire constitué par les éléments de la ligne.

Toutes les activités mises en évidence par cette table de vérité ne sont bien évidemment pas valides. Seules quelques-unes le sont. Elles sont repérées d'une croix dans la colonne $\exists$? La justification de leur "existence" fait l'objet du paragraphe suivant.

n°	Urgence	Reprise	Décision	Diagnostic	Commande	Suivi	Détection	Produit	Ressource	∃?
0	0	0	0	0	0	0	0	0	0	X
1	0	0	0	0	0	0	0	0	1	.
2	0	0	0	0	0	0	0	1	0	.
3	0	0	0	0	0	0	0	1	1	.
4	0	0	0	0	0	0	1	0	0	.
5	0	0	0	0	0	0	1	0	1	.
6	0	0	0	0	0	0	1	1	0	.
7	0	0	0	0	0	0	1	1	1	.
8	0	0	0	0	0	1	0	0	0	.
9	0	0	0	0	0	1	0	0	1	.
10	0	0	0	0	0	1	0	1	0	.
11	0	0	0	0	0	1	0	1	1	.
12	0	0	0	0	0	1	1	0	0	.
13	0	0	0	0	0	1	1	0	1	X
14	0	0	0	0	0	1	1	1	0	.
15	0	0	0	0	0	1	1	1	1	X
16	0	0	0	0	1	0	0	0	0	.
17	0	0	0	0	1	0	0	0	1	.
18	0	0	0	0	1	0	0	1	0	.
19	0	0	0	0	1	0	0	1	1	.
20	0	0	0	0	1	0	1	0	0	.

FIG. 3.2 – *Extrait de la table de vérité*

3.2.1.2 Critères d'admissibilité

Nous nous proposons ici de réduire l'ensemble des états observables (les 512 activités) à l'ensemble des états dits utilisables défini §1.3.2 page 43, c'est à dire ceux exploitables et cohérents. Pour ce faire, nous avons défini un ensemble de critères permettant de rejeter les activités qui ne correspondent pas à des situations utiles pour la surveillance-commande (description de l'ensemble des états dits interdits). Ces critères définissent un sous-ensemble des contraintes de surveillance-commande.

1. $(ressource = 0)$: notre approche concerne la surveillance des ressources, par conséquent, l'absence de ressource entraîne l'absence d'activité de surveillance-commande. Toutefois, nous l'avons déjà signalé, nous faisons une exception pour l'activité numéro 0 qui matérialise l'état précédent la mise en route de la ressource : *ressource hors énergie.*

! **Tous les critères suivants sont appliqués à une activité constituée d'au moins une ressource.**

2. $(suivi = 0)$: cette fonction a pour rôle d'absorber toutes les informations émanant du niveau inférieur de manière à assurer la mise à jour des modèles du procédé (modèle de référence contenu dans le nœud de surveillance-commande et la base de données distante contenue dans le système d'information). Il est donc naturel que cette fonction soit active quelle que soit l'activité en cours. Pour cette raison, nous supprimons toutes les activités ne contenant pas cette fonction.

3. $((detection = 0) \wedge (urgence = 0)) \vee ((detection = 1) \wedge (urgence = 1))$: la *détection* et l'*urgence* sont deux fonctions en mutuelle exclusion. Si une procédure d'urgence est lancée, les comptes rendus émis par le sous-système contrôlé risquent de déclencher une série d'alarmes en cascade. En effet, si une séquence de commande est en cours d'exécution, les comptes rendus envoyés en situation d'urgence vont probablement différer de ceux attendus par la commande. Or ces différences sont exploitées par les mécanismes de détection. Cette fonction ne doit donc jamais être active pendant les traitements d'urgence.

   En revanche, lorsqu'une consigne est envoyée vers le procédé par la commande, un compte rendu d'exécution est attendu. Par ce biais, tout compte rendu différent de celui attendu témoigne d'une déviation de comportement. Par extension de ce mécanisme, dans le cas particulier où aucune consigne n'est envoyée au procédé, aucun compte rendu n'a à remonter du procédé. La détection doit donc être active chaque fois que l'urgence ne l'est pas.

4. $(decision = 1) \wedge (diagnostic = 1)$: si une phase de diagnostic est lancée, la décision concernant la ressource physique incriminée ne pourra être prise que si le diagnostic lui fournit sa conclusion. Pour cette raison, la situation exprimée par cette activité est impossible, nous la rejetons.

5. $(reprise = 1) \wedge (commande = 1)$: il est contradictoire d'effectuer une reprise alors qu'une séquence de commande est en cours d'exécution et vice-versa. En effet, hors situation d'urgence, un élément du procédé ne doit pas être soumis à deux consignes conflictuelles. Par exemple, il n'est pas envisageable d'autoriser la consigne de commande "aller à droite" à un chariot si en même temps la reprise lui impose d'"aller à gauche". Nous rejetons donc toutes les activités présentant ce paradoxe.

6. $(diagnostic = 0) \wedge (décision{=}0) \wedge (urgence = 1)$: une procédure d'urgence peut être lancée de deux manières différentes. Soit après détection d'une défaillance remettant en cause certaines contraintes structurelles préalablement définies par l'utilisateur, soit sur une requête d'urgence émanant du niveau supérieur. Dans le premier cas, il est souhaitable de lancer en parallèle un diagnostic puis une décision de manière à optimiser le traitement de la défaillance: cela permet, lorsque la procédure d'urgence est achevée, d'appliquer la séquence de reprise envisagée par la décision. Dans le deuxième cas, le raisonnement est similaire, toutefois, seule la décision doit être lancée en parallèle à l'urgence car la défaillance n'a pas été détectée dans ce nœud. Ce critère élimine donc toutes les activités dans lesquelles une procédure d'urgence est en cours d'exécution sans qu'un diagnostic ou qu'une décision ne soient également lancés.

En appliquant ces critères à l'ensemble des 512 associations possibles, nous obtenons **31 associations utilisables** durant un cycle de production. Ces associations sont présentées dans le tableau référencé figure 3.3 et interprétées dans le paragraphe suivant.

n° activité	Urgence	Reprise	Decision	Diagnostic	Commande	Suivi	Détection	Produit	Ressource	∃°
0	0	0	0	0	0	0	0	0	0	X
13	0	0	0	0	0	1	1	0	1	X
15	0	0	0	0	0	1	1	1	1	X
29	0	0	0	0	1	1	1	0	1	X
31	0	0	0	0	1	1	1	1	1	X
45	0	0	0	1	0	1	1	0	1	X
47	0	0	0	1	0	1	1	1	1	X
61	0	0	0	1	1	1	1	0	1	X
63	0	0	0	1	1	1	1	1	1	X
77	0	0	1	0	0	1	1	0	1	X
79	0	0	1	0	0	1	1	1	1	X
93	0	0	1	0	1	1	1	0	1	X
95	0	0	1	0	1	1	1	1	1	X
141	0	1	0	0	0	1	1	0	1	X
143	0	1	0	0	0	1	1	1	1	X
173	0	1	0	1	0	1	1	0	1	X
175	0	1	0	1	0	1	1	1	1	X
205	0	1	1	0	0	1	1	0	1	X
207	0	1	1	0	0	1	1	1	1	X
297	1	0	0	1	0	1	0	0	1	X
299	1	0	0	1	0	1	0	1	1	X
313	1	0	0	1	1	1	0	0	1	X
315	1	0	0	1	1	1	0	1	1	X
329	1	0	1	0	0	1	0	0	1	X
331	1	0	1	0	0	1	0	1	1	X
345	1	0	1	0	1	1	0	0	1	X
347	1	0	1	0	1	1	0	1	1	X
425	1	1	0	1	0	1	0	0	1	X
427	1	1	0	1	0	1	0	1	1	X
457	1	1	1	0	0	1	0	0	1	X
459	1	1	1	0	0	1	0	1	1	X
										31 activités

FIG. 3.3 – *ensemble des activités utilisables*

3.2.1.3 Activités utilisables

Nous présentons dans ce paragraphe une interprétation des 31 activités reconnues utilisables en utilisant les abréviations suivantes :

Ressource $= R$,

Produit $= P$,

Détection $= Dt$,

Suivi $= Sv$,

Commande $= Cd$,

Diagnostic $= Dg$,

Décision $= Dc$,

Reprise $= Rp$,

Urgence $= Ug$.

- **0** $= <>$: le système de surveillance-commande est totalement inactif puisque qu'il n'a aucune ressource à contrôler. Cette activité, correspond à une situation dans laquelle la partie commande est hors énergie [GEM, 1981] [Bois, 1991].

- **13** $= < R, Dt, Sv >$: nous avons qualifié cette activité "activité minimale de surveillance-commande". Cette terminologie véhicule bien le sens de cette activité. La situation exprimée par cette activité correspond en effet à la non exploitation des services offerts par la ressource considérée. Toutefois, ce n'est pas parce que cette ressource physique n'est pas exploitée qu'aucune évolution n'est observable. En effet, si un opérateur est amené à intervenir sur cette ressource, cette dernière risque d'évoluer entraînant le déclenchement intempestif des capteurs qui lui sont associés. Le même cas de figure peut se présenter suite à une collision par exemple entre deux chariots ou deux robots, l'un commandé, l'autre non. Ainsi, le système de surveillance-commande doit être capable de détecter d'éventuelles évolutions et de représenter au mieux l'occurrence de ces événements grâce à la fonction suivi.

- **15** $= < R, P, Dt, Sv >$: cette situation traduit un état hors phase de production puisque la fonction commande n'y apparaît pas. Il s'agit donc d'une phase intermédiaire pour laquelle la ressource est allouée au produit sans qu'il n'y ait de transformation en cours. La ressource joue donc le rôle d'un stock intermédiaire.

- **29** $= < R, Dt, Sv, Cd >$: cette activité représente la commande d'une ressource physique sans qu'un produit ne soit transformé. Cela peut donc caractériser deux situations différentes :

  1. soit la commande d'une ressource nécessitant une préparation préalable à la production : mise en position d'un robot, ouverture d'un étau, mise en rotation d'un mandrin, etc. (à ne pas confondre avec la "marche de préparation" définie dans le GEMMA [GEM, 1981] qui concerne le caractère autonome de la ressource et non la production au sens commande),
  2. soit à une activité de maintenance (test, vérification, etc.) hors production (produit=0) de la ressource concernée.

- **31** $= < R, P, Dt, Sv, Cd >$: cette activité représente l'état de production optimale d'un produit.

- **45** $= < R, Dt, Sv, Dg >$: ceci correspond à l'une des deux situations suivantes :

  1. activité de diagnostic de symptôme de défaillance détectée (détection) alors qu'aucune fonction de commande n'était associée. Nous sommes donc ici en présence d'un diagnostic d'une évolution de la ressource alors qu'aucun capteur ne devait être déclenché,
  2. activité de diagnostic détaillé lancé par les niveaux supérieurs.

- **47** $= < R, P, Dt, Sv, Dg >$: il s'agit ici d'une situation classique de diagnostic lancé à la suite de la détection d'une défaillance durant une activité de commande ou de reprise. Cette phase de diagnostic s'accompagne de l'arrêt de la production,

la fonction commande (reprise) n'est donc plus active. Notons que dans cette situation, la détection d'un comportement anormal ne s'opère plus par rapport aux consignes envoyées par la commande (reprise) mais au contraire par rapport au non envoi de consignes car le sous-système associé à la ressource ne doit pas évoluer).

- **61** $= < R, Dt, Sv, Cd, Dg >$:

  1. cette activité présente une situation nouvelle dans la surveillance des systèmes de production. En effet, un diagnostic est lancé alors qu'une commande est en cours et reste active. Dans le cas où produire tout de même [GEM, 1981] [Kermad, 1996] [Deplanche *et al.*, 1995] est essentiel, la détection d'une défaillance ne doit pas forcément entraîner l'arrêt de la commande. La coexistence des fonctions de diagnostic et commande dans cette activité traduit l'existence d'une phase de diagnostic sur la ressource qui était à l'origine en cours de préparation ou de maintenance.

     Par ailleurs, cette activité peut se révéler fort utile lorsque le symptôme détecté n'est pas suffisamment significatif pour permettre au diagnostic d'identifier la défaillance. Dans ce cas, le diagnostic reste actif, en attente d'autres événements lui permettant alors d'atteindre cette conclusion.

  2. lorsqu'un diagnostic détaillé sur une ressource est demandé par le niveau supérieur, cette ressource peut être déjà associée dans une activité autre que l'activité minimale $< R, Dt, Sv >$. C'est ici le cas. L'activité de commande continue donc à être exécutée alors qu'un diagnostic est lancé pour déterminer les causes d'une défaillance détectée dans un autre module.

- **63** $= < R, P, Dt, Sv, Cd, Dg >$: cette activité présente la même caractéristique que la précédente, à savoir diagnostic et commande simultanés. La seule différence réside dans l'association d'un produit qui traduit l'exécution d'une phase de diagnostic pendant la production, sur une ressource en cours de fonctionnement.

- **77** $= < R, Dt, Sv, Dc >$: activité de prise de décision pour résoudre un problème posé par l'occurrence d'une défaillance en cours d'activité de préparation ou de maintenance, soit au cours d'une activité minimale de surveillance-commande (le produit est absent).

- **79** $= < R, P, Dt, Sv, Dc >$: cette activité est similaire à la précédente. Elle concerne une prise de décision après occurrence d'une défaillance en phase de production (présence du produit).

- **93** $= < R, Dt, Sv, Cd, Dc >$: activité de décision dans un contexte de phase de préparation ou de maintenance qui n'a pas été interrompue malgré l'occurrence d'une défaillance. La décision devra concerner la commande en cours toujours active et une éventuelle reprise (continuer à produire, arrêter puis reprendre, etc.). Nous remarquerons que la fonction détection est toujours active. Au cours de cette activité, la détection est faite par rapport aux consignes de commande envoyées vers le procédé.

- **95** $= < R, P, Dt, Sv, Cd, Dc >$: cette activité est la même que la précédente; cependant elle fait intervenir le produit. Il s'agit donc d'une situation similaire, mais cette fois pendant la production.

- **141** $= < R, Dt, Sv, Rp >$: activité de reprise en l'absence de produit. La fonction détection est ici active par rapport à la reprise, donc toute détection se fera sur déviation du comportement attendu par rapport aux consignes de reprise envoyées vers le sous-système contrôlé. Notons enfin que la séquence de reprise ne concernera pas le produit.

- **143** $= < R, P, Dt, Sv, Rp >$: il s'agit de l'activité de reprise décrite précédemment. Elle doit ici s'exécuter en tenant compte de la présence du produit. Cette activité peut donc également être utile lorsqu'une reprise en production est envisagée, c'est à dire reprendre et produire jusqu'à ce que l'efficacité de la solution appliquée soit démontrée.

- **173** $= < R, Dt, Sv, Dg, Rp >$:

  1. notre modèle ne pose pas de limites quant à la répétition des traitements de défaillance. Nous laissons ce soin à l'utilisateur lors de la conception de la stratégie de commande. Pour cette raison, le diagnostic de défaillance pendant l'exécution d'une activité de reprise est représenté ici. C'est une activité dont la seule différence avec $< R, Dt, Sv, Dg, Ct >$ concerne le pilotage de la ressource qui est effectué par la fonction reprise et non par la commande.
  2. durant une activité de reprise similaire à $< R, Dt, Sv, Rp >$, un diagnostic détaillé peut être demandé par le niveau supérieur.

- **175** $= < R, P, Dt, Sv, Dg, Rp >$: activité similaire à la précédente, mais au cours d'une phase de production concernant un produit.

- **205** $= < R, Dt, Sv, Dc, Rp >$: cette activité représente une prise de décision après diagnostic de défaillance tolérée pendant une activité de reprise. C'est pour cette raison que la reprise est encore active. La décision engagée devra déterminer s'il faut continuer à exécuter cette activité de reprise ou bien envisager une reprise d'une reprise ou encore abandonner définitivement toute activité de reprise et confier la résolution du problème posé à l'opérateur. Nous insistons ici sur le désir de représenter dans le modèle de référence pour la surveillance-commande toutes les activités utilisables par l'utilisateur. Donc, même si cette activité peut sembler abusive (lancer une reprise d'une reprise), seul l'utilisateur en sera juge par rapport à ses besoins qu'il exprimera dans sa stratégie de surveillance-commande. Il pourra par exemple autoriser trois reprises de reprise consécutives et lancer une procédure d'urgence s'il y a un quatrième échec (**141**, 173, 205, **141**, 173, 205, **141**, 425 ou 457).

- **207** $= < R, P, Dt, Sv, Dc, Rp >$: c'est le même cas que le précédent, mais le produit intervient ici.

- **297** $= < R, Sv, Dg, Ug >$:

  1. activité d'urgence durant laquelle un diagnostic est lancé de manière à optimiser le traitement de défaillance. Il est en effet inutile d'attendre la fin de l'action d'urgence pour envisager des solutions correctives.
  2. un diagnostic détaillé peut être demandé par le niveau supérieur au cours de cette activité. Le diagnostic étant déjà actif, il ne fait que prendre en compte cette nouvelle demande.

- **299** $= < R, P, Sv, Dg, Ug >$:

  1. cette activité traduit l'apparition d'une situation d'urgence pendant une activité de diagnostic,
  2. un diagnostic détaillé peut être demandé par le niveau supérieur sans que cela change la représentation de cette activité.

- **313** $= < R, Sv, Cd, Dg, Ug >$:

  1. il s'agit ici d'un cas classique de diagnostic d'une défaillance jugée critique [de Bonneval, 1993] pendant l'exécution d'une activité de commande liée à la maintenance ou à la préparation,
  2. un diagnostic détaillé peut être demandé par le niveau supérieur.

- **315** $= < R, P, Sv, Cd, Dg, Ug >$: cette activité présente les mêmes caractéristiques que la précédente. La présence du produit indique une phase de production en cours.

- **329** $= < R, Sv, Dc, Ug >$: cette activité correspond à une activité d'urgence exécutée sur l'activité de base, c'est à dire une ressource qui n'est pas commandée. L'action d'urgence s'accompagne d'une prise de décision pour élaborer une séquence de reprise à exécuter à la fin du traitement d'urgence.

- **331** $= < R, P, Sv, Dc, Ug >$: cette activité représente le même type de comportement que celui exposé précédemment. La seule différence concerne le produit qui indique une phase de production interrompue par le déclenchement de l'urgence.

- **345** $= < R, Sv, Cd, Dc, Ug >$: il s'agit encore d'une activité d'urgence exécutée au cours d'une activité de commande/décision comme $< R, Dt, Sv, Cd, Dc >$. L'absence du produit indique une phase de préparation ou de maintenance.

- **347** $= < R, P, Sv, Cd, Dc, Ug >$: même remarque pour cette activité que pour la précédente. Seule la présence du produit les différencie et indique un contexte de production.

- **425** $= < R, Sv, Dg, Rp, Ug >$: cette activité traduit par exemple la possibilité d'exécuter une procédure d'urgence durant un diagnostic sur une défaillance détectée pendant une reprise.

- **427** $= < R, P, Sv, Dg, Rp, Ug >$: la seule différence qui existe entre cette activité et la précédente réside dans l'association de l'élément produit.
- **457** $= < R, Sv, Dc, Rp, Ug >$: il s'agit de l'avant dernière activité d'urgence mise à jour dans notre approche. Elle peut traduire par exemple la détection d'une défaillance critique au cours d'une prise de décision concernant une reprise en cours.
- **459** $= < R, P, Sv, Dc, Rp, Ug >$: nous ferons ici les mêmes commentaires que pour l'activité précédente. Nous rajouterons simplement que le produit intervient.

Force est de constater qu'après cette étude exhaustive (selon les critères que nous avons définis plus haut) nous avons mis à jour un ensemble d'activités qui jusqu'à présent n'avait jamais été considéré et étendu celui existant déjà. En effet, si nous considérons l'approche LAAS et faisons abstraction de la fonction suivi et des fonctions décision et reprise (regroupées sous le terme de «reprise» dans l'approche LAAS), seules **8** de ces activités étaient utilisables. Il s'agissait des activités :

- $31 =< R, P, Dt, Sv, Cd >$ (activité de commande),
- $47 =< R, P, Dt, Sv, Dg >$ (activité de diagnostic),
- $79 =< R, P, Dt, Sv, Dc >$ (activité de décision),
- $143 =< R, P, Dt, Sv, Rp >$ (activité de reprise),
- $315 =< R, P, Sv, Cd, Dg, Ug >$ (activité d'urgence durant la commande),
- $347 =< R, P, Sv, Cd, Dc, Ug >$ (activité d'urgence durant la commande),
- $427 =< R, P, Sv, Dg, Rp, Ug >$ (activité d'urgence durant la reprise),
- $459 =< R, P, Sv, Dc, Rp, Ug >$ (activité d'urgence durant la reprise).

Au travers des résultats précédents, il apparaît que la surveillance met en jeu des activités bien plus complexes au sein desquelles plusieurs fonctions de surveillance sont actives simultanément. Notre approche de la surveillance s'appuie sur les 31 activités que nous venons de décrire.

Pour satisfaire aux besoins de la surveillance, ces activités doivent être exécutées, enchaînées dans le bon ordre en fonction de la situation de défaillance. Tout au long de cette énumération d'activités de surveillance-commande, nous avons été amenés à prendre en considération les activités préalablement exécutées pour définir le sens et la spécificité de celles considérées. Nous avons donc déjà un peu introduit l'aspect enchaînement d'activités de surveillance-commande qui décrit en fait les processus de surveillance-commande.

3.2.2 Les processus

Les processus de surveillance-commande décrivent les différents enchaînements possibles entre toutes les activités de surveillance-commande énumérées dans le paragraphe précédent. Dans l'absolu, comme nous avons répertorié **31** activités, le nombre d'enchaînements élémentaires s'élève à **930** (nombre d'arrangements de 2 activités parmi 31: $A_2^{31} = 31 \times 30$). Heureusement, le contexte d'application va limiter ce nombre, car la plupart de ces enchaînements ne correspond pas à des traitements de défaillance. La transition $< R, Dt, Sv >$ vers $< R, Dt, Sv, Dc >$ est un exemple de ce qui ne doit être ni envisagé, ni toléré. Cette transition correspond en effet à l'engagement d'une décision sur une ressource au repos. L'activité intermédiaire de diagnostic doit être imposée.

De manière à éliminer les enchaînements élémentaires interdits, nous proposons un ensemble des critères qui caractérisent un processus élémentaire de surveillance-commande. Ces critères vont porter sur des relations étroites existant, compte tenu du contexte de la surveillance, entre les événements qui sont à l'origine des changements d'état du système physique.

Nous proposons dans le paragraphe suivant une démarche qui permet d'éliminer les transitions "indésirables".

3.2.3 Les changements d'états

Lorsque le système réel évolue, il émet vers le système de surveillance-commande un ensemble d'événements qui se traduisent par des changements d'états du système. Ces changements d'états conditionnent les résultats générés par les fonctions de surveillance-commande. Ces résultats provoquent alors des changements d'activités au niveau du modèle de surveillance-commande. Pour exprimer tous ces changements d'activités, et donc caractériser les processus de surveillance-commande, nous nous sommes inspirés du concept d'événements [Benzakour, 1985]. En ce sens, nous avons décrit tous les événements susceptibles de représenter une transition entre deux activités de surveillance-commande. Pour décrire ces événements, nous avons repris la définition d'une activité (association d'éléments de surveillance-commande). Le début d'une activité se traduit par l'ajout d'un ou plusieurs de ces éléments. La fin d'une activité se traduit, quant à elle, par la suppression d'un ou plusieurs éléments. Pour nos besoins, nous définissons l'événement comme étant une combinaison adéquate de couplages ou de découplages de fonctions de surveillance-commande. Par le biais de cette combinaison nous décrivons les critères qui caractérisent un processus élémentaire de surveillance-commande.

Remarque: insistons sur le fait que ces événements n'ont pas de réalité. Ils ne sont utilisés ici que pour nous aider à décrire de manière exhaustive les processus de surveillance-commande, c'est à dire ceux que l'on peut qualifier de "traitement de surveillance-commande".

3.2.3.1 Couplage

Il est souvent plus facile de décrire le normal (connu) que l'anormal. Pour cette raison, nous avons étudié et énuméré ici tous les événements que nous définissons normaux (en terme de couplage et de découplage d'éléments d'une activité de surveillance-commande) et qui sont donc susceptibles de provoquer une évolution décrivant un processus de surveillance-commande. Les événements qui ne sont pas normaux sont donc rejetés.

1. **événement lié au couplage d'une ressource**
   Le couplage d'une ressource est simultané au couplage du suivi et de la détection. En effet, pour qu'une ressource soit contrôlée (surveillée-commandée), les fonctions suivi et détection doivent y être associées, pour représenter l'état courant de la ressource et pour détecter les évolutions anormales (structurelles ou décisionnelles).

2. **événement lié au couplage d'un produit**
   Le couplage d'un produit peut être un événement singulier (détection de la présence d'un produit) ou bien lié au couplage de la fonction commande qui marque le début d'une activité de commande.

3. **événement lié au couplage de la détection**
   Le couplage de la détection est, soit associé au couplage de la ressource et du suivi (cas particulier du passage de l'activité <> à l'activité $<R,Dt,Sv>$), soit associé au découplage de l'urgence. Dans le premier cas, nous retrouvons l'événement 1. Le deuxième cas traduit l'exécution d'une procédure d'urgence. Un ensemble de comptes rendus anormaux va donc probablement remonter du procédé. Pour ne pas les détecter inutilement et déclencher une série d'alarmes en cascade, la détection est découplée de l'activité.

4. **événement lié au couplage du suivi**
   Le couplage du suivi est lié au couplage de la détection et de la ressource pour les raisons décrites en 1.

5. **événement lié au couplage de la commande**
   Le couplage de la commande peut constituer à lui seul un événement, mais il peut être également associé au couplage de la fonction diagnostic lorsqu'un test ou une vérification sont privilégiés, ou encore au produit (cf. événement 2).

6. **événement lié au couplage du diagnostic**
   Le couplage du diagnostic peut être un événement isolé lorsqu'un diagnostic détaillé est demandé par un des niveaux supérieurs. Il peut être également lié au découplage de la commande lorsque la fonction commande est bloquée. Enfin, associé au couplage de l'urgence, le diagnostic est couplé pour déterminer l'origine de la défaillance détectée alors qu'une procédure d'urgence est lancée.

7. **événement lié au couplage de la décision**
   Cet événement peut être constitué soit du couplage de la décision et du découplage du diagnostic, ce qui correspond à un début d'activité de prise de décision et la

fin d'une activité de diagnostic, soit aux couplages respectifs de la décision et de l'urgence. C'est le cas de l'exécution d'une requête d'urgence issue du niveau supérieur.

8. **événement lié au couplage de la reprise**
Le couplage de la reprise est soit un événement à part entière (la commande n'étant pas active), soit un événement de découplage de la commande, de découplage de la décision et de couplage de la reprise. C'est le cas lorsque la décision décide de lancer une séquence de reprise alors que la commande est toujours active.

9. **événement lié au couplage de l'urgence**
Le couplage de l'urgence sera associé à celui du diagnostic ou à celui de la décision. Dans les deux cas, il s'accompagnera obligatoirement du découplage de la détection pour éviter de détecter une avalanche de comptes rendus anormaux pour la commande.

3.2.3.2 Découplage

1. **événement lié au découplage d'une ressource**
Le découplage d'une ressource est forcément lié au découplage du suivi et de la détection. En effet, il est nécessaire d'avoir rejoint l'activité $< R, Dt, Sv >$ avant de découpler la ressource.

2. **événement lié au découplage d'un produit**
Le découplage d'un produit peut être un événement à part entière, ou un événement lié au découplage de la fonction commande.

3. **événement lié au découplage de la détection**
Le découplage de la détection est associé au découplage de la ressource (hors service) ou associé au couplage de l'urgence.

4. **événement lié au découplage du suivi**
Le découplage du suivi est toujours lié au découplage de la ressource, puisque le suivi doit être actif tout au long de la gestion de la ressource par le système de surveillance-commande.

5. **événement lié au découplage de la commande**
Le découplage de la commande peut être un événement indépendant. Associé au couplage de la fonction reprise, il indique le début d'une activité de reprise. Associé au découplage du produit, il traduit la fin de transformation du produit par cette activité de commande.

6. **événement lié au découplage du diagnostic**
Le découplage du diagnostic est toujours lié au couplage de la décision, même dans le cas d'une fin de diagnostic lancé par le niveau supérieur. En effet, si un diagnostic est lancé c'est bien pour prendre une décision pertinente selon l'origine de la défaillance.

7. **événement lié au découplage de la décision**
   Le découplage de la décision est lié au couplage de la reprise ou au couplage de l'urgence (détection d'une défaillance critique au cours d'une prise de décision). Toutefois, à la suite d'un échec de diagnostic détaillé, la décision doit être découplée seule.

8. **événement lié au découplage de la reprise**
   Le découplage de la reprise sera soit un événement singulier dans le cas d'une exécution d'une requête de reprise émanant du niveau supérieur, soit lié au couplage de la commande dans le cas où le point de reprise est atteint [de Bonneval, 1993], soit associé au couplage du diagnostic dans le cas d'un diagnostic de défaillance lié à la détection d'une défaillance, soit enfin associé au découplage du produit.

9. **événement lié au découplage de l'urgence**
   Le découplage de l'urgence s'accompagnera du couplage de la détection. Le découplage de l'urgence signifie que le procédé n'est plus soumis à des consignes anormales pour la commande. Dans ce cas, toute évolution anormale doit à nouveau être détectée.

3.2.3.3 Les processus élémentaires admissibles

Un programme mettant en œuvre ces critères en générant tous les événements de couplage/découplage nous a permis de dresser la liste des processus élémentaires admissibles. Cette liste figure en annexe sous forme de sorties brutes du programme. Il est cependant important de noter que sur les 930 processus élémentaires observables nous en avons retenu **165** comme étant admissibles et donc utilisables.

Nous ferons toutefois une remarque quant à l'enchaînement de l'activité 0 (hors service) vers l'activité 13 (activité minimale de surveillance-commande). Cet enchaînement recouvre implicitement plusieurs autres activités comme les *auto-tests, les pré-chauffages, mises en route diverses* [Biland, 1994] [GFM, 1981]. Ces traitements sont à détailler dans l'activité "hors service". Mais cette étude sortant du cadre de notre travail, nous avons volontairement omis de les représenter.

3.2.4 Le modèle de référence pour la surveillance-commande

3.2.4.1 Le modèle obtenu

Le modèle de référence pour la surveillance-commande représente les 31 activités de surveillance-commande et la façon de passer de l'une à l'autre. Le modèle de Petri à Objets étant relativement conséquent (39 places, 165 transitions et plus de 336 arcs) nous n'en avons représenté qu'un extrait (cf. figure 3.4).

Les principes que nous avons adoptés pour modéliser le système de surveillance-commande sont les suivants :

1. la structure de contrôle du réseau est générique,

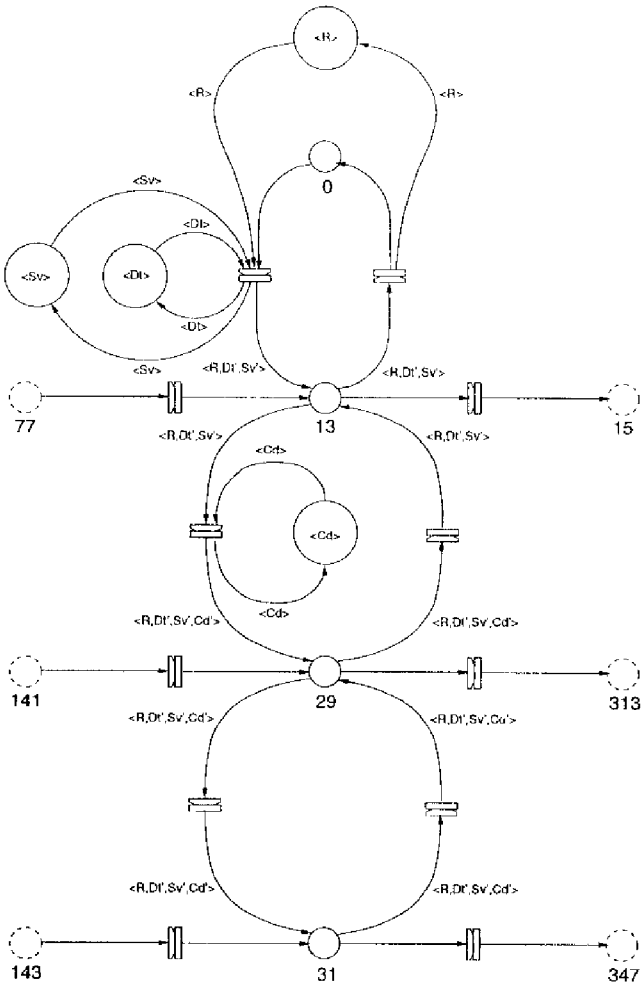


FIG. 3.4 – *extrait du modèle de référence pour la surveillance-commande*

2. le marquage initial du réseau comporte deux volets :
   - les fonctions de surveillance-commande (comme la détection, la commande, etc),
   - les ressources qui dépendent de l'atelier.
3. les activités du système de surveillance-commande sont modélisées par des jetons. Ces jetons sont des n-uplets d'instances d'objets qui caractérisent dynamiquement l'activité en cours d'exécution.
4. toutes les places du réseau (0, 13, 29, etc.) sont des places indiquant la classe d'activité de surveillance-commande en cours d'exécution. Comme nous n'avons pas modélisé de places d'attente dans notre réseau, les événements de fin d'activité sont liés aux événements de début de l'activité suivante.
5. les transitions de ce réseau de référence pour la surveillance-commande représentent les événements de début et de fin d'activité : toute transition "début d'activité" ou "fin d'activité" est encadrée en amont et en aval par des places d'activités.

Ce réseau de référence est en relation exclusive avec le modèle de la stratégie de surveillance-commande. Il va évoluer sur demande de cette stratégie selon les informations reçues par le superviseur ("données traitées", "comptes rendus", "requêtes" diverses, etc. cf. figure 2.3 du chapitre 2) puis traitées par les fonctions de surveillance-commande.

Avant d'aller plus loin, insistons sur le fait que ce modèle est un modèle générique. Seuls les jetons modélisant les ressources physiques et les produits dépendent de l'atelier considéré.

Quelle que soit l'entreprise, le modèle de référence pour la surveillance-commande pourra être intégré à chacun des nœuds de la structure de surveillance-commande. En revanche, le nombre de ces nœuds dépend bien sûr de l'atelier considéré.

3.3 Elaboration du modèle de la stratégie de surveillance-commande

Les entreprises étant différentes, les besoins qui en découlent en terme de surveillance diffèrent forcément. Ces besoins doivent être maintenant intégrés au modèle de la stratégie de surveillance-commande. Le rôle de ce modèle est d'imposer au nœud de surveillance-commande une ligne de conduite répondant au mieux à la politique de surveillance fixée par l'entreprise.

L'élaboration de ce modèle débute par le choix des activités de surveillance-commande utilisables. Ce choix est conditionné par au moins deux paramètres : la politique de surveillance appliquée par l'entreprise et les produits à transformer.

Les contraintes exprimées dans le modèle de référence de la surveillance-commande ne sont quant à elles pas à re-décrire dans le modèle de la stratégie. Cela implique que

lors de l'exécution de la stratégie de surveillance-commande le modèle de référence pour la surveillance-commande soit consulté pour vérifier les contraintes qui y sont décrites. Nous traiterons de cet échange ultérieurement dans le paragraphe §3.5.

Ensuite, la phase de construction de ce modèle de la stratégie de surveillance-commande consiste à décrire les enchaînements entre les activités sélectionnées. Cette étape se fait en respectant les séquences obligatoires ou autres exclusions mutuelles représentées dans le modèle de référence. D'un point de vue modélisation, décrire un enchaînement revient à fusionner les pré-conditions et les post-conditions des activités (cf. figure 3.5) modélisées par RdPO. La fin d'exécution d'une activité marque le début de la suivante.

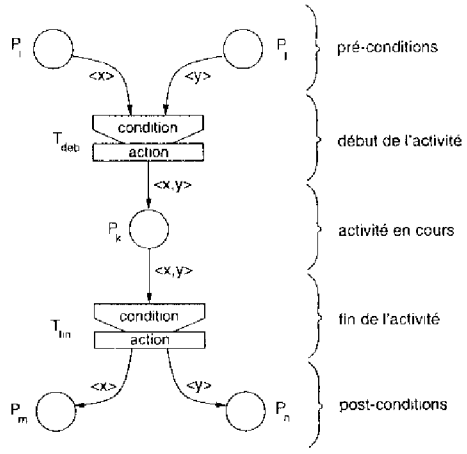


FIG. 3.5 – *modélisation par RdPO d'une activité*

Il est difficile de donner et de définir des règles précises pour élaborer ces stratégies de surveillance-commande. En revanche, nous nous proposons de donner, dans ce qui suit, un guide méthodologique de conception des séquences de surveillance-commande pour l'utilisateur. Ce guide va d'abord aborder le problème de l'intégration des besoins de l'entreprise, pour ensuite conseiller quant à l'intégration des contraintes imposées par la fabrication du produit et enfin montrer comment tenir compte des contraintes modélisées dans le modèle de référence pour la surveillance-commande.

Pour des raisons que nous développons dans le paragraphe §3.3.2 nous nous limitons dans notre approche à une seule politique de surveillance pour plusieurs produits transformés par la même ressource.

3.3.1 Intégration de la politique de l'entreprise

Intégrer la politique de l'entreprise à la stratégie de surveillance-commande, c'est prendre en compte les besoins en terme de surveillance de cette entreprise. Ceux-ci sont en général les suivants.

- Quel que soit le produit à réaliser, la politique de production consiste à produire tout de même en tolérant certaines défaillances. Dans ce cas, des traitements de

défaillances autorisant la prise en compte de ces défaillances (diagnostic, décision) tout en continuant à commander le procédé sont à privilégier. Ainsi, les activités qui possèdent à la fois les éléments commande et diagnostic ou commande et décision comme respectivement les activités $< R, Dt, Sv, Cd, Dg >$, $< R, P, Dt, Sv, Cd, Dg >$ (activités de diagnostic pour une production tout de même), $< R, Dt, Sv, Cd, Dc >$ ou $< R, P, Dt, Sv, Cd, Dc >$ (activités de décision dans un contexte de production tout de même) sont à sélectionner pour construire la stratégie de surveillance-commande.

- Quel que soit le produit à réaliser, la politique de production consiste à produire en limitant les pertes de produits au détriment des coûts (temps, etc.) de production. Dans ce cas, il s'agira de bloquer la commande dès la détection d'une défaillance avant de lancer un traitement de défaillance. Ceci peut contribuer à une perte minimale des produits. Ainsi, les activités ne faisant pas intervenir simultanément la fonction commande et les fonctions diagnostic ou décision sont à choisir. Il s'agit par exemple des activités $< R, Dt, Sv, Dg >$, $< R, P, Dt, Sv, Dg >$ (activités de diagnostic hors commande), $< R, Dt, Sv, Dc >$, $< R, P, Dt, Sv, Dc >$ (activité de décision hors commande).

- Quel que soit le produit à réaliser, la politique de production consiste à tolérer un nombre précis de défaillances. Au delà d'un seuil fixé, il est préférable de stopper la production (par rapport à la ressource considérée) car le nombre de produits perdus devient inacceptable. Dans ce cas, il s'agit de générer une stratégie de surveillance-commande intégrant les deux politiques précédemment décrites. En dessous du nombre de défaillances tolérées les activités suivantes seront privilégiées :

  - $< R, Dt, Sv, Cd, Dg >$,
  - $< R, P, Dt, Sv, Cd, Dg >$,
  - $< R, Dt, Sv, Cd, Dc >$,
  - $< R, P, Dt, Sv, Cd, Dc >$.

  En revanche, en dessus de ce seuil, d'autres activités seront requises :

  - $< R, Dt, Sv, Dg >$,
  - $< R, P, Dt, Sv, Dg >$,
  - $< R, Dt, Sv, Dc >$,
  - $< R, P, Dt, Sv, Dc >$.

  La stratégie de surveillance-commande sera donc définie en enchaînant d'abord plusieurs fois les premières activités puis les autres.

Si la politique de production dépend directement du ou des produits à transformer sans que l'on puisse en tirer de grandes lignes directrices comme dans les trois exemples cités ci-dessus, alors la stratégie de surveillance-commande sera définie en fonction de considérations techniques liées aux produits.

3.3.2 Intégration des besoins de transformation du produit

Nous l'avons vu plusieurs fois tout au long de ce mémoire, le cycle de production dépend essentiellement du produit à transformer. La transformation du produit impose un type de transformation particulier et une surveillance appropriée. En effet, la transformation d'un produit comporte des exigences conditionnant la surveillance. Ces exigences sont de l'ordre de la qualité de transformation, du coût des matières premières, du temps de transformation, etc.

Par exemple, si la qualité de fabrication d'un produit P_i est une contrainte importante, les traitements de surveillance pour entreprendre les reconfigurations adéquates doivent être les plus complets possibles (activités présentant successivement les fonctionnalités détection, diagnostic, décision et reprise) et bloquant d'un point de vue commande (les activités de traitement de défaillance pendant la production doivent donc être écartées). En revanche, si optimiser le temps de transformation du produit, dont la qualité finale n'est pas essentielle, est une contrainte majeure pour améliorer le coût de production, alors les traitements de surveillance devront ne pas bloquer la commande. Bien sûr si le problème mis à jour par la détection de la défaillance impose une réparation, le service délivré par la ressource considérée ne pouvant plus être assuré, l'arrêt de la production sera inévitable. Toutes ces contraintes vont donc amener le concepteur de la stratégie de surveillance-commande à sélectionner certaines activités de surveillance-commande ainsi que certains enchaînements.

Une limite à notre approche doit être signalée à ce stade de la présentation. Lorsque les fabrications de produits différents sont entrelacées sur la même ressource, il n'est pas envisagé, dans l'état actuel de nos travaux, de faire cohabiter simultanément des stratégies de surveillance différentes sur la même ressource. Nous évitons ainsi des incohérences du type de celle que nous proposons d'illustrer par l'exemple suivant.

Considérons une station de lavage et d'égouttage et deux produits P_1 et P_2 simultanément nettoyés dans la même station de lavage et respectivement associés à deux stratégies de surveillance-commande différentes, l'une mettant en avant la qualité de lavage et interdisant donc toute défaillance, l'autre, plus souple, tolérant un lavage médiocre suite à un manque d'eau par exemple. Si l'eau vient à manquer, la deuxième stratégie sera en conflit avec la première.

3.3.3 Respect des contraintes imposées dans le modèle de référence pour la surveillance-commande

Les contraintes qui sont exprimées dans le modèle de référence pour la surveillance-commande (ressources partagées, exclusions mutuelles, enchaînements obligatoires) n'ont pas à être à nouveau modélisées dans le modèle de la stratégie de surveillance-commande. En revanche, elles doivent être respectées lors de la conception de la stratégie de surveillance-commande. Par exemple, la stratégie de surveillance-commande ne devra et ne pourra pas imposer d'exécuter une activité de décision avant une activité de diagnostic (enchaînement obligatoire).

Comme dans l'approche à modèle de référence pour la commande, la modélisation de la stratégie de surveillance-commande est rendue plus simple grâce à l'existence du modèle de référence pour la surveillance-commande. Contrairement à la commande pour laquelle la génération automatique des séquences semble pouvoir être accessible, l'élaboration automatique de la stratégie de surveillance-commande ne semble pas être envisageable. Certaines contraintes qui y sont modélisées ne sont ni formelles (expérience du concepteur), ni génériques (entreprises différentes, donc stratégies de surveillance également).

Nous allons voir dans le paragraphe suivant les différents mécanismes qui permettent la mise en œuvre des stratégies de surveillance-commande.

3.4 Application de la surveillance-commande

Dans un nœud de surveillance-commande, plusieurs activités de surveillance-commande peuvent être exécutées en même temps. Si nous reprenons l'exemple donné dans le chapitre 2 de cette partie et représenté dans la figure 3.6 sous la forme d'un réseau de la stratégie de surveillance-commande, nous remarquons que dans le nœud considéré les activités de commande <*étau, pièce, Dt, Sv, Cd*>, de diagnostic/commande <*robot, outil, Dt, Sv, Cd, Dg,* > et <*centre usinage, Dt, Sv*> sont exécutées en même temps.

Plusieurs types d'informations sont à la fois émis, reçus et générés à l'intérieur du nœud par les différentes fonctions de surveillance-commande.

des requêtes de commande émises "montage outil" vers le niveau inférieur, des requêtes de diagnostic détaillé,

- des requêtes de commande reçues comme "perçage",
- des comptes rendus émis vers les niveaux supérieurs, comme "fin perçage" par exemple,
- des comptes rendus reçus pouvant concerner les exécutions des activités de commande "fin normale de montage outil", "fin anormale de montage outil", etc. et de diagnostic/commande, "fin de diagnostic détaillé", "fin normale de bridage pièce", "fin anormale de bridage pièce", etc.
- des "données traitées" par les fonctions de surveillance-commande comme par exemple "symptôme de défaillance non significative" ou "symptôme de fin anormale de montage outil" générées par la fonction détection, "fin de diagnostic interne" générée par la fonction diagnostic, etc.

La prise en compte et la gestion de ces informations sont assurées par les trois blocs intégrés à chacun des nœuds de notre structure hiérarchique : *acquérir*, *orienter* et *représenter l'exécution du traitement de surveillance-commande.*

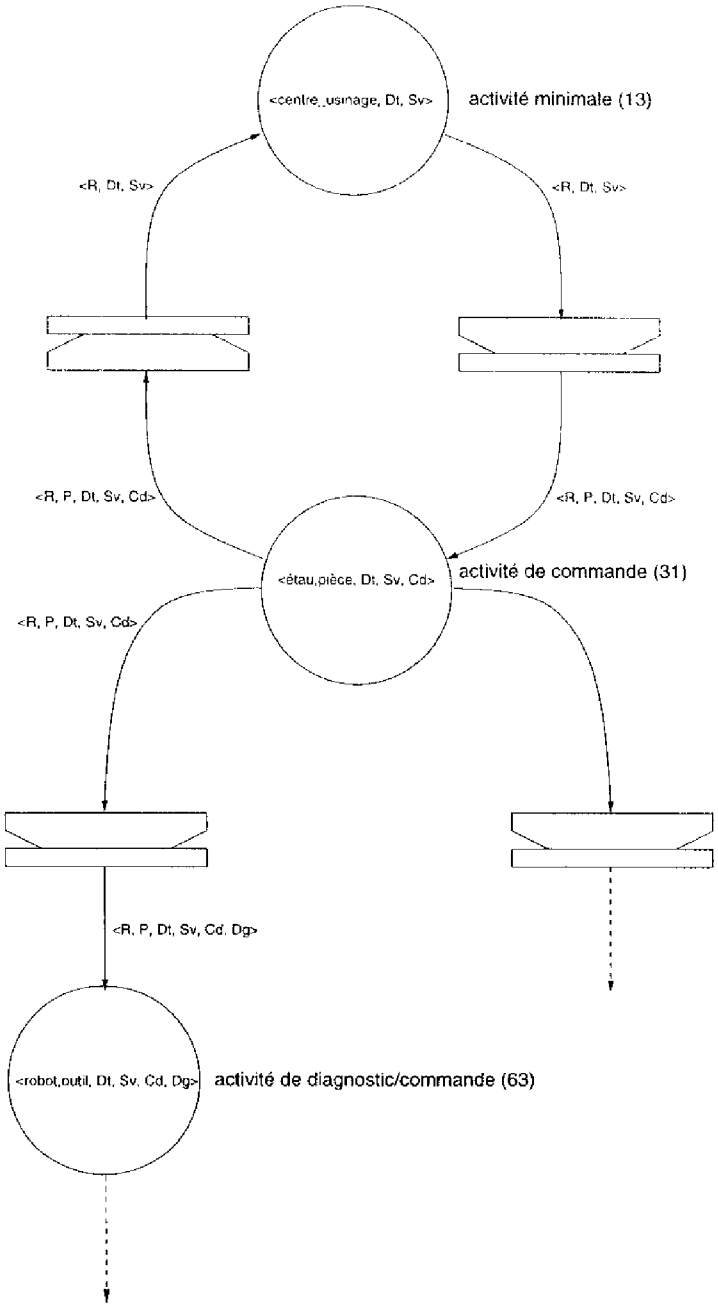


FIG. 3.6 – *exemple de réseau de surveillance-commande*

3.4.1 Acquérir les informations

Le superviseur que nous avons intégré à chacun des nœuds de surveillance-commande doit prendre en compte toutes les informations qui transitent par le nœud. Pour cela, nous lui avons attribué un rôle d'acquisition de l'information. L'élaboration du mécanisme d'acquisition étant plutôt du domaine de la mise en œuvre (par scrutation, par interruption, etc.), nous ne l'aborderons pas ici. Toutefois, nous supposerons dans notre approche que la fréquence d'occurrence d'informations est suffisamment basse pour que le superviseur soit en mesure d'acquérir les informations successivement sans devoir établir des ordres de priorité. Une fois l'information acquise, un traitement d'orientation de l'information est lancé. Les différentes étapes de ce traitement sont décrites dans le paragraphe suivant.

3.4.2 Orientation de l'information

Lorsqu'une information transite dans l'architecture de surveillance-commande, elle véhicule un ensemble de données. Chacune de ces données est pertinente ou non selon la fonctionnalité considérée dans le nœud de surveillance-commande. Par exemple, la donnée "compte rendu de fin normale d'exécution du bridage d'une pièce" est destinée plutôt à la détection (compte rendu normal ou anormal?) et à la commande (la fin d'exécution du bridage de la pièce entraînant par exemple le lancement d'un usinage), alors que la donnée "nom de la ressource concernée par l'information" est pertinente pour le superviseur. Nous en verrons la raison plus loin. Comme chacune des fonctionnalités du nœud de surveillance-commande manipule des données ou parties spécifiques de l'information, nous nous proposons d'utiliser le mécanisme d'encapsulation de l'information similaire à celui utilisé dans l'architecture OSI de l'ISO [Lepage *et al.*, 1991] pour la structurer.

Pour les besoins de notre approche, l'information sera structurée en deux niveaux d'abstraction (cf. figure 3.7). Dans le premier niveau, toutes les données permettant au superviseur d'orienter l'information seront décrites. Dans le deuxième niveau, seront implantées toutes les données ayant trait aux différentes fonctions de surveillance-commande. Toutefois, d'autres niveaux d'abstraction peuvent être ajoutés si besoin est. Leur implantation est possible sans modification des principes décrits. Il s'agira simplement de créer de nouveaux niveaux d'encapsulation dans l'information pour les besoins requis.

La répartition de ces données est faite selon les informations qui circulent dans notre structure :

1. celles associées au flot descendant (requêtes issues du niveau supérieur),
2. celles associées au flot horizontal (opérateur),
3. celles associées au flot ascendant (les comptes rendus),
4. celles constituant les informations internes.

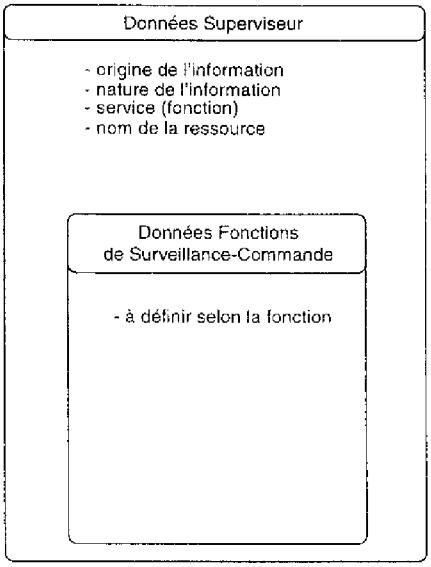


FIG. 3.7 – *structure d'une information : encapsulation des données*

Les informations à traiter émises par le superviseur vers les fonctions de surveillance-commande ne concernent bien évidemment que ces fonctions (deuxième niveau d'abstraction).

Dans le premier cas, les informations issues du niveau supérieur (requêtes) seront constituées des données suivantes :

- **données superviseur :**
  1. *origine* : niveau supérieur.
  2. *service* demandé par le niveau supérieur. Il peut donc s'agir d'une requête de *commande*, d'une requête de *diagnostic*, d'une requête de *reprise* ou enfin, d'une requête d'*urgence*. Cette donnée permet au superviseur de connaître la fonction de commande ou de surveillance capable de désagréger la requête. La nature de l'information n'est pas nécessaire. Il s'agit forcément d'une requête.
- **données fonctions** : elles sont à définir par rapport aux besoins de ces fonctions (hors du cadre de notre travail). Par exemple, pour une requête de commande, ces données seront [Combacau, 1991] :
  1. **nom du service demandé** : "chariotage" par exemple,
  2. **date de fin au plus tôt**,
  3. **date de fin au plus tard**.

Dans le deuxième cas, les informations issues de l'opérateur, véhiculeront les données

suivantes :

- **données superviseur** :
  1. *origine* : opérateur,
  2. *nature* :
     (a) *requête* : dans ce cas, la donnée *service* demandé par l'opérateur sera également présente. Il peut alors s'agir d'une requête de *commande*, d'une requête de *diagnostic*, d'une requête de *reprise* ou enfin, d'une requête d'*urgence*. Cette donnée permet au superviseur de connaître la fonction de commande ou de surveillance capable de désagréger la requête.
     (b) *compte rendu* : dans ce cas, la donnée *nom de la ressource* est implantée pour permettre au superviseur de sélectionner l'activité de surveillance-commande concernée par l'information.
- **données fonctions** : elles sont à définir par rapport aux besoins de ces fonctions (hors du cadre de notre travail).

Dans le troisième cas, les informations issues du niveau inférieur, véhiculeront les données suivantes :

- **données superviseur** :
  1. *origine* : niveau inférieur,
  2. *nom de la ressource* : dans ce cas, la donnée *nom de la ressource* est suffisante pour permettre au superviseur de sélectionner l'activité de surveillance-commande concernée par l'information. Si nous considérons la figure 3.8, nous voyons que le *nœud de coordination* gère quatre ressources physiques : un *tour*,

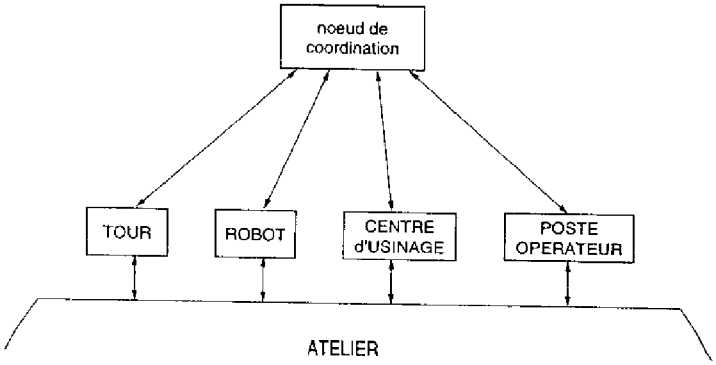


FIG. 3.8 – *liens entre les nœuds de niveau inférieurs et supérieurs*

un *robot*, un *centre d'usinage* et un *poste opérateur*. Lorsqu'une information est par exemple émise par le module *tour*, le superviseur du module coordination doit sélectionner l'activité de surveillance-commande en cours d'exécution sur la ressource tour. La donnée nature n'est pas utile car toute information qui émane du niveau inférieur est considérée ca priori omme un compte rendu.

- **données fonctions** : elles sont à définir par rapport aux besoins de ces fonctions (hors du cadre de notre travail).

Dans le quatrième cas, les informations issues des fonctions de surveillance-commande véhiculeront les données suivantes :

- **données superviseur**

  1. *origine* : interne (générées à l'intérieur du nœud),
  2. *nature* :
     (a) *requête* : cette donnée signale au superviseur qu'une fonction désire émettre une requête vers le niveau inférieur,
     (b) *compte rendu* : cette donnée correspond à la fin d'exécution d'un service demandé. Par exemple, si un diagnostic a été lancé dans un module à partir du niveau supérieur, la fin d'exécution de ce diagnostic correspondra à l'envoi vers le niveau supérieur d'un compte rendu d'exécution.
     (c) *résultat* : cette donnée correspond au résultat fourni par une fonction (*service*). Par exemple, lorsqu'un diagnostic est lancé suite à la détection d'une défaillance, un résultat de diagnostic est fourni.

  données fonctions : elles sont à définir par rapport aux besoins de ces fonctions (hors du cadre de notre travail). Par exemple, pour la fonction détection, un contenu de type "résultat" pourrait être le suivant :

  - copie de l'information en provenance du niveau inférieur,
  - mécanisme de détection associé.

D'un point de vue mise en œuvre, un *moteur* assurera la prise en compte de ces informations. Il réagira de quatre manières différentes, selon les données inscrites dans le premier niveau d'abstraction de l'information considérée. Son algorithme de traitement d'orientation des messages est donné ci-après.

Algorithme du moteur

```
SUIVANT : origine
    niveau supérieur : ORIENTER information vers service
    opérateur : SUIVANT nature :
                requête : ORIENTER information vers service
                compte rendu : ORIENTER information vers fonctions associées
                               à nomressource
            FINSUIVANT
    niveau inférieur : ORIENTER information vers fonctions associées à nomressource
    interne : SUIVANT nature :
                requête : DEBUT
                            REPRESENTER l'état du système
                            EMETTRE information vers le niveau inférieur
                          FIN
                compte rendu : DEBUT
                                 REPRESENTER l'état du système
                                 ORIENTER information vers fonction couplée
                                 EMETTRE information vers le niveau supérieur
                               FIN
                résultat : DEBUT
                             REPRESENTER l'état du système
                             ORIENTER information vers fonction couplée
                           FIN
            FINSUIVANT
FINSUIVANT
```

3.4.3 Représenter l'évolution du traitement de surveillance-commande

Lorsque le *moteur* prend en compte une information, il est souvent amené à faire évoluer l'état du système de surveillance-commande pour représenter l'état courant. Pour cela, il doit transmettre cette information à un autre mécanisme de mise en œuvre de la stratégie qui est chargé de la faire évoluer. Parmi les deux types de mise en œuvre envisageables, compilée [Paludetto *et al.*, 1990] ou interprétée [Benzakour, 1985], nous avons retenu la seconde. Cette technique présente de nombreux avantages comme l'évitement des erreurs de traduction dans un langage de programmation, l'accessibilité à l'état global du système grâce au marquage du réseau ou encore la possibilité d'analyser les évolutions envisageables à tout instant [Combacau, 1991]. Les principes de cette technique reposent sur l'utilisation d'un *joueur de réseau de Petri* [Atabakhche, 1987] dont le fonctionnement global est présenté dans la figure 3.9 [Bako *et al.*, 1990]. Dans son état stable, le joueur est en attente d'un événement (information transmise par le moteur). Dès réception de cette information, le joueur recherche la transition correspondante dans le modèle de la stratégie. Si les conditions (résultats des fonctions de surveillance-commande) portées sur une des transitions correspondent à l'information reçue, alors la

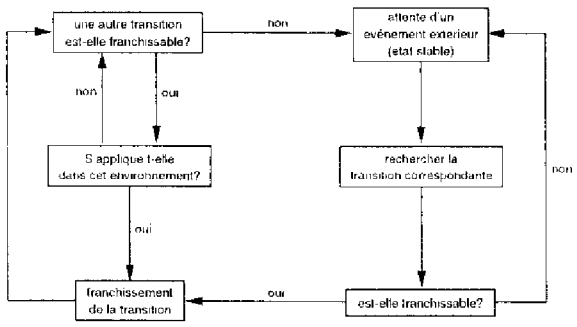


FIG. 3.9 – *mécanisme d'un joueur de réseau de Petri*

transition est sélectionnée. Le joueur détermine ensuite si elle est franchissable. Si tel est le cas, la transition est franchie. Si d'autres transitions deviennent alors franchissables, elles sont à leur tour franchies. Sinon, le joueur retourne dans son état stable.

L'exécution de la stratégie de surveillance-commande est donc cadencée par les informations qui lui sont transmises. Ces informations correspondent aux résultats des fonctions de surveillance-commande et se traduisent par des événements de début et fin d'activités de surveillance-commande.

Le fonctionnement global du système de surveillance-commande correspond à une évolution synchronisée du réseau modélisant la stratégie de surveillance-commande et de son réseau de référence :

1. **Événement de début d'activité** : si les pré-conditions d'une activité prête à être exécutée sont satisfaites dans le réseau de la stratégie de surveillance-commande, un appel est fait vers le réseau de référence associé, de manière à tenir compte des contraintes qui y sont décrites. L'aspect modélisation de cet appel sera développé dans le paragraphe suivant.

2. **Événement de fin d'activité** : comme pour les événements de début d'activité ceux-ci sont initiés par le superviseur. Ils signalent la fin de l'activité de surveillance-commande en cours. Cela se traduit par une sensibilisation de la transition de fin d'activité. Un appel est fait vers le réseau de référence de manière à y représenter la fin de cette activité, en particulier la libération des fonctions de surveillance-commande utilisées par l'activité. Les deux modèles évoluent donc de façon parfaitement synchrone.

L'exécution des traitements de surveillance-commande est donc en partie liée au fonctionnement des procédures d'appel au modèle de référence. Ces procédures d'appel caractérisent une classe des mécanismes de communication.

3.5 Mécanismes de communication

La description des mécanismes de communication dans notre approche est une étape importante puisqu'elle conditionne le fonctionnement de notre architecture de surveil-

lance-commande. Cette description doit être menée sur deux plans, d'une part la communication entre les niveaux de notre structure et d'autre part entre la stratégie de surveillance-commande et le modèle de référence pour la surveillance-commande.

Comme les mécanismes de communication inter-niveaux ont déjà été décrits dans le paragraphe §3.4, nous allons uniquement nous intéresser ici aux mécanismes de communication inter-réseaux.

La synchronisation entre le modèle de la stratégie de surveillance-commande et son modèle de référence est basée sur le principe de fusion de transitions (cf. figure 3.10). Un des avantages certains de cette technique concerne la validation du modèle global

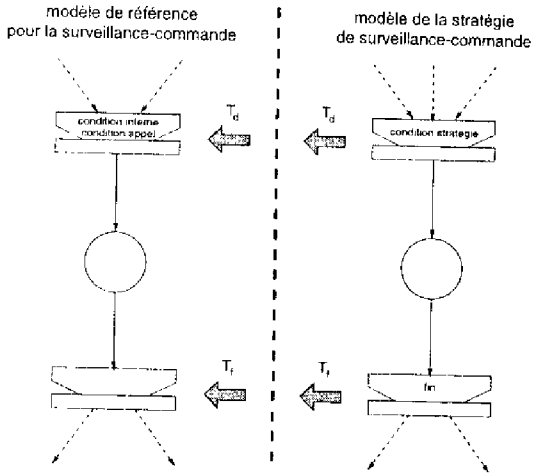


FIG. 3.10 – *lien stratégie/référence*

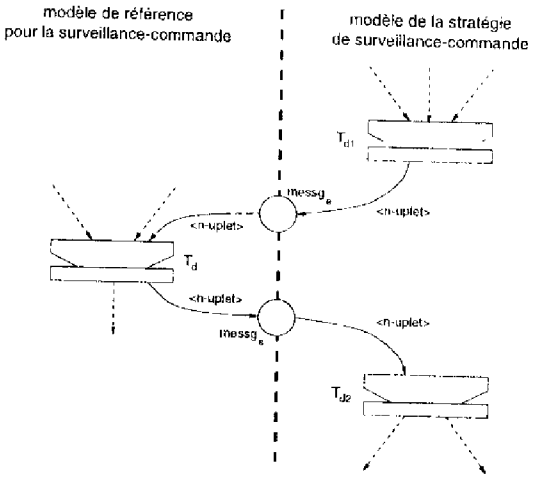


FIG. 3.11 – *spécification d'un lien*

obtenu (réseau de la stratégie de surveillance-commande et réseau de référence associé) par les techniques classiques d'analyse de réseaux de Petri. La spécification du lien entre les transitions du réseau de la stratégie de surveillance-commande et son modèle de référence est largement inspirée de [Combacau, 1991]. En effet, les solutions qui y sont apportées pour spécifier le mécanisme de communications entre le réseau de commande et son modèle de référence conviennent parfaitement à nos attentes.

Le lien obtenu par la fusion de deux transitions (transition stratégie, transition référence) n'est pas symétrique. Comme le montre la figure 3.11, il est constitué d'une *transition appelante* dans le réseau de la stratégie de surveillance-commande qui fournit un ensemble d'informations (via le <n-uplet>) à la *transition appelée* dans le réseau de référence. Cela définit un appel classique de procédure à distance. La transition appelante T_d est décomposée en deux autres transitions T_{d1} et T_{d2} qui sont liées à la transition T_d du réseau de référence par deux places $messg_e$ et $messg_s$. Ces places constituent les canaux au travers desquels les informations sont échangées entre les deux réseaux. Ces informations sont les mêmes que celles véhiculées par le jeton qui circule dans le réseau de la stratégie de surveillance-commande. Ces informations sont capitales pour déterminer si les *conditions internes* (cf. figure 3.10) au modèle de référence sont vraies pour autoriser l'exécution de l'activité demandée ($messg_e$).

Le mécanisme nécessaire à la fusion des transitions de fin d'activité est le même que celui détaillé précédemment. Nous ne le décrirons donc pas.

3.6 Conclusion

Dans ce chapitre nous avons présenté notre approche de la surveillance et de la commande des systèmes à événements discrets complexes. La surveillance qui y est proposée se veut essentiellement ne pas contraindre l'utilisateur à une solution unique. Bien au contraire, elle lui propose toutes les solutions possibles offertes par un système de surveillance cohérent pour traiter tout type d'événements, qu'ils soient normaux ou anormaux. A partir de ces solutions, l'utilisateur peut, à sa convenance, les instancier pour satisfaire ses besoins (élaboration de la stratégie de surveillance-commande hors ligne). Durant la phase de production (en ligne), un superviseur de surveillance-commande se charge de la mise en œuvre de cette stratégie de surveillance-commande selon les événements qu'il reçoit du "monde extérieur". Cette approche est donc différente de celles où le concepteur doit envisager à l'avance tous les symptômes de défaillance (approche de type filtre comme celle développée au LAIL [Cruette, 1991] ou au CRAN): il devra en revanche prévoir ici uniquement le type de fonctionnement global de surveillance (pour un nœud considéré). Selon ce type de fonctionnement, le système de surveillance-commande réagira différemment aux occurrences d'événements sans pour autant les connaître à l'avance. Il est cependant évident, que si les fonctions de surveillance-commande utilisées dans notre approche ne peuvent assurer le service pour lequel elles ont été prévues, alors notre approche est remise en cause. Toutefois, des travaux tels que [Hammami *et al.*, 1995], [Bernauer, 1996], [Valette, 1994] (conception des fonctions détection et diagnostic), [Chaillet, 1995], [Kunzle *et al.*, 1994], [Hammami *et al.*, 1996] (élaboration de la fonction diagnostic) et [de Bonneval, 1993] (fonction décision/reprise), nous encouragent dans cette voie en nous démontrant que de telles fonctions existent et satisfont à leurs spécifications.

La suite de ce mémoire est consacrée à une application de notre système de surveillance-commande sur un exemple inspiré d'une installation réelle.

Partie III

Exemple d'application

Chapitre 1

Présentation de l'atelier et du système de Surveillance-Commande

1.1 Introduction

Dans cette partie, nous développons un exemple d'application de notre approche de surveillance-commande sur une installation réelle : la cellule flexible de l'École Nationale d'Ingénieur de Tarbes (ENIT). Ce choix nous est paru judicieux pour plusieurs raisons. Premièrement, cet exemple avait déjà été traité dans [Combacau, 1991]. Certaines hypothèses avaient due y être faites. En particulier, il était considéré "que chaque commande numérique était capable de retourner une information suffisante pour identifier le mécanisme de détection qui avait entraîné un arrêt de l'opération en cours" [Combacau, 1991]. Grâce à la structure de surveillance-commande que nous venons de présenter, nous verrons que cette hypothèse peut être levée. Deuxièmement, le matériel utilisé dans cet atelier ne permet pas toujours de réaliser des traitements automatiques de défaillances. Un mode manuel est souvent requis. Grâce à sa fonction de suivi, notre système autorise un tel mode de fonctionnement. Il peut en effet se recaler sur l'état réel du système surveillé et de ce fait être en mesure de "reprendre le contrôle" dès le retour en mode automatique.

1.2 Caractéristiques techniques

De nombreux travaux ont été développés sur la base de la cellule flexible de l'ENIT [Belkadi, 1989] [Combacau, 1991] [de Bonneval, 1993]. Pour cette raison, nous nous limiterons simplement à en rappeler les principales caractéristiques.

1.2.1 Machines Outils

- **1 centre d'usinage horizontal (CUH) "GRAFFENSTADEN"** :

  machine quatre axes à broche horizontale à commande numérique (NUM460),

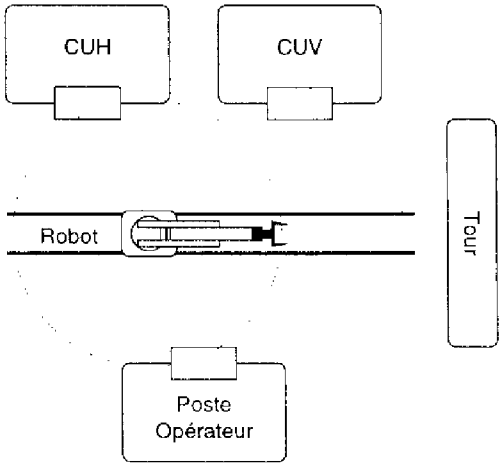


FIG. 1.1 *disposition au sol des éléments de la cellule flexible*

1 changeur d'outils automatique,

1 magasin de trente outils,

1 chargeur de pièces à deux postes.

- **1 centre d'usinage vertical (CUV) "HES-TOYODA"** :

  machine trois axes à broche verticale à commande numérique (NUM760),

  1 changeur d'outils automatique (20 outils disponibles),

  1 changeur de pièces à deux postes.

- **1 tour horizontal "Ernault Somua"** :

  tour à commande numérique (NUM460) pour l'usinage de surface de révolution,

  ce tour est équipé d'un capteur d'usure d'outil. Les signaux qu'il renvoie sont accessibles par un opérateur [Arreguy *et al.*, 1990].

- **1 robot "YACMA Y28"** :

  robot industriel six axes (manipulation, peinture, soudage) à commande intégrée,

  placé sur un axe linéaire numérisé de douze mètres (transfert entre les éléments de l'atelier).

- **1 poste opérateur**

1.2.2 Commandes Numériques

- **NUM460** : cette commande numérique est un système très fermé. Les informations portant sur l'état d'avancement de la transformation ne sont pas fournies. En cas de défaillance, la machine est automatiquement arrêtée et un signal "défaut" est émis.

Le rôle de l'opérateur humain (surveillance en mode manuel) est ici primordial pour la surveillance.

- **NUM760** : plus sophistiquée, cette commande numérique comprend un module LPC (Link Programmable Controller). Il permet la collecte d'informations correspondant aux comptes rendus d'exécution de commande que nous utilisons dans notre structure. Plusieurs programmes peuvent résider en même temps dans sa mémoire.

- **CN robot** : uniquement dédiée à la commande, elle ne permet pas d'accéder aux informations. Seule l'observation du fonctionnement du robot par un opérateur permet de remédier à ce problème.

1.3 Hiérarchisation de la surveillance-commande

La cellule flexible ne présentant pas de complexité particulière, la définition d'une structure hiérarchique différente de celle proposée dans [Combacau, 1991] ne s'impose pas. Elle convient en effet parfaitement à notre approche de surveillance. Cette structure est élaborée selon une démarche ascendante [Combacau, 1991], en partant des services offerts par les éléments du procédé jusqu'au niveau le plus élevé dans lequel toutes les possibilités offertes par l'atelier sont exprimées. Au niveau le plus bas, les services offerts sont ceux proposés par les Commandes Numériques des machines :

- **CUH/NUM460** :
  - *usinage*,
  - *palettisation*,
  - *programmation*,
  - *bridage-pièce*,
  - *débridage-pièce*.
- **Tour/NUM460** :
  - *usinage*,
  - *programmation*,
  - *serrage-mandrin*,
  - *desserrage-mandrin*.
- **CUV/NUM760** :
  - *usinage*,
  - *palettisation*,
  - *programmation*,

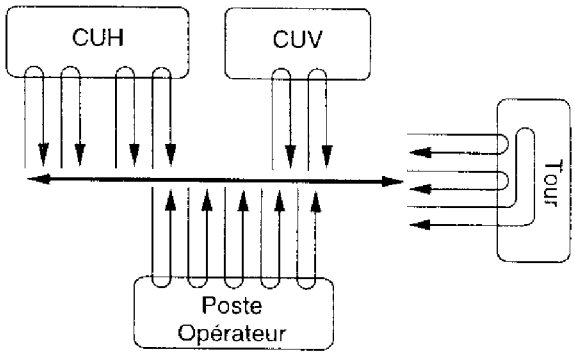


FIG. 1.2 – *trajectoires du robot*

- *bridage-pièce*,
- *débridage-pièce*.

- **robot/CN** : quatorze trajectoires (composées de déplacement, ouverture pince et fermeture pince) ont été créées pour desservir les machines (cf. figure 1.2 extraite de [Combacau, 1991]).

- **Poste Opérateur** :
  - *préparation-pièce*,
  - *chargement-pièce*,
  - *évacuation-pièce*,

Tous ces services sont gérés par le niveau directement supérieur : niveau 1 (cf. figure 1.3). Les modules de ce niveaux proposent à leur tour des services au niveau coordination (niveau le plus haut de la structure). Ces services sont :

- **module CUH** : *usinage horizontal*,
- **module Tour** : *usinage tour*,
- **module CUV** : *usinage vertical*,
- **module Robot** :
  - *transférer pièce sur le tour*,
  - *transférer pièce*.
- **module Poste Opérateur** :
  - *préparer pièce*,
  - *introduire pièce*,
  - *évacuer pièce*.

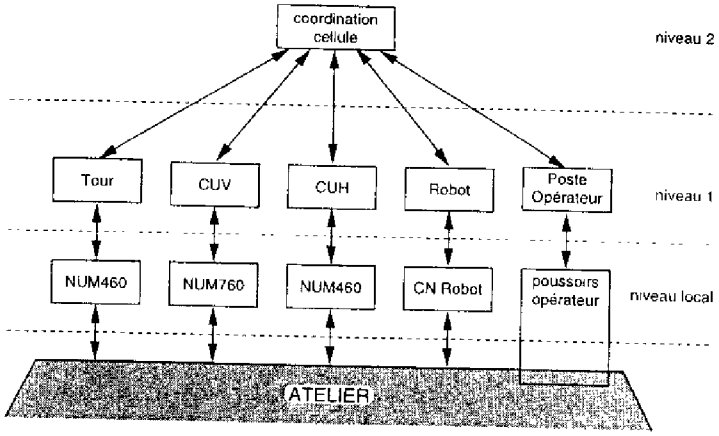


FIG. 1.3 – *structure de surveillance-commande*

Remarque : pour simplifier l'exemple au niveau des modèles de commande, nous avons supposé l'existence d'un chargeur automatique de pièce sur le Tour. Il propose au module coordination les services *accepter-pièce* et *rendre pièce*. Cette hypothèse a quelque incidence sur l'élaboration de notre système de surveillance-commande. Nous verrons dans la conclusion que pour la lever, il faut envisager un système de surveillance-commande hétérarchique. Nous n'en tiendrons pas compte au niveau de la représentation de la structure générale de surveillance-commande.

La structure générale de surveillance-commande obtenue est donnée dans la figure 1.3.

Nous allons maintenant nous focaliser sur l'élaboration des différents modèles de surveillance-commande. En premier lieu, nous compléterons les modèles de référence de surveillance-commande pour chacun des modules de la structure hiérarchique. Ensuite, en nous plaçant en tant qu'utilisateur de la cellule, nous définirons les différentes stratégies de surveillance-commande.

1.4 Modèles de référence pour la surveillance-commande

La structure des modèles de référence étant tout à fait générique, ces modèles ne sont pas à redécrire pour chacun des modules [Zamaï *et al.*, 1998] [Chaillet, 1997]. En revanche, il appartient à l'utilisateur d'y spécifier les ressources physiques ou abstraites qui devront être surveillées et commandées. Dans ce sens, nous déterminons pour chacun des modules de la structure hiérarchique de surveillance-commande quelles sont les ressources qui y sont gérées :

1. **module coordination** : ressources *Tour, CUV, CUH, Robot, Poste Opérateur.*
2. **module Tour** : ressources *mandrin, chariot, chargeur de programme.*

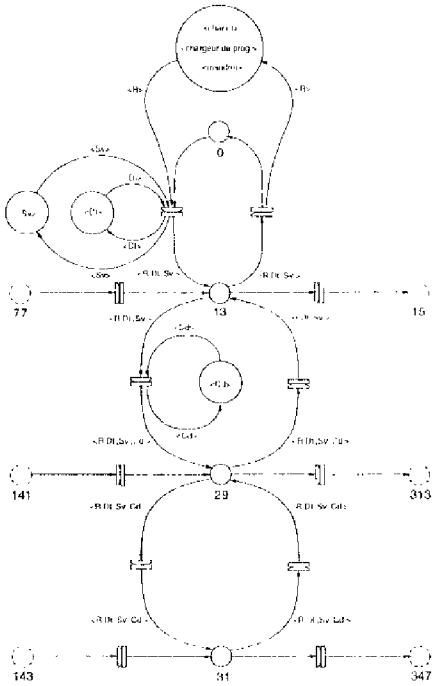


FIG. 1.4 – *extrait du réseau de référence du module Tour*

3. **module CUV** : ressources *porte pièce, palettiseur, broche, chargeur de programme.*
4. **module CUH** : idem.
5. **module Robot** : une seule ressource, la *CN du Robot.*
6. **module Poste Opérateur** : pour les besoins de l'exemple, nous considérons ici l'opérateur comme une "ressource" gérée dans le module poste opérateur. Il s'agit toutefois d'un abus de langage puisque dans ce cas particulier l'opérateur sera également chargé du traitement de surveillance. Ce traitement ne peut être en effet réalisé qu'en mode manuel.

Par souci de simplification et de concision, nous ne donnerons pas ici les six modèles de référence. Seul un extrait du modèle de référence pour la surveillance-commande du module *Tour* est représenté (cf. figure 1.4).

1.5 Modèles de la stratégie

Ce paragraphe est dédié à l'élaboration des séquences d'activités de surveillance-commande qui définissent les différentes stratégies de surveillance, suivant les niveaux, les modules et les produits considérés.

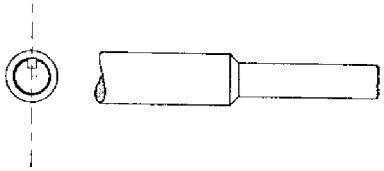


FIG. 1.5 - *pièce à réaliser : arbre de transmission*

1.5.1 Module Coordination

La démarche que nous avons proposée dans ces travaux consiste à spécifier, d'une part, les contraintes issues de la politique de l'entreprise, d'autre part, celles issues de la transformation du produit. Ensuite, à partir des activités offertes par le modèle de référence de surveillance-commande et en tenant compte des contraintes mises en évidence, le modèle de la stratégie peut être construit.

1. **Politique de l'entreprise** : dans un contexte où la productivité est une contrainte forte, nous allons supposer pour cet exemple que la politique générale consiste à "produire tout de même". Cette politique impose que l'occurrence d'une défaillance du procédé peut ne pas aboutir systématiquement à un bloquage de la séquence opératoire en cours d'exécution (sauf bien sûr en cas d'urgence [de Bonneval, 1993]). Un traitement complet de surveillance devra d'abord permettre de trouver une solution au problème posé. Ce n'est qu'en fonction de cette solution que la séquence de commande sera bloquée ou non pour réparation.

2. **Contraintes imposées par la transformation du produit** : pour les besoins de l'exemple, nous avons décidé de simuler une production en petite série d'arbres de transmission. La figure 1.5 nous donne une représentation du produit fini souhaité. Pour la fabrication de ce produit nous avons défini des tolérances de qualité très larges. Le produit peut donc être considéré fini même si son état de surface est "très moyen". Ce choix n'entraîne aucun conflit avec la politique de production envisagée par l'entreprise. En conséquence, la transformation du produit n'impose aucune contrainte sur la stratégie de surveillance-commande à ce niveau d'abstraction.

La prise en compte des contraintes issues de la politique de l'entreprise impose des choix particuliers à faire parmi les activités proposées dans le modèle de référence. Ces choix sont basés sur une étude des différents types de fonctionnement du système de surveillance-commande :

- en **fonctionnement normal**, les activités qui doivent être considérées ne peuvent être constituées que des fonctions *commande*, *détection* et *suivi*. L'association de toute autre fonction représente nécessairement le traitement en cours d'une défaillance. De ce fait, seules les activités $13 =< R, Dt, Sv >$ et $31 =< R, P, Dt, Sv, Cd >$ sont à considérer.

en **fonctionnement anormal**, la politique de l'entreprise impose de produire tout de même. Dans ce cas, toutes les activités mettant en évidence l'association de la fonction *commande* et des fonctions *décision* ou *diagnostic* sont à utiliser dans la stratégie. Ces activités sont :

$63 =< R, P, Dt, Sv, Cd, Dg >$

$95 =< R, P, Dt, Sv, Cd, Dc >$

Au cours de ces activités, la commande peut se terminer normalement (sans échec visible). Le service est alors rendu. Les activités $47 =< R, P, Dt, Sv, Dg >$ et $79 =< R, P, Dt, Sv, Dc >$ sont alors requises.

Enfin, quelles que soient les activités de décision $95 =< R, P, Dt, Sv, Cd, Dc >$ ou $79 =< R, P, Dt, Sv, Dc >$ il faudra toujours être en mesure d'appliquer la solution envisagée. L'activité $143 =< R, P, Dt, Sv, Rp >$ proposée dans le modèle de référence est prévue à cet effet.

Durant cette activité, l'occurrence d'une propagation de défaillance est envisageable : non exécution de la séquence de reprise. Des activités de surveillance doivent alors être prévues, il s'agit des activités faisant intervenir le diagnostic et la décision : $175 =< R, P, Dt, Sv, Dg, Rp >$ et $207 =< R, P, Dt, Sv, Dc, Rp >$.

évolutions possibles hors production : tout en étant hors production (aucune requête de commande n'est émise vers les niveaux inférieurs), le procédé est tout de même surveillé (activité $13 =< R, Dt, Sv >$). Des évolutions intempestives peuvent se produire suite à une intervention de l'opérateur par exemple. Si ces évolutions provoquent une propagation de traitement de défaillance, le module coordination doit être en mesure de la prendre en compte. Pour cela, les activités $45 =< R, Dt, Sv, Dg >$, $77 =< R, Dt, Sv, Dc >$ et $141 =< R, Dt, Sv, Rp >$ doivent apparaître au niveau de la stratégie.

possibilité de diagnostic détaillé : le module coordination étant le plus haut de la hiérarchie, aucune requête de diagnostic détaillé ne peut y être reçue. Aucune activité n'est donc prévue dans ce cas.

situations d'urgence : quelle que soit la situation dans laquelle le module se trouve, des défaillances jugées critiques par l'utilisateur peuvent se produire. Des procédures d'urgence doivent alors être lancées à partir de toutes les activités décrites précédemment :

$315 =< R, P, Sv, Cd, Dg, Ug >$,

$347 =< R, P, Sv, Cd, Dg, Ug >$,

$297 =< R, Sv, Dg, Ug >$,

$329 =< R, Sv, Dc, Ug >$,

$331 =< R, P, Sv, Dc, Ug >$,

$427 =< R, P, Sv, Dg, Rp, Ug >$,

$459 =< R, P, Sv, Dc, Rp, Ug >$.

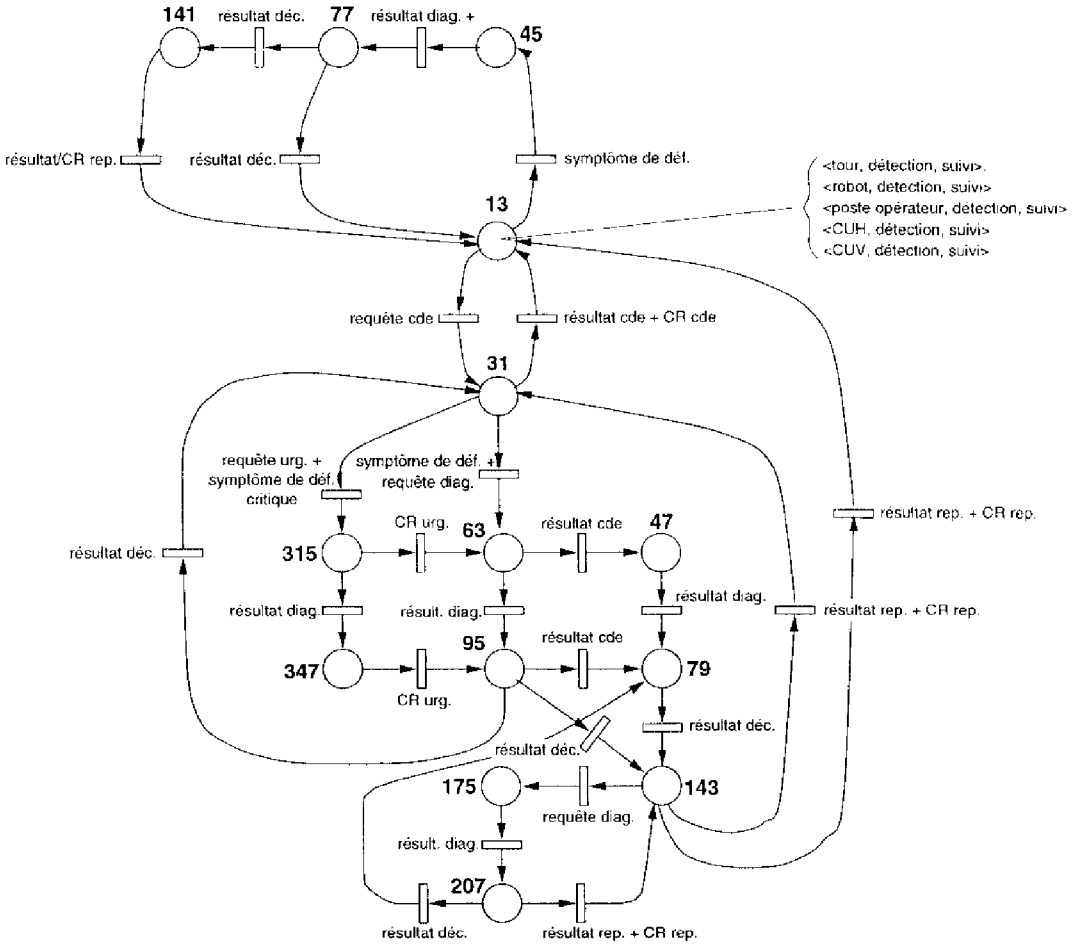
Tenant compte des contraintes imposées par le modèle de référence, en particulier la séquentialité obligatoire de certaines activités (cf. liste des enchaînements élémentaires présentée en annexes 2.6), nous obtenons le réseau de stratégie représenté dans la figure 1.6. Une fois le modèle de la stratégie établi, il s'agit d'y intégrer les prédicats portés sur chacune des transitions. Ces prédicats définissent les événements qui conditionnent le passage d'une activité à une autre. Par exemple, le passage de l'activité $< tour, Dt, Sv >$ (place 13) à $< tour, Dt, Sv, Cd >$ (place 31) est conditionné par l'occurrence d'un événement *requête de commande liée à la ressource mandrin*. Ou encore, l'occurrence de l'événement *résultat de diagnostic sur la ressource robot* conditionne le passage de l'activité courante $< robot, Dt, Sv, Cd, Dg >$ (place 63) vers l'activité $< robot, Dt, Sv, Cd, Dc >$ (place 95).

Par souci de clarté, nous n'avons pas représenté, dans les modèles de la stratégie, tous les prédicats des transitions. Nous n'avons pas non plus modélisé tous les processus d'urgence.

1.5.2 Module Tour

De la même façon que pour le module coordination, nous avons défini la stratégie du module tour.

1. **Politique de l'entreprise** : la politique générale étant de "produire tout de même", quel que soit le niveau considéré, elle doit être appliquée si possible.
2. **Contraintes imposées par la transformation du produit**. Etant donné le niveau d'abstraction considéré, nous pouvons ici distinguer des contraintes plus particulièrement liées aux ressources chargées de la transformation et non directement au produit à transformer.
   - **chariot** : la transformation "usinage" est réalisée par cette ressource jusqu'à ce qu'un "défaut" soit détecté ne permettant plus de produire. En revanche, l'information "usure anormale de l'outil à charioter", générée par un opérateur, peut être tolérée. Un traitement de surveillance peut alors être lancé de manière à déterminer les causes exactes de cette usure et d'envisager ainsi une quelconque reprise. Ce n'est qu'à ce moment là que la production doit être bloquée. Le chariotage du produit permet donc ici de tenir partiellement compte de la politique de l'entreprise. En revanche, l'occurrence d'un signal "Défaut et arrêt de la machine" qui va à l'encontre de la politique de l'entreprise sera accepté et traité par le système de surveillance. Ce non respect de la politique de l'entreprise est bien plus évident lorsque l'on considère la ressource "mandrin".
   - **mandrin** : quelle que soit la politique de l'entreprise, le serrage du mandrin doit être correctement effectué avant tout usinage. Il conditionne en effet



13 = <Ressource, Détection, Suivi>
31 = <Ressource, Produit, Détection, Suivi, Commande>
45 = <Ressource, Détection, Suivi, Diagnostic>
47 = <Ressource, Produit, Détection, Suivi, Diagnostic>
63 = <Ressource, Produit, Détection, Suivi, Commande, Diagnostic>
77 = <Ressource, Détection, Suivi, Décision>
79 = <Ressource, Produit, Détection, Suivi, Décision>
95 = <Ressource, Produit, Détection, Suivi, Commande, Décision>
141 = <Ressource, Détection, Suivi, Reprise>
143 = <Ressource, Produit, Détection, Suivi, Reprise>
175 = <Ressource, Produit, Détection, Suivi, Diagnostic, Reprise>
207 = <Ressource, Produit, Détection, Suivi, Decision, Reprise>
315 = <Ressource, Produit, Suivi, Commande, Diagnostic, Urgence>
347 = <Ressource, Produit, Suivi, Commande, Décision, Urgence>

FIG. 1.6 – *stratégie de surveillance-commande pour le module Coordination*

la transformation future. Pour cette raison, toute occurrence de défaillance doit impérativement entraîner le bloquage de la commande. Un traitement classique de surveillance pourra alors être lancé.

- **chargeur de programme** : "En ce qui concerne le chargement du programme d'usinage, nous avons supposé que le système informatique n'introduit jamais de dysfonctionnement" [Combacau, 1991] (simple transfert d'informations entre mémoire de masse et la mémoire de la commande numérique). Cette ressource ne sera donc jamais couplée à un traitement de surveillance (diagnostic, décision ou reprise).

Comme pour le module coordination, nous allons énumérer ci-après l'ensemble des activités de surveillance-commande que nous devons prendre en compte dans la stratégie du module tour.

Remarque: nous ne justifierons ici que les choix qui diffèrent de ceux effectués dans le module coordination.

- en **fonctionnement normal** :

  $13 =< R, Dt, Sv >$,

  $31 =< R, Dt, Sv, Cd >$,

  $15 =< R, P, Dt, Sv >$ (cas d'une pièce sur le mandrin).

- en **fonctionnement anormal** :

  $63 =< R, P, Dt, Sv, Cd, Dg >$, sauf pour le mandrin (pas de production tout de même),

  $95 =< R, P, Dt, Sv, Cd, Dc >$,

  $47 =< R, P, Dt, Sv, Dg >$,

  $79 =< R, P, Dt, Sv, Dc >$,

  $95 =< R, P, Dt, Sv, Cd, Dc >$,

  $143 =< R, P, Dt, Sv, Rp >$,

  $175 =< R, P, Dt, Sv, Dg, Rp >$,

  $207 =< R, P, Dt, Sv, Dc, Rp >$.

- **évolutions possibles hors production** :

  $45 =< R, Dt, Sv, Dg >$,

  $63 =< R, P, Dt, Sv, Cd, Dg >$,

  $175 =< R, P, Dt, Sv, Dg, Rp >$,

- **possibilité d'accepter de faire un diagnostic détaillé** : une requête de diagnostic détaillé peut être envoyée par le niveau coordination. Cela doit impérativement être pris en compte. Pour cela, les activités suivantes sont à sélectionner :

  $45 =< R, Dt, Sv, Dg >$,

$77 =< R, Dt, Sv, Dc >,$

$141 =< R, Dt, Sv, Rp >.$

- **situations d'urgence :**

$315 =< R, P, Sv, Cd, Dg, Ug >,$

$347 =< R, P, Sv, Cd, Dg, Ug >,$

$297 =< R, Sv, Dg, Ug >,$

$329 =< R, Sv, Dc, Ug >,$

$331 =< R, P, Sv, Dc, Ug >,$

$427 =< R, P, Sv, Dg, Rp, Ug >,$

$459 =< R, P, Sv, Dc, Rp, Ug >.$

A partir de ces activités, le réseau de la stratégie du tour est élaboré (cf. figure 1.7). Il consiste à enchaîner les activités sélectionnées en tenant compte des contraintes imposées dans le réseau de référence.

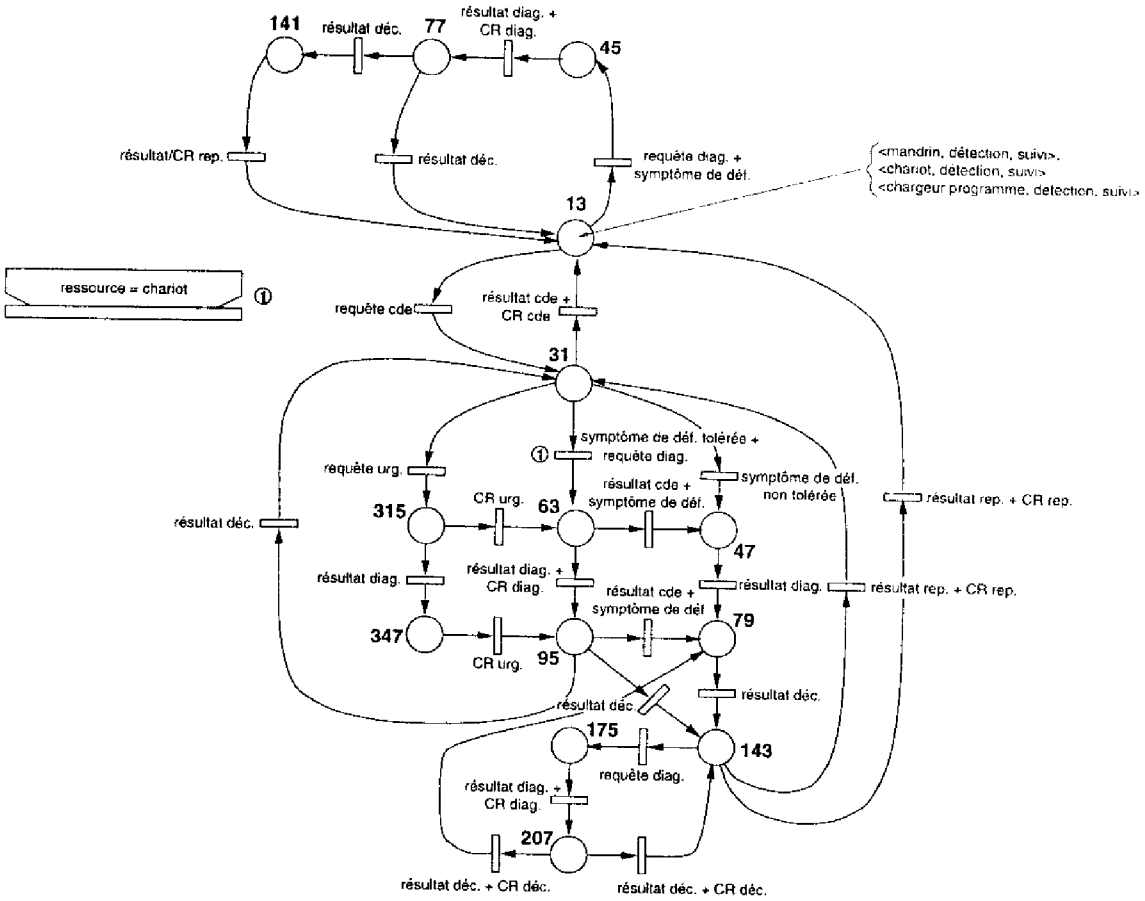
Toutes les transitions contiennent des prédicats. Ces prédicats concernent ici la ressource et l'événement permettant l'évolution du modèle. Par exemple, dans la stratégie représentée figure 1.7 seules les activités contenant la ressource "chariot" peuvent suivre la séquence "$31 \longmapsto 63$" sur occurrence de l'événement "usure anormal de l'outil".

1.5.3 Autres modules

De la même façon que pour les modules *Coordination* et *Tour*, nous obtenons les stratégies des modules *CUV*, *CUII*, *Robot* et *Poste Opérateur*. Nous ne les avons pas développées ici pour alléger le manuscrit.

1.6 Conclusion

Dans ce chapitre nous avons présenté la cellule flexible de l'ENIT sur laquelle nous allons analyser le comportement de notre système de surveillance-commande. A partir des éléments qui la composent, nous avons proposé une structure hiérarchique de surveillance-commande. Nous avons ensuite élaboré l'ensemble des modèles requis pour la surveillance au sein de cette structure. Cette étape de modélisation a montré que notre système de surveillance n'est plus considéré comme un palliatif à la commande. Nous n'avons pas eu besoin de construire les modèles de commande pour réaliser les modèles de surveillance. L'intérêt du modèle de référence pour la surveillance-commande a également été mis en avant pour construire les modèles de stratégies. Les différentes stratégies alors mises en évidence pour surveiller et commander la cellule ont permis de découvrir une nouvelle facette de la surveillance. Nous allons voir dans le chapitre suivant les intérêts qui en découlent.



13 = <Ressource, Détection, Suivi>
31 = <Ressource, Produit, Détection, Suivi, Commande>
45 = <Ressource, Détection, Suivi, Diagnostic>
47 = <Ressource, Produit, Détection, Suivi, Diagnostic>
63 = <Ressource, Produit, Détection, Suivi, Commande, Diagnostic>
77 = <Ressource, Détection, Suivi, Décision>
79 = <Ressource, Produit, Détection, Suivi, Décision>
95 = <Ressource, Produit, Détection, Suivi, Commande, Décision>
141 = <Ressource, Détection, Suivi, Reprise>
143 = <Ressource, Produit, Détection, Suivi, Reprise>
175 = <Ressource, Produit, Détection, Suivi, Diagnostic, Reprise>
207 = <Ressource, Produit, Détection, Suivi, Decision, Reprise>
315 = <Ressource, Produit, Suivi, Commande, Diagnostic, Urgence>
347 = <Ressource, Produit, Suivi, Commande, Décision, Urgence>

FIG. 1.7 – *stratégie de surveillance-commande pour le module Tour*

Chapitre 2

Etude du fonctionnement de la surveillance-commande

2.1 Introduction

Dans ce chapitre, nous allons réaliser deux simulations pas à pas du fonctionnement de notre approche. Ces simulations sont basées sur les deux scénarii (cf. figure 2.1) de traitement de défaillances décrits dans le paragraphe suivant. L'étude de ces deux scénarii

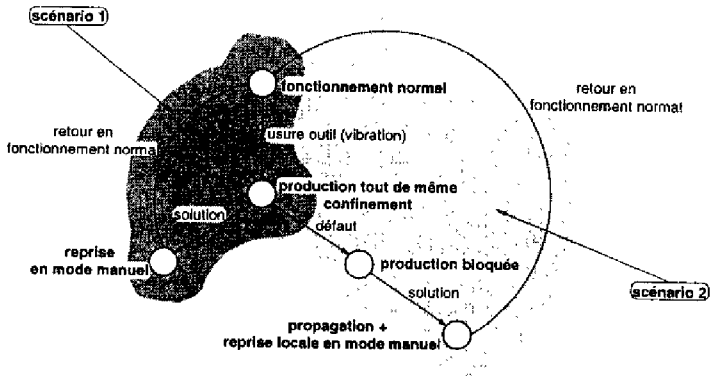


FIG. 2.1 – *scénarii basés sur la détection de défaillance dans le module Tour*

permettra de mettre en valeur les apports de notre approche.

2.2 Présentation du scénario 1

Après avoir reçu une requête (cf. figure 2.3) issue du niveau pilotage, le module coordination exécute la séquence de commande représentée dans la figure 2.2.

A son tour, ce module (coordination) émet successivement vers le niveau inférieur une série de requêtes (Introduire, Transférer vers Tour, Usinage-Tour, etc.) correspondant à la désagrégation de celle reçue (cf. figure 2.3).

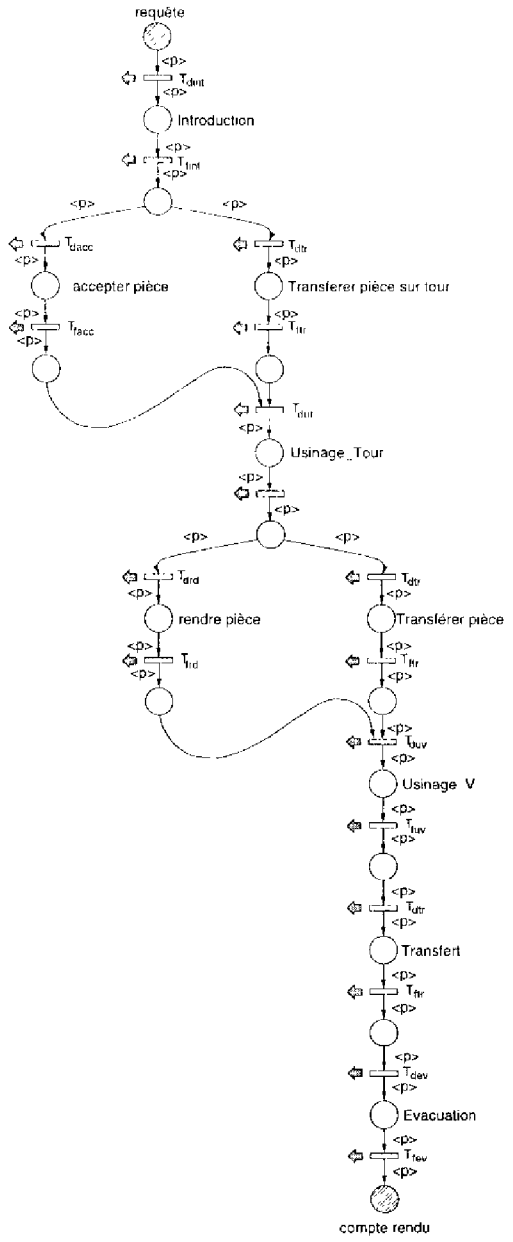


FIG. 2.2 - *séquence de commande du module Coordination*

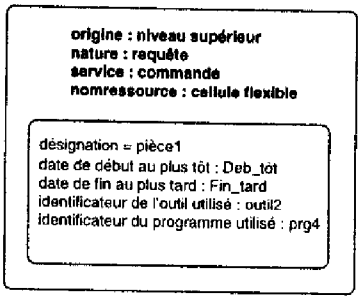


FIG. 2.3 – *requête reçue par le module coordination*

Pendant l'exécution de l'usinage dans le module Tour, supposons que l'opérateur signale une **usure anormale de l'outil** à charioter (détection d'une vibration importante de l'outil). Puisque cette défaillance est tolérée par la stratégie (produire tout de même), son traitement est confiné dans le module Tour jusqu'à ce que les causes en soient trouvées. Un mauvais serrage du mandrin étant à l'origine de cette défaillance, la décision impose le changement de l'outil, le serrage du mandrin et enfin de terminer la transformation du produit en cours (retour en fonctionnement normal). Lorsque la reprise s'achève, nous supposons que le module Tour est revenu en fonctionnement normal sans avoir transgressé les contraintes temporelles imposées par le niveau supérieur (cf. figure 2.3).

Nous allons maintenant suivre pas à pas les réactions du système de surveillance-commande proposé dans notre approche.

2.3 Fonctionnement

Avant de recevoir la requête issue du niveau pilotage, les activités en cours dans le réseau de la stratégie de surveillance-commande du module coordination sont les mêmes pour toutes les ressources qui y sont gérées : $< tour, Dt, Sv >$, $< robot, Dt, Sv >$, $<poste\ opérateur, Dt, Sv>$, etc. La réception de la requête de commande (cf. figure 2.3) par le superviseur du module coordination déclenche la série d'actions suivante :

1. l'algorithme du moteur, donné page 87 de ce manuscrit, permet au superviseur d'orienter la requête de commande vers la fonction commande.

2. la fonction commande désagrège alors la requête en la séquence de commande représentée par le réseau de commande de la figure 2.2.

3. le franchissement de la transition du réseau de commande T_{dint} débouche sur l'émission d'une information (cf. figure 2.4) vers le superviseur et la mise à jour du système d'information [Chaillet, 1995].

4. l'occurrence de cette information issue de la fonction commande provoque l'évolution de l'état du système de surveillance-commande. La transition de fin d'activité

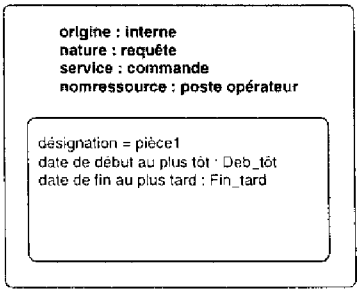


FIG. 2.4 – *information émise par la fonction commande*

<*poste opérateur,Dt,Sv*> conditionnée par l'événement **requête cde** est sensibilisée dans la stratégie et dans le modèle de référence. Son franchissement produit le marquage de la place **31** : <*poste opérateur,Dt,Sv,***Cd**>. Le jeton <*poste opérateur,Dt,Sv*> est retiré de la place **13**.

5. l'information issue de la fonction commande est ensuite propagée vers le niveau inférieur (EMETTRE). Bien entendu, les champs de cette information sont modifiés avant l'émission pour prendre les valeurs indiquées figure 2.5.

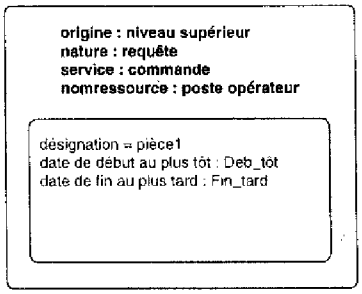


FIG. 2.5 – *requête de commande émise par le superviseur du module coordination*

Un processus similaire se répète au niveau du module "Poste Opérateur".

Nous allons maintenant porter notre attention sur le module Tour. La requête de commande ("tournage") issue du module coordination y est en cours d'exécution. La stratégie de surveillance-commande est dans l'état courant suivant :

- place **31** marquée du jeton <*chariot,Pièce,Dt,Sv,Cd*>,
- place **13** marquée des jetons :

  <*mandrin,Dt,Sv*>, l'opération de serrage est achevée,

  <*chargeur programme,Dt,Sv*>.

Le modèle de commande est quant à lui dans l'état "Usinage" (place usinage marquée, cf. figure 2.6).

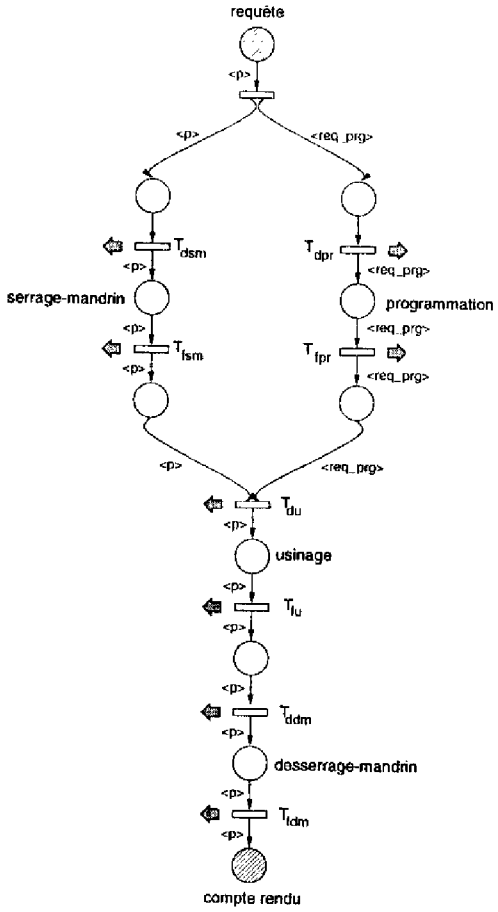


FIG. 2.6 – *séquence de commande dans le module Tour*

Supposons maintenant qu'un opérateur détecte l'usure de l'outil à charioter grâce au capteur d'usure [Arreguy *et al.*, 1990] et qu'il prenne l'initiative d'émettre vers le module Tour l'information représentée dans la figure 2.7.

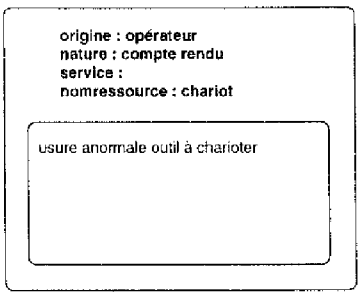


FIG. 2.7 - *signal d'usure outil*

1. Le superviseur du module Tour reçoit cette information.
2. L'algorithme qu'il contient lui impose d'orienter cette information vers les fonctions couplées à la ressource "chariot" : fonctions *commande, détection et suivi.*
3. La détection caractérise l'information de symptôme de défaillance tolérée (compte rendu anormal car il ne correspond pas à celui attendu). Ce résultat est renvoyé

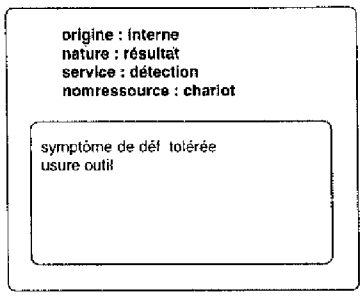


FIG. 2.8 – *résultat de la détection*

vers le superviseur (cf. figure 2.8).

4. Au niveau de la stratégie, la transition "symptôme de déf. tolérée + requête diag" est franchie (cf. figure 1.7, page 105). L'état courant de la stratégie évolue pour représenter la phase de diagnostic lancée sur la ressource "chariot", la commande n'étant pas découplée: <*chariot,P,Dt,Sv,Cd*,**Dg**> (place 63). Simultanément, le résultat de la détection est orienté vers la fonction diagnostic. Cette activité montre que la politique de production tout de même est parfaitement respectée par le système de surveillance-commande. La commande est en effet toujours active et agit sur le procédé. Cependant, un traitement de surveillance est lancé de manière à trouver une solution au problème posé.

5. Grâce aux règles [Combacau, 1991] associées à chacune des opérations de commande gérées dans le module, les causes qui peuvent être à l'origine de cette défaillance sont analysées. Le recours à l'opérateur est nécessaire pour guider le diagnostic dans ses choix. L'opération réalisée sur le chariot est finalement mise hors de cause, une propagation de défaillance est suspectée ce qui se traduit par l'exécution d'un processus de diagnostic sur l'opération précédant "usinage". Il s'agit de l'opération "serrage-mandrin". Une requête de diagnostic sur la ressource mandrin est alors adressée au superviseur.

6. La stratégie évolue pour représenter le couplage du diagnostic à l'activité liée au mandrin. L'état courant global de la stratégie est à cet instant représenté par le marquage suivant :

   - place **45** : <*mandrin,Dt,Sv,***Dg**>,
   - place **63** : <*chariot,Pièce,Dt,Sv,Cd,Dg*>,
   - place **13** : <*chargeur de programme,Dt,Sv*>.

7. La fonction diagnostic met en évidence un mauvais serrage de la pièce effectué lors de l'opération "serrage-mandrin". Cette mise en cause entraîne l'émission d'un résultat de diagnostic vers le superviseur (cf. figure 2.9). Le processus de diagnostic se termine puisque la cause de la défaillance a été trouvée.

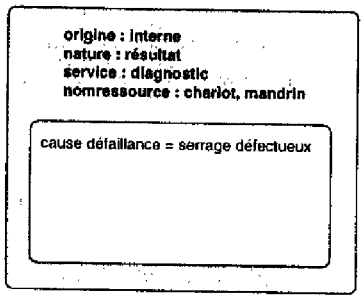


FIG. 2.9 – *résultat du diagnostic : serrage défectueux*

8. Le modèle de la stratégie représente alors l'évolution du processus de surveillance-commande :

   - <*mandrin,Dt,Sv,***Dg**> (place **45**) $\longmapsto$ <*mandrin,Dt,Sv,***Dc**> (place **77**),
   - <*chariot,Pièce,Dt,Sv,Cd,***Dg**> (place **63**) $\longmapsto$ <*chariot,Pièce,Dt,Sv,Cd,***Dc**> (place **95**).

9. La décision possède maintenant tous les éléments pour élaborer une solution de reprise. En supposant que les contraintes temporelles imposées par le niveau supérieur sont encore respectées, une solution peut être envisagée. Elle consiste à arrêter le *tour* pour changer l'outil et re-serrer le mandrin.

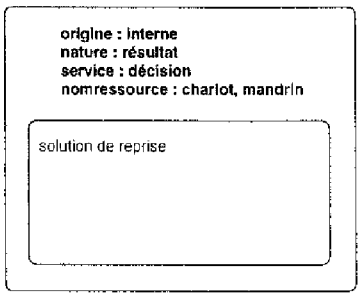


FIG. 2.10 - *information de type résultat produit par la décision*

10. Une fois l'arrêt du "chariot" obtenu, la reprise est exécutée en mode manuel. L'appel à un opérateur est effectué dans le but de changer l'outil et de serrer le mandrin.

11. La fin de la reprise est signalée par l'opérateur. Elle correspond à la réception de deux messages, l'un concernant la ressource *mandrin*, l'autre la ressource *chariot*. Ces deux messages sont acceptés par la fonction reprise qui émet alors un message de fin de reprise (résultat rep.).

12. Ce retour en fonctionnement normal se traduit par une évolution du marquage de la stratégie :

    - $<chariot, Pièce, Dt, Sv, \mathbf{Rp}>$ (place **143**) $\longmapsto$ $<chariot, Pièce, Dt, Sv, \mathbf{Cd}>$ (place **31**),
    - $<mandrin, Dt, Sv, \mathbf{Rp}>$ (place **141**) $\longmapsto$ $<mandrin, Dt, Sv>$ (place **13**).

13. Un nouveau cycle d'usinage peut alors être relancé.

De l'étude du fonctionnement de notre système de surveillance sur un tel scénario plusieurs remarques s'imposent. La politique de "production tout de même" souhaitée par l'entreprise a été respectée dans la mesure du possible. Au niveau du module Tour, sous l'effet d'une défaillance, la commande n'a pas été bloquée pendant le traitement de défaillance. Le séquencement des fonctions diagnostic et décision a permis de trouver une solution au problème posé par la défaillance. Ce n'est qu'à cet instant que la production a été momentanément stoppée pour engager une reprise. Ce type de fonctionnement est garanti par l'existence de la fonction **suivi**. Elle permet de recaler les modèles du procédé sur son état réel. Ainsi, connaissant l'état réel du procédé (tour reconfiguré et pièce montée sur le mandrin), un deuxième cycle d'usinage a pu être relancé.

2.4 Présentation du scénario 2

Ce scénario est une variante du précédent. Il s'en écarte lors du diagnostic de la défaillance "usure anormale de l'outil" dans le module Tour, cette usure étant supposée causée par une mauvaise pièce. Au cours du diagnostic sur l'opération "serrage" (ressource mandrin), un signal de défaut (arrêt tour) est reçu par le superviseur du module

Tour. Nous reprenons l'étude du fonctionnement du système de surveillance-commande à cette étape. L'état de la stratégie du système de surveillance, dans le module Tour, est le suivant (cf. figure 1.7 page 105) :

- place **45** : <*mandrin,Dt,Sv,Dg*>,
- place **63** : <*chariot,Pièce,Dt,Sv,Cd,Dg*>,
- place **13** : <*chargeur programme,Dt,Sv*>. Comme nous l'avons supposé dans le chapitre 1, le chargement d'un programme est une opération considérée comme parfaite. Aucun traitement de surveillance n'est alors à lancer sur cette ressource.

2.5 Fonctionnement

2.5.1 Module tour

T1. Le défaut signalant l'arrêt du Tour, émis par la NUM460/TOUR et reçu par le superviseur, est orienté vers les fonctions couplées à la ressource *chariot* : détection, suivi, commande et diagnostic.

T2. La commande rejette ce compte rendu car il ne représente pas une évolution possible du modèle de commande. La détection émet vers le superviseur un résultat caractérisant un symptôme de défaillance non tolérée.

T3. Ce résultat provoque l'évolution de l'état du système de surveillance-commande. L'activité <*chariot,Pièce,Dt,Sv,***Cd**,*Dg*> (place **63**) passe alors dans l'état <*chariot,Pièce,Dt,Sv,Dg*> (place **47**) pour indiquer l'arrêt de la commande en cours et le début du diagnostic qui en découle.

T4. Le diagnostic, en cours sur les ressources chariot et mandrin, prend en compte cette nouvelle information. Le diagnostic peut alors être achevé sans vraiment attendre une conclusion mettant en évidence la cause de l'arrêt de la production : l'opérateur peut constater l'usure de l'outil à charioter sans réellement pouvoir l'expliquer. Ce résultat d'échec est alors transmis au superviseur.

T5. Il provoque l'évolution des modèles de surveillance-commande pour les activités liées aux ressources *chariot* et *mandrin*. La fonction décision qui leur est affectée doit déterminer une solution pour remédier à cet échec. L'origine de la défaillance n'ayant pas été trouvée à ce niveau d'abstraction, une propagation de traitement de défaillance est exécutée. Simultanément, une reprise locale est élaborée de manière à changer l'outil. Ces solutions étant établies, la décision émet alors vers le superviseur deux messages :

(a) un compte rendu de décision pour propager le traitement de défaillance au niveau supérieur et signaler qu'une reprise locale est lancée,

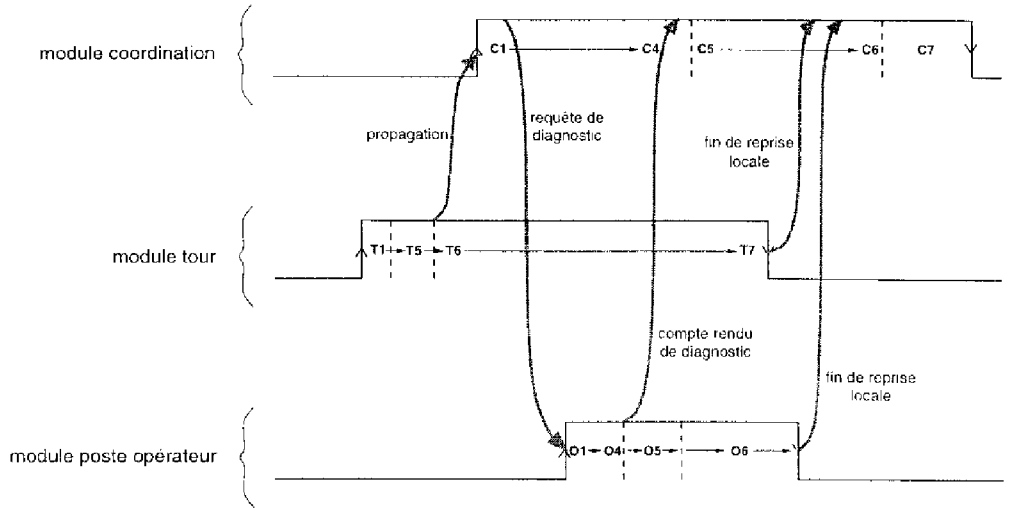


FIG. 2.11 – *vue générale des traitements de défaillance*

(b) un résultat de décision correspondant à la solution de reprise locale à appliquer.

A partir de cette étape, plusieurs traitements de défaillance vont s'exécuter en parallèle dans les niveaux coordination et machine. Une vue synthétique de la simultanéité de ces traitements est donnée figure 2.11 à l'aide d'un chronogramme. Cette représentation est essentielle à la compréhension du fonctionnement global car les différentes étapes de traitement dans le module Tour (T_m), le module Poste Opérateur (O_n) et Coordination (C_p) ne peuvent être discutées que séquentiellement dans le texte qui suit.

T6. Le deuxième message (cf. figure 2.12) provoque l'évolution de l'état du système vers les activités <*chariot,Pièce,Dt,Sv*,**Rp**> (place **143**) et <*mandrin,Dt,Sv*.**Rp**> (place **141**). La fonction reprise applique alors la solution préconisée par la décision. Une reprise similaire ayant déjà été traitée dans le scénario 1, nous ne la redécrirons pas ici.

T7. Le superviseur du module Tour reçoit de l'opérateur le compte rendu de fin de réparation et la fonction reprise s'achève. Cela se traduit par l'émission d'un compte rendu de reprise qui provoque l'évolution de l'état du système. La place **13** est alors marquée par tous les u-plets du réseau : <*chariot,Dt,Sv*>, <*mandrin,Dt,Sv*>, <*chargeur programme,Dt,Sv*>. Cette évolution se traduit également par l'émission d'un compte rendu de fin de reprise locale attestant de la disponibilité du Tour en direction du module coordination.

Plaçons nous maintenant au niveau du module coordination.

2.5.2 Module coordination

C1. Lorsque le module coordination reçoit le message envoyé par le module Tour dont le format est rappelé figure 2.12, l'activité associée à la ressource *tour* est dans

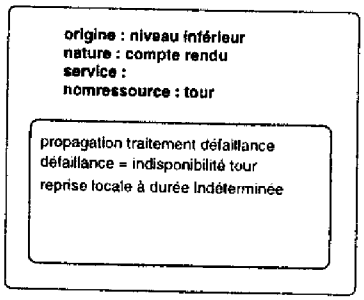


FIG. 2.12 - *propagation du traitement de défaillance*

l'état production optimale <*tour,Dt,Sv,Cd*>. Le message est donc orienté vers les fonctions détection, suivi et commande.

C2. La détection caractérise le message reçu de symptôme de défaillance et l'activité associée à la ressource tour évolue vers l'activité <*tour,Dt,Sv,Cd*,**Dg**> (diagnostic pendant production).

C3. Le diagnostic alors couplé à la ressource Tour consiste à l'utiliser l'historique des opérations et des heuristiques de recherche [Combacau, 1991] [Chaillet, 1995] afin d'établir une liste des opérations précédant l'usinage tour. Celles-ci sont en effet susceptibles d'être à l'origine de la défaillance (Introduction pièce et Transfert). La

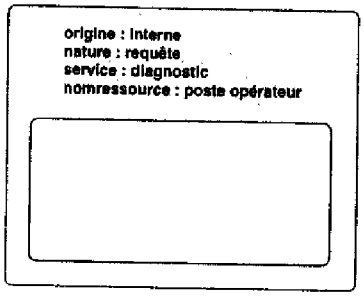


FIG. 2.13 - *requête de diagnostic*

première opération diagnostiquée est celle réalisée par le poste opérateur.

C4. Une requête de diagnostic détaillé est émise faisant passer l'activité courante associée à la ressource "Poste Opérateur" à l'activité <*Poste Opérateur,Dt,Sv*,**Dg**>. Une requête de diagnostic est dirigée vers le niveau inférieur (cf. figure 2.14). Le diagnostic se met alors en attente d'une confirmation ou d'une infirmation de la responsabilité du module "Poste Opérateur" pour la défaillance.

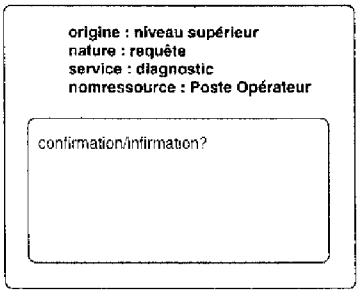


FIG. 2.14 *requête de diagnostic*

2.5.3 Module poste opérateur

O1. Le superviseur reçoit la requête de diagnostic et l'oriente vers le diagnostic.

O2. Celui-ci récupère les informations le concernant dans le système d'information via le centre de gestion en s'aidant de l'historique contenu dans le système d'information [Chaillet, 1995]. Le diagnostic se focalise sur l'analyse de l'opération d'introduction de pièces. L'opération consistant à introduire des pièces doit être diagnostiquée. Cela se traduit par l'émission vers le superviseur d'une requête de diagnostic (cf. figure 2.15) sur la ressource *opérateur*.

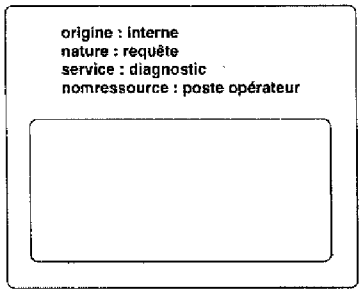


FIG. 2.15 *résultat de diagnostic*

O3. L'activité liée à cette ressource (place **13**) évolue alors vers l'activité <*opérateur,Dt,Sv,***Dg**> (place **77**) représentant ainsi qu'un diagnostic concerne l'opérateur.

O4. Un compte rendu de diagnostic est alors envoyé vers le superviseur qui provoque le passage à l'activité <*opérateur,Dt,Sv,***Dc**> (place **79**). Simultanément, un compte rendu de diagnostic est envoyé vers le niveau coordination confirmant que le module Poste Opérateur est à l'origine de la défaillance.

O5. La suite du traitement consiste à enchaîner un processus de décision <*Poste Opérateur,Dt,Sv,Dc*> au cours duquel il est décidé de préparer une pièce du bon type puis la reprise effective <*Poste Opérateur,Dt,Sv,Rp*>, qui le fait réellement.

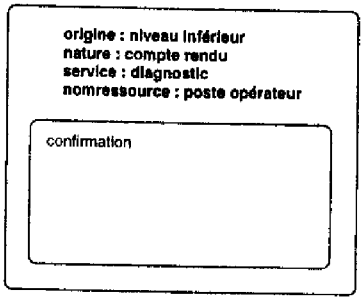


FIG. 2.16 – *compte rendu de diagnostic : confirmation*

O6. A la fin de ce traitement, effectué en mode manuel par l'opérateur, un compte rendu est retourné vers le niveau coordination indiquant le retour en fonctionnement normal du Poste Opérateur (figure 2.17).

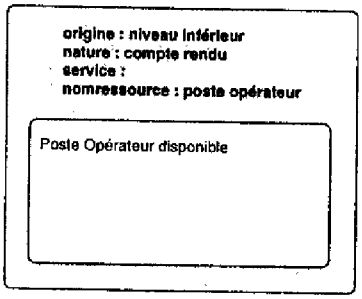


FIG. 2.17 – *compte rendu : retour en fonctionnement optimal du Poste Opérateur*

2.5.4 Module coordination

La description du scénario reprend à la fin de l'attente déjà évoquée au moment où le Poste Opérateur retourne le compte rendu de diagnostic. Le Poste Opérateur étant dans l'activité <*Poste Opérateur,Dt,Sv,Dg*> le message de confirmation de diagnostic est envoyé vers les fonctions associées à la ressource "poste opérateur".

C5. Le diagnostic est sensibilisé par cette information. Le module "Poste Opérateur" étant identifié à l'origine de la défaillance, le diagnostic émet alors le résultat de diagnostic. Le superviseur fait évoluer l'état du système vers l'activité <*poste opérateur,Dt,Sv,***Dc**>.

C6. La décision élabore la solution de reprise qui consiste à mettre la pièce, en cours de traitement sur le tour, au rebut puis à relancer un nouvel usinage dès que les diverses reprises locales seront terminées (maintenance tour et préparation d'une nouvelle pièce par le poste opérateur).

C7. A la réception des deux comptes rendus de reprise locales, la reprise du niveau coordination peut être appliquée.

2.6 Conclusion

L'exemple que nous venons de traiter dans cette partie est basé sur une simulation du comportement du système de Surveillance-Commande que nous avons proposé. Cette simulation s'appuie sur une cellule flexible réelle, celle de l'Ecole Nationale d'Ingénieur de Tarbes (ENIT).

Comme nous avons pu le constater, une seule adaptation ou supposition quelconque a dû être envisagée pour les besoins de l'exemple. Le superviseur que nous avons implanté dans chacun des modules de surveillance-commande est un "véritable outil de traitement de l'information et de communication" [Vidril. 1995]. A ce titre, il autorise une prise en compte de toutes les informations requises durant un cycle de production. Des fins de reprises locales ne sont plus maintenant ignorées, les évolutions intempestives du procédé sont "suivies", des requêtes de diagnostic peuvent désormais déclencher un traitement particulier dans un module de surveillance-commande, etc.

De plus, les exemples que nous avons développés ont permis de montrer que la surveillance n'est plus maintenant un système totalement figé, limité à la traditionnelle séquence "détection, diagnostic, reprise". D'autres séquences, plus complexes comme la production tout de même, ont été mises en évidence.

Bien que notre étude n'ait pas réellement porté sur l'opérateur humain, nous avons montré dans cet exemple que notre approche lui laisse une place importante. L'opérateur peut intervenir en effet à plusieurs niveaux. Dans le module Tour, il a appliqué la solution de reprise imposée par la fonction décision. Dans le module Poste Opérateur, il s'est substitué aux fonctions de commande, diagnostic, décision et reprise. L'opérateur humain n'est pas considéré dans notre approche comme une fonction supplémentaire de surveillance-commande. Il peut se substituer à toute ou partie du système.

Cet exemple présente toutefois un paradoxe. Nous avons fait au tout début de notre présentation l'hypothèse que la commande est parfaite. Or, dans le module Poste Opérateur, l'opérateur qui assure la fonction commande introduit une autre pièce que celle demandée ; il s'agit donc d'une erreur de commande. Il est important de noter que notre approche, bien que non destinée à traiter ce genre de situation, y apporte une réponse satisfaisante.

Enfin, l'hypothèse de l'existence d'un dispositif de chargement automatique des pièces dans le mandrin du tour doit être discutée. En effet, cette hypothèse est impliquée par la structure hiérarchique. Dans la réalité, les mouvements du robot et les opérations de serrage du mandrin sont intimement liées. Le robot place la pièce entre les mors du mandrin et, avant que l'opération "transférer pièce sur tour" ne soit terminée, le serrage du mandrin doit être effectué. Cette synchronisation entre les opérations de transfert et de serrage ne peut pas être résolue dans le contexte hiérarchisé. De telles situations sont courantes dès que l'on s'intéresse à des procédés réels. La résolution des problèmes qui en découlent passe par l'étude de mécanismes de surveillance dans une architecture hétérarchique, voire distribuée. Ceci sort largement du cadre de ces travaux.

Conclusion Générale

Les travaux que nous avons présentés dans ce document traitent de la surveillance et de la commande temps réel des systèmes à événements discrets complexes. Ils font suite à ceux déjà réalisés [Sahraoui, 1987] [Combacau, 1991] [De Bonneval, 1993] [Chaillet, 1995] dans ce domaine au sein du groupe Décision et Conduite des Systèmes de Production du Laboratoire d'Analyse et d'Architecture des Systèmes.

La contribution de nos travaux est l'élaboration d'un nouveau module de surveillance-commande proposant toutes les activités réalisables par un système de surveillance-commande et intégrant les mécanismes nécessaires à la gestion des informations. La prise en compte des besoins de l'entreprise en terme de surveillance et la capacité d'associer à toute information un traitement de surveillance adapté sont les deux principales caractéristiques de ce module.

La première caractéristique est liée à l'utilisation de deux modèles coopérants pour la surveillance-commande.

Le premier modèle, le modèle de référence pour la surveillance-commande, présente trois avantages.

- Premièrement, sa structure de contrôle étant générique, le modèle de référence n'a pas à être redéfini selon l'entreprise considérée. Seules les ressources et les produits dépendent de l'atelier. Le modèle est donc facilement réajustable.
- Deuxièmement, il fournit une aide précieuse à l'utilisateur pour l'élaboration du modèle de la stratégie de surveillance-commande.
- Troisièmement, durant un cycle de production, il garde en permanence une image fidèle de l'état général du système de surveillance-commande.

Le deuxième modèle, le modèle de la stratégie de surveillance-commande, représente les contraintes de surveillance imposées par l'utilisateur. Il en résulte des traitements de surveillance, adaptés à l'entreprise, qui ne décrivent plus obligatoirement la séquence figée "détection, diagnostic, décision et reprise". Un guide méthodologique est fourni dans ce document pour aider l'utilisateur dans sa phase d'élaboration.

La deuxième caractéristique concerne la réceptivité du module. Ce dernier peut en effet prendre en compte toutes les informations qui lui sont transmises. En plus des informations classiques comme les requêtes de commande et comptes rendus d'exécution,

des informations telles que les événements provoqués par des interventions en mode manuel sur le procédé, des fins de reprises locales, des requêtes de diagnostic, etc, sont autant d'informations maintenant acceptées. A chacune de ces informations, le module est en mesure de lui associer le traitement de surveillance-commande adéquat décrit par l'utilisateur.

Comme nous l'avons souligné dans la dernière partie de ce mémoire, l'opérateur trouve naturellement sa place dans notre approche. Il peut en effet se substituer à toute ou partie du système sans que cela nuise à son fonctionnement. L'opérateur peut, tour à tour, être la fonction commande, détection, diagnostic, décision et/ou reprise.

Au terme de ces travaux, plusieurs axes de recherche se dégagent pour envisager, du point de vue des perspectives, de prolonger l'étude menée pendant ces trois ans.

A court terme, une réalisation complète de l'approche de surveillance-commande s'impose et pourrait voir le jour dans les deux années à venir. Un prototype du superviseur en serait la première phase. Une simulation de son fonctionnement, en substituant un opérateur à toutes les fonctions du système, permettrait de le valider. Dans une deuxième phase, l'intégration des fonctions détection et commande peut être envisagée. La réalisation de ces deux fonctions est actuellement bien avancée [Combacau, 1991] [Hammani, 1996]. L'étude des fonctions de diagnostic, décision et reprise développées respectivement dans les travaux de [Chaillet, 1995] et [De Bonneval, 1993] devrait prochainement déboucher sur leur réalisation. L'accomplissement d'un tel travail nous permettrait alors de valider entièrement notre approche sur un atelier réel.

Jusqu'à présent, nous avons supposé que le modèle de la stratégie de surveillance-commande était parfait. Toutefois, si nous levons cette hypothèse, le non-bloquage de notre système n'est plus garanti. Dans une telle situation, le recours au modèle de référence devrait pouvoir aider l'opérateur dans sa phase de reconfiguration. Nous rejoindrions en cela les travaux effectués au LAMIH [Mabrouk, 1996].

Sur un plan plus technique, au moins quatre axes de recherche devraient naître.

- Une première étude doit en effet être menée sur l'interaction de plusieurs stratégies de surveillance-commande associées à la transformation de produits simultanément couplés à la même ressource. La résolution des conflits entre ces stratégies en serait le principal centre d'intérêt.

  Deuxièmement, l'analyse détaillée des implications liées aux ressources partagées doit être faite. Par exemple, comment considérer dans notre approche une ressource partagée par plusieurs activités de surveillance-commande?

- Troisièmement, la mise au clair des conséquences de plusieurs ressources participant à une seule activité de surveillance-commande doit être faite.

- Enfin, nous avons considéré dans notre approche que l'activité numérotée 0 correspondait à la ressource hors service d'un point de vue production. Cependant, cette

activité recouvre plusieurs autres activités dans lesquelles la ressource est *testée*, *pré-chauffée* ou encore *mise en route*. Une étude de toutes ces activités doit être menée pour compléter cette approche. Elle devra s'appuyer sur d'autres travaux comme ceux développés dans le cadre du GEMMA [GEM, 1981].

A plus long terme, il faut envisager la surveillance dans des structures non-entièrement hiérarchiques. Comme nous l'avons souligné dans l'illustration de notre approche sur un procédé réel, la synchronisation des opérations effectuées sur certains éléments du procédé ne peut être résolue que dans un contexte hétérarchique, voire même distribué.

Références

[Abdallah, 1996] Imed Ben Abdallah. "*Méthodes d'Allocation de Ressources dans les Systèmes Flexibles de Production Manufacturière Fondées sur l'Analyse Structurelle des Réseaux de Petri*". Thèse de Doctorat, Ecole Centrale de Paris, Juillet 1996.

[Ahmed *et al.*, 1996] S. Ben Ahmed, M. Moalla, et M. Courvoisier. "*Approche Multimodèles pour la Commande des Ateliers Flexibles*". RAIRO-APII, Journal Européen des Systèmes Automatisés, vol. 30, pages 1201–1232, 1996.

[Andreu, 1996] D. Andreu. "*Commande et Supervision des Procédés Discontinus : une Approche Hybride*". Thèse de Doctorat, Université Paul Sabatier de Toulouse, Novembre 1996.

[Arreguy *et al.*, 1990] D. Arreguy, S. de Carvalho, A. E. K. Sahraoui, A. Serrano, F. Soler, et M. Trilhe. "*A monitoring system for NC machine tool*". Dans IECON'90, pages 622–626, Pacific Grove, Californie, Novembre 1990.

[Atabakhche, 1987] H. Atabakhche. "*Utilisation conjointe de l'intelligence artificielle et des réseaux de Petri: Application au contrôle d'exécution d'un plan de fabrication*". Thèse de doctorat, Université Paul Sabatier, Toulouse, Décembre 1987.

[Ayache, 1995] M. Ayache et A. Flory. "*Etude Comparative des Méthodes de Conception*". Revue des Méthodes, pages 15–19, 1995.

[Bako *et al.*, 1990] B. Bako, R. Valette, et M. Courvoisier. "*A controlled rule-based system interpreter: an application to FMS simulation*". Dans AI, Simulation and Planning in High Autonomy Systems, pages 22–24, Tucson, Arizona, Mars 1990.

[Bako, 1990] B. Bako. "*Mise en œuvre et simulation du niveau coordination de la commande des ateliers flexibles: une approche mixte réseaux de Petri et systèmes de règles*". Thèse de Doctorat, Université Paul Sabatier, Octobre 1990.

[Belkadi, 1989] T. Belkadi. "*Réalisation et intégration de commande de cellule flexible*". Projet de mastère production automatisée, ENIT, Septembre 1989.

[Benzakour, 1985] K. Benzakour. "*SICLOP: Simulateur de commandes logiques et de procédés*". Thèse de doctorat, Université Paul Sabatier, Toulouse, 1985.

[Bernauer, 1996] E. Bernauer. "*Les Réseaux de Neurones et l'Aide au Diagnostic : un Modèle de Neurones Bouclés pour l'Apprentissage de Séquences Temporelles*". Thèse de Doctorat, Université Paul Sabatier, 1996.

[Biland, 1994] P. Biland. "*Modélisation des Modes de Marche d'un Système Automatisé de Production*". Thèse de Doctorat, Ecole Centrale de Nantes, Février 1994.

[Billaut, 1993] J.C. Billaut. "*Prise en Compte des Ressources Multiples et des Temps de Préparation dans les Problèmes d'Ordonnancement en Temps Réel*". Thèse de Doctorat, Université Paul Sabatier, Décembre 1993.

[Bois, 1991] S. Bois. "*Intégration de la gestion des modes de marche dans le pilotage d'un système automatisé de production*". Thèse de Doctorat, Université des Sciences et techniques de Lille Flandres Artois, Novembre 1991.

[Bucci *et al.*, 1995] G. Bucci, M. Campanai, et P. Nesi. "*Tools for Specifying Real-Time Systems*". Real-Time Systems, vol. 8, pages 117-172, 1995.

[Chaillet *et al.*, 1993a] A. Chaillet, M. Combacau, et M. Courvoisier. "*Specification of FMS real-time control based on Petri nets with objects and process failure monitoring*". Dans International Conference on Industrial Electronics, Control, and Instrumentation, IECON'93, Hawaii, USA, Novembre 1993.

[Chaillet *et al.*, 1993b] A. Chaillet, M. Courvoisier, M. Combacau, et A. De Bonneval. "*Merging Petri Nets and Data-Based Models for Control and Monitoring Requirements in FMS*". Octobre 1993.

[Chaillet *et al.*, 1994] A. Chaillet, M. Combacau, E. Zamaï, et M. Courvoisier. "*Tools and Models for Control and Monitoring of Discrete Events Systems*". Dans Congreso de la Asociation Colombiana de Automatica, Cali, Colombi, Novembre 1994.

[Chaillet *et al.*, 1997] A. Chaillet, M. Combacau, M. Courvoisier, A. De Bonneval, A.E.K. Sahraoui, et E. Zamaï. "*Concepts et Outils pour les Systèmes de Production*", chapitre Structures de surveillance pour les systèmes à événements discrets, 1997.

[Chaillet-Subias *et al.*, 1997] A. Chaillet-Subias, E. Zamaï, et M. Combacau. "*Information Flow in a Control and Monitoring Architecture*". Dans IEEE International Symposium on Industrial Electronics, Guimares, Portugal, Juillet 1997.

[Chaillet-Subias, 1996] A. Chaillet-Subias et M. Courvoisier. "*An Architecture and its Mechanism for Real-Time Control and Monitoring of Discrete Events Systems*". Dans IMACS Multiconference, Computational Engineering in Systems Applications, CESA'96, Lille, France, Juillet 1996.

[Chaillet, 1994] A. Chaillet et M. Courvoisier. "*An Information System for Control and Monitoring Purposes in F.M.S.*". Dans IECON'94 Twentieth Annual Conference of the IEEE Industrial Electronics Society, Bologna, Italy, Septembre 1994.

[Chaillet, 1995] A. Chaillet. "*Approche multi modèles pour la commande et la surveillance en temps réel des systèmes à événements discrets*". Thèse de Doctorat, Université Paul Sabatier, Décembre 1995.

[Chapurlat, 1996] V. Chapurlat et F. Prunet. "*Un Modèle pour la Spécification, la Conception et la Validation des Systèmes de Contrôle/Commande répartis*". RAIRO-APII, Journal Européen des Sytèmes Automatisés, vol. 30, pages 25-62, 1996.

[Combacau, 1990] M. Combacau et M. Courvoisier. "*A hierarchical and modular structure for FMS control and monitoring*". Dans 1st International Conference on AI Simulation and Planning in high autonomy systems, Tucson, Arizona, Mars 1990.

[Combacau, 1991] M. Combacau. "*Commande et surveillance des systèmes à événements discrets complexes : application aux ateliers flexibles*". Thèse de Doctorat, Université Paul Sabatier, Décembre 1991.

[Conan, 1996] D. Conan. "*Tolérance aux Fautes par Recouvrement Arrière dans les Systèmes Informatiques Répartis*". Thèse de Doctorat, Université Paris 6, Septembre 1996.

[Cruette, 1991] D. Cruette. "*Méthodologie de conception des systèmes complexes à événements discrets: application à la conception et à la validation hierarchisée de la commande de cellules flexibles de production dans l'industrie manufacturière*". Thèse de doctorat, Université de Lille, Lille, Février 1991.

[de Bonneval *et al.*, 1991] A. de Bonneval, M. Courvoisier, et M. Combacau. "*Real-time diagnosis and recovery in hierarchical F.M.S. control*". Dans IFAC/IMACS International Conference on "Fault Detection, Supervision & Safety for Technical Processes, SAFEPROCESS'91, Baden Baden, Germany, Septembre 1991.

[de Bonneval *et al.*, 1992] A. de Bonneval, M. Combacau, et M. Courvoisier. "*Rôle de l'opérateur humain dans une boucle de surveillance automatique de systèmes à événements discrets*". Dans Canadian Conference and Exhibition on Industrial Automation, Montreal, Canada, Juin 1992.

[de Bonneval, 1993] A. de Bonneval. "*Mécanismes de Reprise dans les Systèmes de Commande à Evénements Discrets*". Thèse de doctorat, Université Paul Sabatier, Toulouse, Septembre 1993.

[Deplanche *et al.*, 1995] A-M. Deplanche, P. Biland, et J-P. Elloy. "*Modes de Marche d'un Système Automatisé, Modélisation Comportementale par Systèmes de Transition*". Dans Journées d'Etude sur la Conception de Systèmes Automatisés de Production à Intelligence Distribuée, SAPID, Paris, France, Mai 1995.

[Feuvrier, 1971] C.V. Feuvrier. "*La simulation des systèmes*". DUNOD, 1971.

[GEM, 1981] "*Le GEMMA, Guide d'Etude des Modes de Marches et d'Arrêts*". Adepa, 17 rue Perier, 92120 Montrouge, 1981.

[GRP, 1988] Groupe de Réflexion Temps Réel du CNRS GRP. "*Le Temps Réel*". Technique et Sciences Informatiques (TSI), vol. 7, pages 493–500, 1988.

[Hammami *et al.*, 1995] S. Hammami, I. Tnazefti, M. Moala, et A. Chaillet. "*Designing control and diagnosis for flexible manufacturing systems as a multi-agent system using blackboard and object Petri nets*". Dans 4th INRIA/IEEE Symposium on Emerging Technologies and Factory Automation, ETFA'95, Paris, France, Octobre 1995.

[Hammami *et al.*, 1996] S. Hammami, I. Tnazefti, N. Doggaz, et M. Moalla. "*Approche Multi-Agents pour la Conception de la Commande-Surveillance des Systèmes Flexibles de Production*". Dans Computer Integrated Manufacturing and Automation Technology, Grenoble, France, Mai 1996.

[Hetreux, 1996] G. Hetreux. "*Structures de Décision Multi-Niveaux pour la Planification de la Production : Robustesse et Cohérence des Décisions*". Thèse de Doctorat, Université Paul Sabatier, Décembre 1996.

[Holloway, 1991] L. E. Holloway et B. H. Krogh. "*Monitoring behavioral evolution for on-line fault detection*". Dans IFAC/IMACS International Conference on "Fault Detection, Supervision & Safety for Technical Processes, SAFEPROCESS'91, Baden Baden, Germany, Septembre 1991.

[Holoway, 1990] L. E. Holoway et B. H. Krogh. "*Fault detection and diagnosis in Manufacturing Systems: a behavioral model approach*". Dans IEEE International Conference on Computer Integrated Manufacturing, Mai 1990.

[Huvenoit, 1994] B. Huvenoit. "*De la conception à l'implémentation de la commande modulaire et hiérarchisée des systèmes flexibles de production manufacturière*". Thèse de Doctorat, Université de Lille 1, Octobre 1994.

[J. Laffont, 1997] R. Ortalo J. Laffont. "*Editeur de politiques de sécurité utilisant le formalisme des logiques modales*". Février 1997.

[Jones, 1989] A. Jones. "*A multi-layer/multi-level control architecture for Computer Integrated Manufacturing Systems*". Dans IEEE Int. Symp. on Intelligent Control, Albany, NY, Septembre 1989.

[Kermad, 1996] L. Kermad. "*Contribution à la Supervision et à la Gestion des Modes et des Configurations des Systèmes Flexibles de Production Manufacturière*". Thèse de Doctorat, Université des Sciences et Technologie de Lille, Janvier 1996.

[Khattabi, 1993] S. El Khattabi. "*Intégration de la Surveillance de Bas Niveau dans la Conception des Systèmes à Evénements Discrets : Application aux Systèmes de Production Flexibles*". Thèse de Doctorat, Université des Sciences et Technologies de Lille, Septembre 1993.

[Kunzle *et al.*, 1994] L-A. Kunzle, B. Pradin-Chezalviel, F. Girault, et R. Valette. "*Synthesis of monitoring functions based on a formal specification*". Dans European Workshop on Integrated Manufacturing Systems Engineering (IMSE'94), Grenoble, France, Décembre 1994.

[Lepage *et al.*, 1991] F. Lepage, F. Afilal, P. Antoine, E. Bajic, J-Y. Bron, et T. Divoux. "*Les Réseaux Locaux Industriels*". Traité des Nouvelles Technologies, série Automatique. Hermes édition, 1991.

[Lhoste, 1991] P. Lhoste. "*Surveillance des M.S.A.P.: les Atouts de la Modélisation de Comportement, journée surveillance du Pôle SED (GT2) du GR Automatique*". Février 1991.

[Lopez, 1991] P. Lopez. "*Analyse énergétique pour l'ordonnancement de tâches sous contraintes de temps et de ressources*". Thèse de Doctorat. Université Paul Sabatier. Toulouse, Septembre 1991.

[Mabrouk *et al.*, 1996] M. Mabrouk, Y. Sallez, et R. Soenen. "*Toward a Tool for Aiding Reconfiguration of Production Automated Systems*". Dans IMACS Multiconference. Computational Engineering in Systems Applications, CESA'96, Lille, France, Juillet 1996.

[Mabrouk, 1996] M. Mabrouk. "*Proposition d'une Méthode et d'un Outil d'Aide à la Reconfiguration des Systèmes Automatisés de Production*". Thèse de Doctorat, Université de Valenciennes et du Hainaut-Cambresis, Mai 1996.

[Marty, 1994] J.C. Marty. "*Utilisation des Statecharts pour une Spécification Structurée du Contrôle des Cellules Flexibles*". Thèse de Doctorat, Institut National des Sciences Appliquées de Toulouse, Octobre 1994.

[Niel *et al.*, 1994] E. Niel, N. Resg, J-M. Diosse, et J. Favrel. "*Système de Commande-Supervision Orientée Sécurité Opérationnelle*". Automatique Productique Informatique Industrielle, vol. 28, pages 539–549, 1994.

[O'Grady *et al.*, 1994] P-J. O'Grady, Y. Kim, et R-E. Young. "*A Hierarchical Approach to Concurrent Engineering Systems*". Computer Integrated Manufacturing, vol. 7, pages 152–162, 1994.

[Paludetto *et al.*, 1990] M. Paludetto, R. Valette, et M. Courvoisier. "*Génération de code ADA à partir d'une approche orientée objet HOOD/Réseau de Petri*". Dans Journées internationales "Le génie logiciel et ses applications", Toulouse, France, Décembre 1990.

[Parayre, 1992] T. Parayre. "*Le MESAP : vers une Méthodologie d'Exploitation des Systèmes Automatisés de Production*". Thèse de Doctorat, Université de Valenciennes et du Hainaut Cambresis, 1992.

[Rezg, 1996] N. Rezg. "*Contribution à la Sécurité Opérationnelle des Systèmes : Mise en œuvre d'une Structure de Surveillance Basée sur les Réseaux de Petri*". Thèse de Doctorat, Institut National des Sciences Appliquées de Lyon, 1996.

[Roche, 1988] A. Roche et M. Jubin. "*Les Cellules et les Ilots Flexibles d'Usinage*". Hermes, Technologies de Pointe, 1988.

[Ross, 1977] D-T. Ross. "*Structured Analysis (SA): A Langage for Communication Ideas*". IEEE TRansaction on Software Engineering, vol. 3, pages 16–34, 1977.

[Sahraoui, 1987] A. E. K. Sahraoui. "*Sur la surveillance des ateliers flexibles*". Thèse de Doctorat, Université Paul Sabatier, Toulouse, Octobre 1987.

[Sibertin-Blanc, 1988] C. Sibertin-Blanc. "*Le modèle de donnée Objet comme formalisme de modélisation de Base de Donnés*". Revue MDB, AFCET, vol. 9, Juin 1988.

[Tawegoum *et al.*, 1994] R. Tawegoum, E. Castelain, et J-C. Gentina. "*Real Time Piloting of Flexible Manufacturing Systems*". European Journal of Operational Research, vol. 78, Number 2, pages 252–261, 1994.

[Tawegoum, 1995] R. Tawegoum. "*Contrôle Temps Réel du Déroulement des Opérations dans les Systèmes de Production Flexibles*". Thèse de Doctorat, Université des Sciences et Technologies de Lille, Avril 1995.

[Technologie, 1989] I.G.L. Technologie. "*SADT, un Langage pour Communiquer*". Eyrolles édition, 1989.

[Toguyeni *et al.*, 1996] A-K-A. Toguyeni, S. El Kahttabi, et E. Craye. "*Functional and/or Structural Approach for the Supervision of Flexible Manufacturing Systems*". Dans IMACS Multiconference, Computational Engineering in Systems Applications, CESA'96, Lille, France, Juillet 1996.

[Toguyeni, 1992] A-K-A. Toguyeni. "*Surveillance et Diagnostic en Ligne dans les Systèmes Flexibles de l'Industrie Manufacturière*". Thèse de Doctorat, Université de Lille I, Lille, Novembre 1992.

[Valette *et al.*, 1988] R. Valette, D. Dubois, et J. Cardoso. "*Représentation de l'état d'un atelier de fabrication avec prise en compte des incidents*". Dans Cougrès AFCET Automatique 1988, Grenoble, France, Octobre 1988.

[Valette, 1994] R. Valette et L-A. Künzle. "*Réseaux de Petri pour la Détection et le Diagnostic*". Dans Journées Nationales : Sûreté, Surveillance, Supervision, Paris, France, Novembre 1994.

[Valette, 1995] R. Valette. "*Les Réseaux de Petri*". Janvier 1995.

[Verlinde, 1989] C. Verlinde. "*Contribution à l'étude des architectures de systèmes automatisés*". Thèse de Doctorat, Institut National Polytechnique de Lorraine, NANCY, 1989.

[Vidril, 1995] M. Vidril. "*Pilotage Hiérarchisé et Réparti des Systèmes Flexibles de Production dans l'Industrie Manufacturière*". Thèse de Doctorat, Université des Sciences et Technologies de Lille, 1995.

[Zamaï *et al.*, 1997] E. Zamaï, A. Chaillet-Subias, M. Combacau, et A. de Bonneval. "*A Hierarchical Structure for Control Of Discrete Events Systems and Monitoring of Process Failures*". Studies in Informatics and Control, vol. 6, Number 1, pages 7–15, 1997.

[Zamaï *et al.*, 1998] E. Zamaï, M. Combacau, et A. Chaillet-Subias. "*Models for Monitoring and Control of Discrete Events Systems*". Soumis dans 9th Symposium on Information Control in Manufacturing, Nancy-Metz, France, Juin 1998.

[Zamaï, 1996] E. Zamaï et M. Combacau. "*Inter Levels Communication Requirements for Hierarchical Control and Monitoring Structures*". Dans IEEE International Conference on Systems, Man and Cybernetics, Lille, France, Juillet 1996.

[Zamaï, 1994] E. Zamaï. "*Commande et surveillance des systèmes à événements discrets complexes: utilisation d'un modèle du procédé*". Juin 1994. Université Paul Sabatier, Rapport de DEA.

Annexes

Spécification complète d'un nœud de surveillance-commande

Flot d'informations et fonctionnalités pour un nœud

Données en entrée

- *requêtes de commande*, données émises par le niveau supérieur et reçues par le nœud de surveillance-commande considéré (cf. figure 2.1) de manière à ce que celui-ci assure le service de commande demandé. Ce mécanisme correspond à un classique affinement d'une requête de commande. Un exemple tiré de [Combacau, 1991] est donné dans la figure 2.18.

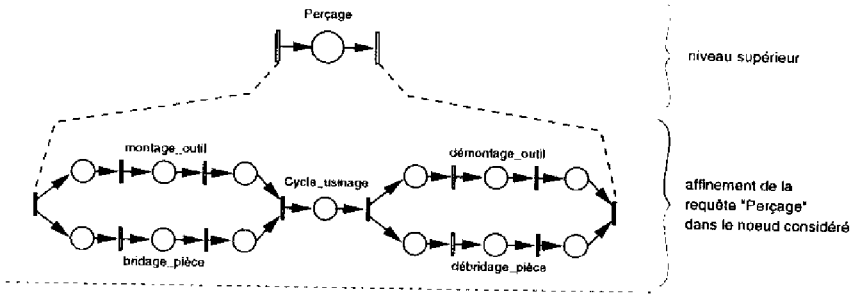


FIG. 2.18 – *affinement d'une requête de commande*

- *requêtes de diagnostic*, le nœud de surveillance-commande considéré reçoit l'ordre de lancer un diagnostic [Chaillet, 1995].
- *requêtes de reprise*, ces données s'apparentent fortement aux requêtes de commande. Toutefois, le contexte dans lequel elles sont reçues et satisfaites est totalement différent puisque directement lié à un traitement de défaillance.
- *données opérateur*, ce sont les données envoyées par l'opérateur lorsque le système de surveillance-commande du nœud considéré a fait appel à lui [de Bonneval, 1993]. C'est par exemple le cas lors d'un diagnostic au cours duquel l'état de l'outil (bon état, détérioré, hors service) est demandé à l'opérateur (les capteurs ne permettant pas d'accéder à ce type d'information).

- *informations centre de gestion*, données renvoyées via le centre de gestion au système de surveillance-commande du nœud qui en a fait la demande (par exemple, lors d'un diagnostic détaillé dans le but de récupérer les informations pertinentes).

  comptes rendus (d'exécution, propagation de défaillance, évolutions incontrôlées, etc.), ils témoignent de l'état réel du sous-système contrôlé.

Données en sortie

requêtes de commande: après avoir reçu une requête de commande (donnée en entrée) du niveau supérieur, le nœud de surveillance-commande demande à son tour l'exécution d'un ensemble d'autres requêtes. Il s'agit du mécanisme classique d'affinement de requête de commande déjà introduit plus haut (cf. figure 2.18).

- *requêtes de diagnostic détaillé*, lorsqu'un mécanisme de propagation ascendante de traitement de défaillance est envisagé (suite à une propagation de défaillance sur une autre activité de commande), la conclusion qui est élaborée au niveau de connaissance adéquat est souvent grossière et peut manquer de précision [Chaillet, 1995]. Appliquer une reprise à ce niveau de connaissance peut donc être brutal par rapport à la défaillance détectée au niveau local. Une propagation descendante de diagnostic détaillé est alors envisagée et matérialisée sous forme d'une requête d'affinement de diagnostic vers les modules candidats [Chaillet-Subias, 1996], c'est à dire ceux pouvant être à l'origine de la défaillance.

  requêtes de reprise, émission d'une requête de reprise pour résoudre le problème posé par la défaillance. En effet, tant que le système est en fonctionnement normal (production optimale), il suit une trajectoire particulière. La défaillance se traduit par une déviation du système de commande. La requête de reprise a alors pour rôle d'imposer un retour vers la trajectoire d'origine.

- *requêtes opérateur*, le système de surveillance-commande du nœud a besoin d'une aide extérieure. Pour cela, il lance une requête vers l'opérateur.

- *requêtes centre de gestion*: ces requêtes sont de cinq types [Chaillet, 1995]:

  1. requête de Recherche de l'activité de commande responsable de la défaillance,
  2. requête de Vérification de cette recherche,
  3. requête de Récupération de données,
  4. requête de Mise à jour de données,
  5. requête de Détermination de chemin en vue d'un diagnostic détaillé dans un nœud cible.

  comptes rendus (d'exécution, propagation de défaillance, évolutions incontrôlées, etc.) vers les niveaux supérieurs,

Supports

- *modèle de commande*, il s'agit de la séquence opératoire conçue pour réaliser le service de commande demandé par le niveau supérieur. Cette spécification a été élaborée en prenant en compte les possibilités offertes par le procédé (contraintes modélisées dans le modèle du procédé), le produit à transformer et enfin la "politique" de production de l'entreprise.

- *modèle du procédé*: ce support a la même vocation que le précédent, mais il est plutôt dédié à la commande. Néanmoins, son rôle sera également très important lors d'un traitement de défaillance car il représente l'état réel dans lequel se trouve le procédé. Ceci permettra par exemple, connaissant cet état, de pouvoir appliquer des séquences de reprise.

- *modèle de Surveillance-Commande*: il représente l'ensemble des activités de surveillance-commande offertes à l'utilisateur ainsi que les contraintes qui les lient (séquencements obligatoires, exclusions mutuelles, etc.). Ce modèle décrit l'ensemble des "états utilisables" §1.3.1.

- *stratégie de surveillance-commande*: décrit la séquence d'activités de surveillance-commande respectant les contraintes imposées par la fabrication du produit, les contraintes de surveillance décrites dans le modèle de surveillance-commande et enfin la politique de l'entreprise. Ceci correspond aux "états autorisés" §1.3.1.

Interfaçage des fonctionnalités Superviser et Surveiller-Commander

Interface associée à "Superviser"

- Données en entrée
  - les **comptes rendus** issus du niveau inférieur peuvent exprimer d'une part la bonne ou la mauvaise exécution du service demandé, ou bien signaler une évolution intempestive alors que rien n'a été demandé, ou encore un retour en fonctionnement normal du sous-système contrôlé.
  - **requêtes opérateur**,
  - **requêtes de commande**,
  - **requêtes de diagnostic**,
  - **requêtes de traitement d'urgence**,
  - **requêtes de reprise**,
  - **données traitées** par la fonctionnalité "Surveiller-Commander". Il s'agit des résultats fournis par les différentes fonctionnalités contenues dans "Surveiller-Commander".
- Flot de contrôle: ON-OFF. Cette donnée de contrôle permet d'activer ou de désactiver la fonctionnalité Superviser et donc tous les éléments quelle contient.

Données en sortie

- **comptes rendus** émis vers le niveau supérieur,
- **requêtes de commande**,
- **requêtes de diagnostic**,
- **requêtes de reprise**,
- le **contexte** est fourni à cette fonctionnalité en prévision d'un apport d'information sur le type de ressources et de produits sur lesquels le système de surveillance-commande travaille. Ces informations, nous le verrons plus loin, définirons parfois un contexte susceptible d'influencer le comportement des fonctionnalités contenues dans la fonctionnalité "Surveiller-Commander".

  fonction(s) de Surveillance-Commande sélectionnée(s), indiquant quelles sont les fonctions de surveillance-commande qui doivent prendre en compte les **données à traiter**,
- les **données à traiter** sont les mêmes données que celles reçues par la fonctionnalité "Superviser", c'est à dire les comptes rendus, les requêtes opérateur, les requêtes de commande, les requêtes de reprise, les requêtes de traitement d'urgence et enfin celles d'affinement de diagnostic. Ces données sont ensuite transmises vers la (les) fonction(s) de surveillance-commande adéquate(s) qui sont regroupées dans la fonctionnalité "Surveiller-Commander".

Supports

- **stratégie de surveillance-commande**,
- **modèle de surveillance-commande.**

Interface associée à "Surveiller-Commander"

- Données en entrée
  - **les données opérateur** donnent aux fonctionnalités contenues dans Surveiller-Commander les informations qu'elles ont demandé par l'intermédiaire de leurs fonctionnalités "Dialoguer" avec l'opérateur.
  - **données centre de gestion**,
  - **données à traiter**,
    activité de surveillance-commande en cours.

  Flot de contrôle
  - **ON-OFF**,
  - **fonction(s) de Surveillance-Commande sélectionnée(s)**,
- Données en sortie
  - **requêtes opérateur**,
  - **requêtes centre de gestion**,
  - **données traitées**,

- Supports
  - **modèle de commande**,
  - **modèle du procédé**.

Diagramme A01 : Superviser

Fonctionnalités à réaliser

Interfaçage des fonctionnalités de Superviser

Interface associée à "Acquérir"

- Données en entrée
  - les **données traitées** sont les données générées par les fonctions de surveillance-commande contenues dans la fonctionnalité "Surveiller-Commander". On y retrouvera par exemple, les symptômes de défaillances (résultat de la détection), les conclusions de diagnostic, les fins d'activités de commande, etc,
  - les **comptes rendus** sont les données émises par le niveau supérieur,
  - les **requêtes/CR opérateur** doivent obligatoirement passer par la fonctionnalité "Superviser" car l'opérateur, n'ayant pas une vision suffisamment précise de l'état du système, risque de prendre des décisions risquant de transgresser des contraintes importantes exprimées dans le modèle de surveillance-commande.
  - les données telles que les **requêtes de commande**, les **requêtes de reprise**, les **requêtes d'affinement de diagnostic** et les **requêtes de traitement d'urgence**,
- Flot de contrôle : néant,
- Données en sortie
  - les **données reçues** correspondent à une des données d'entrée de la fonctionnalité "Acquérir",
  - dès qu'une donnée d'entrée est prise en compte, la fonctionnalité "Acquérir" envoie la donnée **choix destination** de manière à activer la fonctionnalité "Orienter" qui se chargera d'orienter les "données reçues" vers la fonctionnalité correspondante.

Interface associée à "Orienter"

- Données en entrée
  - les **données reçues** transmises par la fonctionnalité "Acquérir" doivent être orientées soit vers le niveau inférieur en tant que requêtes diverses, soit vers le niveau supérieur en tant que comptes rendus, soit enfin vers la fonctionnalité "Surveiller-Commander" qui va se charger de les traiter,

- Flot de contrôle : la donnée **choix destination** permettra de déclencher la fonctionnalité "Orienter" la "donnée reçue".
- Données en sortie

  si les "données reçues" correspondent à un affinement de requêtes diverses demandées par nœud de surveillance-commande alors elles seront transformées soit en **requêtes de commande**, soit en **requêtes de reprise**, soit en **requêtes de diagnostic** ou encore en **requêtes de traitement d'urgence**,

  au contraire, si ces "données reçues" sont des comptes rendus d'exécution générés par le nœud de surveillance-commande, alors "données reçues" sont transformées en **compte rendu** pour le niveau supérieur,

  sinon, les "données reçues" ne correspondant à aucun de ces cas, la fonctionnalité "Orienter" est en présence d'une information issue d'un autre nœud de la hiérarchie. Elle sera alors transformée en **données à traiter** qui devront être traitées par une ou plusieurs des fonctions de surveillance-commande fournies par le nœud dans la fonctionnalité "Surveiller-Commander".

  là où les fonctions adéquates de surveillance-commande seront alors sélectionnées par la fonctionnalité "Orienter" et imposées à la fonctionnalité "Surveiller-Commander" grâce à la donnée **fonction(s) de Surveillance-Commande**.
- Supports : l'**activité de surveillance-commande en cours d'exécution** est nécessaire à cette fonctionnalité. En effet, si nous nous référons au §1.3.2 page 43 un processus est constitué d'une suite logique d'activités. Chacune de ces activités est composée d'une association d'éléments, en particulier les fonctions de surveillance-commande (commande, diagnostic, détection, etc.).

Interface associée à "Représenter le traitement de Surveillance-Commande"

- Données en entrée
  - la seule donnée d'entrée requise par cette fonctionnalité est la **donnée couplage/découplage**. Cette donnée correspond aux événements évoqués précédemment. Ces événements provoquent des changements d'états ou plutôt des changements d'activités. Le processus ainsi décrit étant celui modélisé dans la "stratégie de commande".
- Données en sortie : **activité en cours** d'exécution dans la stratégie de surveillance-commande,
- Supports
  - pour déclencher un traitement de surveillance-commande il est nécessaire d'en posséder un modèle. C'est pourquoi, la **stratégie de surveillance-commande** est fournie à la fonctionnalité "Représenter le Traitement de Surveillance-Commande".

- les contraintes non opérationnelles sont rejetées dans le **modèle de Surveillance-Commande**, ce modèle doit être également fourni. Le lancement d'une activité modélisée dans la stratégie de surveillance-commande est conditionné par la vérification de ces contraintes.

Interfaçage des fonctionnalités de Surveiller-Commander

Interface associée à "Détecter"

- Données en entrée
  - **comptes rendus ou requêtes de commande ou requêtes de reprise** qui sont envoyés par la fonctionnalité "Superviser" pour y être caractérisés (symptômes de défaillance ou commandes erronées).
- Flot de contrôle :
  - **détecter sélect.**. La fonctionnalité "Détecter" a été sélectionnée pour caractériser les "données à traiter",
  - **ON-OFF**, active/désactive la fonctionnalité "Détecter".
- Données en sortie
  - un **symptôme de défaillance** sera renvoyé vers la fonctionnalité "Superviser" si la donnée transmise ne correspond pas à la donnée prévue. Dans le cas contraire, aucune donnée n'est émise.
- Supports
  "Détecter" consiste à déceler toute évolution anormale par rapport à ce qui est prévu. Il est donc nécessaire de disposer d'une représentation de ce que l'on qualifie de "normal". Pour cette raison, la fonctionnalité "Détecter" disposera de deux modèles du fonctionnement normal d'une représentation de l'état courant du système de surveillance-commande :
  - le **modèle de commande** ou modèle de commande représentant le comportement attendu du procédé soumis à des consignes de commande,
  - le **modèle du procédé** représentant l'ensemble des états utilisables du procédé. Ce modèle sert ici à détecter d'une part les erreurs de commande (réaliser le service perçage alors que la perceuse n'est pas disponible) ainsi que la transgression de contraintes structurelles graves comme le partage de certaines ressources (si un carrefour pouvant accueillir un seul chariot est occupé par deux chariots, cela implique nécessairement une collision entre ces deux chariots),
  - l'**activité en cours**. Par exemple, si les fonctionnalités "commander" et "reprendre" ne sont pas associées à l'activité en cours, alors le sous-système ne reçoit pas de consigne. Il ne doit donc pas évoluer. Si l'occurrence d'un compte rendu d'évolution émanant de ce sous-système démontre le contraire, alors la détection caractérisera une défaillance.

Interface associée à "Commander"

- Données en entrée
  - **requêtes de commande ou comptes rendus d'exécution.**

  Flot de contrôle

  commander sélect. permettra la prise en compte de la donnée "données à traiter",
  - la fonctionnalité "Commander" est activée ou désactivée selon l'information portée par **ON-OFF**,
- Données en sortie
  - **requête de commande ou C.R. d'exécution.** Les requêtes de commande correspondent à l'affinement d'une requête de commande issue du niveau supérieur. Les C.R. d'exécution attestent de la bonne ou mauvaise exécution du service rendu par le nœud pour le niveau supérieur.

  Supports

  modèle de commande : comme nous l'avons vu dans la partie I de ce mémoire, la commande nécessite l'utilisation d'un modèle de commande qui spécifie la façon de réaliser le service de commande demandé par le niveau supérieur,
  - le **modèle du procédé** : représente l'ensemble des contraintes qui doivent être vérifiées pour exécuter une activité de commande. Ces contraintes, nous l'avons déjà vu, sont de type exclusions mutuelles, disponibilité des ressources, capacité des ressources, etc.

Interface associée à "Diagnostiquer"

- Données en entrée

  symptôme de défaillance : dans le cas le plus fréquent, la fonctionnalité "diagnostiquer" devra déterminer l'origine d'une défaillance décelée dans le nœud de surveillance-commande considéré (cf. figure 2.1). C'est la raison pour laquelle, un symptôme de défaillance lui est fourni en entrée.
  - **requêtes de diagnostic** : la fonctionnalité "Diagnostiquer" doit pouvoir assurer du diagnostic sur demande d'un niveau supérieur. Ce besoin a été mis en évidence dans les travaux de [Chaillet, 1995] pour préciser des résultats de diagnostic de haut niveau, parfois grossiers.
  - **données centre de gestion** : durant son activité, le diagnostic devra faire appel au système d'information pour rapatrier des données permettant de déterminer l'origine exacte de la défaillance. Pour cette raison, la fonctionnalité "Diagnostiquer" traitera des données émanant du centre de gestion.
  - **données opérateur** : lorsque le diagnostic n'est pas en mesure d'achever sa tâche par manque d'information (sur l'état réel du procédé par insuffisance de capteurs par exemple), il doit faire appel à l'opérateur qui lui fournira ces informations.

- Flot de contrôle
  - **diagnostiquer sélect.** entraîne le traitement de la donnée "données à traiter" par la fonctionnalité "Diagnostiquer",
  - l'activation ou désactivation de cette fonctionnalité est assurée par la donnée de contrôle **ON-OFF**,
- Données en sortie
  Nous venons de le voir, la fonctionnalité "Diagnostiquer" peut faire appel à l'opérateur et au système d'information via le centre de gestion. Il est donc évident de retrouver en sortie de cette fonctionnalité les deux données suivantes :
  - **requêtes centre de gestion**,
  - **requêtes opérateur**.
- Supports
  - le **modèle du procédé** est une base de connaissance sur l'état réel du procédé. Cela constitue une information importante pour le diagnostic.
  - le **modèle de commande** est également une base d'information importante pour le diagnostic, puisqu'elle représente l'état de la commande au moment de la défaillance,
  - l'**activité en cours** est une information utile au diagnostic pour analyser la situation dans laquelle la défaillance a été détectée.

Interface associée à "Décider"

- Données en entrée
  - nous trouverons bien entendu en donnée d'entrée la **conclusion du diagnostic**, principale donnée permettant de prendre la décision qui s'impose. Toutefois, pour prendre cette décision (propagation, attente, élaboration d'une séquence de reprise), la fonctionnalité "Décider" aura besoin à la fois des données contenues dans le système d'information mais également de l'avis de l'opérateur. Pour ces raisons, les deux données suivantes seront utiles.
  - **données centre de gestion**,
  - **données opérateur**.
- Flot de contrôle
  - **décider sélect.** impose le traitement de la donnée "données à traiter" à la fonctionnalité "Décider",
  - la fonctionnalité "Décider" est activée ou désactivée par la donnée de contrôle **ON-OFF**,
- Données en sortie
  Le rôle principal de cette fonctionnalité est de produire une
  - **prise de décision**. Toutefois, pour prendre cette décision, les appels à un opérateur et/ou au système d'information seront parfois utiles,

requêtes opérateur,

- **requêtes centre de gestion.**

- Supports
  - le **modèle du procédé** représentant l'état réel dans lequel se trouve le procédé au moment de la prise de décision permettra d'élaborer une séquence de reprise cohérente. En effet, sans cette représentation le point de départ de la séquence de reprise reste inconnu.

    le **modèle de commande** sera utilisée de manière à connaître l'état vers lequel il faut se diriger pour reprendre la phase de production.
  - **activité en cours.**

Interface associée à "Reprendre"

Données en entrée

- la fonctionnalité "Reprendre" doit assurer l'exécution d'une **requête de reprise** émanant du niveau supérieur, soit mettre en œuvre la **solution à appliquer** élaborée par la fonctionnalité "Décider".

Flot de contrôle

reprendre sélect. impose de traiter la donnée "données à traiter" avec la fonctionnalité "Reprendre",

- cette fonctionnalité est activée ou désactivée selon la donnée **ON-OFF**,

Données en sortie

des **requêtes de reprise** correspond à l'affinement d'un service de reprise ou des **C.R. d'exécution de reprise** (fin normale, anormale),

- des requêtes de **mise à jour** de la base de données contenue dans le système d'information.

- Supports
  - le **modèle du procédé**,
  - le **modèle de commande**.

Interface associée à "Appliquer Traitement d'Urgence"

- Données en entrée
  - **requête d'urgence** envoyée par le niveau supérieur.

  Flot de contrôle
  - **urgence sélect**ionnée pour traiter la donnée "donnée à traiter",
  - **ON-OFF** active ou désactive la fonctionnalité "Appliquer Traitement d'Urgence" ,
- Données en sortie

  mise à jour de la base de données contenue dans le système d'information,

- **requêtes d'urgence ou C.R. d'exécution d'urgence.**

- Supports
  - **modèle du procédé.** Nous offrons ce support à cette fonctionnalité de manière à ce qu'elle puisse constater s'il est possible ou non d'appliquer une procédure d'urgence dans l'état actuel du procédé. Par exemple, il est impensable d'arrêter un réacteur nucléaire alors que les barres sont en position haute! Si tel est le cas, il faut envisager préalablement de rejoindre un état dans lequel l'arrêt du réacteur est possible. Nous n'avons pas pour l'instant spécifié cette fonctionnalité.

Interface associée à "Suivre"

- Données en entrée
  - **comptes rendus** émanant du niveau inférieur traduisant ou non une évolution possible du modèle du procédé.
- Flot de contrôle
  - **suivre sélectionnée** pour prendre en compte la donnée "donnée à traiter",
  - cette fonctionnalité est activée ou désactivée selon la valeur de la donnée **ON-OFF**,
- Données en sortie
  - **mise à jour** de la base de connaissance contenue dans le système d'information et du modèle du procédé.
- Supports
  - **modèle du procédé.**

Liste des processus élémentaires de surveillance-commande

Nous présentons dans ce chapitre la liste des enchaînements d'une activité de surveillance-commande à une autre. L'ensemble des processus décrits permet d'élaborer le **modèle de référence** de la surveillance-commande.

Pour obtenir cette liste, nous avons fait subir à chacune des 31 activités de surveillance-commande tous les couplages et découplages mis en évidence à la page 72. Pris séparément, ces couplages ou découplages apparaissent dans la liste ci-dessous sous forme de nombres. Un nombre positif (respectivement négatif) correspond au couplage (respectivement découplage) de l'élément *R, P, Dt, Sv, Cd, Dg, Dc, Rp* ou *Ug*. Un 0 indique que l'élément considéré est ni couplé ni découplé. La somme de ces couplages et/ou découplages forme l'événement qui provoque le passage d'une activité à l'autre et qui permet ainsi de décrire les différents processus de surveillance-commande.

Dans tout ce qui suit, les activités ne sont repérées que par leur numéro.

```
Activité suivante (As) ========================================\
                                                                |
Activité courante (Ac) ================================\       |
                                                        |       |
Evénement couplage/découplage (Ev) ============\         |       |
                                              |         |       |
     R   P   Dt   Sv   Cd   Dg   Dc    Rp    Ug    |   |       |
     |   |   |    |    |    |    |     |     |     |   |       |
%%%% | % | % | %% | %% | %% | %% | %%% | %%% | %%% | %%%% | %% | %%%
%%%% | % | % | %% | %% | %% | %% | %%% | %%% | %%% | %%%% | %% | %%%
%%%% V % V % V %% V %% V %% V %% V %%% V %%% V %%% V %%%% V %% V %%%

     1   0   4    8    0    0    0     0     0 =   13 +   0 -> 13
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 13 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

  -1   0  -4   -8    0    0    0     0     0 =  -13 +  13 -> 0
   0   2   0    0   16    0    0     0     0 =   18 +  13 -> 31
   0   2   0    0    0    0    0     0     0 =    2 +  13 -> 15
   0   0  -4    0    0   32    0     0   256 =  284 +  13 -> 297
   0   0  -4    0    0    0   64     0   256 =  316 +  13 -> 329
   0   0   0    0   16   32    0     0     0 =   48 +  13 -> 61
   0   0   0    0   16    0    0     0     0 =   16 +  13 -> 29
   0   0   0    0    0   32    0     0     0 =   32 +  13 -> 45
   0   0   0    0    0    0    0   128     0 =  128 +  13 -> 141

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 15 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0    0    0    0     0     0 =   -2 +  15 -> 13
   0   0  -4    0    0   32    0     0   256 =  284 +  15 -> 299
   0   0  -4    0    0    0   64     0   256 =  316 +  15 -> 331
   0   0   0    0   16   32    0     0     0 =   48 +  15 -> 63
   0   0   0    0   16    0    0     0     0 =   16 +  15 -> 31
   0   0   0    0    0   32    0     0     0 =   32 +  15 -> 47
   0   0   0    0    0    0    0   128     0 =  128 +  15 -> 143

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 29 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0   2   0    0    0    0    0     0     0 =    2 +  29 -> 31
   0   0  -4    0    0   32    0     0   256 =  284 +  29 -> 313
   0   0  -4    0    0    0   64     0   256 =  316 +  29 -> 345
   0   0   0    0  -16   32    0     0     0 =   16 +  29 -> 45
   0   0   0    0  -16    0    0     0     0 =  -16 +  29 -> 13
   0   0   0    0    0   32    0     0     0 =   32 +  29 -> 61

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 31 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0  -16    0    0     0     0 =  -18 +  31 -> 13
   0  -2   0    0    0    0    0     0     0 =   -2 +  31 -> 29
   0   0  -4    0    0   32    0     0   256 =  284 +  31 -> 315
   0   0  -4    0    0    0   64     0   256 =  316 +  31 -> 347
   0   0   0    0  -16   32    0     0     0 =   16 +  31 -> 47
   0   0   0    0  -16    0    0     0     0 =  -16 +  31 -> 15
   0   0   0    0    0   32    0     0     0 =   32 +  31 -> 63
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 45 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0   2   0    0   16    0    0    0     0 =   18 +  45 -> 63
   0   2   0    0    0    0    0    0     0 =    2 +  45 -> 47
   0   0  -4    0    0    0    0    0   256 =  252 +  45 -> 297
   0   0   0    0   16    0    0    0     0 =   16 +  45 -> 61
   0   0   0    0    0  -32   64    0     0 =   32 +  45 -> 77
   0   0   0    0    0    0    0  128     0 =  128 +  45 -> 173

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 47 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0    0    0    0    0     0 =   -2 +  47 -> 45
   0   0  -4    0    0    0    0    0   256 =  252 +  47 -> 299
   0   0   0    0   16    0    0    0     0 =   16 +  47 -> 63
   0   0   0    0    0  -32   64    0     0 =   32 +  47 -> 79
   0   0   0    0    0    0    0  128     0 =  128 +  47 -> 175

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 61 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0   2   0    0    0    0    0    0     0 =    2 +  61 -> 63
   0   0  -4    0    0    0    0    0   256 =  252 +  61 -> 313
   0   0   0    0  -16    0    0    0     0 =  -16 +  61 -> 45
   0   0   0    0    0  -32   64    0     0 =   32 +  61 -> 93

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 63 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0  -16    0    0    0     0 =  -18 +  63 -> 45
   0  -2   0    0    0    0    0    0     0 =   -2 +  63 -> 61
   0   0  -4    0    0    0    0    0   256 =  252 +  63 -> 315
   0   0   0    0  -16    0    0    0     0 =  -16 +  63 -> 47
   0   0   0    0    0  -32   64    0     0 =   32 +  63 -> 95
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 77 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0   2   0    0   16    0    0    0     0 =   18 +  77 -> 95
    0   2   0    0    0    0    0    0     0 =    2 +  77 -> 79
    0   0  -4    0    0    0    0    0   256 =  252 +  77 -> 329
    0   0   0    0   16    0    0    0     0 =   16 +  77 -> 93
    0   0   0    0    0    0  -64  128     0 =   64 +  77 -> 141
    0   0   0    0    0    0  -64    0     0 =  -64 +  77 -> 13
    0   0   0    0    0    0    0  128     0 =  128 +  77 -> 205

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 79 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0  -2   0    0    0    0    0    0     0 =   -2 +  79 -> 77
    0   0  -4    0    0    0    0    0   256 =  252 +  79 -> 331
    0   0   0    0   16    0    0    0     0 =   16 +  79 -> 95
    0   0   0    0    0    0  -64  128     0 =   64 +  79 -> 143
    0   0   0    0    0    0  -64    0     0 =  -64 +  79 -> 15
    0   0   0    0    0    0    0  128     0 =  128 +  79 -> 207

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 93 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0   2   0    0    0    0    0    0     0 =    2 +  93 -> 95
    0   0  -4    0    0    0    0    0   256 =  252 +  93 -> 345
    0   0   0    0  -16    0  -64  128     0 =   48 +  93 -> 141
    0   0   0    0  -16    0    0    0     0 =  -16 +  93 -> 77
    0   0   0    0    0    0  -64    0     0 =  -64 +  93 -> 29

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 95 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0  -2   0    0  -16    0    0    0     0 =  -18 +  95 -> 77
    0  -2   0    0    0    0    0    0     0 =   -2 +  95 -> 93
    0   0  -4    0    0    0    0    0   256 =  252 +  95 -> 347
    0   0   0    0  -16    0  -64  128     0 =   48 +  95 -> 143
    0   0   0    0  -16    0    0    0     0 =  -16 +  95 -> 79
    0   0   0    0    0    0  -64    0     0 =  -64 +  95 -> 31
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0   2   0    0    0    0    0    0     0 =    2 + 141 -> 143
   0   0  -4    0    0   32    0    0   256 =  284 + 141 -> 425
   0   0  -4    0    0    0   64    0   256 =  316 + 141 -> 457
   0   0   0    0   16    0    0 -128     0 = -112 + 141 -> 29
   0   0   0    0    0   32    0    0     0 =   32 + 141 -> 173
   0   0   0    0    0    0    0 -128     0 = -128 + 141 -> 13

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0    0    0    0 -128     0 = -130 + 143 -> 13
   0  -2   0    0    0    0    0    0     0 =   -2 + 143 -> 141
   0   0  -4    0    0   32    0    0   256 =  284 + 143 -> 427
   0   0  -4    0    0    0   64    0   256 =  316 + 143 -> 459
   0   0   0    0   16    0    0 -128     0 = -112 + 143 -> 31
   0   0   0    0    0   32    0    0     0 =   32 + 143 -> 175
   0   0   0    0    0    0    0 -128     0 = -128 + 143 -> 15

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0   2   0    0    0    0    0    0     0 =    2 + 173 -> 175
   0   0  -4    0    0    0    0    0   256 =  252 + 173 -> 425
   0   0   0    0   16    0    0 -128     0 = -112 + 173 -> 61
   0   0   0    0    0  -32   64    0     0 =   32 + 173 -> 205
   0   0   0    0    0    0    0 -128     0 = -128 + 173 -> 45

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0    0    0    0 -128     0 = -130 + 175 -> 45
   0  -2   0    0    0    0    0    0     0 =   -2 + 175 -> 173
   0   0  -4    0    0    0    0    0   256 =  252 + 175 -> 427
   0   0   0    0   16    0    0 -128     0 = -112 + 175 -> 63
   0   0   0    0    0  -32   64    0     0 =   32 + 175 -> 207
   0   0   0    0    0    0    0 -128     0 = -128 + 175 -> 47
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 205 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0   2   0    0    0    0    0     0     0 =    2 + 205 -> 207
    0   0  -4    0    0    0    0     0   256 =  252 + 205 -> 457
    0   0   0    0   16    0    0  -128     0 = -112 + 205 -> 93
    0   0   0    0    0    0  -64     0     0 =  -64 + 205 -> 141
    0   0   0    0    0    0    0  -128     0 = -128 + 205 -> 77

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 207 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0  -2   0    0    0    0    0  -128     0 = -130 + 207 -> 77
    0  -2   0    0    0    0    0     0     0 =   -2 + 207 -> 205
    0   0  -4    0    0    0    0     0   256 =  252 + 207 -> 459
    0   0   0    0   16    0    0  -128     0 = -112 + 207 -> 95
    0   0   0    0    0    0  -64     0     0 =  -64 + 207 -> 143
    0   0   0    0    0    0    0  -128     0 = -128 + 207 -> 79

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 297 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0   2   0    0   16    0    0     0     0 =   18 + 297 -> 315
    0   2   0    0    0    0    0     0     0 =    2 + 297 -> 299
    0   0   4    0    0    0    0     0  -256 = -252 + 297 -> 45
    0   0   0    0   16    0    0     0     0 =   16 + 297 -> 313
    0   0   0    0    0  -32   64     0     0 =   32 + 297 -> 329
    0   0   0    0    0    0    0   128     0 =  128 + 297 -> 425

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 299 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0  -2   0    0    0    0    0     0     0 =   -2 + 299 -> 297
    0   0   4    0    0    0    0     0  -256 = -252 + 299 -> 47
    0   0   0    0   16    0    0     0     0 =   16 + 299 -> 315
    0   0   0    0    0  -32   64     0     0 =   32 + 299 -> 331
    0   0   0    0    0    0    0   128     0 =  128 + 299 -> 427
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 313 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0   2   0    0    0    0    0     0     0 =    2 + 313 -> 315
    0   0   4    0    0    0    0     0  -256 = -252 + 313 -> 61
    0   0   0    0  -16    0    0     0     0 =  -16 + 313 -> 297
    0   0   0    0    0  -32   64     0     0 =   32 + 313 -> 345

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 315 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0  -2   0    0  -16    0    0     0     0 =  -18 + 315 -> 297
    0  -2   0    0    0    0    0     0     0 =   -2 + 315 -> 313
    0   0   4    0    0    0    0     0  -256 = -252 + 315 -> 63
    0   0   0    0  -16    0    0     0     0 =  -16 + 315 -> 299
    0   0   0    0    0  -32   64     0     0 =   32 + 315 -> 347

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 329 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0   2   0    0   16    0    0     0     0 =   18 + 329 -> 347
    0   2   0    0    0    0    0     0     0 =    2 + 329 -> 331
    0   0   4    0    0    0    0     0  -256 = -252 + 329 -> 77
    0   0   0    0   16    0    0     0     0 =   16 + 329 -> 345
    0   0   0    0    0    0    0   128     0 =  128 + 329 -> 457

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 331 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0  -2   0    0    0    0    0     0     0 =   -2 + 331 -> 329
    0   0   4    0    0    0    0     0  -256 = -252 + 331 -> 79
    0   0   0    0   16    0    0     0     0 =   16 + 331 -> 347
    0   0   0    0    0    0    0   128     0 =  128 + 331 -> 459

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 345 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

    0   2   0    0    0    0    0     0     0 =    2 + 345 -> 347
    0   0   4    0    0    0    0     0  -256 = -252 + 345 -> 93
    0   0   0    0  -16    0    0     0     0 =  -16 + 345 -> 329
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 347 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0  -16    0    0     0     0 =  -18 + 347 -> 329
   0  -2   0    0    0    0    0     0     0 =   -2 + 347 -> 345
   0   0   4    0    0    0    0     0  -256 = -252 + 347 -> 95
   0   0   0    0  -16    0    0     0     0 =  -16 + 347 -> 331

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 425 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0   2   0    0    0    0    0     0     0 =    2 + 425 -> 427
   0   0   4    0    0    0    0     0  -256 = -252 + 425 -> 173
   0   0   0    0   16    0    0  -128     0 = -112 + 425 -> 313
   0   0   0    0    0  -32   64     0     0 =   32 + 425 -> 457
   0   0   0    0    0    0    0  -128     0 = -128 + 425 -> 297

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 427 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0    0    0    0  -128     0 = -130 + 427 -> 297
   0  -2   0    0    0    0    0     0     0 =   -2 + 427 -> 425
   0   0   4    0    0    0    0     0  -256 = -252 + 427 -> 175
   0   0   0    0   16    0    0  -128     0 = -112 + 427 -> 315
   0   0   0    0    0  -32   64     0     0 =   32 + 427 -> 459
   0   0   0    0    0    0    0  -128     0 = -128 + 427 -> 299

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 457 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0   2   0    0    0    0    0     0     0 =    2 + 457 -> 459
   0   0   4    0    0    0    0     0  -256 = -252 + 457 -> 205
   0   0   0    0   16    0    0  -128     0 = -112 + 457 -> 345
   0   0   0    0    0    0    0  -128     0 = -128 + 457 -> 329

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% -- 459 -- %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% R % P % Dt % Sv % Cd % Dg % Dc %% Rp %% Ug %% Ev %% Ac %% As %%

   0  -2   0    0    0    0    0  -128     0 = -130 + 459 -> 329
   0  -2   0    0    0    0    0     0     0 =   -2 + 459 -> 457
   0   0   4    0    0    0    0     0  -256 = -252 + 459 -> 207
   0   0   0    0   16    0    0  -128     0 = -112 + 459 -> 347
   0   0   0    0    0    0    0  -128     0 = -128 + 459 -> 331
```

Thèse de Monsieur Eric ZAMAI

Architecture de Surveillance-Commande pour les Systèmes à Evénements Discrets Complexes

Le travail présenté dans ce mémoire s'inscrit dans le contexte de la supervision des ateliers flexibles de production manufacturière. Il traite plus particulièrement de l'intégration de la surveillance temps réel des défaillances du procédé. L'approche se distingue en considérant la commande et la surveillance sur un même plan et non la surveillance comme un palliatif à la commande. Une structure d'un module de surveillance-commande est proposée. Le module est constitué de deux modèles coopérants basés sur le concept d'activités : l'un, appelé modèle de référence pour la surveillance-commande, modélise toutes les fonctionnalités mises en œuvre par le système de surveillance, l'autre, appelé modèle de la stratégie de surveillance-commande, modélise les contraintes imposées par les objectifs propres de l'entreprise et des utilisateurs. L'exécution d'un traitement de surveillance (séquence d'activités) n'est pas limitée au strict enchaînement des fonctions détection, diagnostic, décision puis reprise. D'autres séquences mettant en œuvre des activités de surveillance-commande plus élaborées (par exemple, activation de plusieurs fonctions de surveillance-commande du module simultanément) sont autorisées et réalisables par le module. En fonction des activités en cours d'exécution, un superviseur implanté dans chacun des modules gère l'ensemble des informations qui transitent par le module en les orientant vers la ou les fonctions de surveillance-commande aptes à les prendre en compte : détection, diagnostic, décision, reprise, urgence, suivi, commande.

Un exemple d'application basé sur un processus manufacturier réel, la cellule flexible de l'Ecole Nationale d'Ingénieur de Tarbes (ENIT), illustre les apports de notre approche. Ils se traduisent en terme de flexibilité de surveillance, de réactivité aux diverses évolutions du procédé, de prise en compte et de respect des contraintes imposées par l'entreprise d'un point de vue surveillance.

Mots-clés : système à événements discrets, surveillance, commande, supervision, modèle de surveillance, atelier flexible.

Monitoring and Control Architecture of Complex Discrete Events Systems

The work presented deals with the supervision of flexible manufacturing systems. It presents the integration of real-time monitoring of process failures. The originality of this approach is to consider the control and the monitoring on the same level and not the monitoring as a palliative for the control. A structure of a monitoring-control module is proposed. This module is made up of two cooperating models based on the activity concept. The first one, the reference model for monitoring and control, gives all the functionalities of the monitoring system. The second one, the strategy model for monitoring and control, models the constraints imposed by the objectives of the firm and the users. The execution of a monitoring treatment is not limited to the strict sequence « detection, diagnosis, decision and recovery ». Other elaborated sequences are allowed and can be executed by the module. Taking the executed activity into account, a supervisor integrated in each module of the hierarchy manages the information by directing each one to the suitable monitoring or control functions: detection, follow, control, diagnosis, decision, recovery or emergency.

An example based on a manufacturing process illustrates the main benefits of the approach: monitoring flexibility, reactivity to the process evolutions, respect of the monitoring constraints imposed by the factory.

Keywords: discrete events systems, monitoring, control, supervisor, monitoring model, flexible manufacturing systems.