

N° d'ordre : 1263

50376  
1994  
101

# THESE

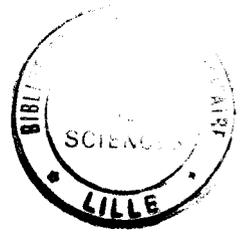
présentée à

**L'UNIVERSITÉ DES SCIENCES ET TECHNOLOGIES  
DE LILLE**

pour obtenir le grade de

**DOCTEUR DE L'UNIVERSITÉ**

en



**PRODUCTIQUE,  
AUTOMATIQUE ET INFORMATIQUE INDUSTRIELLE**

par

**DIPL.-ING. (UNIV)  
CHRISTIAN AUSFELDER**

**CONTRIBUTION À LA CONCEPTION D'UN SYSTÈME DE  
CONDUITE POUR LES SYSTÈMES FLEXIBLES DE  
PRODUCTION MANUFACTURIÈRE :  
MODÉLISATION ET VALIDATION DE LA COMMANDE**

soutenu le 4 mars 1994 devant la commission d'examen :

- |        |              |  |
|--------|--------------|--|
| M J.P. | BOUREY       | <b>Examineur</b>                       |
| M E.   | CASTELAIN    | <b>Examineur, Directeur de travail</b> |
| M M.   | COURVOISIER  | <b>Rapporteur</b>                      |
| M J.C. | GENTINA      | <b>Examineur, Directeur de thèse</b>   |
| M X.   | DATIN        | <b>Examineur</b>                       |
| M R.   | SOENEN       | <b>Rapporteur</b>                      |
| M G.   | SCHMIDT      | <b>Examineur</b>                       |
| M M.   | STAROSWIECKI | <b>Examineur, Président de jury</b>    |



## AVANT PROPOS

*Le travail présenté dans ce mémoire a été effectué au Laboratoire d'Automatique et Informatique Industrielle de Lille (LAIL), sous la direction scientifique de Monsieur J.C. GENTINA, Professeur et Directeur de l'Ecole Centrale de Lille et avec la collaboration de Monsieur E. CASTELAIN, Maître de Conférences à l'Ecole Centrale de Lille. Je tiens à leur témoigner toute ma reconnaissance. Cette aventure "thèse" n'aurait jamais eu lieu sans l'engagement pour la cause européenne de M. J.C. GENTINA. Les relations amicales avec M. E. CASTELAIN ont été un soutien précieux lors des phases difficiles de ces années.*

*Je suis très reconnaissant à Monsieur M. Courvoisier, Professeur à l'Université Paul Sabatier à Toulouse et Monsieur R. Soenen, Professeur à l'Université de Valenciennes de l'honneur qu'ils me font en acceptant de juger ce travail et d'être rapporteurs de cette thèse.*

*Je tiens à remercier de leur présence dans ce jury Monsieur M. Staroswiecki, Professeur à l'Université de Lille I qui a accepté d'être Président de jury, ainsi que Monsieur X. Datin, responsable du Laboratoire Productique au Département de Recherche et Développement de la société Télémécanique S.A. à Nanterre. Je suis particulièrement flatté par la présence de Herrn Professor Dr.-Ing. G. Schmidt, Ordinarius am Lehrstuhl für Steuerungs-und Regelungstechnik der Technischen Universität München, qui m'a ouvert la porte du LAIL dans le cadre d'un échange Erasmus en 1990 / 1991. Je remercie particulièrement Monsieur J.P Bourey, Maître des Conférences à l'Ecole Centrale de Lille pour l'intérêt qu'il a montré pour mon travail, la soigneuse lecture de ce mémoire et sa présence dans ce jury.*

*Je remercie tout particulièrement mes amis et mes collègues du L.A.I.L pour leur soutien, nos vives discussions et leur bonne humeur.*

*Last but not least, je tiens à exprimer ma gratitude envers mes parents et mon frère, mes amis et ma compagne pour leur compréhension, leur soutien et surtout pour leur affection.*



## SOMMAIRE

	<b>Page</b>
<b>Introduction Générale :</b>	<b>7</b>
<b>Les Principales Caractéristiques des Systèmes Flexibles de Production Manufacturière</b>	
<b>Chapitre I :</b>	<b>13</b>
<b>Un Modèle de la Conduite des SFPM</b>	
<b>Chapitre II :</b>	<b>43</b>
<b>Une Méthode pour la Conception de la Commande de Coordination des SFPM</b>	
<b>Chapitre III :</b>	<b>115</b>
<b>La Validation des Modèles de la Commande de Coordination</b>	
<b>Conclusion Générale :</b>	<b>165</b>
<b>Résumé et Perspectives</b>	
<b>Annexes</b>	<b>167</b>
<b>Références Bibliographiques</b>	<b>197</b>
<b>Table des Matières</b>	<b>207</b>
<b>Liste des Figures</b>	<b>213</b>



# **INTRODUCTION GÉNÉRALE :**

## **LES PRINCIPALES CARACTÉRISTIQUES**

### **DES SYSTÈMES FLEXIBLES DE**

### **PRODUCTION MANUFACTURIÈRE**

L'évolution de la situation des marchés vers une diversification des produits adaptés aux besoins et à la demande presque individuelle du consommateur nécessite des structures de production adéquates [Meffert 1986]. Ces structures doivent permettre une adaptation rapide de la production à l'évolution des marchés en gardant les avantages d'une production à grande échelle.

L'intégration des calculateurs a fait évoluer l'automatisation de la production au-delà d'une simple mécanisation des sites vers une automatisation des fonctions de contrôle et de surveillance des processus de fabrication [Fandel et al 1990]. Les gains de productivité d'une automatisation poussée sont, de ce fait, devenus réalisables dans les productions de type "atelier". Ce développement joint à l'évolution de la demande vers des produits individualisés a ainsi donné naissance aux SFPM (Système Flexible de Production Manufacturière).

"Un système flexible de production manufacturière est un atelier automatisé et intégré qui est composé d'un ensemble de machines-outils à commande numérique munies d'une gestion d'outils automatisée, qui dispose de systèmes de métrologie et d'assurance de qualité. Les composants de l'atelier sont reliés par un système de manutention automatisé et l'ensemble est commandé, contrôlé et surveillé par un système informatique." [Schmidt 1989].

Cette définition souligne trois caractéristiques des SFPM que nous examinons par la suite :

- \* le caractère “atelier”,
- \* le haut degré d’automatisation,
- \* l’aspect intégration du point de vue des composants matériels et de la conduite par l’intermédiaire d’un système informatique.

Dans une organisation de type atelier, les opérations transformationnelles de même type sont regroupées dans un même site de fabrication (cf groupement par technologie). Par exemple, seront regroupées dans le même atelier toutes les machines réalisant des opérations de tournage.

Dans les SFPM, un regroupement de machines en sous-structures est souvent réalisé par rapport à des critères d’exploitation, c’est-à-dire par rapport aux produits à fabriquer. Les sous-structures, nommées cellules ou îlots, peuvent comporter, dans ce cas, des machines de fonctionnalités différentes [Ferreira et Pradin 1991].

Le processus de fabrication d’un type de produit est déterminé par la séquence des transformations et non par la séquence des lieux opératoires sur lesquels passe le produit. Une séquence de transformation désigne donc un type de produit. Comme un produit peut être amené à passer plusieurs fois sur une même machine et comme tous les produits dans l’atelier n’ont pas la même séquence d’opérations, un système de transport reliant l’ensemble des lieux opératoires (lieux dans l’atelier sur lesquels les produits subissent les transformations) est indispensable.

Le principe de la fabrication de type atelier crée des problèmes de synchronisation entre les différentes étapes de la fabrication. Les temps opérationnels varient en fonction des produits, le transport par “rafales” des produits entre les lieux d’opération introduit des tampons de stockage intermédiaire [Fandel et al 1990]. Les groupements de machines en sous-structures sont ainsi découplés les uns des autres et obtiennent un certain degré d’autonomie (composants autonomes).

La production d’un type de produit peut être très flexible. Les transformations sont, en général, réalisables sur différents lieux opératoires, l’évolution vers d’autres opérations transformationnelles est relativement simple, ainsi que l’introduction de nouveaux produits. Cette flexibilité demande un haut degré de polyvalence des machines. Mais ce haut degré de flexibilité ne se limite pas seulement à la variation qualitative des types de produits : les

quantités relatives des différents produits subissent également de fortes variations dans le temps.

Les perturbations du processus de fabrication sont compensables de la même manière. La polyvalence des machines permet l'exécution d'une transformation par plusieurs machines et, en cas de panne d'une machine, les transformations sont exécutées sur une autre. La flexibilité des ateliers permet aussi une adaptation temporelle de la production, la fabrication des produits non affectés par la panne est augmentée dans un premier temps et les capacités ainsi libérées serviront ensuite à une récupération rapide du retard des produits affectés par la panne (i.e. re-planification de la production).

La fabrication de type atelier est donc caractérisée par une organisation regroupant les machines dans des composants autonomes et un routage individuel des produits assuré par un moyen de transport flexible. Les produits sont spécifiés par leur séquence d'opérations, la polyvalence opérationnelle des machines permet d'adapter la production d'une manière qualitative, quantitative et temporelle et augmente ainsi la disponibilité de l'atelier.

Par rapport aux ateliers "classiques", les SFPM se distinguent par leur haut degré d'automatisation. Outre le remplacement de l'homme dans les tâches fastidieuses, l'automatisation du processus de fabrication intègre la réalisation des tâches de contrôle et de surveillance par des moyens informatiques. L'homme ne prend qu'une position de supervision et éventuellement de maintenance [Fandel et al 1990].

Mais le remplacement d'une machine et de son opérateur par une machine-outil à commande numérique desservie par un robot a plusieurs effets, qui influencent l'exploitation de l'atelier. Un premier effet est une perte de flexibilité, car, par exemple, de légères modifications d'un produit demandent une re-programmation complète de la machine. Un deuxième effet est une perte de disponibilité de la machine, car l'opérateur accomplit aussi des fonctions de surveillance, de diagnostic et de maintenance. Son expérience lui permet de réagir d'une manière flexible aux différentes défaillances qui se produisent pendant les opérations. Un troisième effet est une augmentation des coûts fixes du poste, car le poste représente un investissement considérable.

Les deux effets primaires, perte de réactivité et coûts croissants, ont un effet secondaire sur la production dans les SFPM. Elle évolue vers la fabrication de petite et de moyenne série afin d'augmenter le taux d'utilisation de chaque machine et de diminuer les temps de reconfiguration des machines entre deux opérations différentes [Ranky 1990, Bonetto 1985]. Ainsi une

exploitation optimale des ateliers, qui augmente la disponibilité du système en exploitant son potentiel (flexibilité), qui optimise les critères économiques de la production (objectifs) et qui assure un fonctionnement sûr de l'atelier, devient importante.

Le troisième aspect de la définition de SCHMIDT, l'intégration, est né du souci d'une exploitation optimale du système. Seule une vue globale du potentiel du système permet de réagir aux évolutions inattendues de l'environnement, d'augmenter la disponibilité du système et d'assurer ainsi les objectifs de production. Cet aspect intégration est donc une composante essentielle pour **la conduite** d'un système.

Comme dans la phase de conception d'un SFPM le plan directeur de production, c'est-à-dire l'ensemble des produits et des quantités à fabriquer sur un horizon donné n'est pas connu, l'intégration de nouvelles ressources de production et de produits nouveaux doit être prévue dans le système et dans sa conduite. Cette flexibilité à long terme [Slack 1988] permet l'adaptation du programme de production à une demande changeante.

Le projet CASPAIM (Conception Assistée de Systèmes de Production Automatisée dans l'Industrie Manufacturière) du Laboratoire d'Automatique et Informatique de Lille (L.A.I.L) s'est fixé comme but le développement d'une méthode de conception du système de conduite et d'un outil d'aide à la conception des SFPM [Bourey et al 1989].

Les travaux, présentés dans ce mémoire, s'inscrivent dans le cadre du développement d'un système de conduite et proposent une méthode de conception de la commande de coordination des SFPM. Notre hypothèse de départ est donc un atelier existant pour lequel nous concevons la commande.

La conception de la commande de coordination doit respecter les spécificités des SFPM : leur organisation en composants autonomes, l'existence d'un système de transport reliant les composants, la définition des produits à partir des séquences d'opérations transformationnelles et leur fabrication en petite et moyen série.

La coordination n'est qu'une partie du système de contrôle / commande. Il est indispensable de tenir compte des interactions et des interdépendances de la commande de coordination avec les autres éléments de la conduite pour réaliser une conduite performante.

De ce fait, le premier chapitre de ce mémoire est une présentation des fonctions de la conduite d'un SFPM et leur intégration dans un système de contrôle / commande du point de

vue du projet CASPAIM.

Le deuxième chapitre présente une méthode de conception de la commande de coordination d'un SFPM (conception détaillée) par réseaux de Petri qui répond aux besoins spécifiques des SFPM.

Le troisième chapitre est consacré à la validation qualitative des modèles conçus suivant la méthode proposée. Nous présentons une démarche de validation qui permet, d'une part, d'affirmer les bonnes propriétés des modèles, d'autre part, la déduction des bornes supérieures de l'encours garantissant un comportement sans blocage du système en phase d'exploitation.



# CHAPITRE I:

## UN MODÈLE DE LA CONDUITE DES

## SFPM

### 1. INTRODUCTION

Dans ce chapitre, nous présentons le **concept** que nous avons utilisé pour l'étude préliminaire du système de conduite d'un atelier flexible. Ce concept propose une synthèse de la conduite à partir de ses sous-fonctions et s'intègre aux travaux menés au L.A.I.L dans le cadre du projet CASPAIM.

La description des fonctions de la conduite reste au niveau conceptuel, c'est-à-dire que nous décrivons chaque fonction par son activité et par les données échangées avec les autres fonctions. Nous proposons donc une modélisation conceptuelle du système de contrôle / commande un utilisant une représentation de type SADT (Structured Analysis Design Technique) [Ross 1985].

Les travaux, menés dans les domaines de la surveillance et de la gestion de production autour du projet CASPAIM, demandent une intégration dans l'ensemble du projet. Une méthode de conception d'une commande globale pour les SFPM nécessite la prise en compte de l'intégralité des fonctions dès la phase de spécification [Bourey 1993].

En effet, l'identification des interactions et des relations des différentes fonctions de la conduite nous a permis d'établir un cahier des charges détaillé pour la conception de la commande de coordination.

## 2. LE BESOIN D'UNE APPROCHE INTÉGRÉE POUR LA CONCEPTION DE LA CONDUITE

Le Système de contrôle / commande conditionne à un point important l'exploitation optimale d'un SFPM. Le problème posé par la recherche d'une conduite optimale est complexe. En effet, il doit garantir le respect des objectifs de production le mieux possible par une disponibilité maximale du système tout en assurant la sécurité de l'homme et du matériel, ce qui nécessite une démarche rigoureuse de conception. Cette démarche constitue l'un des buts principaux du projet CASPAIM.

### 2.1. LA PROBLÉMATIQUE DE LA CONDUITE

La conduite de l'atelier consiste à faire exécuter par l'atelier l'ensemble des opérations de fabrication qui lui sont affectées [Archimède 1991] :

- \* en respectant au mieux les objectifs de production fixés, tout en satisfaisant les contraintes spatiales, temporelles et de coût d'exploitation,
- \* en s'assurant que chaque ordre transmis est cohérent vis-à-vis du caractère éventuellement perturbé de l'environnement dans lequel évolue l'atelier,
- \* en instaurant une collaboration étroite et fidèle entre la gestion de la production et le procédé.

Sur la figure I.1 nous présentons les fonctions principales d'un système de contrôle / commande et ses principales interactions avec les autres systèmes.

L'**allocation** des ressources est intégrée dans le processus de conduite globale. Elle réalise des affectations de ressource garantissant la réalisation des services requis par les tâches. La **commande** répartit et transmet les ordres aux ressources de production en tenant compte de leur état. Le **suiti** recueille l'ensemble des événements survenant dans l'atelier, et met à jour les modèles du système opérant. La **réaction** corrige les déviations induites par les aléas de fabrication.

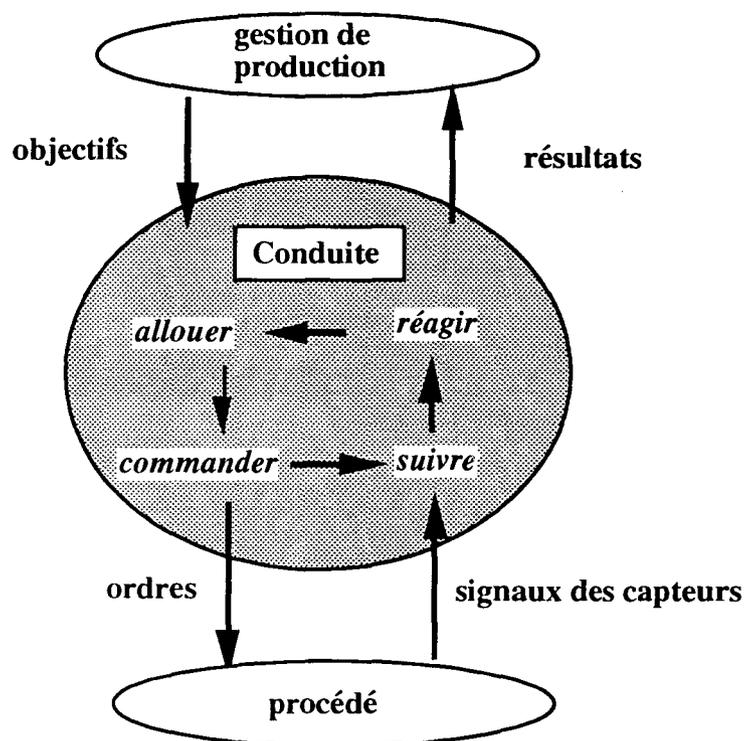


Figure I.1 : les fonctions de la conduite

Le système de conduite transmet les ordres vers le procédé et les résultats vers la gestion de l'atelier. Il reçoit les signaux des capteurs et les objectifs de production. Outre ces informations **externes**, il existe des informations **internes** de conduite, c'est-à-dire des informations, qui sont échangées entre les différentes fonctions. Le flux des informations internes détermine la cohésion entre les différentes activités et établit leurs interdépendances. Outre une réalisation de chaque fonction adaptée à l'atelier et à ses objectifs d'exploitation, une conduite performante demande leur intégration au sein d'un système de contrôle/ commande.

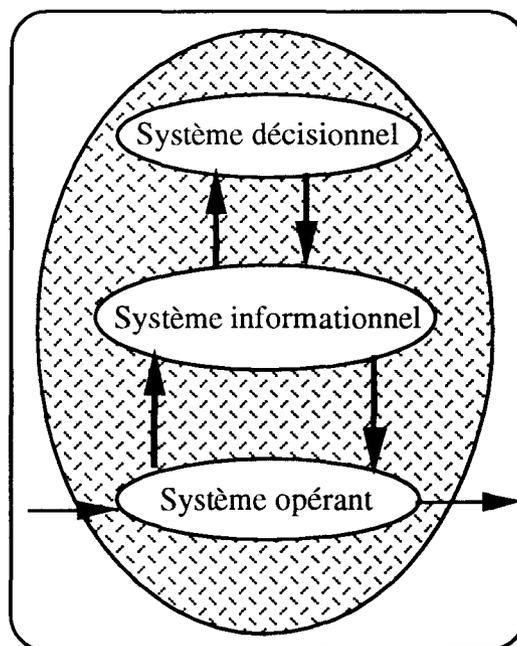
## 2.2. UNE APPROCHE SYSTÉMIQUE POUR LA CONDUITE

Un système peut être défini comme un ensemble d'éléments liés par une relation [Daenzer 1988]. L'approche systémique en distingue deux types. Les premiers ont une taille qui donne une impression de complexité, mais dont les constituants ne sont que faiblement reliés entre eux. Nous les appelons systèmes "compliqués". Ceux-ci peuvent être étudiés par des approches analytiques, c'est-à-dire par une simple décomposition du système.

En revanche, le deuxième type de système est caractérisé par un fort couplage de ses éléments. Ce sont les systèmes dits "complexes". Par conséquent, une simple décomposition en sous-systèmes omettrait la prise en compte de ces liens importants entre les différents éléments et mènerait donc à une perte d'information ainsi qu'à des difficultés de compréhension. Nous étudierons donc ces systèmes dits "complexes" en les modélisant de façon à les comprendre dans leur intégralité, c'est ce que propose l'approche systémique [Le Moigne 1990].

Les systèmes de contrôle / commande des SFPM sont du deuxième type, car leurs éléments sont liés par le flux d'information. Une collaboration étroite des différentes fonctions établit leur forte cohésion et leur caractère de systèmes complexes.

La figure I.2 met en évidence la description la plus générale d'un système complexe où apparaissent trois sous-systèmes comportant des caractéristiques bien dissociables : le système opérant, le système informationnel et le système décisionnel (modèle O.I.D).



**Figure I.2 : modèle O.I.D. d'un système complexe [Le Moigne 1990]**

**Le système décisionnel** définit les stratégies d'exploitation du système. Ces stratégies d'exploitation sont rendues opérationnelles sous forme d'un plan. Les décisions liées aux stratégies d'exploitation ont pour données des informations qui proviennent du système informationnel.

**Le système d'information** traite les données représentant les décisions et les propage vers le système opérant (flèche descendante). Le système opérant génère des données, qui représentent son état, et qui sont enregistrées au niveau du système d'information. Le système d'information dispose par conséquent de modèles du système opérant, qui sont mis à jour par ce flux remontant de l'information, qui mémorisent le comportement du système opérant, et qui sont les données de base du système décisionnel.

**Le système opérant** transforme les opérations logiques (informations) en opérations tangibles et transmet des données concernant son état vers le système d'information.

En appliquant cette description aux systèmes flexibles de production manufacturière, nous pouvons établir les correspondances suivantes :

- \* **Système décisionnel - gestion de production** : le système décisionnel trouve son équivalence dans la gestion de production au sens large. Elle définit les stratégies de production et détermine la nature et la quantité des produits à fabriquer (programme de production).
- \* **Système d'information - système de contrôle / commande ou conduite** : le système d'information dans l'approche systémique correspond au système de contrôle / commande en productique. Il dispose des modèles du système opérant, propage les ordres de fabrication vers les actionneurs des ressources et remonte les informations des capteurs.
- \* **Système opérant - procédé** : le système opérant correspond à la partie "hard" de l'atelier, appelée aussi procédé ou système physique. Le procédé est composé des ressources de transformations et des moyens de transports. Les actionneurs réalisent les ordres (opérations logiques) envoyés par le système d'information et les capteurs génèrent, en retour, les données nécessaires à la mémorisation de l'état actuel du procédé.

C'est dans l'esprit d'une approche systémique, que nous proposons une modélisation du système de contrôle / commande. Ce modèle représente la conduite dans son intégralité, car seule une approche globale nous permet la compréhension de ce système complexe. Le modèle résultant met en évidence les différentes fonctions nécessaires pour une conduite performante.

Ce modèle conceptuel de la conduite peut être exploité pour la conception d'un système de contrôle / commande. Il correspond en fait à un cadre d'étude préliminaire pour la conception de

la conduite d'un atelier spécifique.

Par rapport à ce cadre, le concepteur peut déterminer les fonctions à réaliser par logiciel et celles prises en compte par le pilote du système. L'étude détaillée portera ensuite sur la conception des fonctions automatisées et sur la conception de l'interface avec le pilote.

Le projet CASPAIM propose des solutions ou des méthodes de conception pour des fonctions, qui, en général, sont automatisées, comme la commande de coordination, la surveillance et l'allocation des ressources. Ces solutions peuvent être appliquées pour la conception de chacune des fonctions.

L'identification des fonctions à réaliser par le pilote du système, qui peuvent varier d'un atelier à l'autre, permet la définition d'un cahier des charges pour la conception d'une interface ergonomique [Milot 1988].

La réalisation par logiciel des fonctions automatisées et leur intégration dans le système de contrôle / commande termine la phase de conception. Dans le cadre du projet CASPAIM, cette étape de "construction" du système est couplée à une méthode d'implantation sur une architecture informatique répartie [Huvenoit et al 1994].

Le modèle peut donc être utilisé non seulement pour la compréhension du fonctionnement de la conduite, mais aussi comme cadre pour l'étude préliminaire de la conception d'un système de contrôle / commande. Le modèle générique de la conduite proposé par la suite reste au niveau conceptuel, il décrit donc les fonctions de la conduite par leur activité. Seules des connaissances relatives aux spécificités d'un atelier et de son exploitation permettent de choisir des solutions adaptées, qui mènent à une conduite performante.

### 3. LE MODÈLE DE LA CONDUITE (AUSFELDER ET AL 1993A)

#### 3.1. LA CONDUITE ET SON ENVIRONNEMENT

La figure I.3 décrit le système de contrôle / commande et son environnement par un modèle de type SADT. La description du formalisme utilisé, les ACTIGRAMMES, est mise en annexe 1.

L'approche systémique propose, outre une modélisation du système global, la définition des limites du système à concevoir et l'analyse des liens avec le monde extérieur [Daenzer 1988]. De ce fait, une discussion et une analyse des systèmes annexes, système opérant et système décisionnel, s'impose.

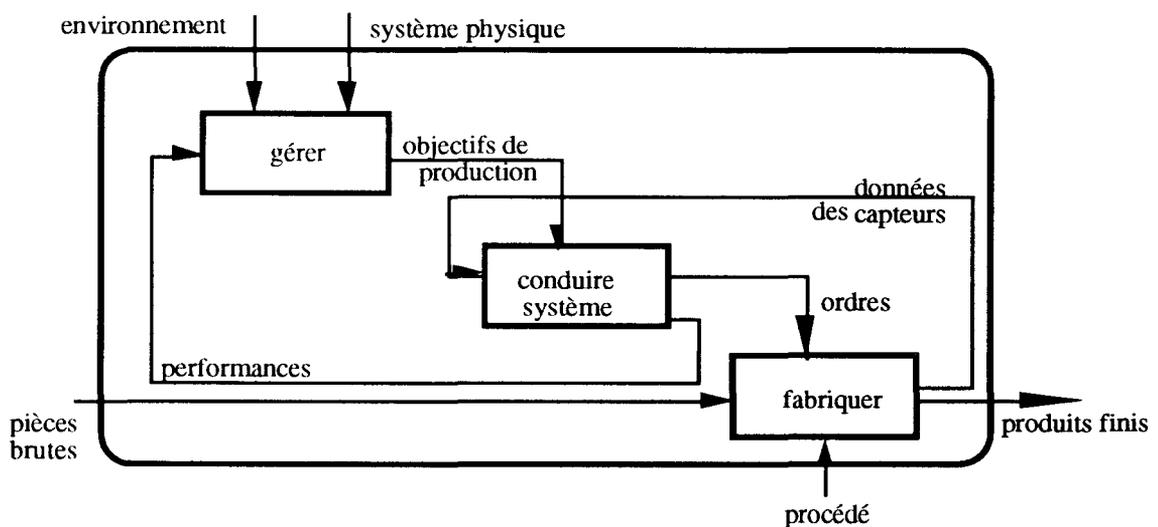


Figure I.3 : la conduite et son environnement

##### 3.1.1. Gérer la Production

La gestion de la production décrit les activités du système décisionnel. A partir des demandes et des prévisions des ventes, donc des contraintes imposées par l'environnement de l'entreprise, cette activité détermine les objectifs de la production sur un horizon donné.

Conformément à notre hypothèse de départ, qui concerne la conception d'un système de contrôle / commande pour un atelier existant, le système physique existe et reste inchangé. L'organisation de l'atelier et sa capacité forment ainsi les contraintes imposées par le système physique.

Les objectifs de productions, en tant que résultats de l'activité "gérer", définissent un plan directeur de production en terme de type de produits à fabriquer, de quantités, de dates dues et de dates de lancement en fabrication. Ce plan directeur de production peut être plus ou moins détaillé. Un plan détaillé fixe le routage de certains produits et détermine l'ordre de passage des produits sur les machines menantes [Chassang et Tron 1983] (les machines menantes représentent les goulets d'étranglement pour une production donnée et déterminent de ce fait la cadence de la fabrication). Un plan très détaillé réduit la complexité du problème de l'ordonnancement, c'est-à-dire de l'allocation spatiale et temporelle d'un produit à une machine précise, mais il réduit aussi les degrés de liberté du système de contrôle / commande, qui lui permettent de réagir aux aléas de la production [Archimède 1991].

La gestion de la production travaille normalement en horizon glissant et à chaque mise à jour du plan directeur de production, les objectifs de production sont réactualisés en fonction de la production effectivement réalisée. Ces performances ou résultats de la production concernent aussi bien des informations qualitatives ( qualité des produits) que quantitatives (délais de fabrication). Les résultats fournis par le système de conduite correspondent donc à un "feedback" pour la gestion.

### **3.1.2. Fabriquer**

Cette activité décrit le système opérant, qui transforme les pièces brutes et des produits semi-finis en produits finis. Les actionneurs des ressources reçoivent les ordres de la conduite et les capteurs du procédé retournent des signaux, qui permettent aux modèles du système de contrôle / commande de suivre l'évolution dynamique du système physique.

Les systèmes physiques que nous proposons d'aborder dans notre étude concernent les SFPM. Ces systèmes ont trois caractéristiques principales :

- \* une grande variété de produits fabriqués,
- \* une polyvalence importante des ressources de production et de transport,
- \* une variabilité de la production dans le temps.

La flexibilité des ateliers permet facilement de modifier la nature des productions ainsi que les quantités respectives des produits, qui concernent en général, des petites séries. Il en résulte une forte variation dynamique des flux de produits (non-régularité des flux sur un horizon significatif).

La flexibilité permet également de réagir à d'éventuelles défaillances des ressources de production par une adaptation ou une reconfiguration du système dans le but d'accroître la disponibilité de l'atelier.

Mais de tels degrés de liberté n'ont d'intérêt, que s'ils sont exploitables en temps réel à l'initiative du système de conduite, car un système de production ne peut être flexible que si son contrôle lui conserve toute sa flexibilité potentielle [Slack 1988].

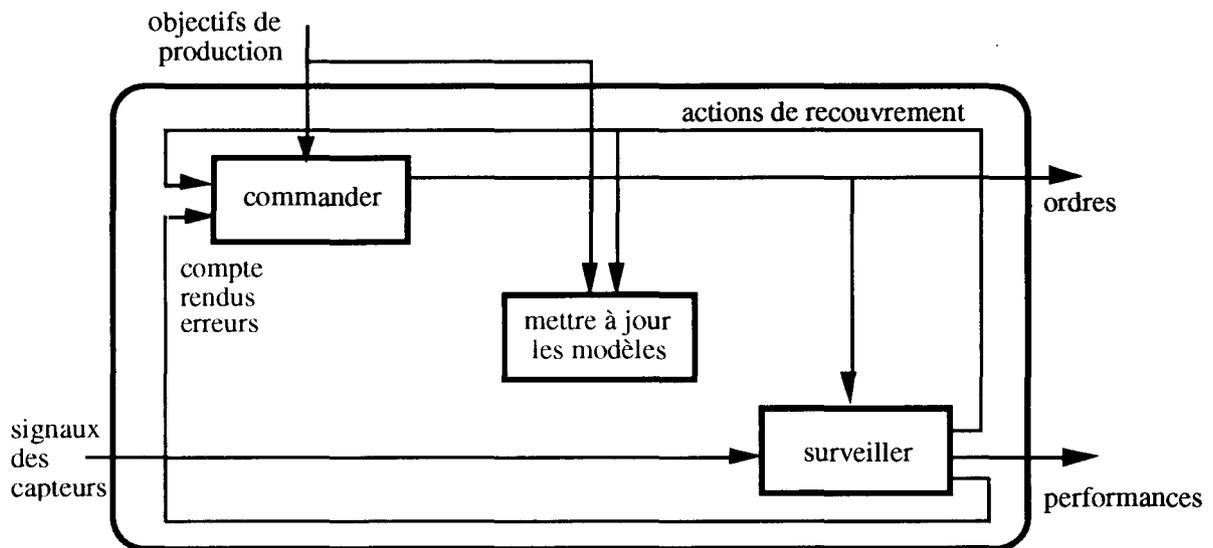
### 3.1.3. Conduire

L'activité "conduire" décrit le système de contrôle / commande. Comme tout système d'information, il dispose de modèles qui mémorisent l'évolution dynamique du système physique. A l'aide de ces modèles, la conduite du système, d'une part, va transformer les objectifs de production en commandes destinées aux actionneurs de la partie opérative, c'est à dire qu'elle analyse et décompose l'information. D'autre part, cette conduite va également synthétiser l'information en regroupant les signaux des capteurs. Cette synthèse permettra de connaître l'évolution réelle du système (performances du système).

Il en résulte deux flux d'information opposés : un flux descendant qui propage l'information vers la partie opérative et un flux ascendant qui retourne les résultats des actions sollicitées par le flux descendant.

La communication entre la **conduite** et le **procédé** est soumise à des fortes contraintes de temps réel [André et al 1993]. Les informations échangées (ordres et signaux de capteurs) sont élémentaires et déterministes. Les informations échangées entre la **conduite** et la **gestion** sont des informations agrégées (objectifs et résultats), qui couvrent une certaine durée temporelle en fonction de la longueur de l'horizon choisi. Le système de contrôle / commande se caractérise donc à la fois par un caractère synchrone / déterministe et un caractère asynchrone vis-à-vis de l'évolution du procédé.

### 3.2. LA MODÉLISATION DE LA CONDUITE



**Figure I.4 : l'activité "conduire"**

Sur la figure I.4 nous présentons le premier niveau de décomposition de l'activité "conduire". Nous identifions les deux activités concernées par la gestion des flux opposés d'informations. L'activité "commander" gère le flux descendant d'information et décompose donc les objectifs de production en commandes effectives envoyées à la partie opérative. L'activité "surveiller" reconstitue l'information issue des signaux des capteurs afin d'établir le bon fonctionnement du système, d'assurer sa sécurité et de mémoriser ses performances.

Les deux activités travaillent sur des données qui représentent le procédé sous des points de vue différents. L'activité "commander" se sert des modèles pour déterminer les commandes à envoyer au procédé. Ces modèles représentent donc l'état actuel du système par rapport aux objectifs de production. L'activité "surveiller" a besoin des modèles pour déterminer la bonne exécution d'un ordre envoyé aux actionneurs de la partie opérative et d'en examiner les conséquences.

Un ordre fait évoluer le procédé. Le résultat de l'évolution est transmis par des capteurs vers le système de surveillance. Cette activité détermine si l'évolution s'est déroulée de la manière attendue ou si une erreur est survenue. Le résultat de ce raisonnement est transmis vers l'activité "commander". Un compte-rendu positif traduit la bonne exécution de l'ordre et permet

l'évolution de la commande. Si par contre une erreur s'est produite sur la partie opérative, la surveillance met en œuvre des stratégies de recouvrement. Ces actions modifient soit les modèles de la commande, soit l'état du procédé à travers l'activité "commander". Le flux descendant d'information signale par conséquent un changement voulu de l'état du procédé, le flux ascendant transmet les résultats obtenus par l'exécution de ces ordres.

Les objectifs de production déterminent les modèles nécessaires à la conduite du système dans un **mode opératoire** donné. Ce mode opératoire, définit donc les modèles de référence pour une production demandée. Le projet CASPAIM étant basé sur une approche produit, le mode opératoire du système est défini par la nature des produits à fabriquer.

Les actions de recouvrement modifient les modèles des activités. Leurs modèles de référence sont tenus à jour afin de représenter à chaque instant l'état courant du système par rapport du mode opératoire courant.

Dans les paragraphes qui suivent nous présenterons d'abord l'activité "surveiller". Dans un deuxième temps nous décrirons l'activité "mettre à jour", qui relie les activités "surveiller" et "commander". En fin nous détaillerons l'activité "commander".

### 3.2.1. Surveiller

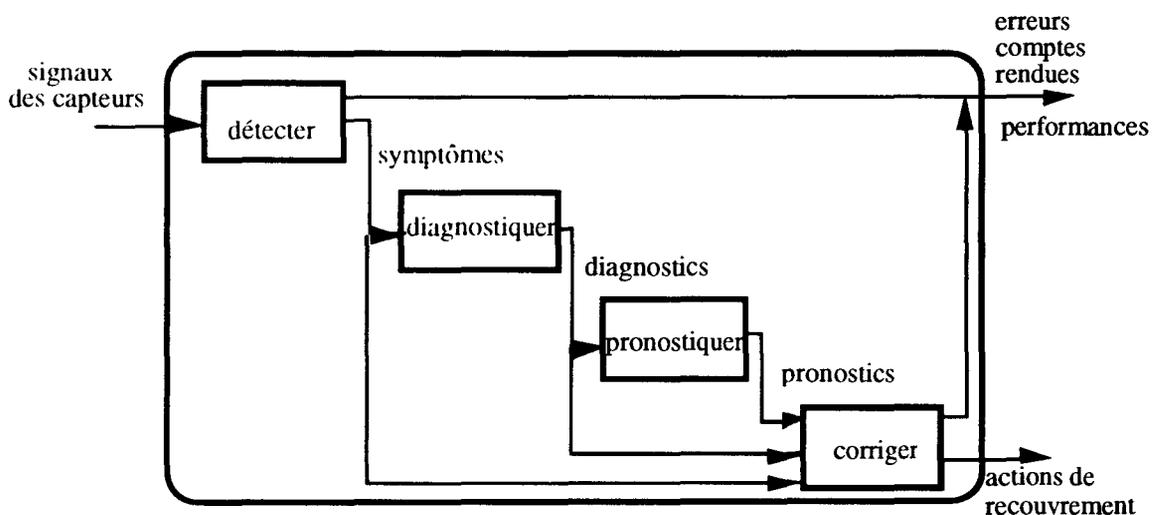


Figure I.5 : l'activité "surveiller"

La figure I.5 illustre la décomposition de l'activité "surveiller" au sens des propositions du projet CASPAIM [Toguyeni 1992].

La détection est l'activité la plus proche du procédé. Elle établit les conséquences d'un ordre envoyé au procédé, c'est à dire le changement d'état de la partie opérative suite à cet ordre et est par conséquent soumise à de fortes contraintes temporelles.

Dans le cadre d'une surveillance séparée, c'est-à-dire d'une surveillance qui n'est pas intégrée à la commande, la détection compare le comportement attendu du procédé avec son comportement réel [Combacau 1991]. Les résultats de cette comparaison sont ensuite transmis à la commande et, en cas d'erreur, aux autres activités de la surveillance sous formes de symptômes.

L'activité "diagnostiquer" localise et identifie le composant qui a causé l'erreur à partir de ces symptômes. L'activité "pronostiquer" établit finalement la portée de la défaillance et prédit ses conséquences.

L'activité "corriger" définit les actions de recouvrement. Elle combine les différentes informations issues des autres activités pour établir le recouvrement adéquat. Les actions de recouvrement assurent, d'une part, la sécurité du système, et, d'autre part, augmentent la disponibilité du système par la mise en œuvre de stratégies de recouvrement "intelligentes".

Ces stratégies exploitent de façon opportune les différentes formes de flexibilité du système : la polyvalence des ressources de production, qui permet la réalisation d'une opération sur plusieurs ressources, et la flexibilité des moyens de transfert, grâce à laquelle un produit peut emprunter différents chemins reliant ces ressources. Cette propriété des SFPM donne la possibilité de fabriquer un produit suivant une multitude de séquences. Elle permet également de continuer la production malgré les défaillances et les dégradations.

L'occurrence d'une erreur sur le procédé nécessite la mise en œuvre des actions de recouvrement en fonction de la gravité de l'erreur. Si, par exemple, l'ouverture d'une butée a échoué, le recouvrement peut simplement consister en une relance de l'ordre d'ouverture. Si par contre l'outil d'une machine s'est cassé pendant une opération de tournage, le recouvrement de cette erreur nécessitera un arrêt de l'avance du porte-outil ou un arrêt d'urgence de la machine. Dans le premier cas, l'action de recouvrement va modifier le comportement de la commande, alors que dans le second cas, elle va modifier directement le comportement du procédé.

Nous distinguons par conséquent les actions de recouvrement qui agissent directement sur le procédé, c'est à dire qui **commandent** le procédé, des actions qui modifient la commande, c'est-à-dire qui paramètrent ses modèles afin de mettre en œuvre les stratégies de maintien de la

production.

Les actions qui commandent directement le procédé comportent les différentes formes d'arrêt d'une ressource (arrêt en fin de cycle, arrêt dans un état déterminé, arrêt d'urgence) [Toguyeni 1992].

Les actions de recouvrement qui modifient les modèles de la commande sont réparties en trois familles [Sahraoui 1987, Craye 1989, Sahraoui 1992] :

- \* reprise,
- \* gel,
- \* reconfiguration dynamique.

Ces actions mettent en œuvre des procédures de maintien de la production (cf. Guide d'Étude des Modes de Marches et d'Arrêt [ADEPA 1979]). Afin de préciser quelles sont les interdépendances existant entre le recouvrement, donc la surveillance, et la commande, nous présenterons ci-après les actions correspondantes.

## **Reprise**

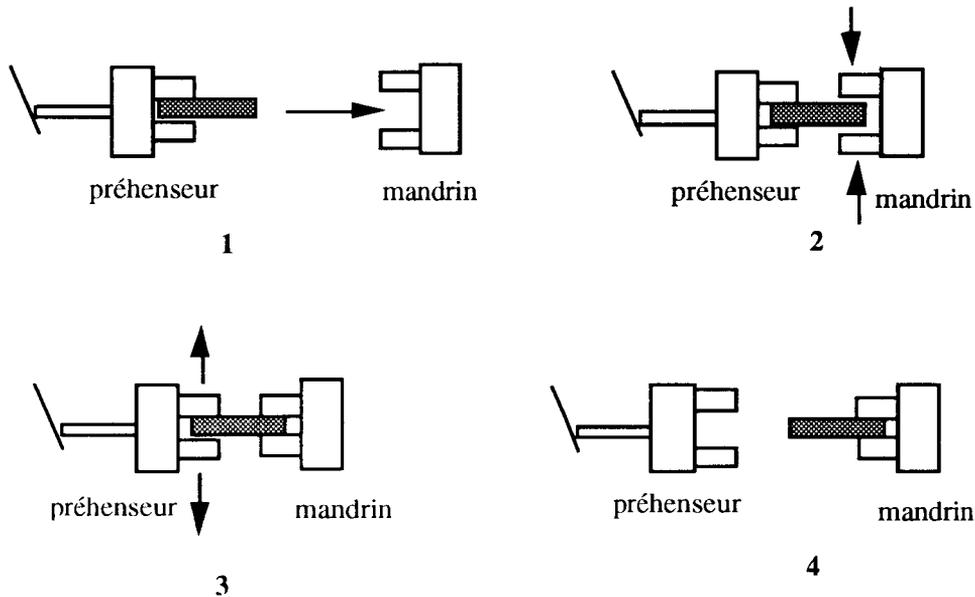
Dès détection d'une faute, le système est remis dans l'état précédent, exempt d'erreur, qui constitue l'un des points de reprise. Cette possibilité de recouvrement dépend du comportement du procédé, c'est-à-dire de l'existence d'actions réversibles.

Par la suite, nous détaillons des actions de recouvrement qui permettent d'atteindre des points de reprise. Nous distinguons deux types différents de points de reprise : les points de reprise d'une ressource et les points de reprise d'un produit. Dans les deux cas, la ressource et le produit sont ramenés dans un état stable et défini qui permet par la suite de continuer la production en exploitant la flexibilité du procédé.

### **a) Points de Reprise d'une Ressource**

Considérons l'exemple du chargement d'une pièce sur un tour par un manipulateur (figure I.6). Le manipulateur positionne la pièce dans le mandrin du tour (1), qui ferme ensuite ses mors autour de la pièce par un mouvement de translation (2). Après la fixation de la pièce dans le mandrin du tour, le manipulateur relâche la pièce (3) et s'écarte (4).

Nous nous plaçons dans le cas où le chargement ne s'est pas réalisé suite à un dysfonctionnement du tour : la fermeture des mors a échoué. Si les mors ne sont que partiellement fermés autour de la pièce, le recouvrement peut effectuer une deuxième tentative de fermeture, ou il peut commander une réouverture.



**Figure I.6 : chargement automatique d'un tour**

Si le recouvrement commande l'ouverture des mors et si cet ordre est réalisé, le procédé est ramené dans un état stable et exempt d'erreur, car la pièce est entièrement tenue par le manipulateur. Cet état correspond à l'échec complet de la fermeture.

Au niveau de la commande il s'agit, par conséquent, de représenter d'une part l'état actuel de la ressource et, d'autre part, de mettre à disposition les différentes solutions envisageables à partir de cet état. Dans le cas de notre exemple, la représentation cohérente de l'état de la ressource est conditionnée par une modélisation fine de son comportement. Les solutions restantes à partir d'un état donné sont représentées par une mise en évidence des actions réversibles. Nous constatons, que la réversibilité de certaines actions n'est qu'une forme de flexibilité du procédé qui est liée aux opérations positionnelles.

#### b) Points de reprise d'un produit

Reprenons l'exemple du chargement par un manipulateur (figure I.6). Si la défaillance du tour rend l'ordre de chargement définitivement impossible, nous devons ramener le produit dans un état stable afin d'envisager de nouvelles solutions et de libérer le manipulateur. Puisque

le manipulateur est dans un état stable et comme il est capable d'exécuter des transferts bidirectionnels, nous pouvons commander un retour en arrière et un rechargement de la pièce, sur son point de départ.

Cette possibilité de recouvrement permet de ramener le produit dans un état stable et défini, sur son point de départ. A partir de cet état stable le traitement du produit sur une autre ressource est possible. Le retour en arrière permet donc d'atteindre un point de reprise, mais cette fois du point de vue produit.

L'exploitation de cette possibilité de recouvrement au niveau de la commande est liée à la modélisation fidèle de la flexibilité des ressources de transport. Si nous avons prévu le retour en arrière, possible grâce à la flexibilité du manipulateur, et si nous l'avons intégré dans le modèle de la commande, la reprise peut être commandée et le produit peut être ramené dans un état stable.

Comme nous avons pu le montrer, la possibilité d'atteindre un point de reprise est liée à la représentation des opérations positionnelles réversibles au niveau des modèles de la commande. Il est nécessaire de représenter les retours en arrière, qui permettent de ramener le système dans un état stable et exempt d'erreur. Dans ce cas, les modèles indiquent l'état obtenu et les actions qui mènent dans cet état. A partir de ce point de reprise l'exploitation de la flexibilité du procédé est possible.

## **Gel**

Le gel de la commande est une action de recouvrement, qui interdit d'envoyer certains ordres au procédé afin de protéger les équipements à la suite d'une erreur. Dans le cas de notre exemple de chargement, si le dysfonctionnement du tour persiste et si nous arrêtons de ce fait le tour, il est inutile d'envoyer d'autres commandes de serrage ou de tenter un chargement de cette ressource de production. Les ordres de transfert vers le tour doivent être interdits et les séquences de commande correspondantes temporairement verrouillées.

Ce verrouillage de certaines commandes porte exclusivement sur les modèles de la commande. Les modèles sont mis à jour afin de n'indiquer que les solutions réellement restantes et intègrent, par conséquent, un mécanisme de verrouillage traduisant le gel d'une partie de la commande.

Ce mécanisme de verrouillage peut se traduire par une condition fonctionnelle d'envoi de

commande (prédicat) ; le verrou consiste à rajouter une condition supplémentaire. [Sahraoui 1992].

### **Reconfiguration Dynamique**

Nous entendons par reconfiguration dynamique le remplacement d'un module de commande en cours d'exécution par un autre qui est utilisé en cas d'anomalie [Sahraoui 1992].

La configuration dynamique correspond à un traitement d'exception, qui est sollicité par l'occurrence d'une erreur provenant du procédé. Au niveau de la commande, ce traitement est équivalent à l'exécution d'une séquence possible de commandes, grâce à la nature et l'état actuel du procédé. Si nous modélisons fidèlement la flexibilité du procédé, le traitement d'exception sera intégré dans les modèles de la commande.

La reconfiguration dynamique de la commande devient alors un reparamétrage de la commande dans le cas d'une défaillance. Ce reparamétrage remplace la séquence des ordres prévue par une nouvelle séquence. Puisque cette nouvelle séquence est représentée sur les modèles de la commande, on aurait donc pu la choisir dès le départ. Des stratégies opportunes de reconfiguration permettent l'exploitation de la flexibilité potentielle du système afin de maintenir la production, elles sont mises en œuvre par cette forme de recouvrement.

Nous avons mis en évidence les interdépendances entre la flexibilité du système, sa modélisation au niveau de la commande et son exploitation par les mécanismes de recouvrement. Ces mécanismes caractérisent l'activité "recouvrir" et sont mis en œuvre à travers la commande. Il est évident, que la disponibilité du système dépend essentiellement de la réalisation de cette activité. D'un autre point de vue, seule la prise en compte des interactions entre la commande et la surveillance dès la phase de spécification peut conduire à une modélisation du système au niveau commande qui puisse satisfaire les besoins du recouvrement.

Bien que l'activité "surveiller" soit vue le plus souvent par rapport aux erreurs des ressources de production, elle doit être aussi envisagée du point de vue des produits sous forme du suivi de leur comportement [Chassang et Tron 1983]. Il s'agit dans ce cas de détecter le retard d'un produit par rapport à la production planifiée et d'en pronostiquer les répercussions sur les autres produits, ou encore de détecter un manque de qualité, qui peut solliciter une action de recouvrement sous forme d'une retouche.

### 3.2.2. Mise à Jour des Modèles

Cette activité modifie les modèles en conformité avec l'évolution réelle du procédé. En effet, seule une information **fiable** concernant l'état actuel du procédé permet la mise en œuvre des stratégies de recouvrement, comme nous l'avons vu dans le paragraphe précédent.

En fonction des quantités relatives des produits à fabriquer, c'est-à-dire du point de fonctionnement désiré du système, les activités "commander" et "surveiller" ont besoin de modèles, qui mémorisent les informations nécessaires à la gestion du système. Mais le point de fonctionnement de ces systèmes est sujet à des modifications incessantes. Ces modifications sont issues, soit d'un changement qualitatif et/ou quantitatif de la production, soit d'une dégradation des performances du système de production.

Un changement qualitatif de la production affecte la nature des produits et correspond donc à un changement du mode opératoire. Un changement uniquement quantitatif de la production affecte les quantités produites et induit une modification du point de fonctionnement à l'intérieur d'un mode opératoire.

Le changement de mode opératoire est par conséquent un changement **commandé** suite à une modification qualitative des objectifs de production. Nous avons introduit un mode opératoire comme la donnée de la nature des produits à fabriquer. Le changement du mode opératoire correspond donc à l'arrêt ou au lancement de la fabrication d'une ou de plusieurs gammes de produit.

Le lancement de fabrication demande une configuration adéquate de l'atelier, qui permet de réaliser les opérations positionnelles (transports) et transformationnelles (usinages, assemblages) nécessaires pour la fabrication du produit demandé. L'arrêt d'une gamme de production nécessite l'évacuation de toutes les pièces restantes de ce type dans l'atelier et éventuellement des procédures de clôture pour certaines ressources de production qui ne sont plus utilisées.

Il s'agit donc d'une part de mettre le **procédé** dans un état cohérent avec la production demandée. Il s'agit, par exemple, de mettre à disposition les outils nécessaires dans le cas des machines outils à commande numérique, ou des supports de fixation et de transport adaptés aux produits dans les cas des moyens de manutention. D'autre part, il y a nécessité de configurer la **commande** dans l'état demandé. Au niveau des machines à commande numérique par exemple, les programmes correspondants aux transformations à réaliser sur le nouveau produit

doivent être télé-chargés et des cycles de tests avant une reprise sont éventuellement effectués.

Un mode opératoire détermine de ce fait l'ensemble des données nécessaires à la conduite du système dans ce mode. Ces données sont mémorisées au niveau des modèles de commande et de surveillance. Un changement de mode sollicite une mise à jour des modèles de conduite, donc une définition des modèles courants de référence.

Cependant l'activité "mettre à jour" des modèles est aussi sollicitée par les flux d'information remontant de la surveillance : les actions de recouvrement qui agissent sur le procédé et les actions qui modifient les modèles (verrouillage, reparamétrage). Une réalisation de cette activité doit, par conséquent, tenir compte de cette combinaison de flux descendants et ascendants d'information : le flux descendant propage l'information concernant un changement de mode jusqu'au niveau des modèles des actionneurs, le flux ascendant regroupe les différentes informations liées à une défaillance afin d'en établir les répercussions sur le fonctionnement du système entier.

Les différentes activités de la conduite ont besoin de modèles qui représentent le même système sous des points de vues différents et à des niveaux d'abstraction différents. L'activité de mise à jour doit assurer la cohérence de l'information dans tous les modèles, c'est-à-dire, si la surveillance a engagé l'arrêt d'une ressource, l'état de cette ressource doit être instantanément déclaré à "en arrêt" dans tous les modèles.

La modélisation du système physique à différents niveaux d'abstraction nécessite une prise en compte des liens de dépendance entre les différentes entités. Si un actionneur d'une ressource de production est en panne, la ressource est hors service. L'information de la panne de l'actionneur, présente à son niveau de modélisation, doit être propagée au niveau des modèles des ressources le plus rapidement possible.

La mise à jour des modèles est donc soumise à de fortes contraintes de temps, car la propagation de l'information concernant les conséquences d'une défaillance, par exemple d'une ressource, doit être immédiate, afin de disposer d'une vue cohérente et actuelle de l'état du procédé au niveau des modèles.

Afin de prendre en compte la forte contrainte de temps, nous favorisons une approche de type langage synchrone [Berry et al 1987] pour la mise à jour des modèles comme elle a été présentée dans [Ausfelder et al 1993b]. Cette approche exploite des liens de dépendance fonctionnelle entre les différents constituants du procédé afin de diffuser de manière instantanée

les informations concernant le changement de mode du système.

La représentation de l'information a donc besoin d'une identification de ces liens de dépendance. Dans le cadre du projet CASPAIM, la description des liens de dépendance adoptée introduit la configuration d'une ressource comme concept de base [Kermad 1992]. Dans le cas d'une machine outil à commande numérique, ce concept rajoute à la prise en compte de l'état d'une ressource et de ses actionneurs, la disponibilité des programmes définis pour cette configuration et la disponibilité des outils nécessaires.

La réalisabilité d'une opération est déterminée par la disponibilité de la configuration correspondante. Un produit peut être fabriqué si toutes ses opérations sont réalisables. Finalement un mode opératoire est réalisable si tous les produits le composant peuvent être fabriqués.

Les liens de dépendance identifiés par cette modélisation permettent ainsi la mise en oeuvre des procédures de configuration de l'atelier à partir de la définition d'un mode opératoire. Ces procédures comportent la configuration matérielle de l'atelier et la configuration logique, c'est-à-dire la définition des modèles de référence. Par ailleurs, ces liens sont exploités pour maintenir la cohérence de l'information dans les différents modèles de la commande et de la surveillance.

Une autre approche exploite les liens structurels existant entre les ressources de production et leurs unités de commande [Bois 1991, Parayre 1992]. Dans ces approches, un regroupement des ressources en cellules et ateliers est effectué. A chaque élément structurel est associé un gestionnaire des modes de marche et d'arrêt qui, suivant les propositions du GEMMA, met en oeuvre les actions de configuration des ressources. Le gestionnaire associé à un niveau donné a pour charge d'assurer la cohérence de l'information dans les gestionnaires qui lui sont subordonnés.

Outre la modélisation de la flexibilité, l'intégration d'un mécanisme de mise à jour des modèles de la commande, qui rend l'information fiable, permet la mise en oeuvre des stratégies de maintien de production, qui, face à des défaillances, augmentent la disponibilité du système et contribuent à une conduite performante.

### 3.3. L'ACTIVITÉ "COMMANDER"

Nous proposons une décomposition de cette activité en trois sous-activités (figure I.7) :

- \* allouer,
- \* coordonner,
- \* commander les actionneurs.

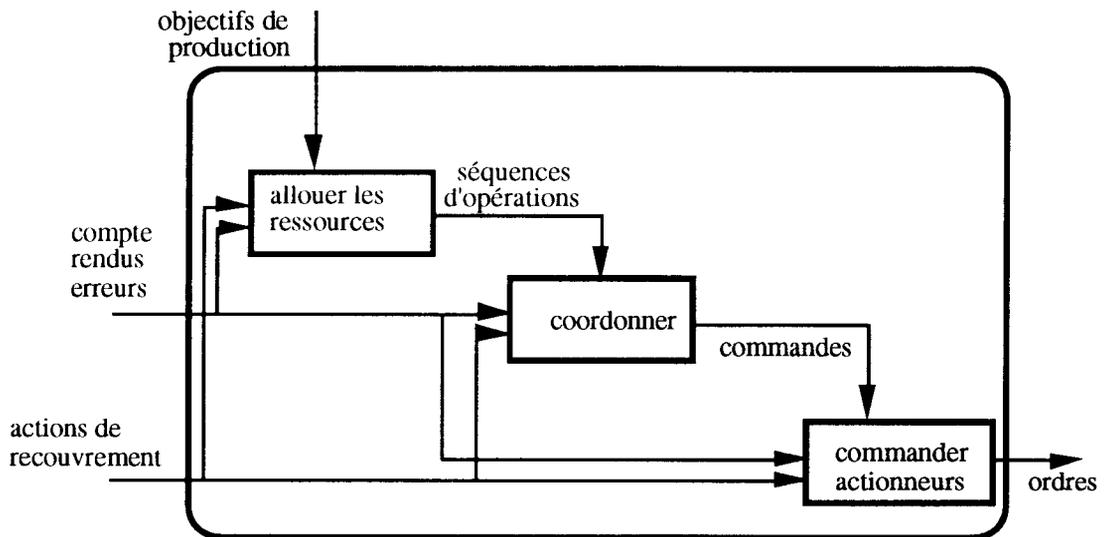


Figure I.7 : l'activité "commander"

#### 3.3.1 Allouer les Ressources

L'allocation des ressources transforme les objectifs de production en séquences d'opérations. Ces séquences définissent l'ordre de passage des produits sur les ressources. Outre le choix d'ordre existant entre plusieurs produits, la flexibilité des ateliers introduit des choix pour chaque produit concernant la ressource sur laquelle une opération sera réalisée. Ces choix existent aussi bien au niveau des machines qu'au niveau des moyens de transport pour ce qui concerne le routage des produits. L'allocation définit donc l'affectation temporelle et spatiale de tous les produits du plan de fabrication sur les ressources de l'atelier.

Du fait de ce caractère décisionnel, l'allocation conditionne le bon fonctionnement du système. Les degrés de liberté du procédé peuvent être utilisés pour optimiser son

fonctionnement. Cette optimisation est effectuée par rapport à des critères temporels exprimant des critères économiques [Fandel et al 1990]. Parmi les critères les plus courants citons la minimisation des retards (= satisfaction des clients), ou la minimisation de la durée de fabrication pour l'ensemble des ordres défini dans le plan de fabrication (= limitation de l'encours) [Bruns et Appelrath 1991].

Une optimisation de l'exploitation de l'atelier par rapport à des objectifs donnés définit l'affectation prévisionnelle de chaque opération, à une et une seule ressource.

Mais d'autres solutions sont envisageables. Il existe des solutions qui optimisent, de manière locale le fonctionnement du système, comme, par exemple, une allocation sur requête et par calcul de priorité des produits [Panwalker et Iskander 1977]. D'autres solutions proposent une allocation par règles de production, qui, en cas de conflit, définissent le produit prioritaire [Castelain et Gentina 1988].

Si nous choisissons une optimisation globale par affectation prévisionnelle, le problème d'allocation des ressources dans les SFPM peut être classé dans les problèmes d'ordonnancement de type "job-shop" [Bénassy 1987]. En effet :

- \* les opérations de chaque produit sont liées par un ordre total,
- \* cet ordre n'est pas nécessairement identique pour tous les produits,
- \* une ressource ne peut exécuter qu'une opération à la fois (contrainte de non-chevauchement).

Le problème de l'ordonnancement de type "job-shop" est un problème NP-complet, c'est-à-dire que le temps de calcul d'une solution optimale croît d'une manière exponentielle avec le nombre d'opérations à planifier.

Afin de maîtriser la complexité de calcul, des approches de décomposition ont été proposées pour résoudre le problème d'allocation d'une manière prévisionnelle. Nous identifions globalement trois types de décomposition :

- \* la décomposition temporelle,
- \* la décomposition spatiale,
- \* la décomposition fonctionnelle, séparant l'allocation des transformations de l'allocation des transports.

Les approches de **décomposition temporelle** réduisent la complexité du problème

d'affectation par une limitation des ordres à ordonnancer en même temps. Dans un premier temps un sous-ensemble d'opérations est défini, qui fait partie d'une même fenêtre temporelle. Toutes les opérations de ce sous-ensemble sont ensuite ordonnancées ensemble. Au lieu de rechercher une solution globale pour toutes les opérations définies dans les objectifs de production, des sous-ensembles d'opérations, éventuellement permutable, sont ainsi affectés successivement [Roubellat et Thomas 1988].

Les approches de **décomposition spatiale** affectent les opérations sur un sous-ensemble de ressources de l'atelier. Nous pouvons distinguer les approches qui recherchent des groupes à partir des opérations définies pour une famille de produits (groupements en îlots), de celles qui s'inspirent de l'organisation de l'atelier.

La recherche d'îlots part de la séquence d'opérations à effectuer sur chaque produit. On définit comme îlot "logique" un groupe de machines, qui est toujours visité dans le même ordre par une famille de produits pour réaliser un ensemble d'opérations. Un produit appartenant à une famille est ainsi affecté à l'îlot correspondant [Portmann 1988].

Le groupement en îlots nécessite une répétitivité de la production afin de déterminer les familles de produits. Cette répétitivité existe dans les SFPM de par la production en petite série.

Le deuxième type de décomposition spatiale du problème d'allocation s'inspire de l'organisation des ateliers en structures autonomes, nommées cellules flexibles et postes de travail. Au niveau atelier une première étape d'allocation est effectuée, qui permet d'affecter la réalisation d'une ou de plusieurs opérations dans les cellules [Brandimarte et al 1992]. Une deuxième étape réalise l'affectation des opérations aux ressources à l'intérieur de chaque cellule.

Cette approche décentralisée répond au souci d'augmenter l'autonomie des cellules. Elle tient ainsi compte des propriétés spécifiques des SFPM : leur organisation en unités autonomes.

La décomposition **fonctionnelle** est presque toujours combinée avec les autres approches présentées et sépare l'allocation des ressources de transformation de l'allocation des moyens de transport. Cette approche réduit la complexité du problème initial par une limitation des opérations à effectuer à chaque étape. Mais le problème de l'affectation des transformations reste toujours un problème de type job-shop.

Les modèles en tant que données de l'allocation doivent représenter, d'une part, les contraintes d'un problème de type job-shop et, d'autre part, les approches de décomposition

choisies.

Dans le cadre du projet CASPAIM, une allocation prévisionnelle des transformations (ordonnancement), dans un premier temps, puis une allocation des transports, dans un deuxième temps, est réalisée [Hammadi 1991]. Les modèles de l'allocation représentent donc cette décomposition fonctionnelle. Le modèle de l'ordonnancement représente le processus de fabrication comme une séquence de transformations à réaliser sur un produit par différentes ressources de production.

Le modèle d'allocation des moyens de transport nécessite une vue globale du procédé, notamment de l'ensemble des moyens de transport présents dans un atelier, leur disponibilité actuelle et éventuellement leur localisation. L'allocation des moyens de transport met ensuite en œuvre des mécanismes de résolution comme une allocation prévisionnelle [Fu et Liu 1992] ou une allocation par calcul de priorité [Tawegoum et al 1992].

Les modèles influencent fortement la réalisation de l'allocation. Un système de transport, par exemple, qui dispose de différents chariots automoteurs ou de palettes à circulation libre, peut demander une gestion sophistiquée. Les modèles comme données des algorithmes d'allocation doivent être adéquats, afin de mettre à disposition des informations telles que les encombrements actuels ou l'évolution attendue du système de transport.

L'allocation des ressources reste également nécessaire en mode dégradé de l'atelier. Ses modèles doivent intégrer de ce fait la flexibilité potentielle du système relative à cette activité, comme par exemple les routages possibles d'un produit. Les modèles forment ainsi la base d'une exploitation optimale du système par les mécanismes de résolution de l'activité "allouer".

### **3.3.2. Coordonner**

L'activité "allouer" revient à définir des séquences d'opérations à réaliser sur chaque ressource et, de ce fait, l'ordre et la date de passage de chaque produit sur les ressources. A partir de ces séquences déterministes, l'activité "coordonner" gère l'interaction entre les différentes ressources de l'atelier et synchronise leurs actions élémentaires afin d'obtenir un fonctionnement cohérent de l'atelier [Combacau 1991].

La coordination comporte deux aspects : la coordination des opérations à effectuer sur un produit et la coordination des actions élémentaires réalisées par les ressources. L'aspect produit de la coordination assure, par exemple, la réalisation de la "bonne" transformation après un

transfert et après cette transformation le "bon" routage du produit vers le lieu opératoire suivant. Cet aspect de la coordination porte donc sur la coordination des opérations par rapport à un même produit.

La succession des opérations est, pour un type de produit, une donnée statique, le routage des produits dépend de la décision de l'allocation. Les modèles de la coordination doivent donc connaître la séquence des opérations transformationnelles de chaque produit et tous les routages possibles entre deux lieux opératoires.

L'autre point de vue concerne la bonne coopération des ressources de production. Cette coopération ne dépend plus du produit pris en charge par la ressource, mais de la nature de chaque ressource. Le chargement automatique d'un tour par un manipulateur demande une étroite coopération de ces deux ressources (tour et manipulateur), un chargement d'un produit sur un stockeur n'en demande aucune.

L'aspect coordination des ressources nécessite donc une modélisation des opérations au niveau des actions élémentaires réalisées par les actionneurs de chaque ressource et doit tenir compte du comportement spécifique de chacune.

La coordination gère ainsi les flux des produits et réalise leur routage dans l'atelier. Elle décompose les opérations en actions élémentaires en tenant compte du comportement spécifique de chaque ressource.

### **3.3.3. Commander les Actionneurs**

"Commander les actionneurs" représente la plus basse couche de la commande. Cette partie de la commande est reliée directement à la partie opérative. Elle met en œuvre les actions engagées par la coordination.

Ces actions sont à caractère déterministe et soumises à de fortes contraintes temps réel. La commande des actionneurs est en fort couplage avec le matériel, ce qui lui confère une importance capitale pour les problèmes de surveillance [Combacau 1991]. En effet, les informations remontant dans l'activité "conduire" sont des résultats de l'interprétation des signaux sollicités par des commandes et émis par des capteurs. Le caractère déterministe et la contrainte de temps et de réponse immédiate justifient donc une approche "synchrone" pour la réalisation de cette fonction [Péraldi 1993, Elkhatabi et al 1992].

Les modèles de cette activité conditionnent de façon importante la mise en œuvre des mécanismes de reprise et de traitement d'erreur. Un point de reprise ne peut être atteint, que si les commandes ont été prévues afin de ramener le procédé dans un état stable et exempt d'erreur. Par ailleurs, les commandes, qui permettent d'atteindre ce point de reprise, doivent respecter les contraintes imposées par un fonctionnement sûr du procédé.

Les modèles doivent donc exprimer, d'une part, le caractère déterministe de cette activité et, d'autre part, indiquer les actions réalisables à partir d'un état, y compris les retours en arrière permettant d'atteindre un point de reprise. Une fidèle modélisation des actions réalisables à partir de l'état d'un actionneur est la condition sine qua non pour la mise en œuvre des mécanismes de recouvrement d'erreur et, par conséquent, pour une gestion satisfaisante des modes opératoires. La séquence réelle des ordres est définie ensuite par la coordination ou évoquée directement par les actions de recouvrement (reconfiguration dynamique).

### 3.3.4. Conclusion

Nous avons mis en évidence à travers les activités "allouer", "coordonner" et "commander les actionneurs" la décomposition de l'information à l'intérieur de la commande. Au niveau de l'allocation, une affectation des opérations est effectuée, qui détermine l'ordre de passage des produits sur les ressources de l'atelier. Les modèles d'allocation représentent donc les contraintes d'ordre de chaque produit et la contrainte de non-chevauchement temporel des opérations sur une machine. Une approche hiérarchisée de l'allocation doit se retrouver au niveau des modèles. Une solution de l'activité "allouer", qui sépare l'allocation des ressources de transformation de celle des ressources de transport, a besoin de modèles représentant cette décomposition.

La coordination comporte deux aspects. L'aspect produit coordonne les opérations à réaliser sur un produit donné et ses modèles représentent la séquence des opérations à réaliser sur un produit. La coordination des opérations inclut la coordination des transferts entre les lieux opératoires de l'atelier. Le deuxième aspect de la coordination concerne la coordination des ressources. Cet aspect de la coordination consiste à synchroniser les actions élémentaires des ressources de production afin de réaliser des transferts de pièces entre les ressources de production. Ses modèles tiennent de ce fait compte du comportement spécifique de chaque ressource.

Les actions élémentaires sont finalement transmises aux actionneurs de la partie opérative. Les modèles de cette activité ne représentent que le comportement des ressources, l'aspect

produit n'apparaît plus à ce niveau de la commande.

## **4. UN CAHIER DES CHARGES POUR LA COORDINATION**

### **4.1. INTRODUCTION**

Ce paragraphe concerne l'exploitation des résultats du modèle de la conduite pour la conception de la coordination. Une analyse similaire permettra la définition d'un cahier des charges détaillé pour la conception des autres activités.

Les deux aspects de la coordination, coordination des opérations par rapport à un produit et coordination des actions élémentaires des ressources, sont, de manière générale, réalisés par le logiciel de commande [Kapusta 1988, Bourey 1988, Craye 1989, Cruette 1991]. De ce fait le cahier de charges de la coordination comporte des aspects issus du génie logiciel.

### **4.2. LES CONTRAINTES DE QUALITÉ D'UN SYSTÈME DE COORDINATION DU POINT DE VUE GÉNIE LOGICIEL**

Le génie logiciel est l'application des méthodes scientifiques au développement de théories, de modèles, de techniques, de langages et d'outils favorisant la production, le développement, l'exploitation et la maintenance de logiciels de qualité [Meyer 1990]. Il couvre l'ensemble des activités de spécification, de conception et de mise en œuvre des produits et des procédures tendant à rationaliser la production des logiciels et leurs suivis.

Le premier souci du concepteur d'un système d'information est, d'après la définition de MEYER, la production de logiciels de qualité. Cette qualité peut être jugée en termes de modifiabilité, efficacité, fiabilité et intelligibilité. [Ross et al 1975, d'après Booch 1988].

Le génie logiciel a défini des techniques de base permettant la conception des logiciels de qualité [Booch 1991] :

- \* l'abstraction et l'encapsulation de l'information : l'objectif est d'extraire les propriétés essentielles en omettant les détails inutiles à un niveau d'abstraction donné et de rendre

visibles seulement les informations nécessaires à ce niveau.

- \* modularité et localisation : l'objectif est de concevoir un ensemble de modules à faible couplage (limitation des échanges inter-modules), à forte cohésion (liaison étroite des éléments du même module) et réalistes (au sens d'un modèle effectif du monde réel),
- \* uniformité, intégralité et validité : l'objectif est d'aboutir à des systèmes cohérents et corrects par l'emploi d'une notation cohérente. Les modules doivent être complets en regard de leurs spécifications et validables.

La démarche de conception de la commande de coordination présentée dans le chapitre suivant tient compte de ces spécificités. Cette démarche aboutit à un modèle hiérarchisé et modulaire en utilisant un outil unique de modélisation, les réseaux de Petri [Brams 1983]. Elle contribue ainsi à la qualité du logiciel de la commande. La qualité de la solution est établie par une démarche de validation des modèles conçus, qui est présentée dans le chapitre III.

### **4.3. LES BESOINS DE LA COORDINATION ISSUS DU MODÈLE DE LA CONDUITE**

Outre les besoins de qualité valables aussi pour les autres fonctions de la conduite réalisées par logiciel, les besoins de la coordination, issus de la modélisation de ce système complexe, au sens où nous l'avons défini dans le paragraphe 2.2, sont, dans un premier temps, spécifiques à cette activité. Par ailleurs, seule l'analyse de la coordination dans son environnement nous a permis de dégager ces besoins.

Nous avons insisté sur l'importance des modèles de la coordination et nous avons mis en évidence les relations existant entre la flexibilité du procédé, sa modélisation, son exploitation en mode dégradé et la mise à jour de ces modèles. Nous résumons ces résultats sous forme de deux caractéristiques souhaitées de la coordination : sûreté et adaptabilité [Ausfelder et al 1994].

#### **4.3.1. La Sûreté de la Coordination**

Le terme de **sûreté** caractérise une commande de coordination qui évite la propagation d'erreurs. Cette caractéristique est liée aux traitements des signaux en provenance de l'activité "détecter" de la surveillance.

L'activité "détecter" compare, à l'aide d'un modèle de référence, le comportement attendu du procédé avec son comportement observé à la suite d'un ordre. Cette comparaison permet de formuler un compte rendu d'exécution ou la détection d'une erreur le cas échéant.

La commande de coordination doit être impérativement synchronisée sur ce compte rendu afin d'éviter des erreurs consécutives à un dysfonctionnement du procédé et difficiles à corriger. Si, dans le cas du chargement automatique d'un tour (figure I.6), une pièce n'est pas bien positionnée dans le mandrin, une ouverture du préhenseur peut provoquer une erreur consécutive à la faute initiale au sens où la pièce peut tomber du tour.

Dans le cas d'une commande sûre, c'est-à-dire synchronisée sur les comptes rendus venant de l'extérieur, le manipulateur et la commande de coordination restent gelés dans un état d'attente jusqu'à nouvel ordre. La commande de fermeture des mors n'étant pas effectuée, l'état de la commande n'évolue pas. De cette manière une coordination sûre évite la propagation d'erreurs et ses modèles représentent à chaque instant l'état réel du procédé [Sahraoui 1992].

### 4.3.2. L'Adaptabilité de la Coordination

Nous qualifions une commande de coordination d'**adaptative**, si elle permet une exploitation de la flexibilité du procédé à chaque instant, c'est-à-dire que le système de coordination est susceptible de s'adapter en fonction de la dégradation des performances du système au sein d'un même mode opératoire.

Nous avons pu montrer, que la modélisation de la flexibilité permet, d'une part, d'atteindre des points de reprises dans le cas d'opérations réversibles (transports), et, d'autre part, de mettre en œuvre des actions de recouvrement, qui augmentent la disponibilité du système par une reconfiguration dynamique ou, plus précisément, par un reparamétrage de la commande.

Une seconde condition de l'adaptabilité concerne l'intégration d'un mécanisme de mise à jour des modèles au niveau de la commande de coordination. Ce mécanisme déclenche le verrouillage (ou déverrouillage) de certaines parties de la commande et garantit ainsi une représentation fidèle de l'état du procédé au niveau des modèles.

Si une commande de coordination respecte ces deux conditions, ses modèles représentent à chaque instant l'état actuel du procédé et les solutions restantes. La commande englobe ainsi la commande des modes dégradés au sein d'un mode opératoire, car les modes dégradés forment un sous-ensemble des modes possibles, représentés sur les modèles et exploitables par l'activité "commander" ou l'activité "recouvrir" de la surveillance.

Au niveau des fonctions de la commande, la sûreté est une caractéristique de la coordination et de la commande des actionneurs. Ces activités suggèrent une synchronisation sur les comptes

rendus et les erreurs, qui assurent une représentation fiable de l'état du procédé au niveau de leurs modèles.

La caractéristique d'adaptabilité, par contre, peut être définie pour l'allocation des ressources et la commande des actionneurs. Un modèle représentant la flexibilité du procédé au niveau de l'allocation et un mécanisme de résolution adéquat forment ainsi les conditions pour une allocation adaptative, c'est-à-dire d'une allocation qui s'adapte automatiquement à la dégradation des performances de l'atelier et qui contribue ainsi à un haut degré de disponibilité de l'atelier.

## 5. CONCLUSION

Le modèle du système de contrôle / commande, présenté dans ce chapitre, permet de souligner l'importance d'une approche globale de conception. Les interrelations des activités "surveiller" et "commander" dans le cas du recouvrement d'erreur ou la problématique de la cohérence de l'information dans les différents modèles en sont deux exemples significatifs. En effet, la définition d'un cahier des charges pour la coordination est une justification de la démarche proposée.

Nous avons choisi ici un formalisme de type SADT comme outil de modélisation, car il est particulièrement bien adapté à la représentation d'un modèle conceptuel. SADT ou son standard dérivé IDEF-0 est aussi proposé en phase d'étude préliminaire dans le projet AMICE (European CIM Architecture), projet ESPRIT visant une méthode de conception des usines intégrées ainsi que dans d'autres approches de conception des systèmes de production et de leur conduite [Ranky 1990, Roboam et al 1989, Parayre 1992, Tahon et Dindeleux 1992].

Le modèle présenté ici tient compte des spécificités des SFPM :

- \* le caractère "systèmes à événements discrets",
- \* la problématique de surveillance et surtout
- \* leur caractère flexible, qui s'exprime par la polyvalence des ressources et la variété des produits et des quantités fabriqués.

Par rapport au projet CASPAIM, le modèle proposé constitue une spécification des objectifs

du projet relative à l'intégration des différentes fonctions au sein d'un système de contrôle / commande. L'aspect de conception de la conduite du projet CASPAIM rejoint ainsi les objectifs du génie automatique [Panetto et al 1991, Morel 1992].

# **CHAPITRE II :**

## **UNE MÉTHODE POUR**

## **LA CONCEPTION DE LA COMMANDE DE**

## **COORDINATION DES SFPM**

### **1. INTRODUCTION**

Nous présentons, dans ce chapitre, une méthode de conception de la commande de coordination des SFPM. Cette méthode propose la représentation du comportement du système (procédé et produits) dans un ensemble de modèles réseaux de Petri qui sont adaptés chacun à un aspect spécifique de la coordination.

Les SFPM appartiennent, du point de vue de leurs modèles, à la classe des systèmes à événements discrets. La modélisation de ces systèmes s'attache à décrire les changements d'état et les conditions d'évolution des variables représentant le système physique. Le fait que l'on sache à tout instant quelles sont les évolutions possibles du système permet une modélisation de son comportement [Ramdage et Wonham 1989].

Si nous décrivons le comportement du système par une approche "systèmes à événements discrets", la fabrication des produits demande une synchronisation de processus parallèles. Un tournage est seulement réalisable, si le tour est libre et disponible et si le produit est présent. Les processus décrivant l'évolution dynamique du tour et le processus modélisant le comportement du produit dans le système doivent être synchronisés pour effectuer le tournage.

Cette propriété et le caractère indéterministe intrinsèque des modèles, introduit par la flexibilité du procédé, ont mené beaucoup d'équipes de recherche à utiliser les réseaux de Petri pour la modélisation des SFPM, aussi bien pour la conception de la commande que pour la conception de la surveillance [Schmidt et Glüer 1988, Silva et Valette 1990, Haralakis et al 1992, Bourey et al 1992]. Les réseaux de Petri sont un outil parfaitement adapté à la modélisation des systèmes à événements discrets [Cao et Ho 1990], qui permet la représentation de la synchronisation des processus parallèles et l'expression d'un comportement indéterministe du procédé. Le graphisme associé aux réseaux de Petri est favorable à une compréhension intuitive des modèles [Reisig 1982], et les propriétés mathématiques exploitées pour la validation formelle rendent cet outil encore plus puissant [Alla et al 1984].

Nous avons retenu des réseaux de Petri de haut niveau, les réseaux à Prédicat / Transition [Genrich 1986], comme outil de modélisation (annexe 3). Les réseaux de haut niveau agrègent la représentation de l'information et les modèles deviennent plus concis [Genrich et Lautenbach 1981]. L'utilisation des réseaux Prédicat / Transition permet, par l'intermédiaire de la notion d'individu associée aux jetons, l'identification individuelle des produits, modélisés par des jetons, à tout instant dans le système. Par rapport aux réseaux de Petri à Objets [Sibertin-Blanc 1985] dans lesquels évoluent aussi des jetons individuels, les réseaux de Petri à Prédicat / Transition représentent un formalisme suffisant dans le cadre de notre modélisation. L'existence d'un ensemble d'outils bien définis pour la validation formelle des réseaux de Petri à Prédicat / Transition a renforcé notre choix.

Dans le premier paragraphe de ce chapitre, nous décrivons d'une manière détaillée la démarche de modélisation. Ce paragraphe constitue ainsi le canevas du chapitre. Les différentes étapes de la modélisation seront illustrées à l'aide de l'exemple de la cellule flexible de l'École Centrale de Lille présentée en annexe 2.

## **2. LA DÉMARCHE PROPOSÉE**

### **2.1. INTRODUCTION**

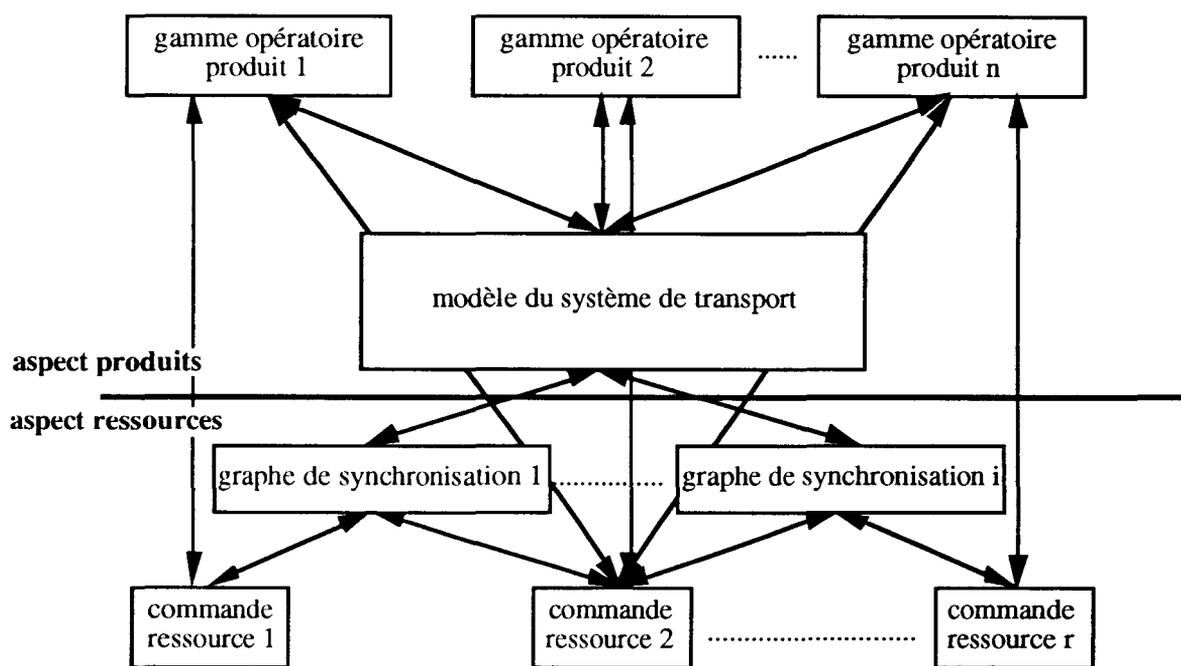
La méthode présentée dans ce chapitre propose une démarche rigoureuse pour la conception de la commande de coordination. Les modèles réseaux de Petri générés s'attachent à décrire le

comportement commandé du procédé à différents niveaux d'abstraction.

Comme nous l'avons vu dans le chapitre I, la commande de coordination au sein d'un système de conduite comporte principalement deux aspects :

- \* la coordination des opérations relatives à un produit afin d'obtenir les produits souhaités,
- \* la coordination des actionneurs des ressources afin d'obtenir un fonctionnement cohérent de l'atelier.

Ces deux aspects de la coordination nous ont amené à développer des modèles différents, qui combinent une approche produit de la description du procédé avec une approche décrivant le comportement des ressources de production (figure II.1).



**Figure II.1 : les quatre types de modèles**

Les modèles décrivant le comportement des produits sont appelés "gammes opératoires" et "modèle du système de transport", les modèles concernés par la coordination des ressources de l'atelier, "graphes de synchronisation" et "commande des ressources". Ces modèles communiquent d'une manière asynchrone, ce qui est représenté par les flèches de la figure II.1.

La méthode proposée couvre la génération de ces quatre modèles et la définition des interfaces de communication (connexion des modèles).

## 2.2. LA SPÉCIFICATION DES OBJECTIFS ET L'ANALYSE DE L'ATELIER

Comme dans toute démarche de conception, la première étape de la génération des modèles est une phase de spécification des objectifs de production (produits) et d'analyse de l'existant, c'est-à-dire de l'atelier.

A partir de la spécification, nous générons un modèle qualitatif, qui représente la fabrication d'un produit comme une succession d'opérations transformationnelles (**gamme logique**). Les gammes logiques expriment les contraintes d'ordres, c'est-à-dire les contraintes logiques du processus de fabrication et font abstraction de toute information quantitative concernant les objectifs de production (cf. paragraphe 3.1).

La spécification des objectifs, sous forme de gamme logique, est complétée par une analyse du procédé. Cette analyse part d'une description des ressources de production et définit leurs principales caractéristiques par une décomposition en sous organes opérants. Le résultat de l'analyse est donné sous forme d'une représentation arborescente de l'atelier (3.2.1).

Dans la deuxième étape, nous effectuons une analyse de l'architecture fonctionnelle de l'atelier basée sur les concepts de **lieux caractéristiques** et de **relation d'accessibilité**. L'information générée est synthétisée sur un modèle topologique du système, nommé **graphe des accessibilités du système**, qui permet l'identification des contraintes imposées par l'organisation de l'atelier (3.2.2).

La définition des principales caractéristiques des ressources est complétée par une description textuelle des contraintes de fonctionnement de chacune d'elles. Ces contraintes décrivent le comportement spécifique d'une ressource qui contraint l'évolution libre de ses sous-organes opérants [Elkhatabi 1993] (3.2.3).

## 2.3. LA GÉNÉRATION DES MODÈLES DE LA COORDINATION

### 2.3.1. Les Gammes Opératoires

Une gamme opératoire décrit, pour un type de produit, le processus de fabrication par une séquence de transformations et de transferts entre zones opératoires.

Les gammes opératoires sont déduites des gammes logiques et modélisent, d'une part la

flexibilité du produit, c'est-à-dire les différentes séquences de transformation, qui mènent au même produit fini. D'autre part elles incluent la polyvalence des ressources transformationnelles, car elles indiquent pour chaque opération d'un produit toutes les machines capables de sa réalisation.

Vis-à-vis des transports, une gamme opératoire décrit la réalisabilité d'un transfert entre deux zones opératoires au sein de l'atelier. Les différentes possibilités de routage et les opérations positionnelles nécessaires pour réaliser un transfert précis sont mises en évidence sur le modèle du système de transport.

Une gamme opératoire définit ainsi un objectif particulier de la production, c'est-à-dire une exploitation voulue du système, par une approche produit et représente les contraintes statiques relatives à son exploitation.

### **2.3.2. Le Modèle du Système de Transport**

Le modèle du système de transport représente, du point de vue statique, les transports possibles à partir d'une localisation d'un produit dans l'atelier et, du point de vue dynamique, le flux des produits dans le système.

La possibilité de transférer un produit d'un lieu vers un autre concerne les relations d'accessibilité qui permettent à deux lieux physiques voisins d'échanger des produits. Du point de vue de l'évolution dynamique, un transfert n'est réalisable que si un emplacement de stockage, capable de recevoir le produit, est libre sur le lieu d'arrivée. La gestion des flux demande donc une gestion des emplacements de stockage de manière à éviter les blocages sur le système physique.

En général, un SFPM comporte des ressources à transferts libres. Ces ressources nécessitent une gestion, qui consiste en un suivi sous forme d'une description des différents transferts indépendamment de leur état (libre ou chargé). Ensuite, elle doit exprimer leur synchronisation lors d'un chargement ou un déchargement d'un produit [Gasnier et Vercauter 1988].

Parmi ces ressources, nous avons les supports de transfert d'un seul produit, tels que les palettes d'un convoyeur, ou les supports de plusieurs produits, tels que les plateaux ou palettes permettant de transporter un ensemble de produits.

Ces ressources de type support sont ou deviennent à des moments donnés **libres**, en cours

d'exploitation, lorsqu'elles ne supportent aucun produit. Le modèle du système de transport comporte la gestion des ressources à transferts libres, car ces ressources libres participent au flux des produits.

### **2.3.3. Les Graphes de Synchronisation**

Lors des échanges de produits entre deux lieux physiques, des liens sont créés entre le moyen de transport, réalisant le transfert, la ressource du lieu de départ et la ressource du lieu d'arrivée. Ces liens demandent une communication et une synchronisation des trois ressources sollicitées, adaptée au comportement de chacune.

En effet, le comportement de chaque ressource, vis à vis du chargement et du déchargement d'un produit, diffère d'une ressource à l'autre. Cela induit des modalités d'échange de produit très différentes en fonction des ressources intervenant pour un échange donné.

Par exemple, un échange de pièce entre un robot et une machine aura un comportement différent, si la machine est un tour horizontal ou un centre d'usinage. Dans le cas d'un chargement de pièce sur le centre d'usinage par le robot, le robot peut lâcher la pièce, donc desserrer la pince avant son bridage. Alors que la même opération, un chargement sur un tour, impose l'inverse, comme nous l'avons vu dans la figure I.6. Cela nécessite la définition de protocoles de coopération spécifiques à chaque type d'échange.

Ces protocoles de coopération sont nécessaires pour chaque transfert élémentaire entre deux lieux fixes dans l'atelier. Ils dépendent non seulement des ressources sur lesquels se trouvent ces lieux fixes mais aussi des ressources de transport (manipulateurs, convoyeur, etc.) qui réalisent le transfert.

Les graphes de synchronisation déterminent ces protocoles de coopération pour chaque échange possible. Les échanges possibles sont mémorisés sous forme de relations d'accessibilité directe entre deux lieux fixes sur le graphe des accessibilités du système, le comportement des ressources, spécifique à cet échange, est déduit des contraintes de fonctionnement.

### **2.3.4. Les Modèles des Commandes des Ressources**

Les modèles des ressources représentent de manière agrégée le comportement des ressources de production simples lors du chargement (et du déchargement) des produits. Ils modélisent la

coordination des actionneurs d'une ressource pendant ces opérations. Nous présenterons les caractéristiques d'une bibliothèque de modèles qui sont adaptés en fonction des possibilités et des contraintes de la ressource modélisée.

## **2.4. LA CONNEXION DES MODÈLES**

L'ensemble des modèles générés est le résultat de cette première étape de conception. Ces modèles échangent des données et doivent de ce fait communiquer. Nous avons défini des modules de communication, qui modélisent la synchronisation de l'évolution du marquage d'un modèle plus abstrait sur l'évolution du modèle représentant les mêmes opérations mais à un niveau plus détaillé. Les données échangées correspondent aux jetons des réseaux de Petri, qui sont de ce fait porteurs de toute information. Cela justifie l'utilisation des jetons individualisés auxquels nous pouvons associer toute information relative à un produit.

## **2.5. CONCLUSION**

Les différents modèles de la coordination sont liés entre eux dans une structure hiérarchisée basée sur le raffinement de l'information. Au niveau des gammes opératoires, un transfert est représenté par une seule transition, au niveau du modèle du système de transport cette transition sera décomposée en transferts élémentaires faisant intervenir chacun un seul moyen de transport. Chaque transfert élémentaire demande la synchronisation des actionneurs des ressources sollicitées, très souvent de nature différente, ce qui nécessite une définition des protocoles d'échange spécifiques.

Cette structure hiérarchique des modèles confie à chaque type de modèle une tâche bien spécifique de la coordination. Les protocoles de communication, que nous présenterons dans la partie 5 de ce chapitre, traduisent cette structure de type "client - serveur".

La figure II.2 représente le processus de modélisation. Dans le paragraphe suivant, nous présentons brièvement l'étape de spécification et d'analyse issue des travaux menés au sein du projet CASPAIM. La génération des modèles, présentée dans le quatrième paragraphe, s'effectue en parallèle. En effet, nous générons la partie statique des réseaux de Petri, les places et les transitions. L'évolution du marquage dans ces modèles correspond à l'activité de la commande de coordination et modélise le comportement dynamique du procédé.

Une démarche pour la validation et la vérification des modèles est présentée au chapitre III de

ce mémoire et une démarche pour la traduction des modèles et leur implantation sur ordinateur a été présentée dans [Huvenoit et al 1994].

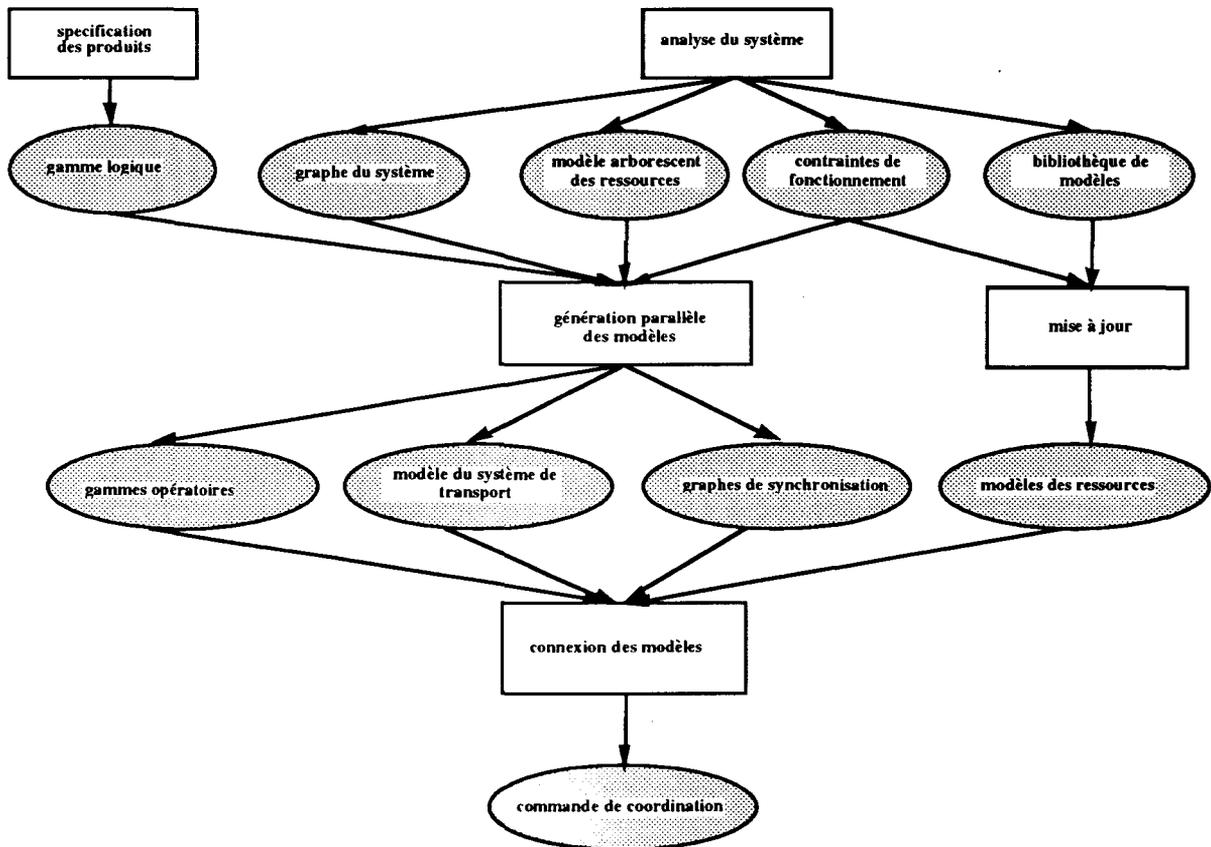


Figure II.2 : la démarche de conception

### 3. LA SPÉCIFICATION DES OBJECTIFS DE PRODUCTION ET L'ANALYSE DU SYSTÈME

#### 3.1. LA SPÉCIFICATION DES PRODUITS : LES GAMMES LOGIQUES

La fonction essentielle d'un système de production est de réaliser une quantité donnée de produits sur un horizon défini. Au niveau de la conception de la coordination nous pouvons

nous limiter à l'aspect qualitatif de la production demandée, car les quantités et les délais ne sont connus en phase d'exploitation du système. De ce point de vue, le processus de fabrication est une succession de traitements, qui changent les caractéristiques du produit. Les traitements qui font évoluer les caractéristiques du produit par rapport à son état initial sont appelées **opérations caractéristiques**.

**Définition d'une Opération Caractéristique** [Amar et al 1992] : Une opération est dite caractéristique, si elle représente une transformation fonctionnelle (usinage, assemblage, conditionnement, contrôle..), c'est-à-dire si elle n'a pas comme unique objectif un transfert et/ou un stockage.

La succession des opérations caractéristiques, qui mène du produit initial au produit final est modélisée pour chaque type de produit par la **gamme logique**.

**Définition d'une Gamme Logique** [Amar et al 1992] : La gamme logique d'un produit décrit le séquençage (contrainte d'ordre) des opérations élémentaires et caractéristiques définissant ainsi le processus de fabrication qui permet d'obtenir le produit fini à partir de son état brut.

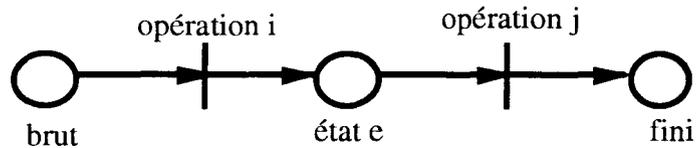
Une gamme logique correspond par conséquent à une description purement fonctionnelle du processus de fabrication sans aucune référence à l'aspect physique ou matériel du système de fabrication.

Dans le cadre du projet CASPAIM une représentation de la gamme logique par réseau de Petri a été retenue [Cruette 1991]. Les places du réseau de Petri indiquent l'état d'avancement du produit et ses transitions modélisent les opérations caractéristiques.

A partir de cette représentation trois types de gammes sont possibles en fonction de la nature du procédé [Cruette 1991] :

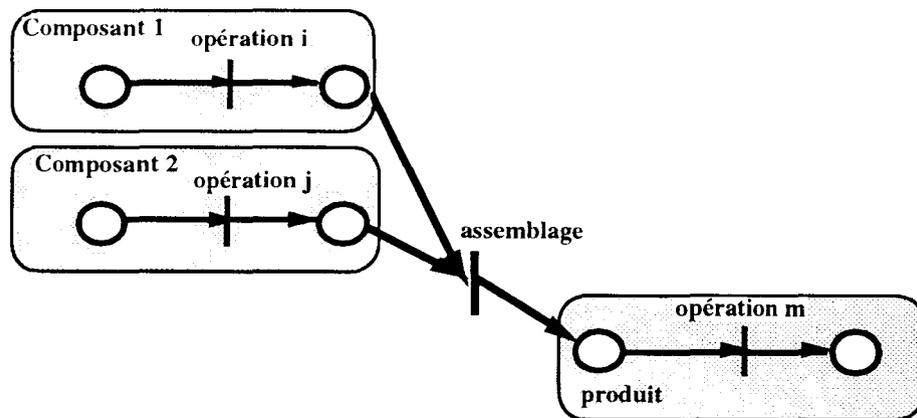
- \* gammes logiques d'un produit isolé
- \* gammes logiques de produits liés entre eux
- \* gammes logiques de produits traités en lots.

Si le processus est constitué d'opérations **unaires** (usinage, métrologie, etc.) faisant intervenir un seul produit à la fois, la gamme logique est élémentaire et purement linéaire (figure II.3).



**Figure II.3 : gamme élémentaire**

Si le processus est constitué d'une ou plusieurs opérations **n-aires** faisant intervenir deux produits au minimum (assemblage), deux phases sont à considérer. La phase en amont de l'opération d'assemblage décrit le processus de fabrication des composants. L'assemblage lui-même est modélisé comme une synchronisation de ces deux processus. En phase aval de la synchronisation, une nouvelle gamme élémentaire représente l'évolution du produit résultant de la synchronisation des gammes amont (figure II.4).



**Figure II.4 : gamme d'assemblage**

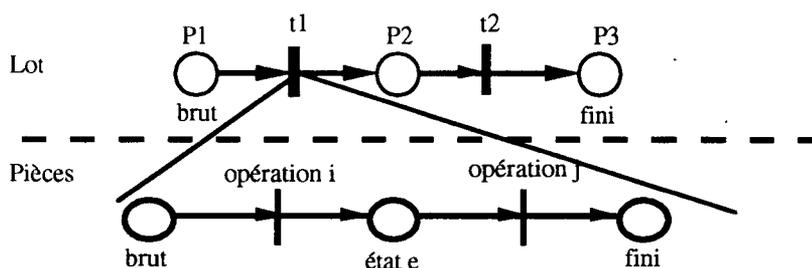
Si le processus de fabrication fait intervenir des traitements de **lots** (ensemble de produits identiques) et des traitements sur les produits élémentaires le constituant, alors la modélisation doit être menée à deux niveaux.

En premier lieu, en considérant le lot en tant qu'entité indivisible comme s'il s'agissait d'un produit élémentaire. Les opérations effectuées sont alors modélisées par un processus élémentaire, le jeton modélise le lot comme unité indivisible.

En second lieu, en considérant les pièces constituant le lot d'une façon élémentaire. Leurs gammes peuvent alors comporter des opérations unaires ou n-aires.

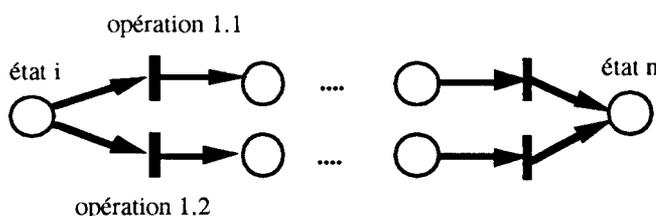
La figure II.5 modélise le traitement du lot et le traitement des pièces le constituant par une approche de type réseaux de Petri hiérarchisés [Huber et al 1990, cf. Annexe 4]. La transition

t1 est une transition d'appel, qui fait exécuter la page, modélisée par la gamme logique des pièces, pour chaque produit du lot. Si tous les produits constituant le lot ont été traités, le lot est reconstitué et le jeton correspondant marque la place P2.



**Figure II.5 : traitement de lot**

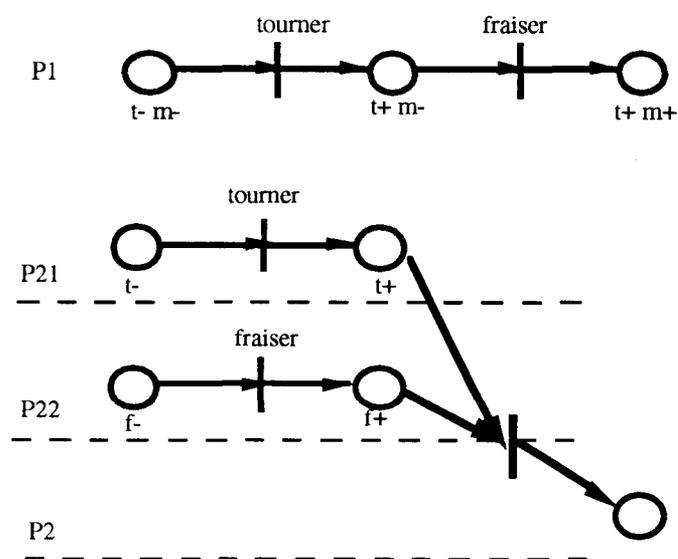
La spécification des produits doit intégrer la flexibilité dans l'ordre des opérations mettant en évidence toutes les combinaisons possibles dans le processus de fabrication. La représentation des permutations d'opérations s'effectue par duplication de la portion où doit apparaître cette flexibilité. Il en résulte un graphe avec alternatives introduisant un indéterminisme directionnel (choix). Dans le cas des opérations unaires, les réseaux de Petri résultant sont des graphes d'état. (figure II.6).



**Figure II.6 : choix des séquences**

Pour un produit donné, sa gamme logique représente donc la flexibilité du séquençage des opérations caractéristiques. La gamme logique indique ainsi les contraintes minimales d'antériorité ou de précédence imposées par le processus de fabrication.

La représentation des gammes logiques sous forme de réseau de Petri définit le format sous lequel les résultats de la spécification des produits sont retenus. La figure II.7 montre les réseaux de Petri correspondant aux gammes logiques des deux produits de notre exemple (cf annexe 2).



**Figure II.7 : exemple de gammes logiques**

La génération elle-même des gammes logiques sort du cadre des travaux présentés dans ce mémoire et fait l'objet de recherches dans d'autres domaines de la productique [Bourjault 1986, Favrel et Campagne 1984, Kapusta 1988], qui sont à ce niveau complémentaires des travaux menés dans le cadre du projet CASPAIM.

Nous supposons par la suite que les permutations des opérations sont limitées et que la gamme logique est donc traçable. D'une part des permutations importantes n'interviennent que dans des rares exemples d'assemblage [Dungern et Schmidt 1990], d'autre part elles peuvent être regroupées au niveau des gammes logiques dans une opération agrégée qui sera détaillée à un autre niveau du modèle.

### 3.2. L'ANALYSE DU PROCÉDÉ

L'analyse des ressources de production conditionne de manière importante les résultats de la modélisation de la commande de coordination. D'une part, une représentation fidèle de la flexibilité des ressources de production à tous les niveaux de la modélisation est indispensable pour une commande adaptative. D'autre part, seule l'intégration des contraintes du procédé dans les modèles permet un fonctionnement sûr du système.

L'analyse du procédé met donc en évidence les caractéristiques des ressources relatives à la commande. Son cahier des charges comporte, entre autres, l'identification des actionneurs des

ressources et leurs contraintes de fonctionnement, mais aussi, au niveau atelier, l'identification des lieux physiques pouvant recevoir des produits et son architecture fonctionnelle.

### 3.2.1. La Décomposition structuro-fonctionnelle des Ressources [Amar et al 1992]

La décomposition structuro-fonctionnelle présentée par la suite à été développée dans le cadre du projet CASPAIM par S. Amar [Amar et al 1992]. L'analyse d'une ressource de production s'effectue par décomposition de la structure fonctionnelle de sa partie opérative. Elle utilise comme point de départ un ensemble de fonctions opératives de base qui ont été définies et classées au sein d'une bibliothèque (figure II.8).

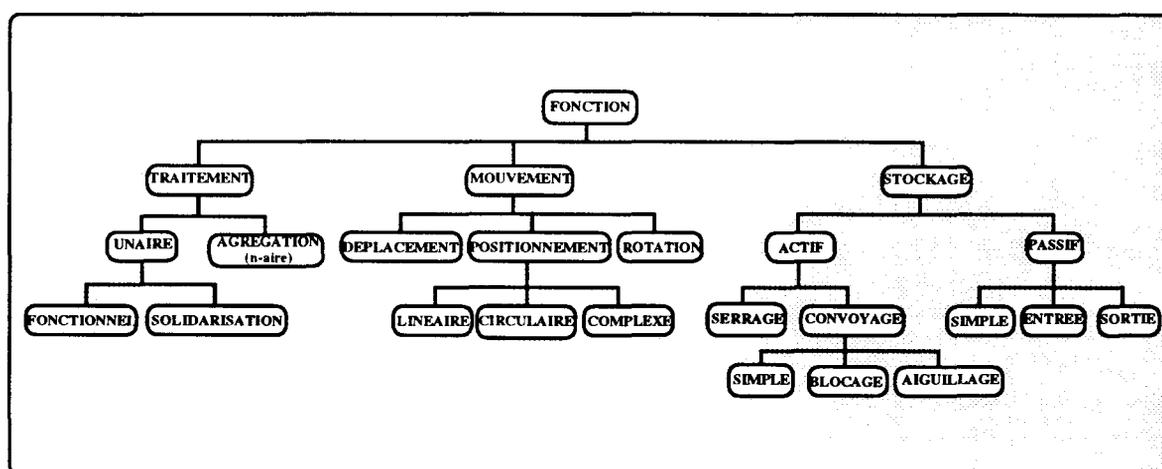


Figure II.8 : bibliothèque fonctionnelle

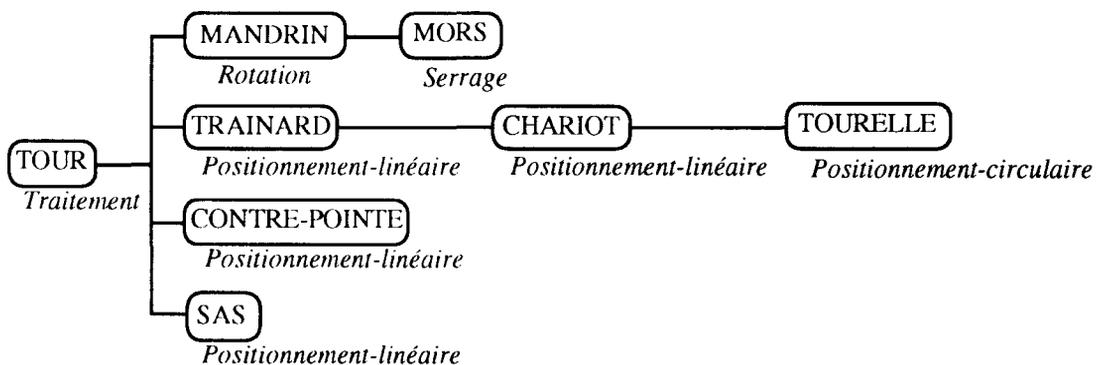
Une fonction opérative, c'est-à-dire une fonction associée à un organe opérant, peut avoir deux facettes par rapport à la commande. Elle peut être soit active, c'est-à-dire que pour être réalisée elle est réceptive et tributaire d'une commande par l'intermédiaire d'un actionneur. Elle peut être soit passive, comme c'est le cas, par exemple, pour un stock simple dont la fonction est uniquement de stocker temporairement un ou plusieurs produits.

Une fonction opérative permet ainsi de définir un type de comportement élémentaire d'un organe opérant vis-à-vis d'un produit, tel que le positionnement, le stockage ou la transformation du produit. La spécification d'une ressource commence par une description de sa structure de sorte à mettre en évidence les différents organes opérants. Le critère de décomposition est basé sur des liens de type supportant / supporté faisant apparaître une arborescence dans laquelle chaque organe appartenant à un niveau donné est le support de la sous-structure inférieure. Tous les organes intervenant dans une même branche sont donc

solidaires mécaniquement. Le comportement d'un organe situé à un niveau donné influence les organes situés à un niveau inférieur. Par contre, deux organes appartenant à deux branches différentes n'ont aucune influence l'un vis-à-vis de l'autre, d'un point de vue comportemental.

Dans la deuxième étape de l'analyse, nous associons à chaque organe une ou plusieurs fonctions de la bibliothèque fonctionnelle (figure II.8). Dans cette bibliothèque les fonctions ont été classées en trois catégories :

- \* les fonctions de type **stockage** définissent les lieux physiques élémentaires pouvant recevoir un produit (pince de robot, mandrin d'un tour, etc.) ou le support d'un produit solidaire avec le produit (palette, plateau, cf. ressource à transfert libre),
- \* les fonctions de type **mouvement** définissent les organes dotés d'un certain degré de mobilité tels que les dispositifs mécaniques animés d'un mouvement spatial (bras d'un robot, coulisse longitudinale),
- \* les fonctions de type **traitement** représentent l'ensemble des opérations unaires (usinage) et n-aires (assemblage), effectuées sur les produits. Ces fonctions ne sont pas élémentaires dans le sens des deux autres catégories et apparaissent par conséquent seulement au niveau initial de la structure arborescente d'un moyen de fabrication.



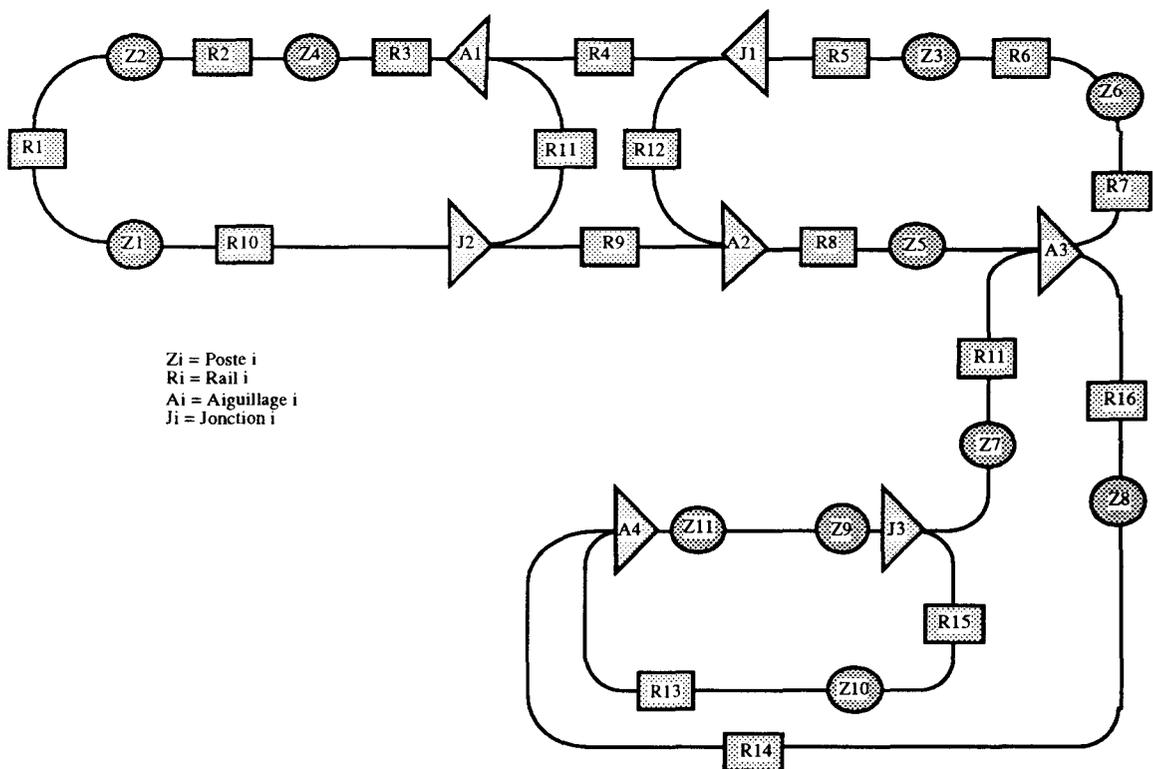
**Figure II.9 : décomposition du tour [Amar et al 1992]**

La figure II.9 montre le résultat de la description structuro-fonctionnelle pour le tour de la cellule de l'École Centrale de Lille. Les feuilles de l'arborescence indiquent les actionneurs dont le comportement nous préoccupe. Le comportement des entités appartenant à une même branche est fortement contraint de par leur appartenance à la même structure physique. Le comportement des feuilles comme comportement "observable" d'une branche est donc d'un intérêt majeur pour la conception de la commande.

La description d'un système de convoyage s'effectue par l'intermédiaire d'une méthode de décomposition bien spécifique [Bourrières et Chevillard 1988]. Elle est réalisée de manière à

mettre en évidence dans un premier temps toutes les sections (rails) ou tronçons de convoyage constituant un point singulier.

Chaque point singulier représente soit une zone munie d'un système de blocage (bloqueur ou indexeur), soit une zone critique constituant une divergence, munie d'un système d'aiguillage (aiguilleur ou pousseur), et / ou une convergence. Chaque zone critique a par définition et par convention une capacité unitaire. Ensuite chaque zone munie d'un bloqueur est dimensionnée (capacité finie) et tous les tronçons restants sont représentés par une zone de convoyage élémentaire.



**Figure II.10 : décomposition du convoyeur**

La figure II.10 illustre le schéma de description du convoyeur de la cellule de l'École Centrale de Lille. Le schéma met en évidence les points singuliers, qui demandent une gestion de par leur capacité limitée et les actionneurs les délimitant, qui doivent être commandés.

L'intérêt particulier apporté aux systèmes de convoyage est dû, d'une part, à leur double fonctionnalité de transport et de stockage des pièces et, d'autre part, à la réalisation des deux fonctions par l'intermédiaire des palettes. Les palettes nécessitent une gestion à part, car elles font parties des ressources à transfert libre.

De par sa structure, un système de chariots automoteurs est comparable à un système de convoyage, car il comporte les mêmes éléments : aiguillages, zones d'arrêt pré-définies etc. La méthode de description s'applique donc de manière identique.

### 3.2.2. La Synthèse du Graphe des Accessibilités du Système

Le graphe des accessibilités du système permet de visualiser la topologie et l'organisation d'un atelier. Il représente, par l'intermédiaire des relations d'accessibilité, les transferts possibles entre les lieux caractéristiques du système, donc entre les ressources de production.

A partir de la définition des liens existants entre les différents moyens de production qui concernent les échanges des pièces, nous identifions les lieux physiques par lesquels les produits transitent.

**Définition d'un lien d'échange :** Un moyen (machine d'usinage) est en lien d'échange avec un autre (manipulateur), si un produit quelconque peut transiter de l'un à l'autre sans passer par un moyen physique intermédiaire (chargement et déchargement de la machine par le robot).

Les liens d'échange déterminent la possibilité de faire transiter des produits entre différents lieux physiques. Ils sont donc à la base du flux des produits dans l'atelier. Les lieux physiques pouvant recevoir un produit ont été identifiés lors de la décomposition des ressources de production par association d'une fonction de type stockage (par exemple pince d'un robot, mandrin d'un tour). La définition des liens d'échange établit, par conséquent, des relations entre les différentes feuilles de la décomposition structuro-fonctionnelle.

**Définition :** Un lieu caractéristique d'un SFPM est un lieu physique fixe ou un lieu virtuel faisant référence à un lieu physique fixe sur lequel un produit change de support (**lieu de synchronisation**) ou sur lequel le produit subit une opération caractéristique (**lieu de transformation**).

La définition des lieux caractéristiques permet d'effectuer un filtrage des lieux physiques. D'abord seuls les lieux fixes sont retenus, car les lieux mobiles accomplissent essentiellement une fonction de transport comme la pince d'un robot ou une palette d'un convoyeur, bien que des produits soient temporairement stockés sur ces ressources.

Les lieux de transformations sont retenus, car sur ces lieux les produits subissent une

opération caractéristique. Cette information est essentielle dans une approche produit, car elle permet de déterminer l'état d'avancement des produits par rapport à leur gamme.

Les lieux de synchronisation sont des lieux fixes, qui sont en relation d'accessibilité externe avec un lieu mobile, c'est-à-dire, avec un lieu mobile appartenant à une autre ressource. Du point de vue du flux des produits, ces lieux représentent donc un intérêt majeur, car c'est sur eux que l'échange des produits est effectué (changement de support), ce qui nécessite une coordination des ressources concernées.

La définition des lieux caractéristiques reprend la notion de lieu virtuel introduit dans [Amar 1994]. Un lieu caractéristique fixe, comme par exemple la zone opératoire d'une machine à outil, reçoit toujours un produit au même endroit. Cependant, certains types de ressources représentent des emplacements de stockage, qui sont mobiles, comme les palettes d'un convoyeur ou les étagères d'un ascenseur. On ne peut accéder à ces emplacements que dans certaines zones prédéfinies, où ces supports sont arrêtés dans une position stable et connue. La notion de lieu virtuel fait référence à ces zones d'accès aux supports mobiles des produits.

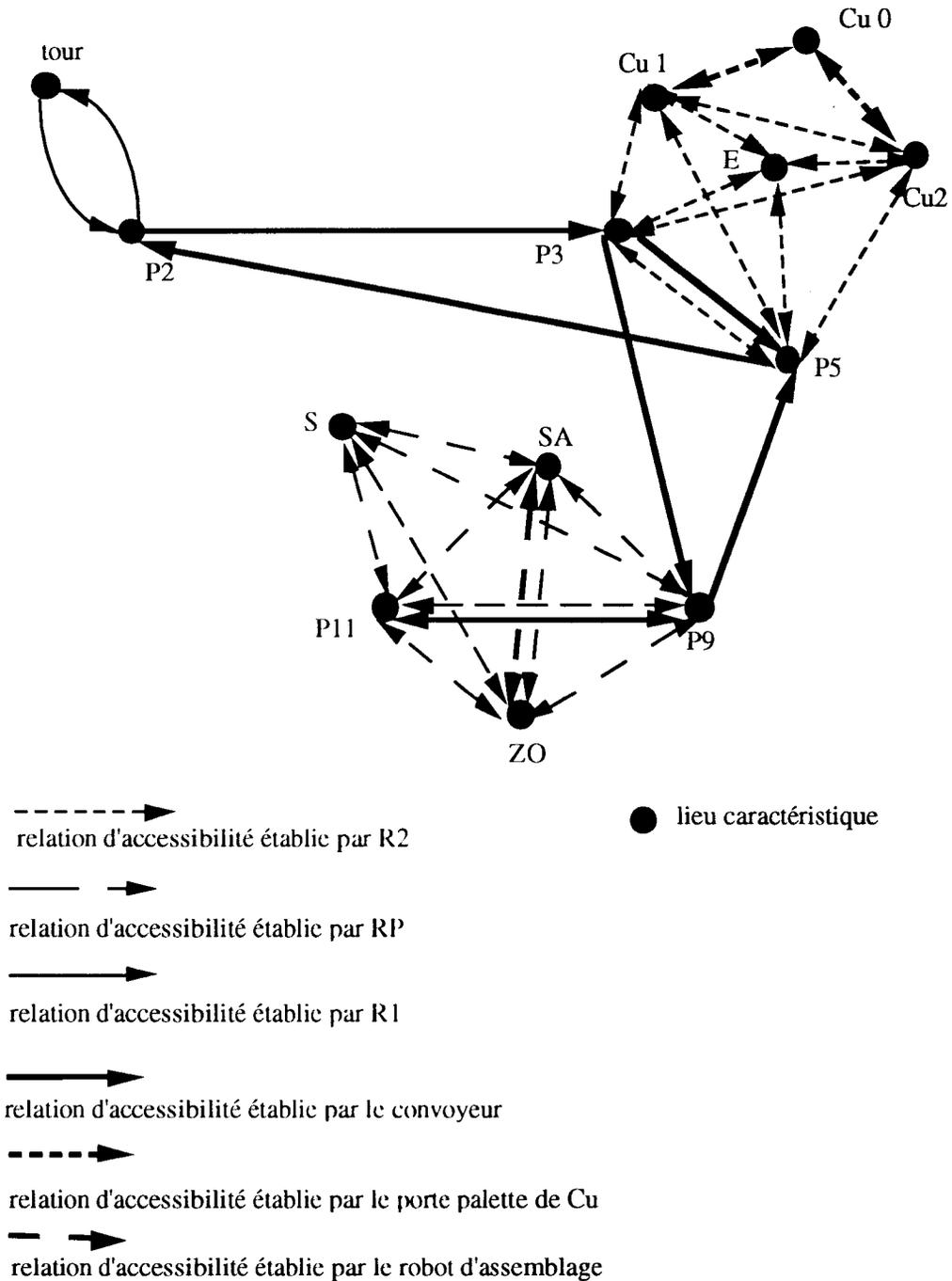
**Définition d'une relation d'accessibilité** : Un lieu caractéristique A est en relation d'accessibilité avec un lieu caractéristique B si un produit localisé en A peut transiter en B. A est en **relation d'accessibilité directe** avec B si un produit peut transiter de A vers B sans passer par un autre lieu caractéristique.

Les relations d'accessibilité directe traduisent les transferts élémentaires entre deux lieux caractéristiques. Ces relations sont orientées (transfert de A vers B) et traduisent les liens potentiels existant entre les différents moyens de production de l'atelier.

Les relations d'accessibilité complètent l'analyse de la topologie. Les lieux caractéristiques indiquent les lieux de transit des pièces, les relations d'accessibilité montrent les différents transferts possibles. Cette information est synthétisée sur **le graphe des accessibilités du système**.

**Définition** : Le Graphe des Accessibilités du Système est un graphe orienté et étiqueté. Deux noeuds A et B sont reliés par un arc orienté de A vers B si le lieu caractéristique A représenté par le noeud A est en relation d'accessibilité directe avec le lieu caractéristique B représenté par le noeud B. L'étiquette indique le moyen de transport reliant les deux lieux caractéristiques.

La figure II.11 montre le graphe des accessibilités de la cellule de notre exemple. Le graphe donne une vue concise de la topologie de l'atelier et des flux possibles des produits entre les moyens de production (flexibilité des moyens de transport).



**Figure II.11 : graphe des accessibilités du système**

### **3.2.3. Les Contraintes de Fonctionnement**

Le dernier élément de l'analyse d'un SFPM concerne les contraintes de fonctionnement existant entre les actionneurs de chaque ressource. Les actionneurs d'une ressource de production peuvent être fonctionnellement contraints, même s'ils ne font pas partie d'une même branche de la décomposition structurelle. Nous avons déjà présenté un cas typique de telles contraintes : le chargement interne du tour de la cellule flexible de l'École Centrale de Lille (figure I.6).

Le chargement d'une pièce ne peut s'effectuer que si la contre-pointe et la tourelle sont reculées, si les mors du mandrin sont ouverts, et si la porte du SAS (Système d'Accès de Sécurité) est ouverte. Les organes opérants "mors", "tourelle", "contre-pointe" et "SAS" sont de ce fait contraints, car ils doivent être dans un état spécifique pour effectuer un chargement.

Nous remarquons, qu'au niveau de la commande de coordination, les contraintes de fonctionnement interviennent lors des opérations de chargement et de déchargement. Ces contraintes sont spécifiques à chaque ressource et leur déduction demande des connaissances détaillées de son fonctionnement.

### **3.3. CONCLUSION**

L'étape de spécification des objectifs d'exploitation d'un système et son analyse conditionnent de façon importante les résultats de la démarche de modélisation [Daenzer 1988]. Pour la spécification des objectifs, nous avons présenté un formalisme réseau de Petri, qui permet une visualisation de la partie logique du processus de fabrication (gammes logiques).

L'analyse du procédé génère des données qui seront ensuite reprises pour les différentes étapes de la modélisation. Aux niveaux des modèles concernés par la coordination des ressources, nous nous intéressons essentiellement aux actionneurs. Au niveau du modèle du système de transport, nous nous intéressons à tout lieu physique pouvant recevoir un produit. La décomposition structuro-fonctionnelle met donc en évidence tout organe opératif d'une ressource, ainsi que les organes pouvant recevoir un produit.

La génération du graphe des accessibilités du système met en évidence les liens entre ces lieux caractéristiques. Il en résulte un modèle de l'architecture fonctionnelle du système, qui

représente la flexibilité des moyens de transport et les contraintes imposées par son organisation.

L'analyse des contraintes de fonctionnement, comme dernier élément, fournit les entrées relatives au comportement des ressources pendant les opérations de chargement et de déchargement. Leur établissement demande des connaissances détaillées sur le fonctionnement des ressources.

## **4. LA GÉNÉRATION DES MODÈLES DE LA COORDINATION**

### **4.1. INTRODUCTION**

Nous présentons dans ce paragraphe la méthode de génération des modèles de la coordination. Nous avons défini quatre types de modèles réseaux de Petri : les gammes opératoires, le modèle du système de transport, les graphes de synchronisation et les modèles des ressources, qui représentent le procédé à différents niveaux d'abstraction. Ces différents modèles sont donc liés dans une structure hiérarchisée, même s'ils peuvent être générés en parallèle.

### **4.2. LA GÉNÉRATION DES GAMMES OPÉRATOIRES**

#### **4.2.1. Introduction**

La gamme opératoire est une description par réseau de Petri de la séquence de transformations et de transports à réaliser sur un même produit. La partie statique du réseau, c'est-à-dire le graphe du réseau, donne ainsi une première vue synthétique des possibilités et des contraintes du procédé.

La dynamique du procédé est représentée par l'avancement des jetons dans le réseau de Petri. La commande de coordination doit garantir un fonctionnement sûr du système, qui dépend de son comportement dynamique. De ce fait, les gammes opératoires apportent une première vue du comportement des ressources de transformation.

Nous présentons, après un bref rappel des entrées, un algorithme qui permet la génération automatique des gammes opératoires. La partie de cet algorithme concernant les étapes 1 à 3 a été exposée dans [Amar et al 1992].

En fonction de la complexité du processus de fabrication et de la taille du système physique, les gammes opératoires deviennent vite compliquées et le modèle illisible. Nos travaux ont permis, face à cette difficulté de la modélisation par réseaux de Petri, de proposer une approche de hiérarchisation de ces gammes. Cette démarche qui redéfinit les notions de lieux caractéristiques et d'opérations caractéristiques sera présentée dans un deuxième temps.

#### 4.2.2. Les Données d'Entrée

Les **gammes logiques** donnent les caractéristiques spécifiques de chaque produit en exprimant la séquence des opérations transformationnelles à appliquer. De manière générale, chacune de ces transformations peut être réalisée sur plusieurs ressources (cf. polyvalence des ressources de transformation). Les ressources susceptibles de réaliser une transformation donnée sont identifiées lors de la **décomposition structuro-fonctionnelle**. Mais ces ressources, bien qu'elles fassent partie de la même installation, ne sont pas forcément accessibles entre elles. Il se peut, par exemple, que deux sous-ensembles de ressources soient reliés par un tapis roulant, qui ne permet le transport des produits que dans une direction. Cette information, déduite des relations d'accessibilité, est extraite du **graphe des accessibilités du système** et constitue le troisième élément d'entrée de la génération.

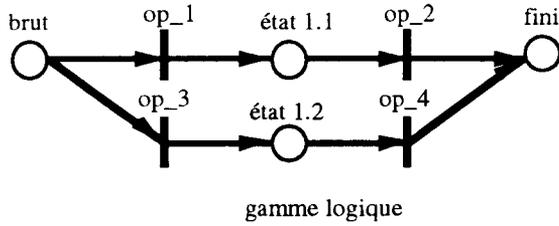
#### 4.2.3. Un Algorithme pour la Génération des Gammes Opératoires

*Étape 1 : Choix d'un lieu de départ et d'arrivée, pour chaque gamme logique (place initiale et place finale), et des machines candidates pour chacun des traitements.*

Les lieux de départ et d'arrivée ainsi qu'une liste de machines candidates pour chaque traitement sont spécifiés pour initialiser la phase de génération. Il est à noter qu'une place supplémentaire ainsi qu'une transition de transfert sont ajoutées systématiquement en fin de chaque gamme logique pour pouvoir réaliser l'étape suivante (figure II.12)

Sur la figure II.12, les lieux **entrée** et **sortie** désignent respectivement les lieux de départ et d'arrivée pour chacune des deux gammes.

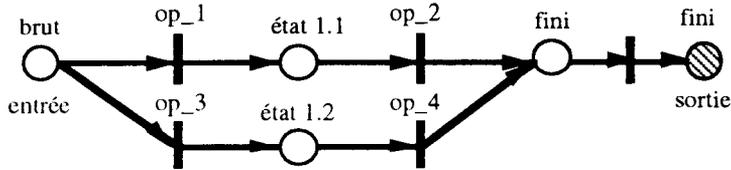
Entrées :



Liste des machines candidates :

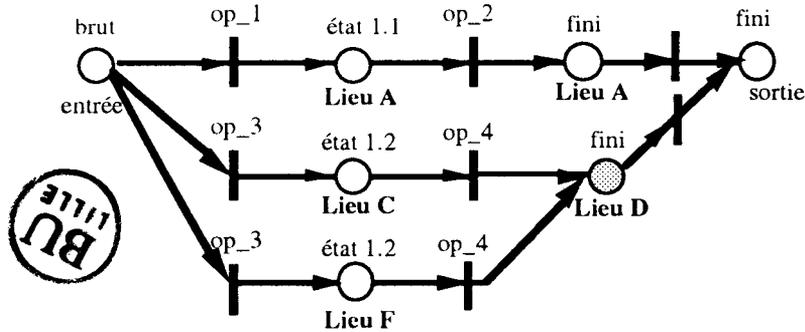
- op\_1 : lieu A
- op\_2 : lieu A
- op\_3 : lieu C, lieu F
- op\_4 : lieu D

Etape 1: choix des lieux de départ et d'arrivé



place rajoutée

Etape 2 : Désignation des lieux opératoires



place dupliquée

Etape 3: Dissociation des transitions

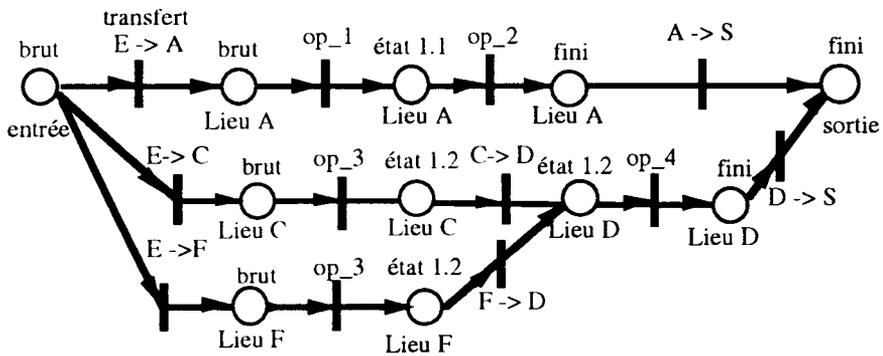


Figure II.12 : la génération des gammes opératoires

*Étape 2 : Désignation des lieux opératoires au niveau des gammes.*

Le lieu opératoire, sur lequel un traitement est effectué, est indiqué au niveau de la place aval de la transition correspondante. Une structure alternative (cf. places “lieu C” et “lieu F” de la figure II.12) est générée pour intégrer la flexibilité de choix des machines pour un traitement donné (plusieurs machines candidates). Pour les structures alternatives représentant une flexibilité dans l’ordre des opérations, un éclatement de la place située en fin de cette structure est effectué.

*Étape 3 : Dissociation des transferts (changement de lieu) et des transformations (changement d’état) simultanés.*

Chaque transition représentant un transfert et un traitement simultanés est dupliquée en deux transitions (transfert puis traitement) générant ainsi une place intermédiaire (figure II.12).

*Étape 4 : Intégration des relations d’accessibilité.*

La génération part, jusqu’à maintenant, de l’hypothèse que toutes les ressources de production sont reliées les unes aux autres et que chaque transfert indiqué sur les gammes opératoires est réalisable. Ceci est traduit sur le graphe des accessibilités du système par la propriété de forte connexité. Si le graphe n’est pas fortement connexe, une étape de filtrage, qui élimine les transferts non réalisables, est nécessaire.

Cette étape assure donc le respect des contraintes statiques imposées par l’organisation de l’atelier.

*Étape 5 : Représentation des Ressources de Transformation*

Nous représentons chaque lieu opératoire par une place, dont le marquage initial correspond à la quantité de produits qui peuvent être traités en parallèle. Cette place est connectée en amont et en aval de chaque transition modélisant une transformation sur ce lieu.

A ce niveau, les informations relatives à une ressource représentent l’ensemble des opérations réalisables par la machine. Le marquage de la place ressource correspondante indique son état actuel qui peut être “libre” ou “occupé”. Nous intégrons donc à ce niveau une première vue des demandes d’allocation des ressources de production.

La figure II.13 indique le résultat de la génération des gammes opératoires pour les deux produits de l’exemple.

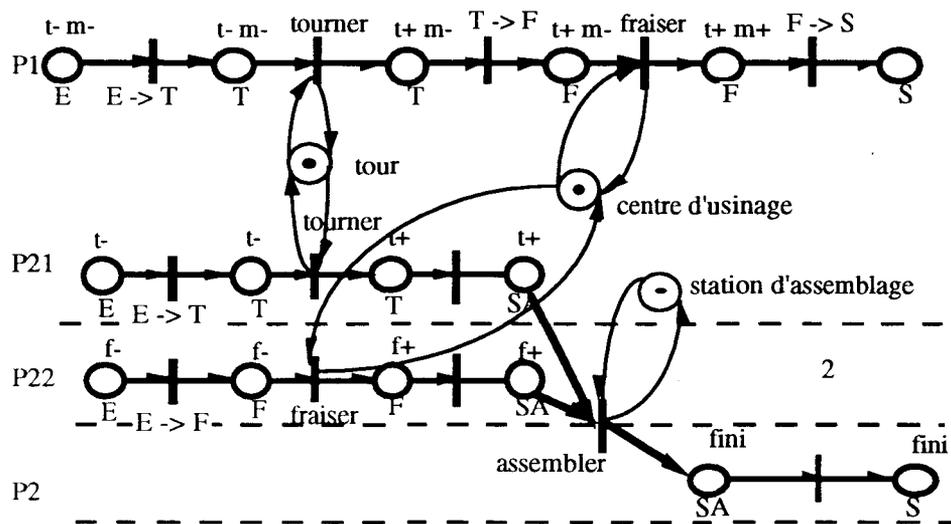


Figure II.13 : les gammes opératoires de l'exemple

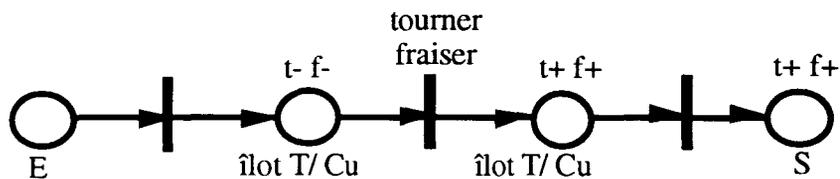
#### 4.2.4. La Hiérarchisation des Gammes Opératoires

Une place des gammes opératoires caractérise la localisation (lieu opératoire) et l'état d'un produit, une transition modélise soit un transfert entre lieux opératoires soit une opération caractéristique réalisée par une ressource spécifique.

Nous proposons une approche hiérarchique, qui redéfinit les opérations caractéristiques et les lieux opératoires pour réduire la taille des modèles. Cette redéfinition regroupe les lieux opératoires d'un atelier en sous-ensembles et permet de ce fait une agrégation de l'information. A un niveau donné, tous les lieux opératoires d'une sous-structure sont regroupés au sein d'un seul lieu. Suivant cette démarche, la structure du niveau supérieur ne "voit" que ce lieu comme lieu opératoire.

La perception des lieux opératoires par rapport à un niveau d'abstraction donné permet ensuite le regroupement des opérations caractéristiques. Nous affectons donc à un regroupement de lieux toutes les opérations caractéristiques, qui sont réalisables sur ces lieux ainsi que les transferts entre ces lieux.

Ce regroupement de l'information permet la réduction de la taille des modèles dans le cas où plusieurs opérations successives sur le même produit sont effectuées dans la même sous-structure. Si le tour et le centre d'usinage de notre exemple faisaient partie de la même sous-structure (donc un regroupement de lieux opératoires), par exemple d'un même îlot, les deux opérations transformationnelles à effectuer sur le produit 1 pourraient être regroupées dans une opération agrégée (figure II.14).



l'opération "tourner + fraiser" est une opération agrégée,  
le lieu "îlot T/Cu est un lieu agrégé.

**Figure II.14 : gamme agrégée**

Nous constatons dans le cas de la figure II.14 une réduction considérable du modèle. Ceci pose le problème des critères d'un regroupement de lieux opératoires adapté aux gammes opératoires. Nous présentons dans le paragraphe suivant un critère structurel pour le regroupement des ressources, adapté à la génération du modèle du système de transport. Cependant, la "philosophie" de modélisation adoptée par les gammes opératoires fait abstraction de l'architecture de l'atelier (seules les relations d'accessibilité sont prises en compte).

Si nous appliquons des critères structurels comme critère de regroupement des lieux opératoires, la réduction de la taille du modèle n'est pas forcément importante. Dans le cas de notre exemple, si nous appliquons la démarche de regroupement structurel, la réduction du modèle se limite à l'élimination d'une seule transition (la première transition (E->F) pour la gamme du produit P 22 (figure II.13) est éliminée du fait de l'appartenance des lieux "E" et "F" au même îlot).

L'effet de la hiérarchisation dépend donc essentiellement du critère choisi pour le regroupement des lieux opératoires. Outre le critère structurel, des critères fonctionnels, des critères d'exploitation, des critères d'implantation, ou encore des critères souhaités par le concepteur sont aussi envisageables.

Les critères fonctionnels regroupent les machines pouvant exécuter les mêmes opérations sur un produit donné et la taille des modèles est réduite par l'élimination des structures alternatives représentant l'exécution de la même opération sur différentes machines [Toguyeni 1992].

Pour un regroupement des lieux opératoires selon des critères d'exploitation, nous pouvons envisager le regroupement de ces machines, qui sont toujours visitées dans un ordre identique par un ensemble de produits. Ce critère de regroupement correspond alors à l'approche de décomposition spatiale d'un problème d'ordonnement présentée dans le chapitre I.

Le choix même d'une hiérarchisation des gammes opératoires reste un choix réservé au concepteur de la commande. Néanmoins, dans le cas de processus de fabrication qui combinent un traitement ou un transport de produits par lots avec un traitement des produits à l'unité, une hiérarchisation s'est montrée avantageuse [Gentina et al 1992].

En effet, la description du processus de fabrication par lot à un niveau et ensuite la vue détaillée sur les opérations à réaliser sur chacun des produits telle qu'elle est symbolisée par la gamme logique de la figure II.5 correspond déjà à une approche hiérarchique. Le critère de hiérarchisation est dans ce cas le lien logique entre les pièces d'un lot.

Un atelier, adapté à la fabrication par lot, intègre des moyens de stockage et de transport adéquats. Ces ateliers comportent des moyens de transport réalisant les transferts par lots et d'autres manipulant les produits individuellement. La structuration d'un atelier, présentée dans la partie suivante met en évidence cette caractéristique de l'architecture d'un atelier. Dans ce cas, la hiérarchie du modèle du système de transport, correspond à la hiérarchie des gammes opératoires, si nous appliquons le critère structurel.

Indépendamment du critère de regroupement choisi, les gammes opératoires hiérarchisées modélisent l'évolution d'un même produit à différents niveaux d'abstraction. Une opération agrégée, représentée par une seule transition, comporte différents transports et transformations élémentaires indiqués sur la gamme de départ.

Une transition modélisant une opération agrégée pour un produit donné représente un ensemble de traitements spécifiques de ce produit. A chaque transition du niveau supérieur du modèle correspond donc un sous-réseau distinct, qui est à insérer à chaque franchissement de cette transition. Si nous utilisons la terminologie des réseaux de Petri hiérarchisés (cf. annexe 4), ce sous-réseau définit la **page** à insérer à chaque franchissement de la transition modélisant l'opération agrégée. (figure II.15). La transition "opération agrégée" de la figure II.15 devient transition de substitution, elle est remplacée par la partie de la gamme de départ (gamme élémentaire).

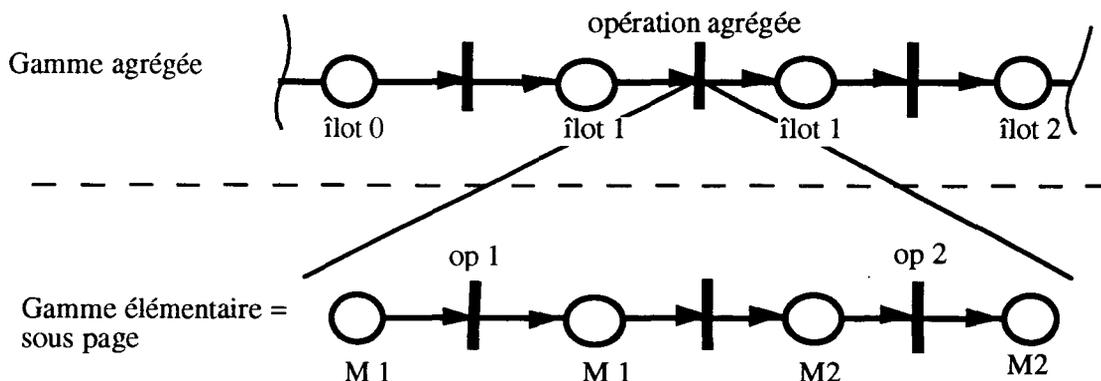


Figure II.15 : gammes hiérarchisées

#### 4.2.5. Conclusion

Les gammes opératoires, hiérarchisées ou non, modélisent la spécificité du processus de fabrication par rapport aux produits. Elles donnent une première vue synthétisée des possibilités et des limites de l'installation vis-à-vis des objectifs qualitatifs de production, c'est-à-dire sans références temporelles. Il en résulte des modèles non autonomes qui, pour réaliser la commande effective, nécessitent la résolution des indéterminismes et des conflits d'accès aux ressources.

Nous avons présenté un algorithme, qui génère, à partir des entrées de l'étape de spécification et d'analyse, un modèle du processus de fabrication de chaque produit, qui intègre la flexibilité de permutation des opérations et la polyvalence des ressources de transformation. Une première représentation des ressources par des places de type "protocole d'accès" modélise le caractère d'exclusion mutuelle de l'accès aux ressources de transformation (non-chevauchement temporel des opérations sur une même ressource).

Le comportement dynamique d'un produit est donc représenté par l'avancement du jeton correspondant. Cet avancement du jeton correspond à un changement d'état ou à un changement de lieu opératoire du produit. Les gammes opératoires représentent ainsi un premier élément de la modélisation des aspects produits de la commande de coordination. Ce modèle est complété par le modèle du système de transport, qui indique les possibilités du procédé par rapport au routage des produits.

La définition d'une hiérarchie des gammes opératoires sert à obtenir une modélisation plus concise du processus de fabrication. L'effet de réduction est largement influencé par les critères

de regroupement utilisés pour la définition d'une hiérarchie des sous-systèmes de l'atelier. Dans le cas des processus de fabrication intégrant un traitement par lot et des opérations sur des produits unitaires, la mise en relation de la hiérarchie logique introduite par les lots et de la structure de la partie opérative mène à une hiérarchie naturelle des modèles .

### **4.3. LA GÉNÉRATION DU MODÈLE DU SYSTÈME DE TRANSPORT**

#### **4.3.1. Introduction**

Le modèle du système de transport complète l'aspect produit de la commande de coordination. Il détaille les transports possibles et correspond à un modèle comportemental décrivant l'aspect de manutention des produits dans un atelier. Ce modèle intègre de fait la gestion des ressources à transferts libres.

La modélisation de tous les transferts sur un modèle dédié est justifiée par notre hypothèse de départ sur les SFPM. La polyvalence des ressources de transformation permet la réalisation de différentes transformations sur différents produits. Ces postes de l'atelier sont reliés par un ensemble de moyens de transport réalisant les transferts.

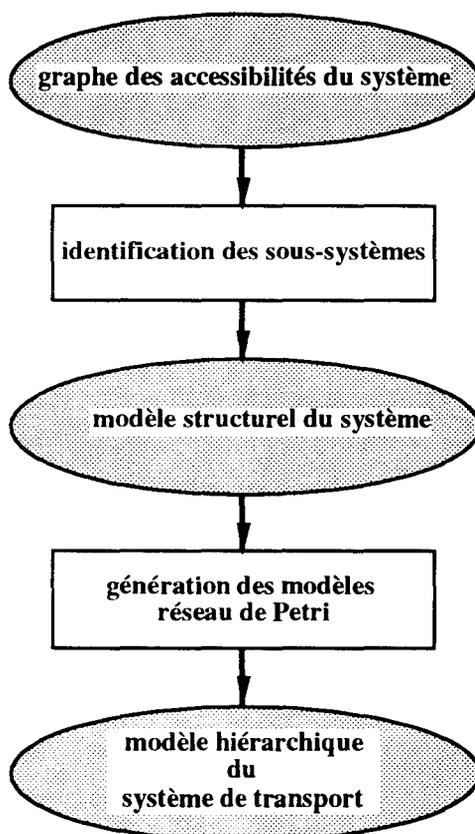
De ce fait des produits différents empruntent les mêmes chemins et certains chemins sont même empruntés par tous les produits. Si nous représentons ces chemins, qui ne sont donc pas caractéristiques pour un produit donné au niveau des gammes opératoires, nous surchargeons ces modèles par une information redondante. Outre la possibilité de modéliser fidèlement la flexibilité du système au niveau des transports sur un modèle dédié, la séparation des deux aspects de la coordination des produits permet la mise en facteur des modèles. Elle traduit en plus la fonction "service" du transport dans la production manufacturière.

La manutention des produits demande une gestion des emplacements de stockage du système. Un produit ne peut transiter d'un lieu vers un autre que si le lieu d'arrivée dispose d'un emplacement susceptible de le recevoir. L'intégration de la gestion des emplacements de stockage du système, de manière à éviter les blocages du système, est donc une tâche fortement liée à la modélisation des transferts dans un système.

Les destinations des transferts sont données pour chaque produit par les gammes opératoires, qui définissent ainsi un objectif pour l'utilisation du système de transport par un produit à un instant donné. Il existe alors un lien hiérarchique entre les deux modèles. Une transition des gammes opératoires indique un transfert spécifique, qui sera décomposé en

transferts élémentaires au niveau du modèle du système de transport.

Le choix d'une modélisation par réseaux de Petri permet de représenter de façon homogène l'évolution dynamique des produits et des ressources à transfert libre dans le système. Du fait de la représentation détaillée de la dynamique et de la flexibilité du système, le modèle conduit à une taille conséquente. Pour pallier ce problème, nous proposons, comme pour les gammes opératoires, une approche hiérarchisée du modèle des transports.



**Figure II.16 : la démarche de génération du modèle**

La figure II.16 représente la démarche de génération du modèle. Nous proposons d'abord un regroupement des ressources d'un atelier en unités structurales à partir d'une analyse du graphe des accessibilités du système. Cette structuration du système est basée sur le concept de cohésion des ressources.

Cette structuration, mène au **modèle structurel du système**. Le modèle structurel d'un SFPM définit la hiérarchie des modèles réseau de Petri, qui seront générés pour chaque sous-structure et en parallèle dans la dernière phase.

### 4.3.2. La Structuration d'un SFPM [Ausfelder et al 1993c]

#### 4.3.2.1. Introduction

La structuration nous permet de dégager des sous-systèmes à l'intérieur d'un SFPM. Cette structuration devient nécessaire dès lors qu'on modélise des systèmes de production de plus en plus complexes, afin d'obtenir des modèles lisibles et compréhensibles. Cette approche se prête d'autant plus aux systèmes de production flexibles, qu'ils sont le plus souvent organisés en unités de production autonomes. Ces unités autonomes sont en général appelées "ateliers", "cellules" ou "îlots" [Bonetto 1985].

Nous supposons par la suite, qu'un SFPM est composé de plusieurs cellules flexibles. Chaque cellule comporte plusieurs îlots de fabrication et des ressources isolées, qui sont reliés par un moyen de transport central, par exemple des chariots automoteurs ou un système de convoyage. Un îlot est composé de plusieurs ressources de transformations, de moyens de stockage et de manipulateurs.

L'hypothèse porte donc sur la structure organisationnelle des ateliers : les ressources de production sont reliées par un moyen de transport. Les termes "îlots" et "cellules" ont été choisis pour introduire une hiérarchie et pour simplifier la compréhension de la suite. On trouvera certainement dans d'autres publications le terme "cellule" pour une installation qui sera selon nos critères un îlot et vice versa, car la terminologie n'est pas unique dans ce domaine.

Les moyens de transport dans les différentes structures déterminent les liens existant entre les composants d'une structure et définissent ainsi une cohésion entre les ressources de production. Ces liens sont mis en évidence par les arcs du graphe des accessibilités du système et seront explorés pour la définition des sous-structures.

Outre les moyens de transports, la deuxième caractéristique de chaque unité (atelier, cellule ou îlot) est l'existence de tampons, qui découplent l'unité de son environnement et assurent, avec le moyen de transport, l'autonomie de chaque unité de production. Ces interfaces des composants sont des lieux de synchronisation appartenant au moyen de transport central ou des lieux de synchronisation situés sur des stockeurs (tampons d'entrée et de sortie). Sur le graphe des accessibilités du système nous pouvons les identifier par leurs arcs adjacents, qui sont étiquetés différemment.

Sur la figure II.11 (graphe des accessibilités de la cellule de l'École Centrale de Lille), les

noeuds appartenant au convoyeur (P2, P3, P5, P9, P11) sont de ce type : ce sont des lieux de synchronisation où les arcs adjacents représentent des relations d'accessibilité établies par le convoyeur et par les robots.

Les difficultés essentielles des démarches de regroupement sont, d'une part, les critères, qui doivent, à partir des objectifs identiques, conduire à des résultats identiques, indépendamment du concepteur (critères uniques). D'autre part, la structuration obtenue ne doit pas influencer les caractéristiques du modèle, en l'occurrence les caractéristiques d'une commande adaptative, ou ne doit pas imposer à priori des solutions pour les autres phases de la conception (par exemple l'implantation des modèles sur calculateurs).

La démarche de structuration, présentée ici, est basée sur une analyse des propriétés du graphe des accessibilités du système (les définitions des propriétés utilisées sont décrites en annexe 5). A l'aide de ces propriétés, nous identifions de manière ascendante les sous-structures d'un atelier.

#### 4.3.2.2. *L'Identification des Îlots*

**Définition d'un Îlot de fabrication** : Un îlot de fabrication est un ensemble de ressources de production dont les lieux de synchronisation forment une composante symétrique et complète du graphe des accessibilités du système ou dont le sous-graphe est la jonction de plusieurs composants symétriques et complets, dont l'intersection n'est pas vide.

La définition distingue deux types d'îlot, le premier type est caractérisé par une composante symétrique et complète du graphe des accessibilités du système (îlot élémentaire). Les noeuds appartenant au domaine d'accessibilité d'un manipulateur forment un tel sous-ensemble de noeuds du graphe des accessibilités du système.

Le domaine d'un manipulateur est l'ensemble des lieux physiques qui sont reliés par des **liens d'échange** par le préhenseur du manipulateur. Il s'agit donc des lieux de synchronisation sur lesquels la pince du robot est chargée et déchargée. La caractéristique des transferts effectués par un manipulateur est sa capacité à réaliser des transports bidirectionnels entre ces lieux de synchronisation. Au niveau du graphe, cela se traduit par la symétrie des arcs reliant les noeuds qui modélisent ces lieux de synchronisation, c'est-à-dire que les noeuds sont mutuellement en relation d'accessibilité.

Dans le cas de notre exemple (figure II.11), nous retrouvons les sous-ensembles de noeuds

suivants :  $E_{R1} = \{P2, T\}$ ;  $E_{R2} = \{E, Cu\ 1, Cu\ 2, Cu\ 0, P3, P5\}$ ;  $E_{RP} = \{P9, P11, S, Sa, Zo\}$ ;  $E_{RA} = \{Sa, Zo\}$ . Chacun des ensembles appartient au domaine d'un robot (indiqué en indice) et leurs sous-graphes sont complets et symétriques.

La cellule de notre exemple comporte alors deux îlots élémentaires : l'îlot 1 composé du tour et  $R_1$  et l'îlot 2 composé de E, Cu, et  $R_2$ . L' îlot 3 composé de S, Sa, Zo, de RA et du robot portique RP est un îlot complexe. Ce deuxième type d'îlot, énoncé par la définition, fait référence à des îlots comportant une ou plusieurs ressources partagées entre deux îlots élémentaires.

Dans le cas des ressources partagées, ou des zones critiques, qui sont accessibles par plusieurs robots, l'intersection des ensembles des nœuds, qui forment des sous-graphes complets et symétriques d'un atelier, n'est pas vide. Dans le cas de l'îlot 3, l'intersection comporte les lieux {Sa et Zo}. L'intersection non vide traduit le besoin de coordination globale des accès à ces ressources, dont les lieux caractéristiques sont les éléments de l'intersection.

Détaillons ce point à l'aide de l'exemple de la figure II.17. Elle montre le cas de deux machines, qui sont chargées par des robots dédiés. Mais ces robots peuvent également s'échanger des pièces par l'intermédiaire du stockeur rotatif (ST1), qui devient ainsi une ressource partagée. Le graphe des accessibilités de ce système comporte alors deux composants symétriques et complets :  $E_{R1} = \{T1, ST1, P1\}$  et  $E_{R2} = \{Cu1, ST1, P2\}$ . L'intersection de ces deux ensembles de nœuds comporte l'élément {ST1}.

En effet, le stockeur rotatif ne doit pas être sollicité par une commande, tant que l'un des deux robots effectue un chargement. L'accès à cette ressource doit être exclusif et nécessite la définition d'un protocole de communication entre les deux robots qui s'exprime sous forme d'une exclusion mutuelle.

La définition d'un îlot reflète bien les hypothèses sur l'organisation des systèmes en composants autonomes. La forte cohésion des ressources d'un îlot se traduit sur le graphe par des composants symétriques et complets. Elle est due à la présence d'un robot (ou plusieurs robots ayant le même domaine) ou à la présence d'une ressource partagée par deux composants symétriques et complets.

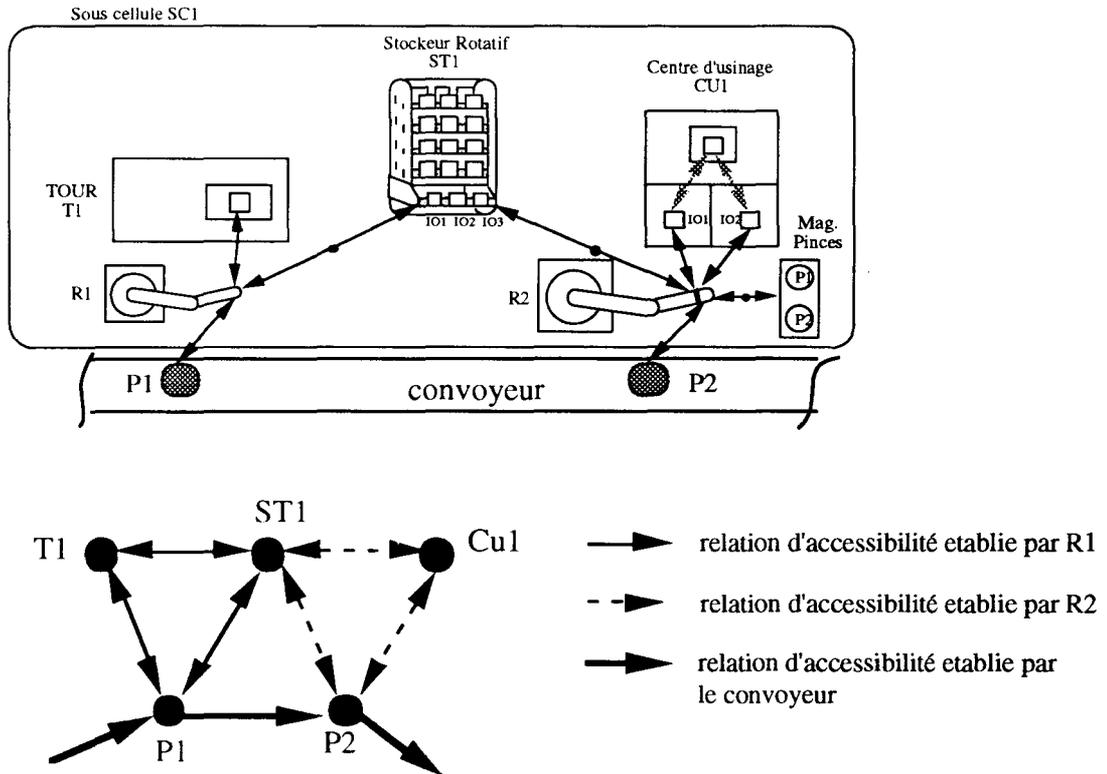


Figure II.17 : ressource partagée

#### 4.3.2.3. L'Identification d'une Cellule

**Définition d'une Cellule :** Une cellule au sein d'un SFPM est un ensemble de ressources, éventuellement groupées en îlots de fabrication, dont les lieux caractéristiques forment une composante fortement connexe du graphe des accessibilités du système et dont la forte connexité est établie par un **seul** moyen de transport.

Le sous-système "cellule" est, lui aussi, défini par rapport à un moyen de transport qui établit une cohésion entre les ressources de production. La cohésion est cependant moins forte que celle que nous avons évoquée pour un îlot (forte connexité des lieux caractéristiques au lieu de composants complets et symétriques).

Conformément à nos hypothèses, le moyen de transport centralisé forme l'ossature d'une cellule. Les composants d'une cellule, îlots ou ressources non groupés, échangent des produits par l'intermédiaire de ce moyen de transport. Les lieux d'interface des îlots sont, par conséquent, des lieux de synchronisation, qui appartiennent au domaine d'un robot et désignent

en même temps un lieu du moyen de transport central.

Dans le cas de l'installation de l'École Centrale de Lille, qui est, en effet, selon notre définition, une cellule, les interfaces des îlots sont donc le lieu P2 pour l'îlot 1, les lieux P3 et P5 pour l'îlot 2 et les lieux P9 et P11 pour l'îlot 3.

#### **4.3.2.4. L'Identification d'un Atelier**

**Définition d'un Atelier :** Un atelier est un ensemble de ressources éventuellement groupées en cellules et îlots de fabrication.

Nous distinguons deux types d'atelier selon la nature du flux des produits entre les cellules le constituant. Dans le premier cas, les différentes cellules d'un atelier sont groupées chacune par rapport à un moyen de transport central de type convoyeur. L'échange des produits entre les cellules est réalisé par l'intermédiaire d'un moyen de stockage. Ce moyen de stockage découple le flux entre cellules, mais permet aux produits de transiter dans les deux sens. Le moyen de stockage, qui peut faire partie d'un îlot, forme ainsi l'interface entre les cellules.

Dans le deuxième cas, les cellules sont reliées par un moyen de transport unidirectionnel. Le graphe des accessibilités du système n'est donc pas fortement connexe. Le moyen de transport établit un flux de produit orienté entre les cellules et impose donc un ordre de passage des produits dans les cellules.

#### **4.3.2.5. Conclusion**

Dans ce paragraphe, nous avons montré un regroupement des ressources par des critères uniques, basé sur leur cohésion. Nous avons associé à ces regroupements des termes désignant les sous-systèmes d'un SFPM. Cette identification est réalisée à partir de définitions strictes permettant l'identification des cellules et des îlots.

Ces définitions conduisent au modèle structurel d'un SFPM. Ce modèle met en évidence le regroupement de ces ressources. La figure II.18 montre ce modèle pour la cellule de l'exemple. Les arcs représentent la relation "est composé de".

Le regroupement mène ainsi à une modélisation concise de la structure d'un SFPM, qui est exploitée pour la génération du modèle du système de transport. Les sous-structures identifiées d'un SFPM montrent une forte cohésion interne, qui nécessite une coordination globale des ressources les composant. Chaque sous-structure est caractérisée par des tampons d'entrée et de

sortie de produits, qui correspondent à des interfaces.

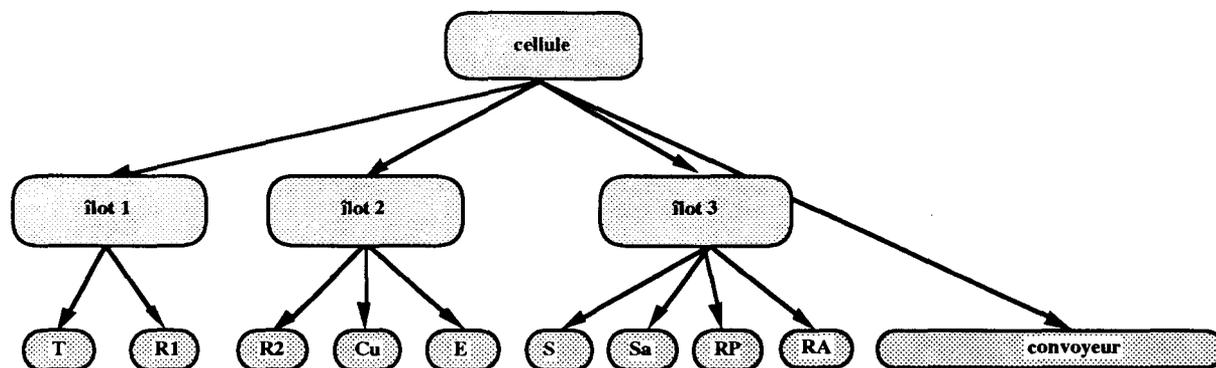


Figure II.18 : modèle structurel de l'exemple

### 4.3.3. La Génération des Modèles Réseau de Petri

#### 4.3.3.1. La Hiérarchie Proposée

Nous proposons une hiérarchie des modèles basée sur l'aspect structurel de l'atelier. Le modèle du niveau atelier est composé à partir des modèles des cellules, le modèle du niveau cellule correspond au modèle du moyen de transport centralisé de chaque cellule et les modèles des flots indiquent les transferts à l'intérieur de chaque flot.

La conception part du modèle structurel du système. Pour chaque sous-structure, ses transferts sont modélisés indépendamment. Le modèle final est obtenu par un assemblage des différents modèles générés en respectant les liens de hiérarchie exprimés sur le modèle structurel. Nous présentons par la suite la génération des différents modèles de manière ascendante.

#### 4.3.3.2. La Modélisation des Îlots

La conception du modèle d'un flot part de la composante du graphe des accessibilités du système décrivant cet flot. Les nœuds de ce graphe correspondent aux lieux caractéristiques (lieux opératoires et lieux de synchronisation) de l'îlot, les arcs mettent en évidence les transports possibles entre ces lieux (relations d'accessibilité) et, par leur étiquette, le moyen de transport qui réalise l'échange.

La modélisation des transferts d'un flot nécessite la gestion de ses emplacements de

stockage. De ce fait la décomposition structuro-fonctionnelle est une deuxième donnée pour la génération des modèles.

Nous exposons par la suite la démarche de génération à l'aide de l'exemple de l'îlot 2 de la cellule de l'École Centrale de Lille. Un deuxième exemple de génération est mis en annexe 6.

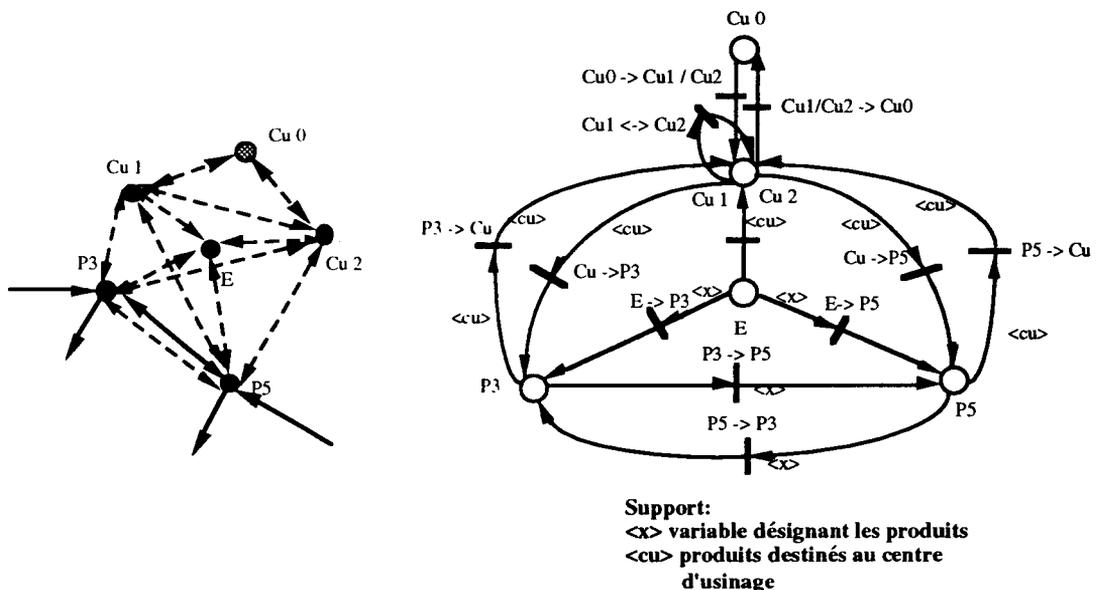
**Étape 1 : Traduction de la composante du graphe correspondant à l'îlot en réseaux de Petri.**

La traduction du graphe en réseaux de Petri substitue aux nœuds du graphe les places et aux arcs les transitions. Une place modélise ainsi les lieux caractéristiques de l'îlot et une transition représente un transfert élémentaire à l'intérieur de l'îlot, réalisé par un moyen de transport spécifique. Nous obtenons finalement un graphe d'état qui traduit la flexibilité des transports représentée sur le graphe des accessibilités du système.

Nous définissons ensuite des prédicats qui testent la valeur des produits. Un prédicat correspond à une destination dans l'îlot, une destination étant l'identificateur d'un lieu opératoire de l'îlot. Les arcs en amont et en aval de chaque transition modélisant un transfert vers ce lieu sont étiquetés par cette variable. Les arcs adjacents à chacune des places qui représentent des lieux appartenant à la même ressource portent de ce fait la même étiquette.

Ainsi, seuls les produits admis sur une ressource peuvent franchir une transition modélisant un transfert qui à cette ressource comme point de départ ou comme destination.

La figure II.19 illustre le résultat de cette première étape de la génération pour l'îlot 2 de la cellule de l'École Centrale de Lille.



**Figure II.19 : la traduction du graphe des accessibilités en réseaux de Petri**

Remarques :

- (1) Afin d'obtenir un modèle plus concis pour la présentation ici, nous avons modélisé les deux lieux de synchronisation du centre d'usinage, Cu1 et Cu2, par une seule place.
- (2) Nous avons éliminé également les arcs adjacents au tampon d'entrée.

**Étape 2** : *Intégration de la gestion des emplacements.*

a) la gestion des emplacements des supports de transport.

Les places qui correspondent aux tampons d'entrée et de sortie des îlots (places P3 et P5 pour l'îlot 2) sont marquées par des jetons de type  $\langle e, x \rangle$  modélisant un emplacement chargé d'une ressource à transfert libre et par des jetons de type  $\langle e \rangle$  (emplacement vide). Le déchargement d'un produit du support de transport dissocie le produit ( $\langle x \rangle$ ) de son emplacement ( $\langle e \rangle$ ) et génère un emplacement vide. Nous modélisons le déchargement par des arcs supplémentaires, étiquetés par  $\langle e \rangle$  qui relient les transitions de déchargement avec les places "entrée / sortie" de l'îlot. Le chargement des supports est représenté de manière équivalente.

b) la gestion des emplacements des ressources de l'îlot.

Nous générons pour chaque lieu de synchronisation de l'îlot (à l'exception, bien sûr, des places d'entrée / sortie) une place déterminant l'accès à ce lieu. Le marquage initial de cette place correspond à la quantité des emplacements de stockage du lieu physique modélisé par cette place (limite de capacité de la ressource). Si le lieu physique comporte plusieurs emplacements de stockage, leur identification par une variable est nécessaire afin de déterminer la localisation exacte des produits. Nous connectons la place ainsi générée en amont à chaque transition indiquant un transfert vers ce lieu et en aval aux transitions modélisant le déchargement.

La figure II.20 illustre le résultat de cette étape de génération pour l'îlot 2 de la cellule de l'École Centrale de Lille.

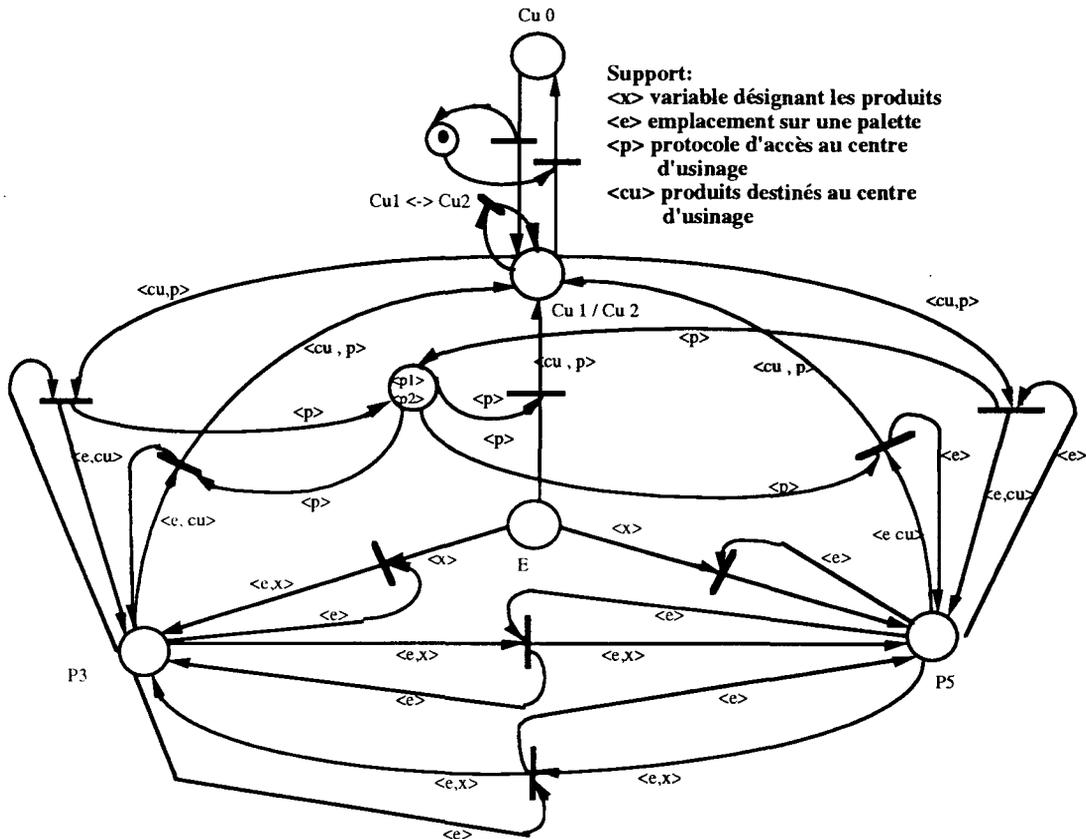


Figure II. 20 : îlot 2 de la cellule

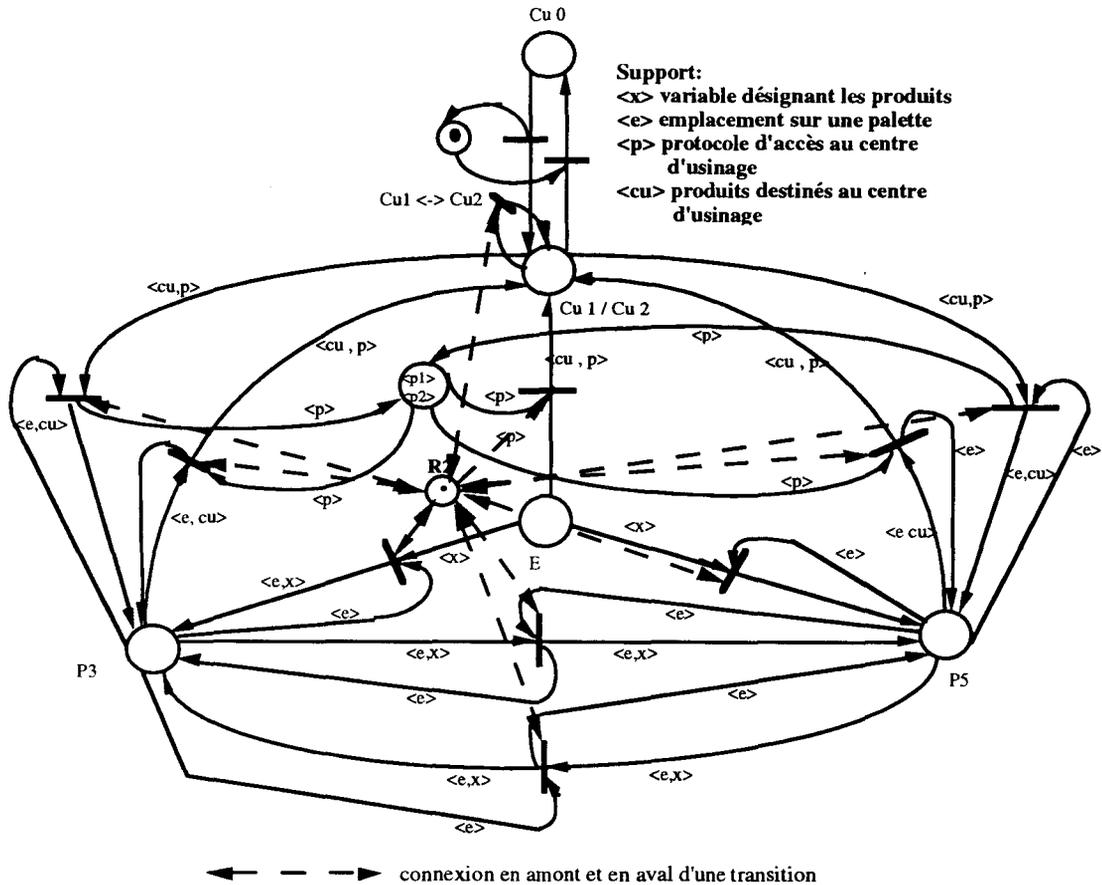
### Étape 3 : L'accès aux ressources partagées.

Les ressources partagées sont, d'une part, le ou les manipulateurs de l'îlot et, d'autre part, des lieux de synchronisation qui font partie du domaine de plusieurs manipulateurs, comme le stockeur rotatif de l'exemple de la figure II.17. Afin d'éviter des collisions, nous introduisons une place "mutex" qui modélise l'exclusion mutuelle d'accès à ces lieux. Elle est marquée par un jeton unique et elle est connectée en amont et en aval à chaque transition modélisant un transfert vers ce lieu ou à partir de ce lieu.

Dans le cas de notre exemple, le robot R2 est le moyen de transport unique de l'îlot 2. De ce fait une collision sur un lieu de synchronisation n'est pas possible dans cet îlot.

Mais le robot R2 peut être sollicité par plusieurs demandes de transfert (cf. marquage simultané de plusieurs places), il s'agit alors d'une ressource partagée. Afin de représenter l'exclusion mutuelle de deux transferts, nous représentons chaque robot d'un îlot par une place, connectée en amont et en aval de chaque transition modélisant un transfert réalisé par ce robot.

La figure II.21 indique le résultat de cette dernière étape de la génération pour l'exemple de l'îlot 2.



**Figure II.21 : accès aux ressources partagées**

Les trois étapes de la génération du modèle d'un îlot ne sont pas forcément exhaustives. Une adaptation des modèles obtenus face à des circonstances spécifiques peut s'avérer nécessaire. Par exemple l'intégration de conditions supplémentaires au niveau des arcs dans le cas des palettes spécialisées qui ne sont admises que pour un certain type de produit constitue une spécificité de cette nature.

Les modifications doivent cependant respecter le souci d'une modélisation rigoureuse, c'est-à-dire une représentation fidèle de toute la flexibilité et faire abstraction de toute politique de gestion ou d'allocation.

Le modèle obtenu par les étapes de génération modélise dans cet esprit les transferts possibles à l'intérieur de l'îlot. Il assure de par les protocoles d'accès un comportement sauf de l'îlot, indique de manière détaillée les chemins possibles dans l'îlot et permet la localisation de chaque produit.

Le modèle n'est pas autonome. Les indéterminismes comme le chargement d'un produit sur

l'un des deux tampons du centre d'usinage nécessitent une résolution, de même que les conflits d'accès résultant d'une demande simultanée d'accès à un lieu partagé d'une ressource.

Ces conflits se traduisent au niveau du modèle par la sensibilisation simultanée d'au moins deux transitions ayant la même place de type "protocole d'accès" ou "mutex" en aval qui est marquée par un seul jeton. Le tir d'une transition inhibe alors le tir de toute autre transition, les transitions sont en conflit de tir. Les conflits de tir se présentent donc en fonction de l'évolution du marquage des modèles, c'est-à-dire en fonction du comportement dynamique du système.

#### **4.3.3.3. La Modélisation d'une Cellule**

Conformément à notre hypothèse sur la structure des SFPM, les îlots sont reliés par un seul moyen de transport central. Ils échangent des produits avec le reste du système par l'intermédiaire de ce moyen et leurs tampons d'entrée et de sortie sont des lieux de synchronisation appartenant à la fois au moyen de transport central et à l'îlot. Les tampons ont donc un caractère d'interface pour l'échange des produits.

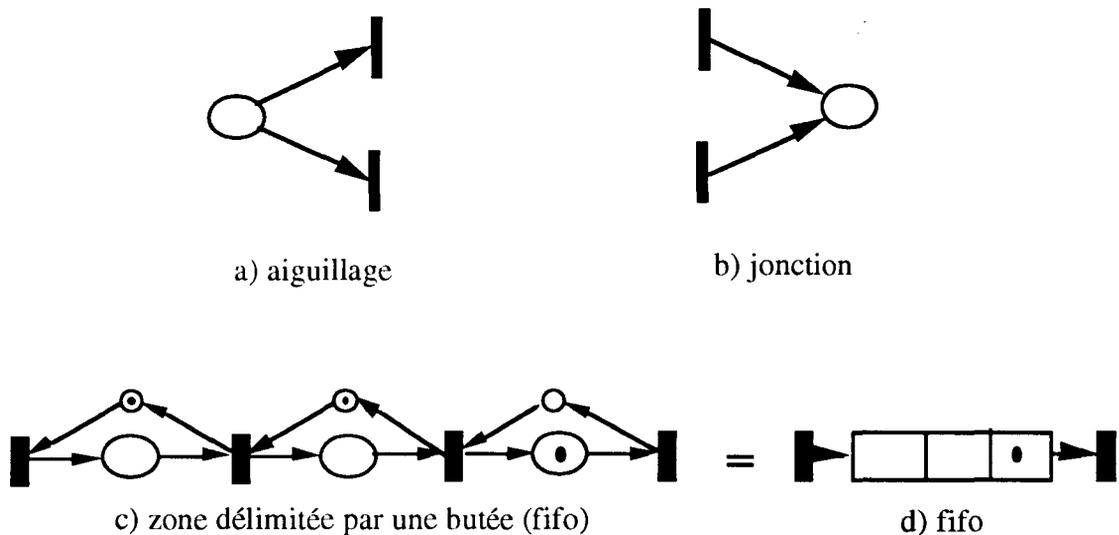
Le modèle d'une cellule correspond de ce fait au modèle du moyen de transport. La hiérarchisation du modèle du système de transport étant basée sur l'encapsulation de l'information, une cellule ne voit donc que les îlots la constituant, ou, plus précisément, ne voit que les interfaces de ces îlots à travers lesquelles l'échange des produits est effectué.

Les tampons d'entrée et de sortie de chaque îlot sont représentés par des places, qui, dans une structure de réseaux de Petri hiérarchisés, correspondent à des places de substitution. Leur marquage mène à l'insertion du modèle de l'îlot correspondant.

L'encapsulation de l'information se traduit aussi au niveau des jetons du réseau de Petri. Comme sur le moyen de transport central de type convoyeur ou système de chariots filoguidés les produits se déplacent sur des supports (palettes ou chariots), nous ne représentons au niveau du modèle de la cellule que ces supports. Nous faisons donc abstraction de la nature de la charge d'un support à ce niveau de la modélisation.

Le marquage d'une place de substitution, c'est-à-dire l'arrivée d'un support dans un îlot, rend la charge de ce support visible au niveau du modèle de l'îlot. Le jeton modélisant le support est en effet décomposé en ces emplacements élémentaires, notés  $\langle e \rangle$  ou  $\langle e, x \rangle$ , s'ils comportent un produit.

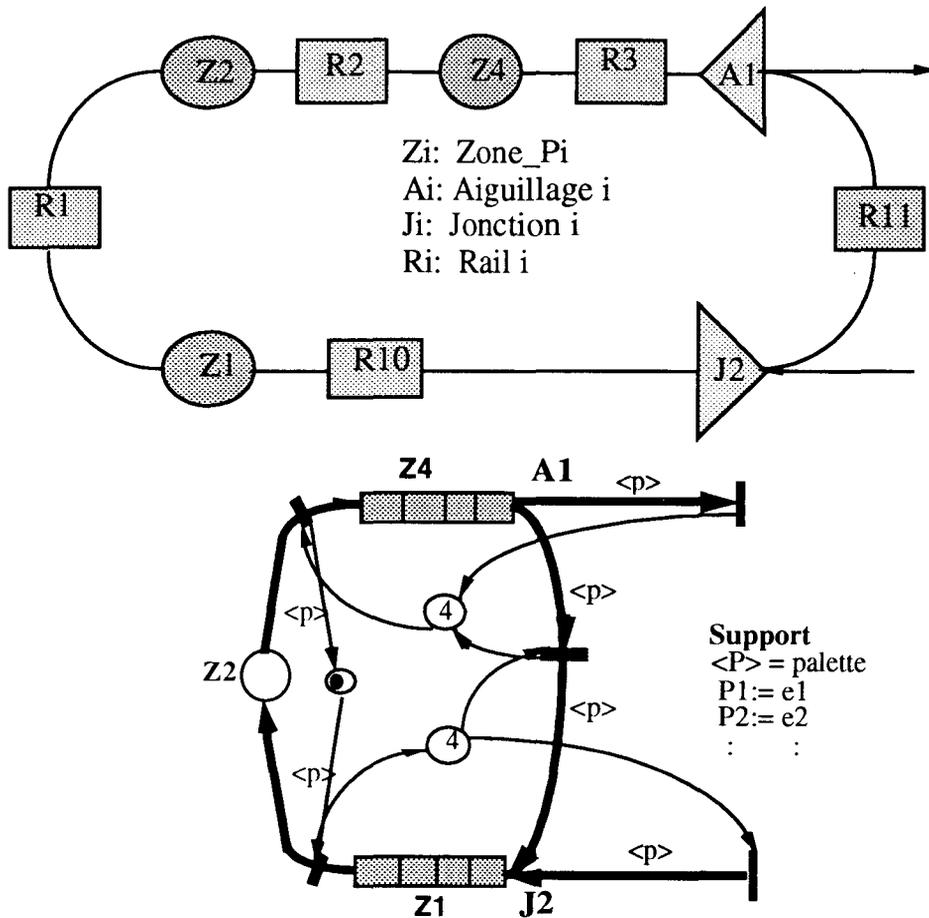
Le modèle d'une cellule, ou le modèle du moyen de transport central est généré par traduction du modèle arborescent obtenu lors de l'analyse du procédé (figure II.10 pour le convoyeur de l'exemple) en réseau de Petri. Les jetons du réseau modélisent ainsi l'évolution dynamique, c'est-à-dire le comportement, des supports de transport dans cette structure. Le modèle intègre de ce fait la gestion de ce type de ressources à transferts libres.



**Figure II.22 : primitives de connexion**

La figure II.22 indique quelles sont les primitives réseaux de Petri, qui modélisent les aiguillages, les jonctions et les zones d'arrêt des supports, qui sont délimitées par des butées. Nous utilisons la représentation agrégée d'une fifo (figure d) qui est équivalente au modèle de la figure II.22 c. La longueur maximale de chaque fifo correspond à la capacité de la zone représentée. La connexion de ces primitives décrit ainsi la topologie du moyen de transport. Afin d'assurer une modélisation du comportement du système sans blocage ni collisions, nous rajoutons des protocoles d'accès aux différentes zones et aux jonctions.

La figure II.23 montre un extrait du modèle du convoyeur de la cellule de l'École Centrale de Lille.



**Figure II.23 : génération du modèle du convoyeur**

De manière générale, le support d'un réseau de Petri à Prédicat / Transition représente les liens statiques existant entre ses différents éléments. La spécification du support de la figure II.23 indique alors le lien entre une palette et ses emplacements. Comme les palettes de la cellule ne peuvent transporter qu'un seul produit à la fois, un emplacement correspond à une palette.

Le marquage de la place  $Z_2$  de la figure II.23 correspond à l'arrivée d'une palette dans le tampon d'entrée et de sortie de l'îlot 1 de la cellule. Le marquage de cette place mène à l'insertion du réseau de Petri, c'est-à-dire la sous-page, modélisant l'îlot. Elle correspond ainsi à une place de substitution. La figure II.24 représente ce lien hiérarchique.

Outre les tampons des îlots, le moyen de transport peut comporter des lieux caractéristiques qui modélisent des lieux de transformations, comme, par exemple une zone de lavage des pièces pendant le transfert. Le modèle de la cellule intègre donc des lieux de synchronisation modélisant les tampons des îlots et des lieux de transformation.

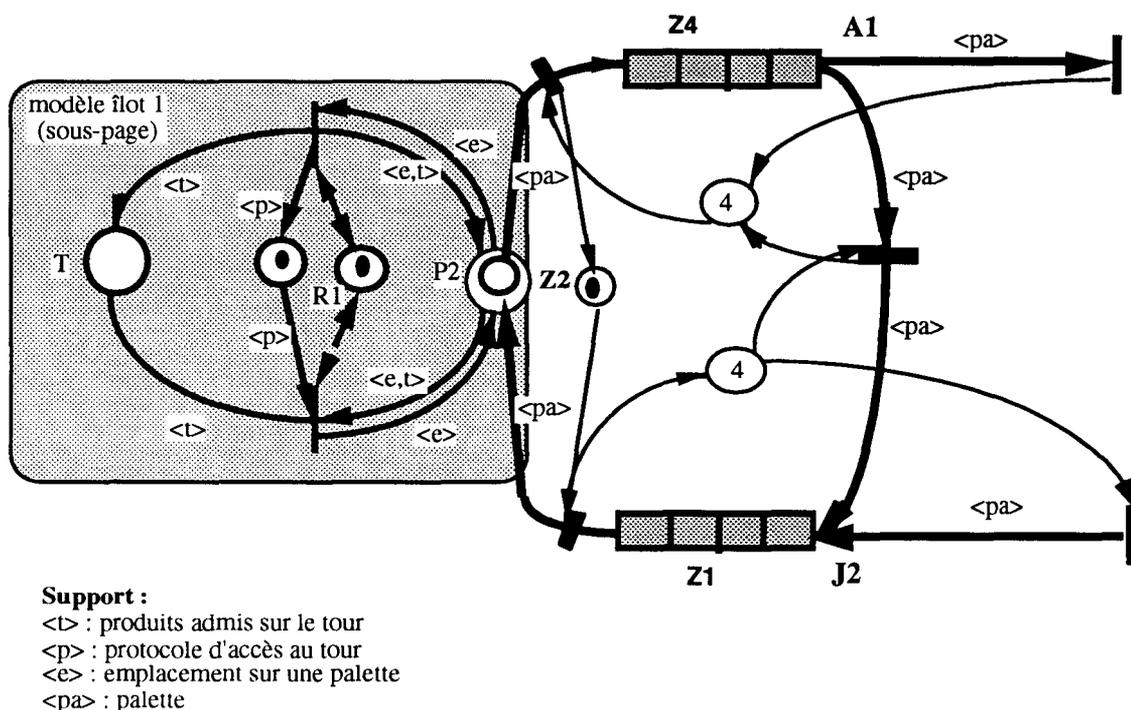


Figure II.24 : hiérarchie entre le modèle de la cellule et le modèle de l'îlot 1

#### 4.3.4. Conclusion

Comme pour le modèle de la cellule, nous pouvons définir une structure de réseaux de Petri hiérarchisés pour un atelier, qui met en relation les modèles des cellules le composant. Ce modèle met en évidence les tampons des cellules formant l'atelier et les flux des produits entre elles.

Le modèle du système de transport est ainsi obtenu par une construction modulaire d'un ensemble de modèles de type réseaux de Petri hiérarchisés. La hiérarchie des modèles correspond à la hiérarchie structurelle des sous-systèmes d'un SFPM.

Cette hiérarchie structurelle est déduite d'un modèle structurel du système construit à partir de définitions strictes qui permettent l'identification des îlots et des cellules comme composants d'un atelier. Les définitions de chaque module font référence à la cohésion de ses ressources établie par des moyens de transport. Cette cohésion est mise en évidence sur le graphe des accessibilités du système par des propriétés spécifiques.

Les transferts dans chaque sous-structure d'un atelier sont ensuite modélisés par réseaux de Petri. Au niveau des îlots, le modèle est obtenu par une traduction de la partie correspondante du graphe des accessibilités; au niveau des cellules, le modèle est une traduction du modèle du moyen de transport central obtenu lors de l'analyse du système. Cette démarche conserve la flexibilité des moyens de transports, mise en évidence pendant la phase d'analyse, au niveau du modèle final et au niveau d'abstraction adéquat.

La construction hiérarchique du modèle mène à l'encapsulation de l'information. Au niveau des modèles des îlots, les transferts possibles entre lieux caractéristiques sont représentés. Le modèle de la cellule indique les transferts entre les tampons des îlots et les lieux opératoires situés sur le moyen de transport central. Ces lieux correspondent en quelque sorte à des lieux caractéristiques vus du niveau cellule.

Le modèle du système de transport représente les contraintes physiques du système par l'intermédiaire des protocoles d'accès aux ressources. Il intègre de ce fait la gestion des emplacements de stockage au niveau approprié.

L'utilisation des jetons individuels modélisant les produits au niveau des îlots nous permet de localiser chaque produit à chaque instant dans le système. La modélisation de l'évolution dynamique des supports au niveau du modèle de la cellule permet une visualisation des routages possibles et de l'encombrement actuel du moyen de transport. En effet, les supports de transport étant individualisés, chaque produit et tous les emplacements vides sur les différents supports sont localisables à chaque instant.

Malgré le nombre important des informations représentées, le modèle du système de transport reste lisible grâce à sa construction hiérarchique.

Le modèle du système de transport complète l'aspect produit des modèles de la coordination. Les gammes opératoires induisent un objectif de transport par la donnée du point de départ et du point d'arrivée. Le modèle du système de transport indique les différents chemins possibles pour réaliser ce transfert. Un chemin est décomposé en transferts élémentaires au niveau de ce modèle et chaque transfert élémentaire demande la synchronisation de plusieurs ressources de production.

La figure II.25 reprend la représentation des quatre types de modèles de la commande de coordination. Les modèles de la coordination des produits présentés ci-dessus sont mis en

évidence par un fond grisé.

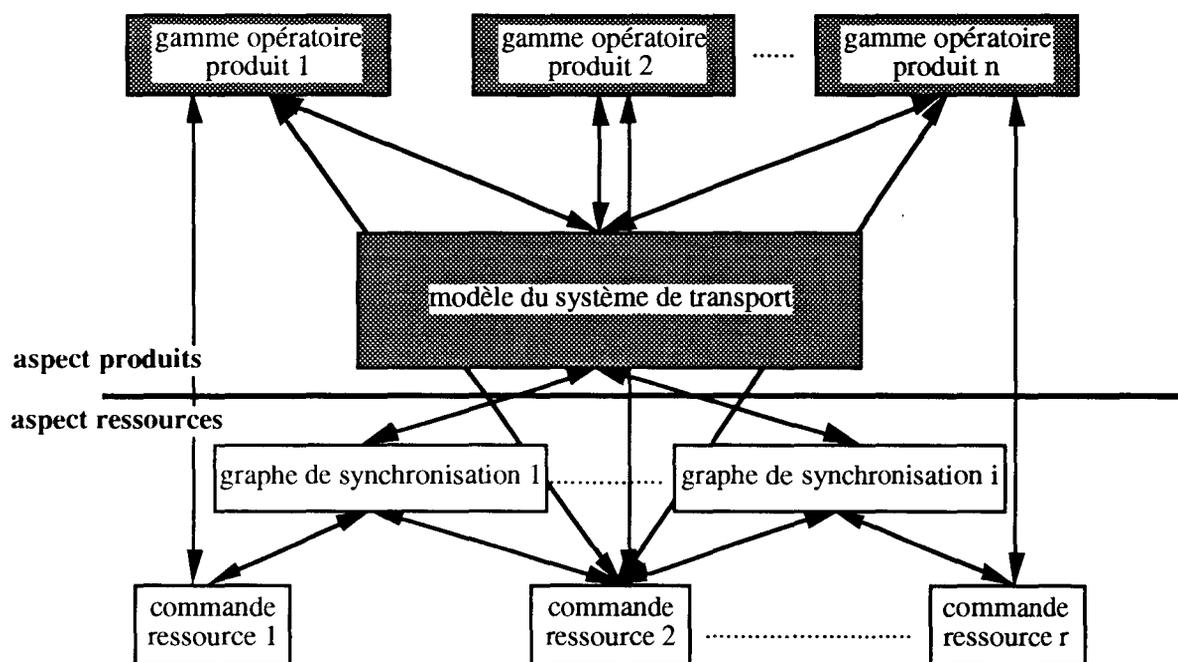


Figure II.25 : les modèles de la commande

#### 4.4. LES GRAPHES DE SYNCHRONISATION

##### 4.4.1. Introduction

Les graphes de synchronisation constituent le premier élément des modèles de la commande de coordination concernés par la coordination des actionneurs des ressources.

Au niveau des modèles du système de transport, un transfert est représenté par une séquence de transferts élémentaires. Chaque transfert élémentaire est modélisé par une transition. La transition représente ainsi un échange de produit entre deux lieux modélisés chacun par une place du modèle réseaux de Petri.

Le comportement de chaque ressource vis à vis de son chargement diffère d'une ressource à l'autre. Ce comportement nécessite la définition des protocoles d'échange spécifiques aux ressources sollicitées par un chargement (cf. moyen de transport, ressource de départ et ressource d'arrivée). Les graphes de synchronisation modélisent la suite des actions

élémentaires à réaliser par les organes opérants de chacune des ressources.

De ce fait, la suite de leurs actions est déterminée par le comportement des ressources concernées et les graphes de synchronisation font abstraction du produit transféré. Nous avons donc choisi une représentation des graphes de synchronisation par réseaux de Petri à Places / Transitions.

Par la suite nous distinguons deux types de graphes de synchronisation : les graphes de synchronisation inter-ressources et les graphes de synchronisation intra-ressources. Le modèle du système de transport montre deux types de transferts élémentaires correspondants à ces deux types de graphes de synchronisation.

Un transfert inter-ressources est caractérisé par le fait que le produit transféré change de support, par exemple le transfert de P3 -> Cu1/Cu2 dans l'îlot 2 de la cellule. Ces types de transfert sont exclusivement réalisés par des manipulateurs.

Pendant un transfert intra-ressource, le produit reste solidaire avec son support. C'est le support qui change dans ce cas de lieu physique dans le système. Les transferts élémentaires sur le convoyeur (par exemple P3 -> P5) ou les transferts de Cu1 /Cu2 -> Cu 0 en sont des exemples.

#### 4.4.2. La Génération des Graphes Inter-Ressources

##### 4.4.2.1. Les Données d'Entrée

Les graphes de synchronisation inter-ressources modélisent les protocoles de coopération des ressources intervenant lors d'un transfert élémentaire effectué par un manipulateur. Ces transferts sont mis en évidence sur le **graphe des accessibilités du système** par des arcs, dont l'étiquette fait référence à un manipulateur.

Les arcs relient deux lieux de synchronisation, donc deux lieux sur lesquels le produit change de support. Les actionneurs des ressources auxquels appartiennent ces lieux et les actionneurs du manipulateur identifié par l'étiquette sont donc à synchroniser.

Les actionneurs de chaque ressource, qui interviennent pendant un chargement, sont mis en évidence sur le **modèle arborescent** de l'atelier. A l'issue de la décomposition structuro-fonctionnelle nous avons associé à ces actionneurs une fonction de type "stockage".

Mais l'évolution libre des sous-organes opérants d'une ressource peut être contrainte par le comportement de la ressource, qui nécessite une séquence déterminée pour le chargement. Cette séquence est décrite par des **contraintes de fonctionnement**.

Au niveau des machines, nous avons déjà rencontré deux comportements différents qui ont été caractérisés par les termes "chargement interne" (par exemple le tour) et "chargement externe" (par exemple le centre d'usinage). Le chargement interne nécessite une séquence d'actions élémentaires qui est représentée sur la figure I.6, et nous pouvons définir une séquence analogue pour le chargement externe.

Au niveau des moyens de stockage, nous pouvons aussi distinguer deux types de stockage. Les stockages fixes, comme des étagères, et les stockeurs qui comportent des organes opérants tributaires de commandes comme par exemple un stockeur rotatif. Sur ce dernier type de ressource, les produits sont stockés sur des étagères qui ont une disposition fixe les unes par rapport aux autres. Les étagères ne sont accessibles que dans une certaine position, le chargement d'un produit sur une étagère donnée ou le déchargement d'un produit spécifique nécessite donc un positionnement préalable de l'étagère.

Le comportement des supports des moyens de transport, comme les chariots, les palettes ou les étagères d'un ascenseur, est identique au comportement des stocks rudimentaires, car, pour effectuer un chargement ou un déchargement, ils doivent être arrêtés, ainsi que nous l'avons vu, par exemple, sur le modèle de l'îlot 2 (figure II.21.). Seul l'arrêt d'une palette chargée en P3, rend la transition P3 -> Cu1 / Cu2 franchissable.

#### ***4.4.2.2. La Conception des Graphes de Synchronisation Inter-Ressources***

Un graphe de synchronisation décrit la séquence des actions élémentaires à réaliser par chaque sous-organe des ressources intervenant lors d'un transfert réalisé par un robot. Le produit se trouve au début et à la fin du transfert sur un lieu de synchronisation appartenant à des ressources différentes. Pendant le transfert il change deux fois de support, il est d'abord saisi par le préhenseur du manipulateur et ensuite chargé sur la ressource de destination.

La situation initiale, pour la génération, est la suivante :

- \* le produit est sur sa ressource de départ, mais il ne peut pas être accessible directement (cas d'un stockeur rotatif ou d'une machine avec chargement interne),
- \* le manipulateur est dans sa position de repli,

- \* la ressource de destination est dans une position de repli, prête à recevoir le produit.

Nous représentons cette situation par une place d'un réseau de Petri. Ensuite, chaque action élémentaire des différentes ressources est modélisée par une seule transition et l'état atteint après l'exécution par une place. La séquence est déduite des contraintes de fonctionnement des ressources.

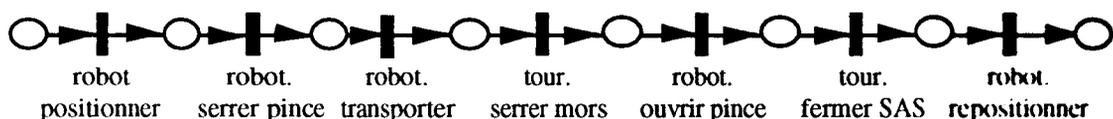
La dernière place modélise la situation suivante :

- \* le chargement du produit est effectué sur la ressource de destination
- \* le manipulateur est dans une position de repli
- \* la ressource de départ est libérée, donc prête à recevoir un nouveau produit.

A titre d'exemple nous présentons la conception du graphe de synchronisation, modélisant le transfert P2 -> T, qui correspond au chargement interne sur un tour. La situation de départ est la suivante :

- \* le produit se trouve sur une palette arrêtée en P2
- \* le manipulateur R1 est dans sa position de repli
- \* le tour est prêt à recevoir le produit, la porte du SAS (Système d'Accès de Sécurité) et les mors du mandrin sont ouverts.

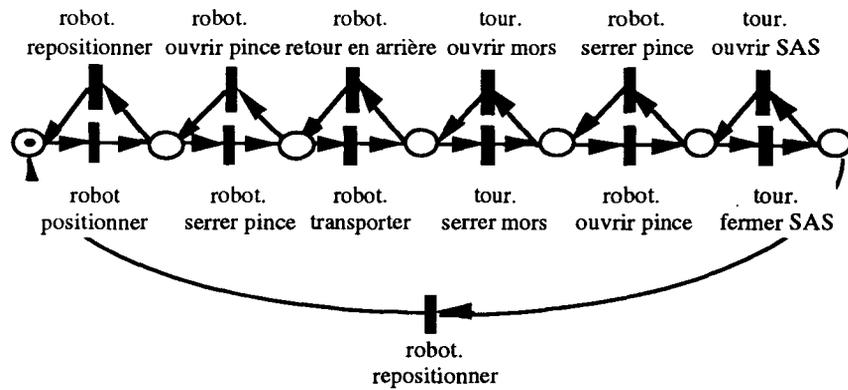
La suite des actions, déduite des contraintes de fonctionnement, se déroule alors comme suit. Le robot R1, pour saisir la pièce doit être **positionné** en P2 et **ferme** ensuite sa pince. Il **transfère** le produit dans les mandrins du tour. Les mors du tour sont **serrés**, le préhenseur du robot peut **lâcher** la pièce et le robot est ensuite **repositionné**. La position de repos atteinte, nous pouvons **fermer** la porte du SAS. La figure II.26 montre le réseau de Petri traduisant la suite de ces actions élémentaires.



**Figure II.26 : synchronisation d'un chargement interne**

Nous remarquons, que les actions élémentaires modélisées par les transitions de la figure II.26 sont des opérations positionnelles. Une des caractéristiques de ces actions est leur réversibilité. Comme nous avons montré dans le chapitre I de ce mémoire, la représentation des actions réversibles au niveau des modèles de la coordination est indispensable pour obtenir une

commande adaptative.



**Figure II.27 : graphe de synchronisation modélisant un chargement interne**

Nous proposons donc de représenter cette propriété des actions positionnelles au niveau des graphes de synchronisation, ainsi que nous l’avons fait pour la figure II.27. Nous modélisons de la même manière chaque transfert élémentaire inter-ressource par un graphe d’état mono-marqué. Le jeton modélise l’activité de la tâche.

Les réseaux de Petri ainsi obtenus correspondent donc à ces protocoles de coopération recherchés. Du fait du caractère réversible des opérations positionnelles réalisant les différentes étapes d’un transfert élémentaire, les graphes de synchronisation deviennent utilisables dans les deux sens et contribuent à une commande adaptative.

#### 4.4.3. Les Graphes de Synchronisation Intra-Ressource

Les transferts élémentaires de type intra-ressource correspondent à un transfert d’un support de produit, vide ou chargé, entre deux lieux d’une même ressource. Dans le cas de la cellule de notre exemple, il s’agit des transferts des palettes entre les différentes zones délimitées par des butées et des transferts des palettes du centre d’usinage entre ses tampons d’entrée / sortie et sa zone opératoire.

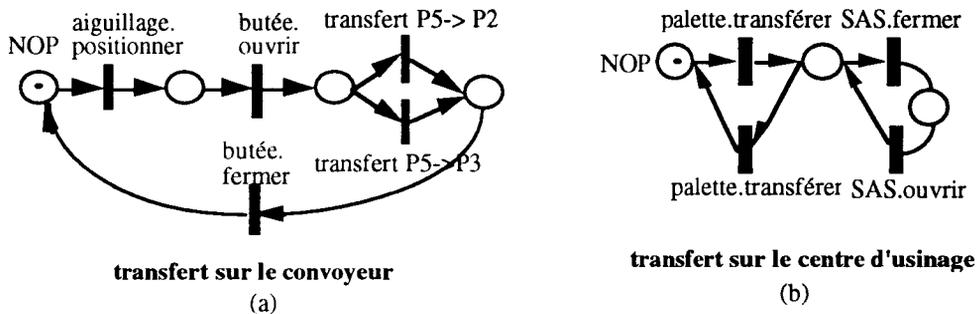
Ces graphes de synchronisation sont générés également en fonction du comportement de la ressource modélisée. Le transfert sur un chariot automoteur nécessite une synchronisation différente de celle utilisée pour le transfert d’une étagère entre deux étages d’un ascenseur.

La conception des graphes de synchronisation intra-ressource s’effectue d’une manière semblable à celle des graphes inter-ressources. A partir d’un état donné, par exemple une palette arrêtée dans une zone du convoyeur, nous recherchons la suite des actions élémentaires, qui est

conforme aux contraintes de fonctionnement et qui mène dans l'état souhaité, c'est-à-dire qui permet de transférer le support à la destination.

Contrairement à la synchronisation inter-ressource, les actions élémentaires ne sont réversibles que s'il s'agit d'un transfert de type "aller-retour" comme sur le centre d'usinage. Une palette d'un convoyeur, libérée par l'ouverture d'une butée, ne peut pas revenir en arrière.

La figure II.28b illustre le graphe de synchronisation retenu pour ce deuxième type de transfert élémentaire intra-ressource.



**Figure II.28 : synchronisation des transferts intra-ressource**

La figure II.28a indique le graphe de synchronisation retenu pour les transferts élémentaires P5->P3 et P5->P2. Si nous voulons réaliser un transfert P5->P2, nous devons d'abord positionner l'aiguillage et ensuite ouvrir la butée de P5. Cette séquence d'actions élémentaires est identique à celle à effectuer pour réaliser un transfert P5->P3, seul le paramètre de positionnement de l'aiguillage change. Nous représentons donc les deux graphes de synchronisation sur un seul modèle.

#### 4.4.4. Conclusion

La génération des graphes de synchronisation pour les transferts inter-ressources et intra-ressource demande une bonne connaissance du comportement des ressources concernées et surtout une bonne connaissance des répercussions du comportement de la ressource sur la suite des opérations élémentaires devant être réalisées pendant les échanges.

La décomposition structuro-fonctionnelle et les contraintes de fonctionnement spécifiées pour chaque ressource doivent permettre d'abord l'identification des actionneurs intervenant dans la synchronisation ou concernés par la synchronisation et ils doivent ensuite permettre la détermination d'un ordre de commandes à leur envoyer.

Les actions élémentaires étant des opérations positionnelles, elles sont souvent réversibles. La représentation de cette propriété dans les graphes de synchronisation contribue ainsi à une commande adaptative.

Nous obtenons ainsi des graphes d'état, qui modélisent par une tâche les protocoles de coopération des ressources. Chaque tâche correspond à une transition du modèle du système de transport. Elle indique les actions élémentaires à réaliser par les éléments opérants de chaque ressource intervenant lors d'un échange de produits (transfert inter-ressource) ou d'un transfert de support (transfert intra-ressource).

## **4.5. LES MODÈLES DES RESSOURCES SIMPLES**

### **4.5.1. Introduction**

Les modèles des ressources simples représentent le deuxième élément de la coordination de l'aspect "ressources". Ils coordonnent les actions élémentaires de chaque ressource et font, comme les graphes de synchronisation, abstraction du produit traité.

Nous avons retenu une modélisation du comportement des ressources par réseaux de Petri. Les modèles représentent l'ensemble des états atteints par la ressource, suite aux commandes envoyées aux actionneurs, et indiquent ainsi la suite des actions à réaliser par la ressource. En ce qui concerne les opérations positionnelles, les modèles indiquent les actions réversibles.

A ce réseau de Petri, nous associons une structure de données, qui mémorise les informations relatives à la configuration de la ressource. Ces informations sont mises à jour par un changement du mode opératoire (cf. chapitre I).

Les modèles des ressources usuelles comme les manipulateurs, les aiguillages d'un convoyeur ou les stockeurs sont mémorisés dans une bibliothèque. Face à une situation spécifique, ces modèles sont adaptés. Une modélisation du comportement des objets physiques ou logiques par réseaux de Petri avec une structure de donnée associée a été formalisée dans les Objets Coopératifs [Bastide 1991], mais est aussi utilisée pour la conception de logiciels temps réel [Valette et al 1988]. Elles sont appliquées dans les méthodes de conception de logiciel telles que MERISE [Tardieu et al 1983] ou HOOD [Vielcanet 1990].

A titre d'exemple, nous exposons par la suite le modèle d'un manipulateur et celui d'un tour.

### 4.5.2. Deux Exemples : Modèles d'un Manipulateur et d'un Tour

La figure II.29a présente le modèle du manipulateur. Le jeton qui réside dans la place NOP (= no operation) représente le manipulateur au repos, les différentes places correspondent aux états de la ressource et les transitions modélisent les commandes à envoyer aux actionneurs de cette ressource pour changer d'état.

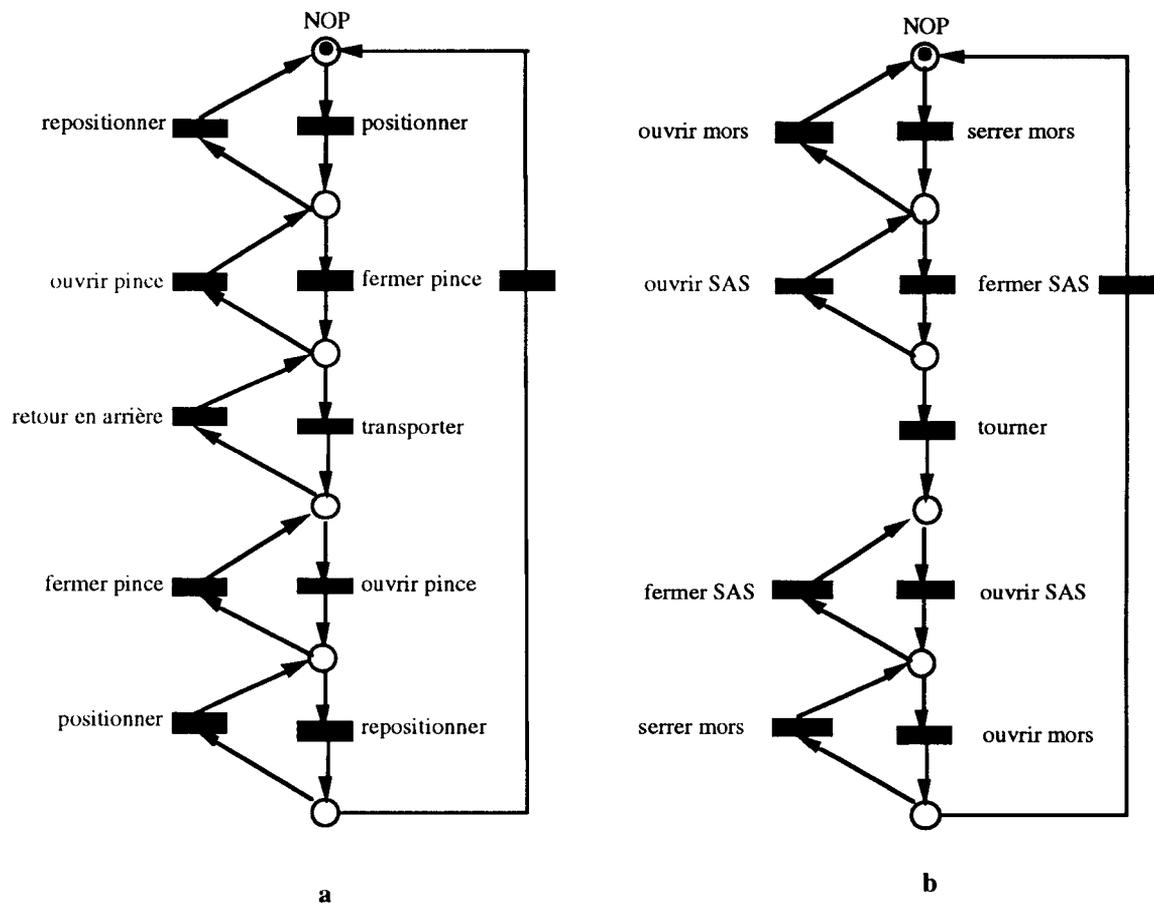


Figure II.29 : modèles d'un manipulateur (a) et d'un tour (b)

Le modèle comporte les allers et les retours des commandes réversibles. Leur intégration contribue, comme au niveau des graphes de synchronisation, à une commande adaptative. La transition "retour en arrière" représente la possibilité pour la ressource d'effectuer des transferts bidirectionnels. Comme nous avons vu dans le chapitre I, les retours en arrière permettent d'atteindre un point de reprise par rapport aux produits. La représentation de cette flexibilité de la ressource au niveau de son modèle contribue de ce fait à l'adaptabilité de la commande.

La structure de données associée à cette ressource mémorise aussi les informations relatives au mode opératoire actuel du système, c'est-à-dire à la configuration de la ressource. Si un

manipulateur doit changer de préhenseur pour saisir un certain type de produits, nous devons adapter le modèle à ce comportement de la ressource. Cela revient à rajouter une transition modélisant le changement de la pince. La structure de données associée au manipulateur indique alors la pince à choisir en fonction du produit à transporter, c'est-à-dire de la configuration demandée.

La figure II.29b représente le modèle d'un tour. Ce modèle comporte les commandes correspondantes aux actions réversibles. La transition "tourner" modélise de manière agrégée les actions élémentaires à réaliser pour effectuer la transformation, comme, par exemple, le choix des outils, leur positionnement et la rotation de la broche.

Les paramètres relatifs à une transformation sont donnés dans le programme de tournage au niveau de la commande numérique du tour. Le programme de tournage à effectuer par le tour est choisi en fonction du produit demandeur et de son état d'avancement dans la gamme opératoire. En effet, c'est la gamme opératoire qui détermine le programme de tournage à réaliser. Les relations programme-produit sont mémorisées au niveau de la structure de données et font donc partie de la configuration de la ressource.

### **4.5.3. Conclusion**

Les modèles des ressources simples représentent la couche la plus basse de la commande de coordination. Ils représentent le comportement de la ressource. Les informations des ressources relatives aux produits se limitent à des données statiques concernant les traitements à réaliser sur les produits. Ces informations sont données par le mode opératoire du système.

Par la mise en œuvre des actions réversibles, les modèles des ressources sont conçus dans l'idée d'une flexibilité maximale, ils répondent aux besoins d'une commande adaptative. La combinaison d'une modélisation du comportement de la ressource par réseau de Petri et d'une représentation des données statiques relatives à son exploitation dans une structure de donnée permet une modélisation adéquate. Elle est inspirée par la programmation orientée objet (association traitement + données) et contribue à une méthode de modélisation basée sur la modularité et la réutilisabilité de modules génériques.

## 4.6. CONCLUSION

La première étape de la conception de la commande de coordination d'un SFPM concerne la génération de quatre modèles différents. Chaque modèle, gamme opératoire, modèle du système de transport, graphes de synchronisation et modèles des ressources met en évidence les différents aspects d'un même procédé et de sa commande.

Ainsi à chaque type de modèle est confiée une tâche spécifique de la coordination, qui permet, d'une part une modélisation fidèle de la flexibilité du procédé, et d'autre part la représentation de la spécificité du procédé à des niveaux d'abstraction convenables.

Les gammes opératoires représentent, dans cet esprit, l'aspect qualitatif des objectifs de la production et indiquent de ce fait une exploitation spécifique du système. Une gamme opératoire caractérise un produit par ses transformations et fait abstraction de la réalisation des transferts entre deux lieux d'opération.

Le modèle du système de transport met en facteur tous les transferts de tous les produits en représentant les transports possibles à l'intérieur de l'atelier. Il intègre les mécanismes de gestion d'emplacement de stockage, qui garantissent la qualité du service de la fonction transport. Il contribue à la représentation du flux des produits dans le système dans une structure hiérarchisée de réseaux de Petri. Cette hiérarchie est issue d'une analyse structurelle du système, qui permet de dégager des sous-systèmes à l'intérieur d'un SFPM.

C'est au niveau des modèles du système de transport et des gammes opératoires que l'on trouve tous les indéterminismes et les conflits d'accès aux ressources du système. Les indéterminismes des gammes opératoires concernent la flexibilité du processus de fabrication, qui permet, d'une part de réaliser une même opération sur plusieurs ressources de transformation et d'autre part de permuter certaines opérations d'un produit donné.

Les indéterminismes traduisant la flexibilité de routage des produits dans le système, sont représentés sur le modèle du système de transport. Par l'intégration des protocoles d'accès dans le modèle, les conflits d'accès aux ressources partagées y sont également représentés. Ces conflits se produisent lors du comportement dynamique du système, en fonction de l'état courant du système de transport.

L'intégration du comportement des ressources au niveau des graphes de synchronisation permet la prise en compte des spécificités des ressources et l'intégration du caractère réversible des opérations. Ils définissent des protocoles d'échange, qui font abstraction du type du produit transporté.

Les modèles des ressources coordonnent finalement les actions élémentaires de chaque ressource. La modélisation des actions réversibles contribue à une commande adaptative.

Cette décomposition de l'information au sein des différents modèles et l'association d'un modèle spécifique à chaque tâche de la coordination combinent les avantages d'une approche produit avec les avantages d'une modélisation du comportement des ressources.

## **5. LA COMMUNICATION ENTRE LES MODÈLES**

### **5.1. INTRODUCTION**

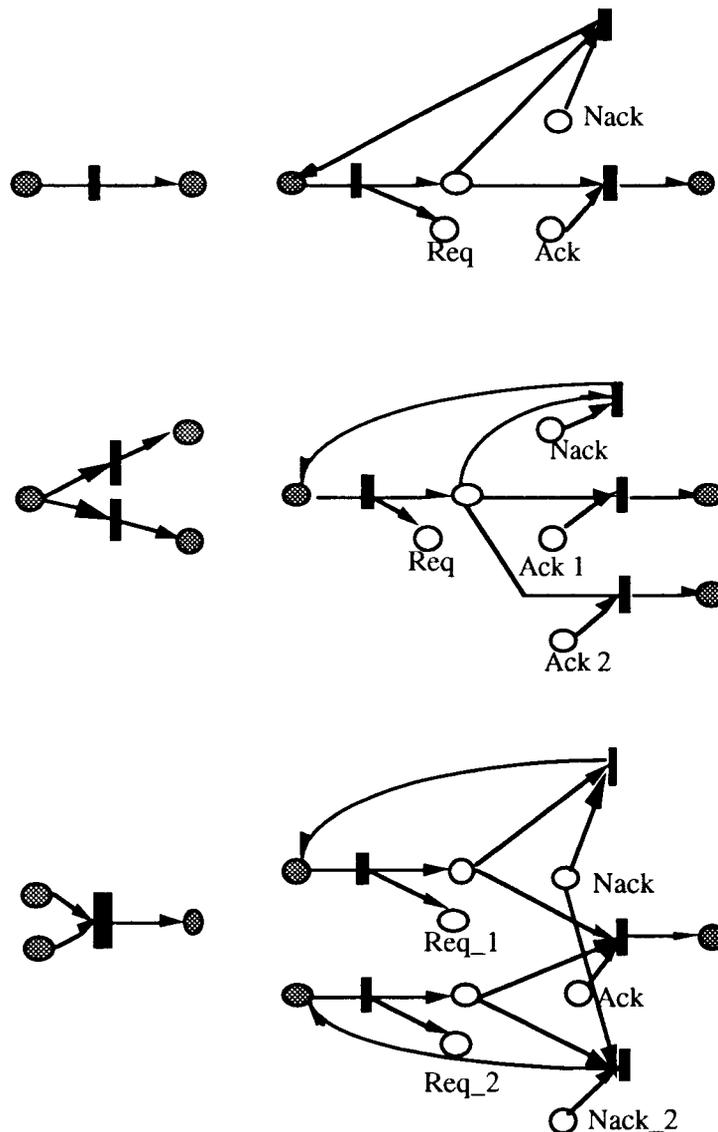
Nous avons conçu les modèles de la commande de manière indépendante. Il convient maintenant de définir les modalités de communication entre les différents modèles de la commande de coordination.

Néanmoins, nous avons identifié certaines relations entre ces différents modèles. Le modèle du système de transport a une fonction de service pour les gammes opératoires, les gammes opératoires déterminent les transports à réaliser sur le système. Ces deux modèles doivent donc communiquer.

Les transferts élémentaires, représentés par des transitions sur le modèle du système de transport sont modélisés par des graphes de synchronisation. Le franchissement d'une transition du modèle du système de transport détermine par conséquent la tâche à exécuter au niveau des graphes de synchronisation.

Enfin, en ce qui concerne les modèles des ressources, ils communiquent à la fois avec les gammes opératoires et avec les graphes de synchronisation.

La définition de ces communications revient à préciser les modalités de communication et de la nature des informations à passer. La décomposition successive de l'information aux différents niveaux de modélisation et les besoins d'une commande sûre, c'est-à-dire synchronisée sur l'exécution des actions par le niveau inférieur, imposent une communication de type "client-serveur". Le niveau supérieur, c'est-à-dire le client, demande l'exécution d'un certain service par le serveur et seule la réalisation du service par le niveau inférieur permet l'évolution du marquage du modèle "client". Nous symbolisons la communication synchronisée par les éléments de la figure II.30.



**Figure II.30 : modules de communication**

Les données échangées correspondent aux jetons modélisant les produits ou, dans le cas d'un système de transport complexe, leurs supports (palettes, chariots etc.). Ils sont ainsi

porteurs de toute information entre les modèles. Les jetons transitent d'un modèle vers un autre par l'intermédiaire des places "ack", "nack" et "req". En effet, ces places sont partagées par les modèles communiquant et les modèles sont de ce fait connectés.

Le marquage d'une place "req" (= requête) traduit la demande d'exécuter une opération ou une suite d'actions par le niveau inférieur. La place "ack" (acknowledgement) est marquée par le serveur après l'exécution de l'opération. Le marquage de la place "nack" (no acknowledgement) indique que le service demandé n'a pas pu être exécuté. Si l'état du procédé le permet, nous injectons ensuite le jeton demandeur en amont de la transition modélisant le service demandé. Le marquage de cette place correspond à un état stable et représente de ce fait un point de reprise. Les modules de connexion contribuent ainsi à une commande sûre permettant à priori la prise en compte des défaillances du niveau inférieur.

Dans la partie suivante, nous présentons la communication des différents modèles de la commande de coordination selon une démarche ascendante.

## 5.2. LA COMMUNICATION ENTRE GRAPHES DE SYNCHRONISATION ET MODÈLES DES RESSOURCES

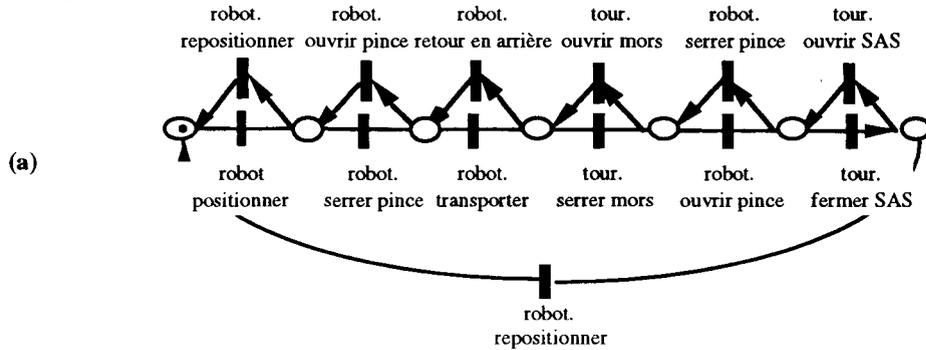
La figure II.31 indique la communication du graphe de synchronisation avec les modèles des ressources. Les transitions des graphes de synchronisation sont remplacées par les modules de la figure II.30. Nous avons remplacé, à titre d'exemple, la transition "robot.serrer pince" par le module correspondant (figure II.31b).

Les places "requête" des graphes de synchronisation sont connectées aux transitions des modèles des ressources correspondants à l'opération requise. Ces places sont marquées par le jeton modélisant le produit qui demande l'action. Les réponses sont générées à partir du modèle de la ressource. Le modèle du manipulateur est connecté à l'interface du procédé, qui génère les réponses permettant le marquage des places de communication. La réponse de l'interface est ainsi propagée au niveau des graphes de synchronisation par l'intermédiaire des modèles des ressources.

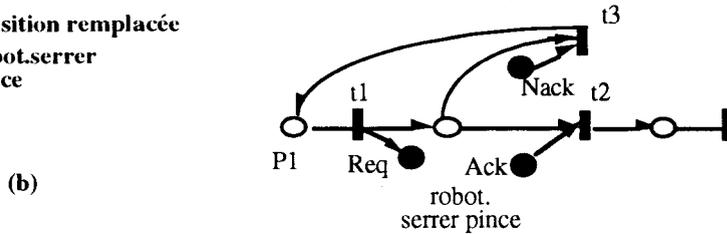
Si l'action "serrer pince", c'est-à-dire le service demandé par le graphe de synchronisation, a échoué, alors la transition t3 du modèle du manipulateur (figure II.31c) est tirée. Le jeton modélisant le produit marque la place "nack serrer pince" au niveau du modèle du manipulateur. Puisque cette place est une place de communication entre les deux modèles, et donc partagée par les deux modèles, son marquage mène au tir de la transition t3 du graphe de synchronisation

(figure II.31b). Le jeton est alors injecté dans la place P1 du même modèle. A partir de cette place, qui correspond à un point de reprise pour le manipulateur, une nouvelle tentative pour la fermeture de la pince peut être commandée ou le robot peut être repositionné.

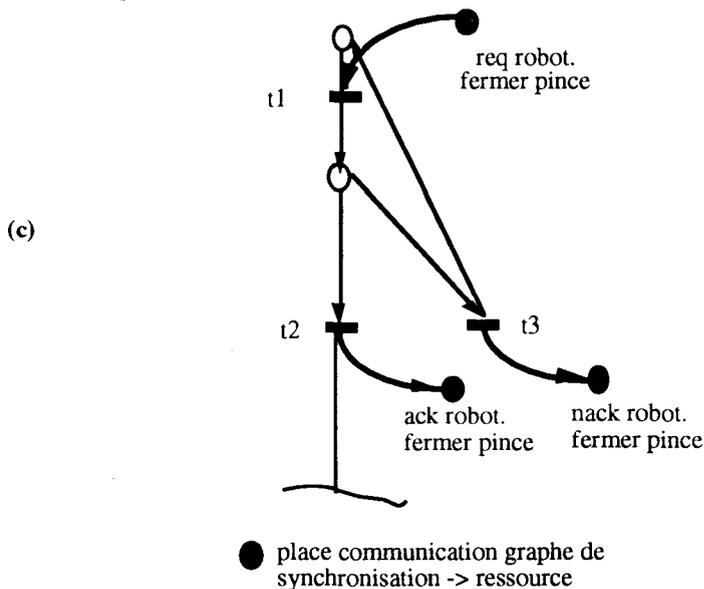
**graphe de synchronisation**



**transition remplacée  
robot.serrer  
pince**



**modèle du manipulateur**



**Figure II.31 : communication ressources / graphes de synchronisation**

Si, par contre, l'interface ne génère aucune réponse, le modèle du manipulateur n'évolue pas et le graphe de synchronisation reste gelé dans son état. L'utilisation des modules de communication contribue ainsi à une commande sûre et permet, par l'intermédiaire des places

“*nack*”, d’atteindre des points de reprise de la ressource.

### 5.3. LA COMMUNICATION ENTRE LE MODÈLE DU SYSTÈME DE TRANSPORT ET LES GRAPHES DE SYNCHRONISATION

Chaque transition du modèle du système de transport correspond à un transfert élémentaire qui est modélisé par un graphe de synchronisation distinct. Nous en avons identifié deux types : les graphes de synchronisation inter-ressources modélisant un transfert élémentaire d’un produit par un robot et les graphes de synchronisation intra-ressource modélisant le transfert d’un support au sein d’une ressource.

Une caractéristique importante d’un manipulateur concerne sa capacité à réaliser des transferts bidirectionnels. Ceci permet, si la commande reflète cette possibilité, de revenir en arrière et de recharger le produit sur son point de départ, au cas où le chargement initial aurait échoué. Le point de départ représente alors un point de reprise.

Du fait de la réversibilité des actions élémentaires, les graphes de synchronisation inter-ressources sont utilisables dans les deux sens. Le recouvrement peut alors exploiter la flexibilité du manipulateur et commander un retour en arrière.

La connexion des places de communication permet ensuite de propager l’information au niveau du modèle du système de transport. La figure II.32 précise la connexion d’un graphe de synchronisation au modèle du système de transport. Comme l’indique cette figure, qui montre le cas du transfert P3 -> Cu1 / Cu2 dans l’îlot 2 de notre exemple, chaque transition du modèle du système de transport est remplacée par son module correspondant de la figure II.30.

Le marquage de la place “*req* P3 -> Cu ” par le franchissement de la transition *t1* du modèle du système de transport (figure II.32), mène au franchissement de la transition “*robot.positionner*” au niveau du graphe de synchronisation. Si le transfert se déroule sans erreur, le franchissement de la transition “*robot.repositionner*” marque la place “*ack* P3 -> Cu. La transition *t2* est alors tirée et le jeton modélisant le produit transporté marque la place Cu.

Le cas échéant, le produit est remis sur la palette en P3 par une suite d’actions élémentaires, qui inverse les actions effectuées jusqu’à l’occurrence de l’erreur. Le marquage de la place “*nack*” au niveau du graphe de synchronisation mène au franchissement de la transition *t3* et le produit est en effet localisé en P3.



graphe de synchronisation.

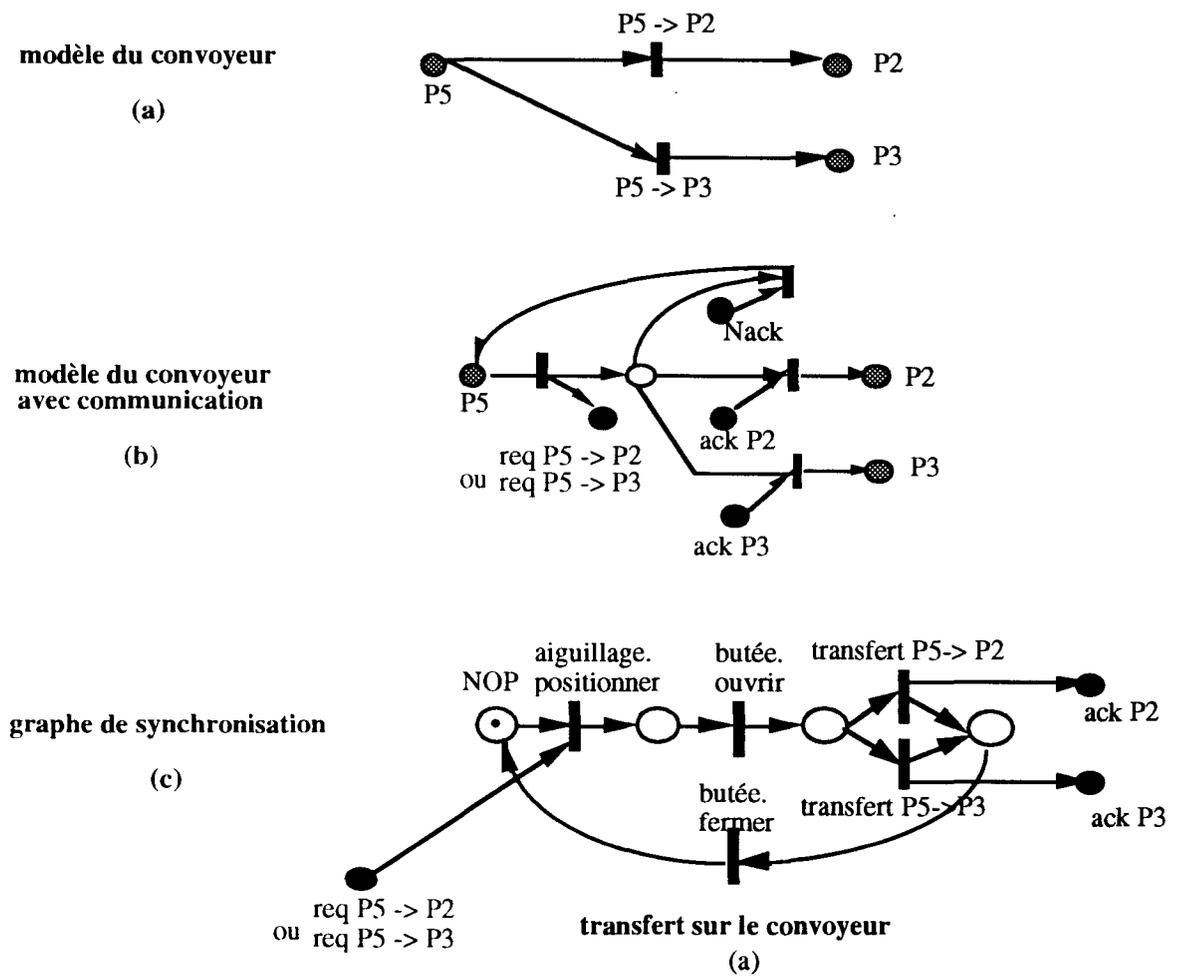
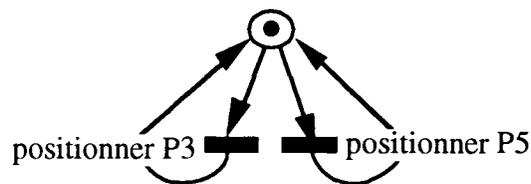


Figure II.33 : communication modèle de la cellule/ graphe de synchronisation

Cependant, l'indéterminisme n'est pas levé à ce niveau de la commande. En effet, c'est le positionnement de l'aiguillage qui met en œuvre le routage de la palette. C'est pour cela que nous retrouvons le choix du positionnement au niveau de la commande de l'aiguillage (figure II.34).

De cette manière le choix de traduire les indéterminismes de nos modèles par le module de communication de la figure II.30b les retransmet à un niveau de la commande où le **procédé** impose un choix, comme nous l'avons montré pour l'aiguillage. Mais cette traduction des indéterminismes n'appréhende aucune politique de prise de décision. Si la destination d'une palette à été déterminée au préalable, par exemple par une allocation prévisionnelle, il suffit de lire cette information au niveau de la commande de l'aiguillage. L'information est, dans ce cas, codée dans le jeton modélisant la palette et propagée par l'intermédiaire des places "req" à ce

niveau du modèle. Si, par contre, la destination d'une palette n'est pas encore connue, elle peut être déterminée au niveau de la commande de l'aiguillage en utilisant les données associées au jeton. Cette solution d'allocation sur requête correspond à une prise de décision décentralisée en temps réel avec une vue locale du procédé.



**Figure II.34 : modèle d'un aiguillage**

La traduction des indéterminismes par notre module (figure II.30b) contribue aussi à une commande adaptative. Supposons, que l'on veuille transférer une palette arrêtée en P5 vers P2 (req P5 -> P2). Selon le graphe de synchronisation, l'aiguillage est positionné avant d'ouvrir la butée. Le plus souvent, les convoyeurs comportent des capteurs qui détectent uniquement le passage des palettes. A partir de cette information, le transfert réellement effectué est déduit. Si l'aiguillage reste dans la position P5 -> P3, suite à un dysfonctionnement, nous détectons donc par l'intermédiaire d'un capteur de passage, que le transfert P5 -> P3 a été réalisé au lieu du transfert demandé (P5 -> P2). Ce signal se traduit par un franchissement de la transition P5 -> P3 au niveau du graphe de synchronisation qui mène au marquage de la place "ack P5 -> P3 ". Au niveau du modèle du convoyeur la transition t3 est franchie et le jeton modélisant la palette marque la place P3.

Remarque : La place "nack" du modèle du convoyeur (figure II.33b) est marquée, si l'ouverture de la butée a échoué. Cette information est générée à partir de l'interface du procédé et transmise vers les graphes de synchronisation. La communication correspondante sera mise en évidence, si on remplace la transition "butée.ouvrir" par son module de communication.

#### **5.4. LA COMMUNICATION ENTRE GAMMES OPÉRATOIRES ET MODÈLE DE TRANSPORT ET GAMMES OPÉRATOIRES ET MODÈLES DES RESSOURCES**

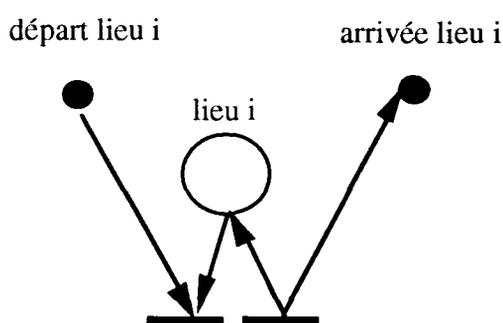
Les transitions des gammes opératoires modélisent soit un transport soit une transformation. Les transitions de transfert s'adressent au modèle du système de transport, les transitions de transformation aux modèles des ressources.

Les transitions de transformation sont remplacées par les modules de communication

standard (figure II.30) et connectées aux transitions des modèles des ressources. La communication entre les deux modèles est identique à celle établie entre les modèles des ressources et les graphes de synchronisation.

Les transitions de transport sont remplacées par le module de communication sans la place "nack". En considérant qu'un produit puisse rester un temps indéterminé dans le système, il peut rester sur un stockeur ou tourner en rond sur le moyen de transport central, il est impossible de déterminer, si un transfert a définitivement échoué.

Sur le modèle du système de transport, toutes les places modélisant un lieu opératoire sont remplacées par le module de la figure II.35, qui modélise l'arrivée d'une pièce sur ce lieu et qui communique son arrivée au niveau des gammes opératoires.

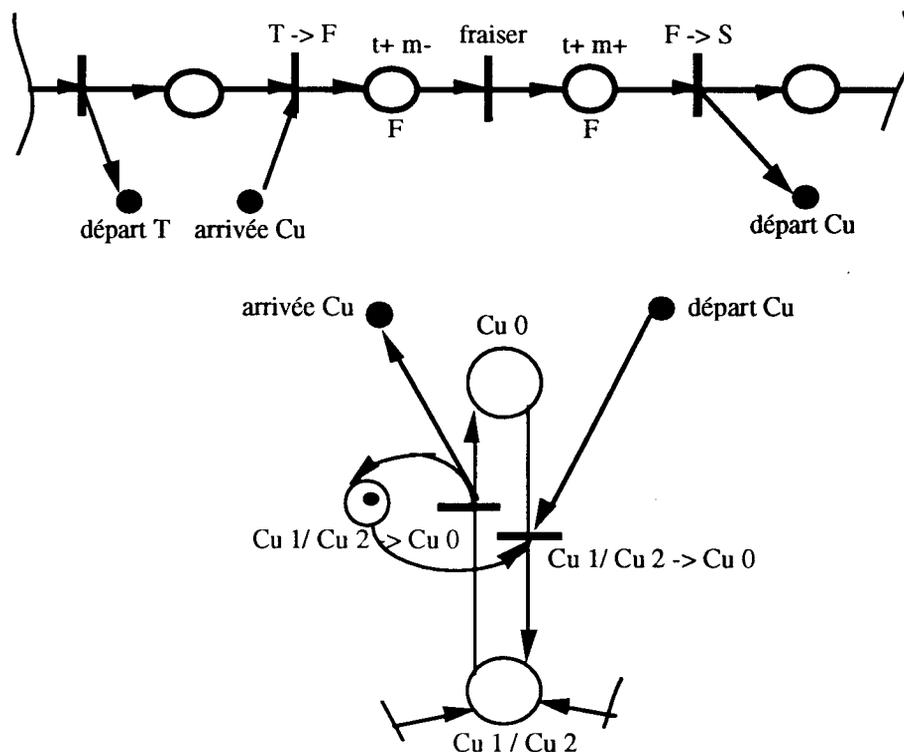


**Figure II.35 : module de connexion**

Sur la figure II.36 on précise l'insertion du module au niveau du centre d'usinage de notre exemple et sa connexion au niveau de la gamme du produit 1.

Le franchissement de la transition  $Cu1/Cu2 \rightarrow Cu0$  mène au marquage simultané de la place  $Cu0$  au niveau du modèle du système de transport, et à la place "arrivée Cu". Le produit est donc chargé sur le lieu opératoire du centre d'usinage.

Au niveau de sa gamme opératoire, le jeton correspondant marque la place  $F$ ,  $t+f-$ , qui modélise l'état du produit avant l'usinage. La transformation effectuée (marquage de la place  $F$ ,  $t+f+$ ) le jeton franchit la transition  $F \rightarrow S$  et la place "départ Cu" est marquée. La transition  $Cu0 \rightarrow Cu1/Cu2$  du modèle de système de transport est sensibilisée et le transfert du produit vers le tampon de sortie peut être commandé à partir de ce modèle.



**Figure II.36 : communication gammes opératoires / modèle du système de transport**

Les places “arrivée lieu  $i$ ” et “départ lieu  $i$ ” sont partagées par les différentes gammes opératoires. La place “arrivée Cu 0” et sa place de départ correspondante sont donc connectées également à la gamme du produit P22.

## 5.5. CONCLUSION

Les modules présentés définissent les modalités de communication entre les différents types de modèles. Les modèles supérieurs sont synchronisés sur l'exécution de la tâche qu'ils ont déléguée aux modèles inférieurs.

Les modèles sont connectés par ces places de communication. Ces places peuvent servir d'indication pour découpler les modèles dans le cas d'une implantation répartie sur calculateurs. Cependant, ces connexions ne préjugent d'aucune solution ni pour l'implantation ni pour la répartition des modèles sur calculateur. Cette approche de la communication laisse un degré de liberté maximal aux concepteurs du code implantable.

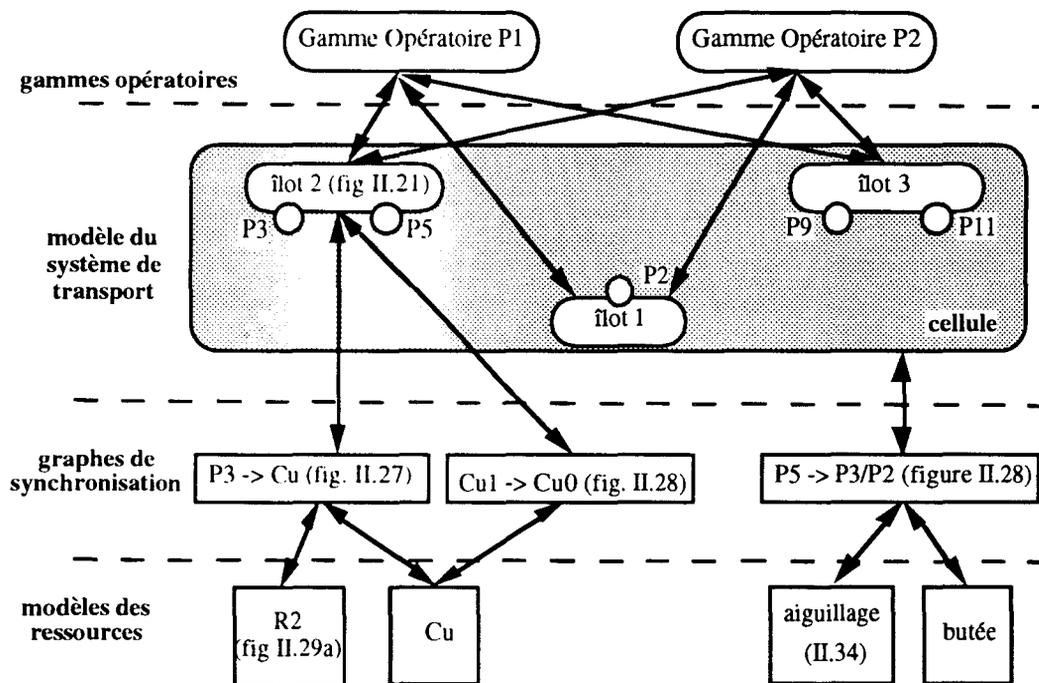
La séparation stricte des aspects transport et synchronisation, des aspects produits et

ressources, se montre également avantageuse pour l'aspect communication entre les modèles. La connexion est systématique. L'utilisation des modules spécifiés, permet d'aboutir à une commande sûre tout en permettant d'atteindre des points de reprise par rapport aux ressources et aux produits.

Le choix de traduire des indéterminismes par la structure de la figure II.30b est dû au souci de ne pas vouloir imposer a priori un type de résolution des indéterminismes et des conflits, ce qui relève de l'exploitation du système. Une résolution en temps réel à la demande, par règles statiques ou dynamiques ou même une exploitation en ordonnancement / pilotage sont compatibles avec ce modèle.

Nous avons présenté deux algorithmes pour la génération des gammes opératoires et le modèle du système de transport. La génération des graphes de synchronisation à partir des données de l'analyse ne pose pas de problème majeur. Pour les modèles des ressources nous avons défini les caractéristiques d'une bibliothèque de modèles.

La connexion des différents modèles de la commande termine ainsi la phase de modélisation de la commande de coordination. La connexion des modèles est systématique. De ce fait une automatisation du processus de conception de la commande de coordination est possible.



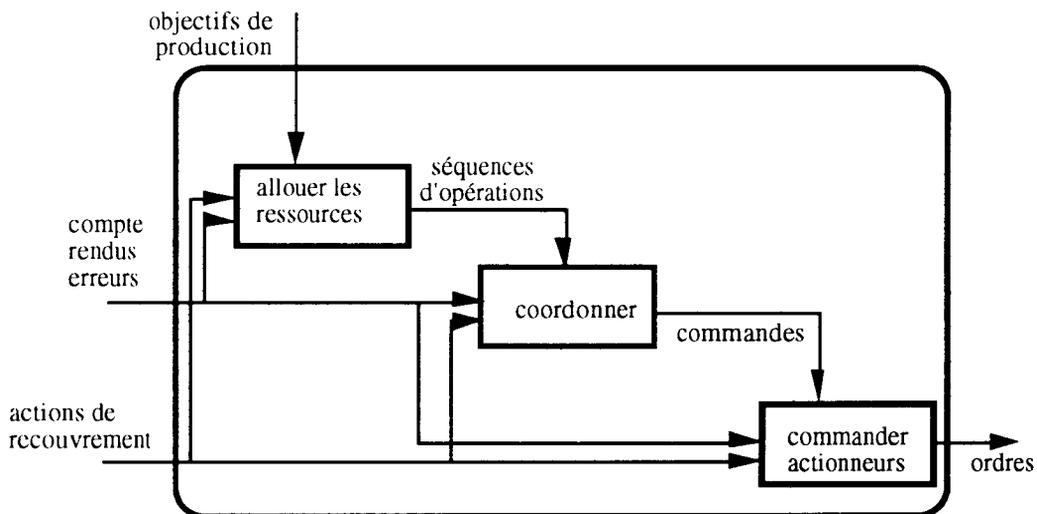
**Figure II.37 : schéma de la commande de coordination de l'exemple**

La figure II.37 présente le schéma global de la commande de coordination pour l'exemple de

la cellule de l'École Centrale de Lille. Les modèles développés dans ce chapitre sont référencés par leurs figures. Les flèches représentent les communications entre ces modèles, développées dans le dernier paragraphe.

## 6. LA COORDINATION AU SEIN D'UN SYSTÈME DE CONDUITE

La figure II.38 reprend le modèle de type SADT développé dans le premier chapitre de ce mémoire, qui décrit la commande de coordination comme sous-activité de l'activité "commander". La commande de coordination reçoit de la détection des comptes rendus ou des erreurs d'exécution. Des actions de recouvrement sont mises en œuvre à travers ses modèles. Elle évolue sous les contraintes de l'allocation des ressources qui rend ses modèles déterministes et elle émet des commandes élémentaires vers la commande des actionneurs.



**Figure II.38 : la coordination au sein de la commande**

Dans ce paragraphe nous établissons les propriétés d'adaptabilité et de sûreté de la commande de coordination présentée. Ces besoins concernent l'intégration de cette activité par rapport aux fonctions de la surveillance. Dans un deuxième temps, nous montrons la connexion de la coordination à l'interface du procédé, tel qu'il a été présenté pour le projet CASPAIM. Finalement, nous mettons en évidence l'intérêt d'utiliser les modèles de la coordination comme système d'information pour l'allocation des ressources et le recouvrement des erreurs.

## 6.1. UNE COORDINATION SÛRE ET ADAPTATIVE

Une commande **sûre** est synchronisée sur l'exécution des ordres par les actionneurs. Elle évite de ce fait l'évolution non cohérente du système de contrôle / commande en cas d'occurrence d'une erreur sur le procédé. Les modules de communication, introduits pour la connexion des modèles de la coordination, traduisent ce besoin par un mécanisme de communication fortement synchronisé de type requête - accusé de réception. L'utilisation systématique d'un tel protocole de communication dans les différents modèles conserve cette propriété à tous les niveaux d'abstraction.

La caractéristique d'**adaptabilité** permet la mise en œuvre d'actions de recouvrement et comporte trois aspects :

- \* une représentation fidèle de la flexibilité du procédé, qui intègre de ce fait la possibilité d'une coordination en mode dégradé,
- \* la modélisation des commandes réversibles permettant d'atteindre les points de reprise par rapport aux produits et par rapport aux ressources,
- \* l'intégration d'un mécanisme de verrouillage de la coordination.

La **flexibilité** potentielle du procédé est intégrée à tous les niveaux des modèles, comme nous avons pu le montrer auparavant. La **réversibilité** est une propriété des opérations positionnelles lors des transferts et des échanges de produits entre ressources de production. Les modèles des ressources ainsi que les graphes de synchronisation représentent cet aspect.

Le troisième élément d'une commande adaptative concerne la définition d'un mécanisme de **verrouillage**. Nous pouvons associer à chaque transition de notre modèle une condition de tir, qui inhibe son franchissement. La valeur de la condition est déterminée par des événements externes à la commande, c'est-à-dire externes au comportement des modèles réseaux de Petri.

En effet, le changement de l'état d'un mode opératoire détermine la valeur de cette condition. Ainsi, en mode normal du système, les transitions modélisant les retours en arrière au niveau des graphes de synchronisation et des modèles des ressources seront verrouillées. Si par contre une panne met une partie du système en mode dégradé ou défaillant, les transitions concernées seront déverrouillées et permettront par la suite l'exploitation de la réversibilité des opérations positionnelles. Les indéterminismes introduits par la modélisation des retours en arrière dans les modèles de la coordination des ressources sont donc soulevés par la gestion des modes

opérateurs.

Si, par exemple, le chargement automatique du tour (figure I.6) échoue à cause d'une panne sur le tour, le tour se trouve en mode dégradé. Le mécanisme de verrouillage interdit d'abord le franchissement de la transition "serrer mors" au niveau de la commande du tour (figure II.29b) et du graphe de synchronisation (figure II.27) et déverrouille les transitions modélisant le retour en arrière vers P2.

Le changement de mode du tour est répercuté au niveau de la gestion des modes opératoires (cf. chapitre I). Cette propagation peut causer le verrouillage des transitions du modèle de transport indiquant les transferts élémentaires vers le tour. La propagation de l'information au niveau de la gestion des modes opératoires détermine ainsi l'état de chaque transition du modèle.

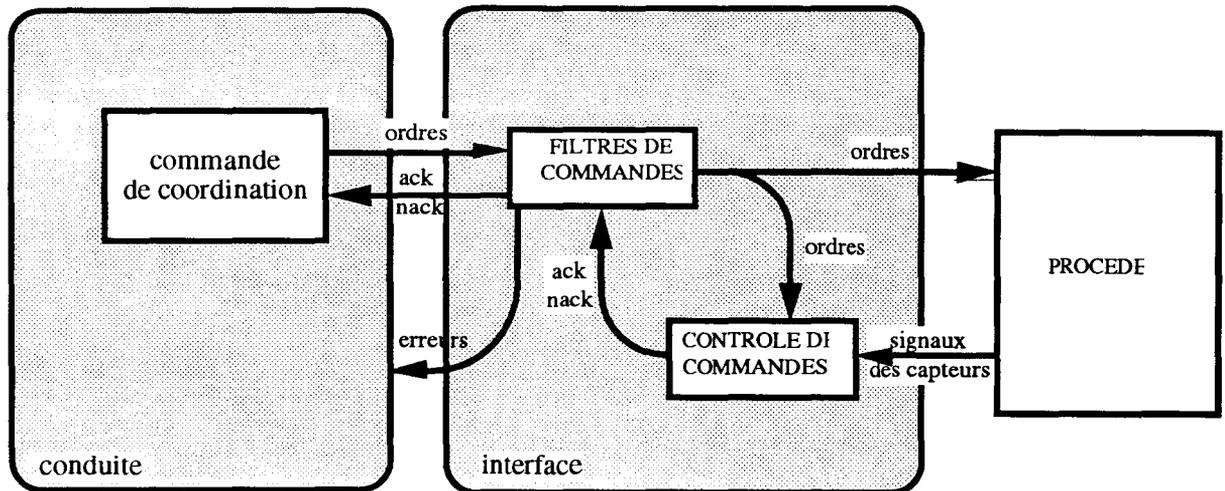
L'intégration du mécanisme de verrouillage définit ainsi les données échangées entre la gestion des modes opératoires et la commande de coordination et définit le dernier élément de modélisation permettant d'obtenir une commande de coordination adaptative.

## **6.2. LA CONNEXION DE LA COORDINATION À L'INTERFACE DU PROCÉDÉ**

Les commandes émises par la commande de coordination sont transmises à la commande des actionneurs (figure II.38). Leur réalisation par les actionneurs du procédé est établie par une fonction de type "détection", proche du procédé. Ces deux activités forment ainsi l'interface du procédé.

Dans le cadre du projet CASPAIM, les travaux de S. Elkhatabi [Elkhatabi 1993] concernant la surveillance de bas niveau proposent la réalisation de cette interface par une approche langage synchrone (ESTEREL [Berry et al 1987]), qui respecte le caractère réactif et déterministe de ces deux fonctionnalités.

Suivant cette approche, l'interface du procédé est composée de deux types de module, comme le montre la figure II.39. A chaque ressource est associé un filtre de commande, qui présente une interface de communication. Le filtre permet, d'abord la détection des ordres erronés, c'est-à-dire des ordres incompatibles avec l'état courant de la ressource. Un ordre "validé" par le filtre est ensuite envoyé à la fois à l'actionneur, auquel l'exécution de l'ordre est confiée, et à son module de contrôle de commande associé.



**Figure II.39 : communication coordination / interface du procédé**

Le contrôle commande surveille l'exécution de l'ordre par l'actionneur. Il dispose d'un modèle de référence, qui lui permet, en fonction des signaux des capteurs, de juger de la réalisation de l'action. Si le contrôle reçoit les signaux correspondant à l'exécution de l'ordre, il transmet cette information sous forme d'un accusé de réception au niveau du filtre, le cas échéant le filtre de commande reçoit un "nack" signalant la non exécution de l'ordre.

Le comportement de la ressource au niveau des filtres de commande et le comportement des actionneurs au niveau des contrôles de commande est représenté par un modèle de type automate à états finis. Le modèle du filtre de commande représente la flexibilité des actions élémentaires au niveau des ressources en respectant les contraintes de sécurité et de bon fonctionnement. Il indique ainsi toutes les actions réalisables à partir d'un état donné et les états atteints suite à son exécution.

Le graphe de commande de la ressource reçoit les signaux émis par le filtre de commande. En utilisant également les modules de connexion, proposés pour la communication des modèles de la commande (figure II.30), pour la communication entre la commande de coordination et l'interface du procédé, nous synchronisons l'évolution de la coordination sur les signaux transmis par l'interface.

Cette connexion conserve les propriétés d'une commande sûre (synchronisation) et adaptative (traitement de la réponse "nack"). La flexibilité du procédé est ainsi exploitable à partir de la commande de coordination, même au niveau des ressources.

### **6.3. LES MODÈLES DE LA COORDINATION COMME MODÈLES DU NIVEAU HIÉRARCHIQUE**

Les modèles de la coordination des ressources, les gammes opératoires et le modèle du système de transport, mettent en évidence, comme nous l'avons vu, les décisions nécessaires lors de l'exploitation du système. De ce fait, ils peuvent aussi servir de base de décision pour l'allocation des ressources et le recouvrement d'erreurs.

Utilisé comme modèle pour l'ordonnancement, les gammes opératoires et le modèle du système de transport dissocient l'aspect transformation de l'aspect transport. Les gammes opératoires intègrent les contraintes de non-chevauchement des opérations. Afin d'intégrer les références temporelles, nous modélisons la durée de chaque opération par des temporisations des transitions au niveau des gammes opératoires [Carlier et Chrétienne 1988]. La date de lancement et la date due d'un produit peuvent être référencés au niveau des jetons individualisés. Les gammes opératoires sont de ce fait un modèle des ordonnancements admissibles [Baptiste et Chappe 1992] pour une production donnée.

Cette représentation du problème d'ordonnancement est parfaitement adaptée aux SFPM, car elle indique les contraintes d'ordre des opérations sur chaque produit, exprime toute la flexibilité du procédé par rapport aux opérations transformationnelles et tient compte de la production en petite ou moyenne série, représenté par un marquage multiple d'une même gamme.

Les approches de hiérarchisation de l'ordonnancement, comme les approches de décomposition spatiale du problème d'ordonnancement se représentent facilement sur les gammes hiérarchisées et le regroupement de différentes opérations dans des opérations agrégées a été introduit.

Le modèle du système de transport permet à chaque instant la localisation des produits et des supports de transport. Il représente tous les chemins possibles pour atteindre un lieu de l'atelier et permet de prédire l'évolution du système. Il indique par le marquage les encombrements des différents chemins de transfert. Ce modèle peut donc servir comme base de décision pour une gestion prévisionnelle et sophistiquée des transports dans le cas d'une stratégie de pilotage centralisé des moyens de transports [Hébrard 1992]. Mais comme le modèle représente aussi tous les problèmes d'accès aux ressources, il peut servir comme modèle de prise de décision dans une stratégie d'allocation sur requête en utilisant les données relatives aux produits qui sont accessibles par l'intermédiaire des jetons.

Les modèles de la coordination représentent à chaque instant les solutions envisageables pour un produit donné. Ces modèles peuvent donc aussi servir de modèle d'information pour le recouvrement. La réalisation d'un ordre par un actionneur ayant échoué, le mécanisme de communication transmet cette information au niveau des graphes de synchronisation. A partir de ce point de reprise, le recouvrement peut définir une séquence de commandes qui correspond à un retour ou à une relance de l'ordre.

Le modèle du système de transport représente toutes les solutions envisageables par rapport au recouvrement des produits. Il correspond à un synoptique modélisant de manière détaillée l'évolution des produits dans le système. A partir de ce modèle, le recouvrement peut commander le routage de chaque produit individuellement, les commandes étant prévues par la modélisation fidèle de la flexibilité du système.

L'intérêt d'utiliser les mêmes modèles pour ces deux activités de caractère décisionnel est de simplifier la communication. Si un ordonnancement s'exprime sous forme réseaux de Petri en définissant par exemple la séquence de tir des transitions ainsi que leur date au niveau des gammes opératoires, l'échange de ces données entre les différentes activités devient assez simple.

## **7. CONCLUSION**

Dans ce chapitre, nous avons montré une méthode de conception de la commande de coordination pour les SFPM. Cette méthode est structurée, modulaire et hiérarchique. Elle intègre les différents aspects de la coordination, coordination par rapport aux produits et coordination des ressources dans des sous-modèles dédiés à chaque tâche.

La construction modulaire du modèle de la commande de coordination est basée sur l'encapsulation de l'information et l'utilisation d'un formalisme unique. Elle contribue à la qualité du logiciel de commande à concevoir. De ce fait, le modèle bénéficie des avantages d'une approche de type génie logiciel. Les modifications du modèle, suite à des changements du procédé, sont faciles à identifier et ont seulement des conséquences locales.

Nous avons pu montrer l'importance d'une analyse détaillée du procédé et des connaissances spécifiques du comportement des ressources de production, afin d'intégrer toutes les

possibilités de l'installation dans le modèle de la conduite et de permettre son exploitation ultérieure.

Nous avons développé une démarche de structuration d'un SFPM en îlots et cellules, qui est basée sur un critère de cohésion des ressources. Cette structure hiérarchique a été exploitée pour la construction modulaire et hiérarchique du modèle du système de transport.

La coordination par rapport aux produits est confiée aux gammes opératoires et au modèle du système de transport. Les gammes opératoires modélisent le processus de fabrication par une approche produit et définissent une utilisation spécifique de l'atelier. Le modèle du système de transport indique toutes les possibilités du système par rapport à la manutention des produits.

Les modèles de la coordination des ressources font abstraction du produit et représentent le comportement des organes opérants des ressources de production lors d'un échange de produit.

La dissociation des différents aspects de la coordination et leur prise en compte dans des modèles spécifiques se montre avantageuse à tout niveau. L'adaptabilité de la commande est, en outre, assurée par la modélisation fidèle de la flexibilité du système. Nous avons représenté les différents types de flexibilité dans les modèles adéquats.

La connexion des modèles par des interfaces de communication modélisant la synchronisation de leur évolution respecte le besoin d'une commande sûre et permet d'atteindre les différents points de reprise.

La commande de coordination s'intègre ainsi facilement dans un système de contrôle / commande comme nous l'avons montré à la fin du chapitre. Cette simplicité de l'intégration est due à l'analyse détaillée des inter-relations et interconnexions des différentes activités et le respect du cahier des charges issu de cette analyse.

# **CHAPITRE III :**

## **LA VALIDATION DES MODÈLES**

### **DE LA COMMANDE DE COORDINATION**

#### **1. INTRODUCTION**

Nous avons présenté, dans le chapitre précédent, un ensemble de modèles réseaux de Petri qui modélisent la commande de coordination et le comportement dynamique du système de production. Afin d'assurer leur qualité, nous proposons dans ce chapitre une analyse des propriétés des modèles en vue de leur validation qualitative.

Cette validation permet d'établir les "bonnes propriétés" des modèles conçus avec la méthode proposée. Ces propriétés sont la vivacité, la finitude et la réinitialisabilité [Valette 1986], dont les définitions sont rappelées en annexe 7. Parmi ces propriétés, nous accordons une plus grande importance à la vivacité, d'une part, en raison du fait que la vivacité assure l'absence de tout blocage sur le système, d'autre part, parce que la finitude des modèles proposés est généralement plus facile à exprimer. En effet, ces modèles sont assez souvent construits sur la base des graphes d'état.

La vivacité, comme la finitude et la réinitialisabilité, appartient à la classe des propriétés comportementales d'un réseau de Petri et dépend, de ce fait, non seulement de la structure du graphe mais aussi du marquage initial. Un des buts de l'analyse des propriétés est donc la définition d'un marquage de non-saturation des différents modèles, qui garantit leur vivacité à charge maximale. Ceci conduit à préciser les conditions extrêmes de blocage à ne pas atteindre.

Nous proposons une démarche ascendante de l'analyse formelle des modèles. Cette

démarche utilise la construction modulaire et hiérarchisée du modèle de coordination qui permet par découpage une analyse individuelle de chacun des modèles.

Rappelons, que la hiérarchie entre les différents types de modèles est basée sur l'encapsulation de l'information. Une transition, à un niveau donné, correspond à l'exécution d'une tâche, modélisée par un réseau du niveau inférieur. Si le réseau du niveau inférieur est vivant, nous en déduisons qu'il ne peut jamais être un obstacle au franchissement de la transition représentant la tâche par abstraction au niveau supérieur.

## 2. L'ANALYSE DES MODÈLES DES RESSOURCES SIMPLES

Les modèles des ressources représentent le comportement des ressources par un réseau de Petri ordinaire. La figure III.1 illustre deux exemples issus de notre bibliothèque de modèles. Les graphes modélisent un manipulateur (a) et une machine à chargement interne (b). Ils représentent l'état atteint de chaque ressource suite à l'exécution d'un ordre par ses actionneurs.

De façon générale, les modèles des ressources simples sont des graphes d'état pour garantir la flexibilité maximale. Les bonnes propriétés des réseaux de Petri, appartenant à cette classe, sont faciles à prouver à l'aide du théorème suivant :

**Théorème** [par exemple Murata 1989] : Un graphe d'état  $(N, M_0)$  est vivant, si  $N$  est fortement connexe et  $M_0$  contient au moins un jeton. Un graphe d'état est sauf (1-borné) si  $M_0$  contient exactement un jeton.

Les graphes d'état des modèles des ressources sont fortement connexes et leur marquage initial contient un jeton, ils sont donc saufs et vivants.

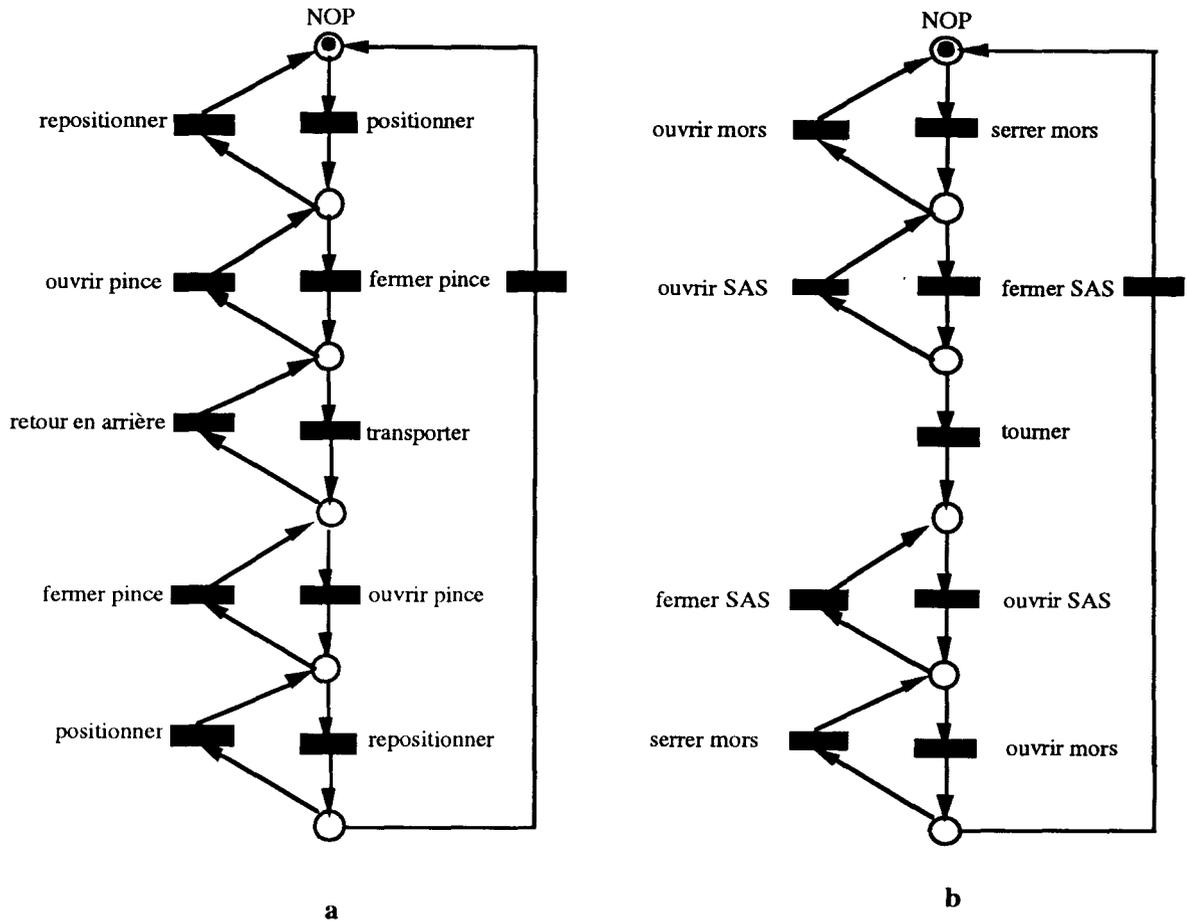


Figure III.1 : modèles des ressources

### 3. L'ANALYSE DES GRAPHES DE SYNCHRONISATION

La figure III.2 représente le graphe de synchronisation, qui modélise le chargement du centre d'usinage à partir du tampon P3 de la cellule. Chaque graphe de synchronisation est, à la base, un processus linéaire décrivant les étapes d'un chargement ou d'un déchargement. L'intégration des opérations réversibles nous a conduit à rajouter des indéterminismes.

Les graphes de synchronisation font abstraction du produit transféré, il s'agit donc de réseaux de Petri ordinaires. Le jeton unique modélise l'activité de la tâche de synchronisation.

Les graphes de synchronisation sont par construction des graphes d'état fortement connexes et le marquage initial contient un jeton. Ils sont donc également vivants et saufs.

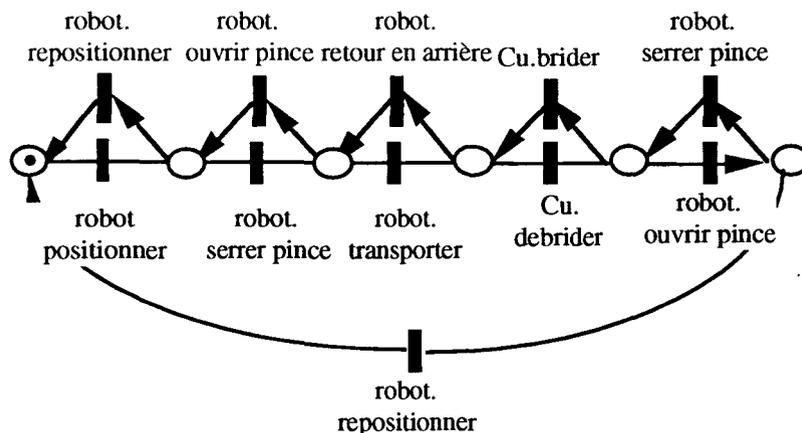


Figure III.2 : graphe de synchronisation

## 4. L'ANALYSE DU MODÈLE DU SYSTÈME DE TRANSPORT

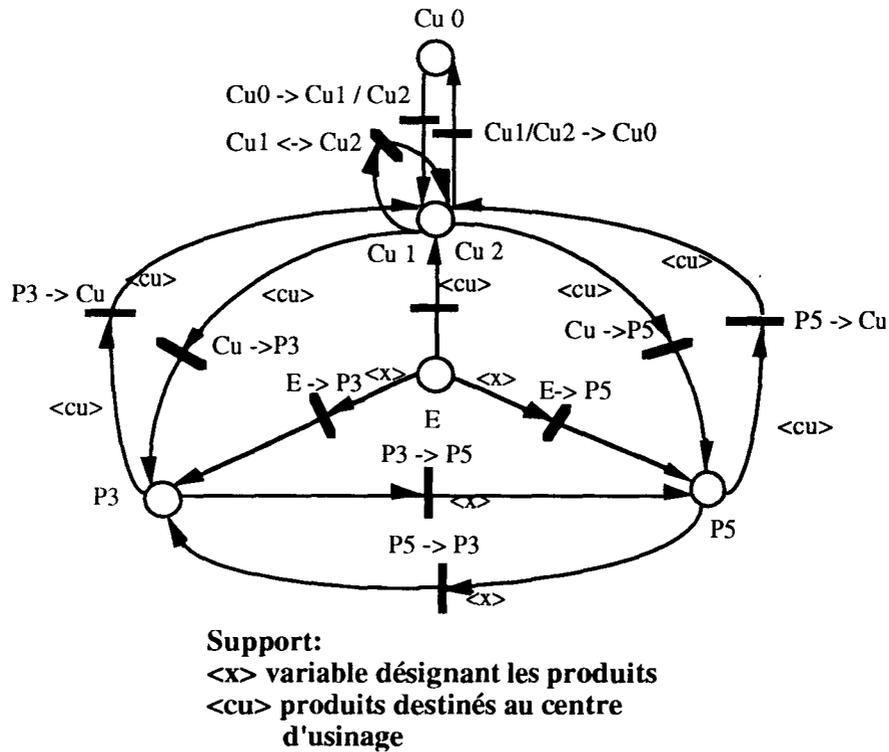
Le modèle du système de transport est le premier élément des modèles de la coordination par rapport aux produits. Le modèle est un réseau de Petri à Prédicat /Transition où les jetons modélisent les produits à l'intérieur des flots et les supports de transport dans le modèle de la cellule.

Le modèle du système de transport est lui-même un modèle hiérarchisé. Pour établir ses propriétés, nous analysons d'abord les modèles des flots et ensuite le modèle de la cellule, qui correspond au modèle du moyen de transport central.

### 4.1. L'ANALYSE DES MODÈLES DES ÎLOTS

#### 4.1.1 La Démarche de l'Analyse

La figure III.3 reprend le résultat de la première étape de la génération du modèle de l'îlot 2 de la cellule de l'École Centrale de Lille. Nous remarquons, que ce réseau de Petri est un graphe d'état.



**Figure III.3 : modèle intermédiaire de l'îlot 2**

Rappelons, que cette première étape de génération consiste en une traduction du graphe des accessibilités du système qui substitue les places aux nœuds du graphe et les transitions aux arcs. Cette traduction conduit donc à des graphes d'état, où chaque place modélise la localisation possible d'un produit.

Afin d'intégrer la capacité limitée des ressources de production, nous avons rajouté, dans la deuxième étape de génération, des places de type protocole d'accès. Le marquage initial de chacune de ces places détermine une condition d'accès à la ressource modélisée. Le marquage d'une place modélisant un lieu physique de l'atelier est donc borné par le marquage initial d'une place "protocole".

Les transitions du modèle d'un îlot représentent des transferts élémentaires. Un transfert élémentaire correspond à l'échange d'un produit entre deux lieux caractéristiques d'un îlot. Une transition n'est plus tirable, si un produit, dont le jeton marque la place amont à la transition, ne peut plus transiter vers le lieu modélisé par la place aval à la transition. Cela se traduit par le fait que la place de type protocole associée à ce lieu est vide. Un blocage local peut alors se produire si, à partir d'une localisation donnée, un produit ne peut être transporté vers aucun lieu physique dans l'îlot. En d'autres termes, un blocage du modèle correspond à la saturation de

chaque lieu physique.

Reprenons le modèle de la figure III.3. Si nous éliminons la place E et ses transitions en aval, nous obtenons un graphe d'état fortement connexe. Ce graphe d'état est structurellement vivant, c'est-à-dire qu'il existe un marquage pour lequel il devient vivant [Murata 1989].

La place E, modélisant le tampon d'entrée de la cellule, n'est pas accessible. Si elle est marquée, le modèle de la figure III.3 est quasi vivant, les transitions  $E \rightarrow P3$ ,  $E \rightarrow P5$  ou  $E \rightarrow Cu$  peuvent être tirées autant de fois que E contient de marques.

Les modèles intermédiaires des îlots possèdent donc les bonnes propriétés, seul l'ajout des protocoles peut mener à des blocages locaux ou globaux. Ces blocages ne peuvent être obtenus que par une saturation de toutes les ressources modélisées par des places. Nous proposons donc une analyse des modèles, dans le but de définir leur marquage maximal de non-saturation, qui garantit leur vivacité.

Nous recherchons pour cette analyse une représentation plus concise du comportement dynamique de chaque module, qui conserve les informations essentielles vis-à-vis des propriétés analysées. Cette **réduction** des modèles sera obtenue par des transformations d'équivalence définies pour les réseaux de Petri à Prédicat / Transition (annexe 8). Ces transformations regrouperont les lieux d'un même îlot, qui sont caractérisés par un comportement identique vis-à-vis de l'échange des produits, en une seule place.

Le modèle réduit sera ensuite analysé dans une situation extrême. Cette situation correspond à un marquage du modèle où un blocage est très probable. Nous analyserons donc son comportement dans un mode opératoire du système qui ne comporte que des produits admis dans l'îlot.

#### 4.1.2. La Réduction du Modèle

Un système modélisé par un réseau de Petri à Prédicat / Transition peut être représenté à plusieurs niveaux d'abstraction. Places et transitions peuvent être fusionnées, ou séparées afin de mettre en évidence un aspect plus abstrait ou plus détaillé du système modélisé. Ces transformations d'équivalence conservent les propriétés du modèle d'origine au niveau du modèle transformé, car les deux modèles sont sémantiquement équivalents.

Nous proposons une réduction des modèles des îlots à partir des places modélisant des lieux

physiques. Nous regrouperons d'abord les places de significations comparables, ce qui donne une interprétation plus abstraite de la localisation d'un produit. Dans un deuxième temps, nous unifions les places de type "protocoles" qui sont associées aux places fusionnées précédemment. La troisième étape consistera en l'élimination des transitions en parallèle.

Nous présentons par la suite les différentes étapes de regroupement à l'aide de l'exemple de l'îlot 2 de la cellule de l'École Centrale de Lille. Leur application sera étendue, dans un deuxième temps, à deux autres exemples ce qui permettra de formuler des résultats plus généraux à la fin de ce paragraphe.

#### 4.1.2.1 La Réduction du Modèle de l'îlot 2

La figure III.4 reprend le modèle final de l'îlot 2 obtenu par notre méthode de modélisation présenté dans le paragraphe II.4.3.3.2 (figure II. 21).

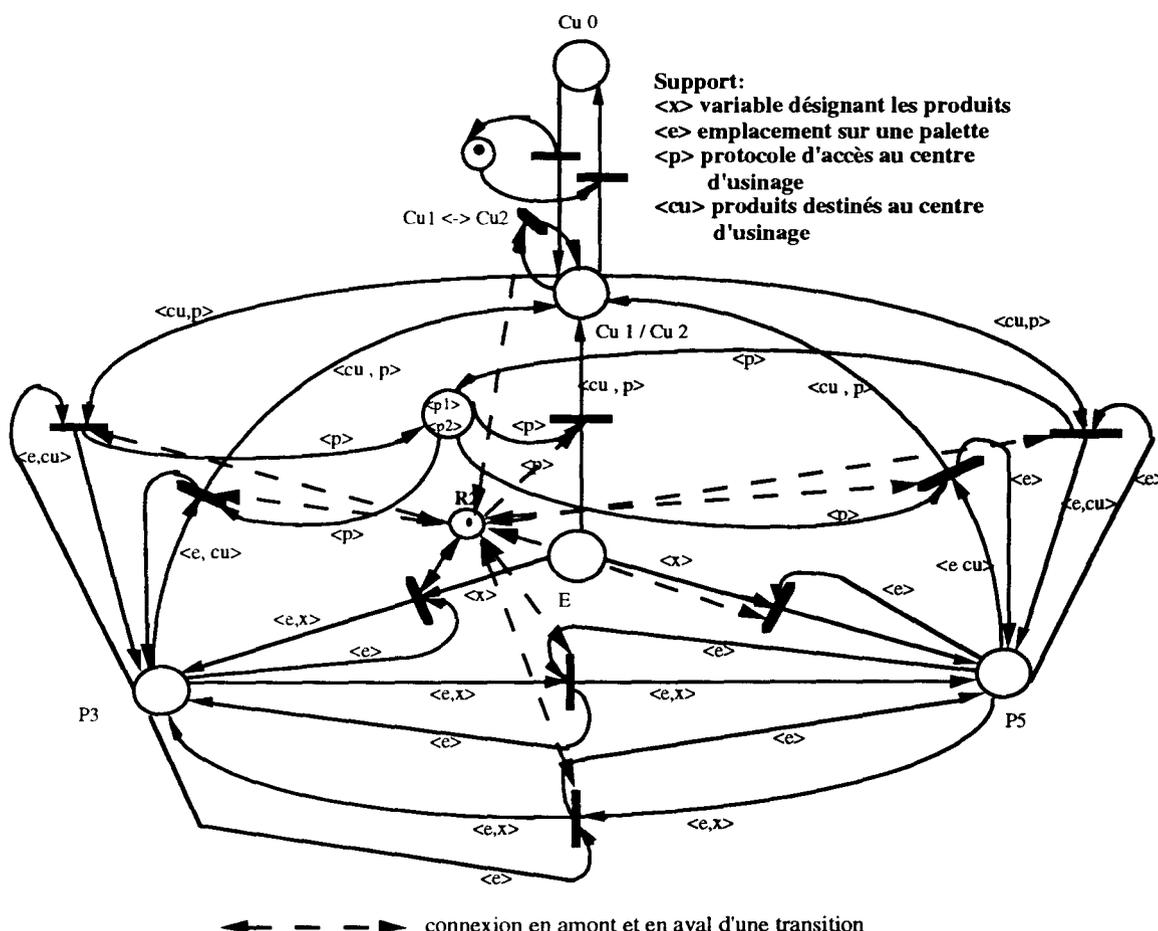


Figure III.4 : modèle final de l'îlot 2

### Étape 1: Elimination des Places Implicites

Dans la dernière étape de génération des modèles des îlots (chapitre II, 4.3.3.2), nous avons introduit des places de type “mutex”. Ces places modélisent l’exclusion mutuelle d’accès simultané à des lieux de synchronisation appartenant à des ressources partagées entre plusieurs îlots.

Une place de type “mutex” est mono-marquée et elle est connectée en amont et en aval de chaque transition modélisant un transfert qui a comme point de départ ou comme point d’arrivée le lieu en accès mutuellement exclusif.

Ce type de connexion est identique à celui choisi pour la place modélisant le ou les robot(s) de l’îlot (cf. connexion de la place “R2” au niveau de la figure III.4).

Les places de type “mutex” et les places modélisant les robots au niveau des îlots sont des places implicites. Le marquage de ces places n’est jamais un obstacle au franchissement de leurs transitions de sortie, car elles sont toujours marquées. Nous pouvons donc supprimer ces places et leurs arcs correspondants (figure III.5). Cette réduction conserve la vivacité du réseau initial [David et Alla 1989].

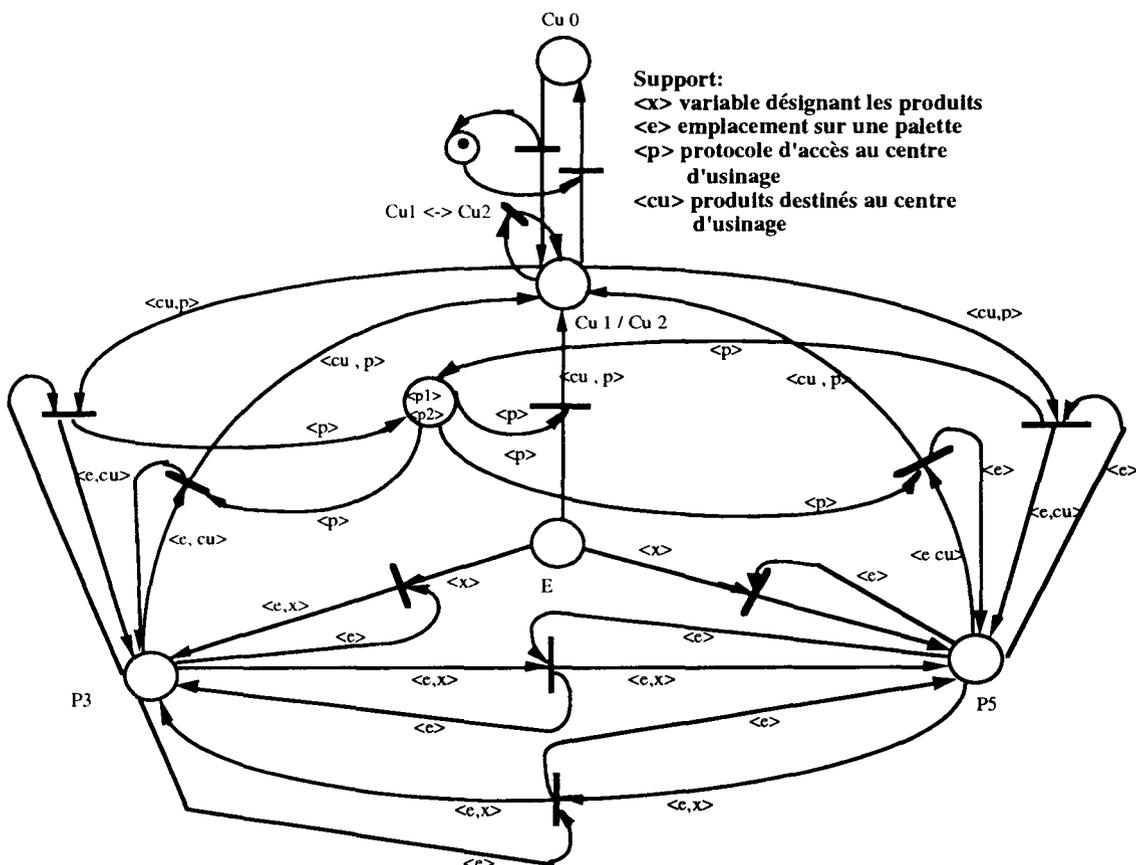


Figure III.5 : réduction du modèle de l’îlot 2 : étape 1

**Etape 2 : Regroupement des Places de Signification Comparable**

La deuxième étape de réduction consiste en la fusion des places modélisant des lieux physiques qui ont un comportement identique vis à vis du chargement et du déchargement des produits.

Nous proposons donc de regrouper :

- \* les lieux de synchronisation formant les **tampons** de l'îlot dans une place, référencée par la suite comme place de type 1,
- \* les lieux de synchronisation des ressources de type **stockage** et les lieux de synchronisation des ressources de transformation avec **chargement interne** dans une deuxième place (place de type 2),
- \* les lieux de **synchronisation** des ressources de transformation avec **chargement externe** dans une place (place de type 3),
- \* ainsi que leurs lieux de **transformation** dans une place (place de type 4).

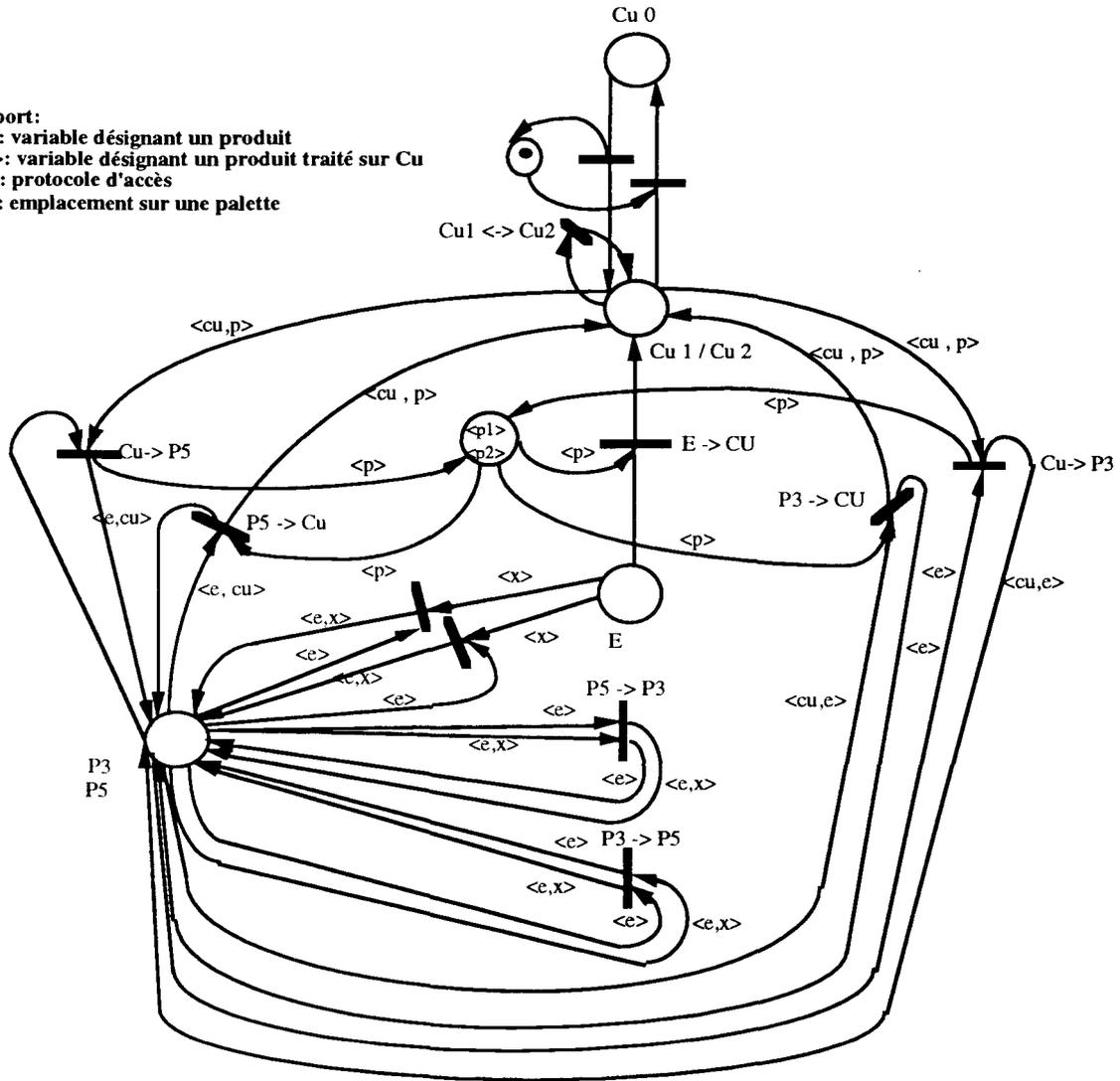
Par rapport au modèle de l'îlot 2, nous établissons les correspondances suivant :

- \* les tampons de l'îlot (places de type 1) : P3 et P5,
- \* lieux de synchronisation des ressources de type stockage (type 2) : E,
- \* lieux de synchronisation des ressources de transformation avec chargement externe (type 3) : Cu1/Cu2,
- \* lieux de transformation des ressources avec chargement externe (type 4) : Cu 0.

Le modèle de la figure III.5 comporte deux instances de places du type 1, les deux tampons de l'îlot. Les autres types de place ne sont représentés que par une seule occurrence. La deuxième étape du regroupement se résume donc dans la fusion des deux places P3 et P5 dans une seule, nommée P3 v P5 (figure III.6).

Ce regroupement est en effet une transformation d'équivalence qui conserve les propriétés du modèle de départ. Les deux places indiquent des accessibilités identiques, ce qui s'exprime sur le modèle de la figure III.6 par la double existence de chaque transition. Les places unifiées sont donc des copies l'une de l'autre, comme le requiert la règle d'unification des nœuds. Leur fusion n'influence pas le comportement du modèle, il en résulte un modèle équivalent qui fait abstraction de la localisation actuelle d'un produit : un produit qui marque la place P3 v P5 peut être sur P3 ou P5.

**Support:**  
 <x> : variable désignant un produit  
 <cu> : variable désignant un produit traité sur Cu  
 <p> : protocole d'accès  
 <e> : emplacement sur une palette



**Figure III.6 : réduction du modèle de l'îlot 2 : étape 2**

**Etape 3 : Unification des places de type "protocoles d'accès"**

Les places de type "protocole d'accès" modélisent la capacité limitée de chaque lieu physique dans un îlot. Dans le cas général, une telle place est associée aux places de type 2, 3, et 4.

Si nous avons unifié des places de type 2, 3, ou 4 dans l'étape précédente, nous unifions, en appliquant la même transformation d'équivalence, leurs places de type "protocole d'accès". Le marquage de chacune des places ainsi générées correspond à la somme des marques contenues dans les places fusionnées. Ceci permet de garder les informations quantitatives relatives à la saturation des ressources de l'îlot sur le modèle réduit.

Par rapport à l'îlot 2, seule la place Cu 1 / Cu 2 possède une place de type "protocole d'accès". Comme nous n'avons fusionné dans l'étape précédente que les tampons de l'îlot (cf places de

type 1), le modèle de la figure III.6 reste inchangé.

**Etape 4 : Elimination des Transitions Identiques**

Deux transitions sont identiques, si elles ont les mêmes places en aval et en amont et si leurs arcs adjacents ont les mêmes étiquettes. Les étiquettes caractérisent les produits qui peuvent franchir une transition.

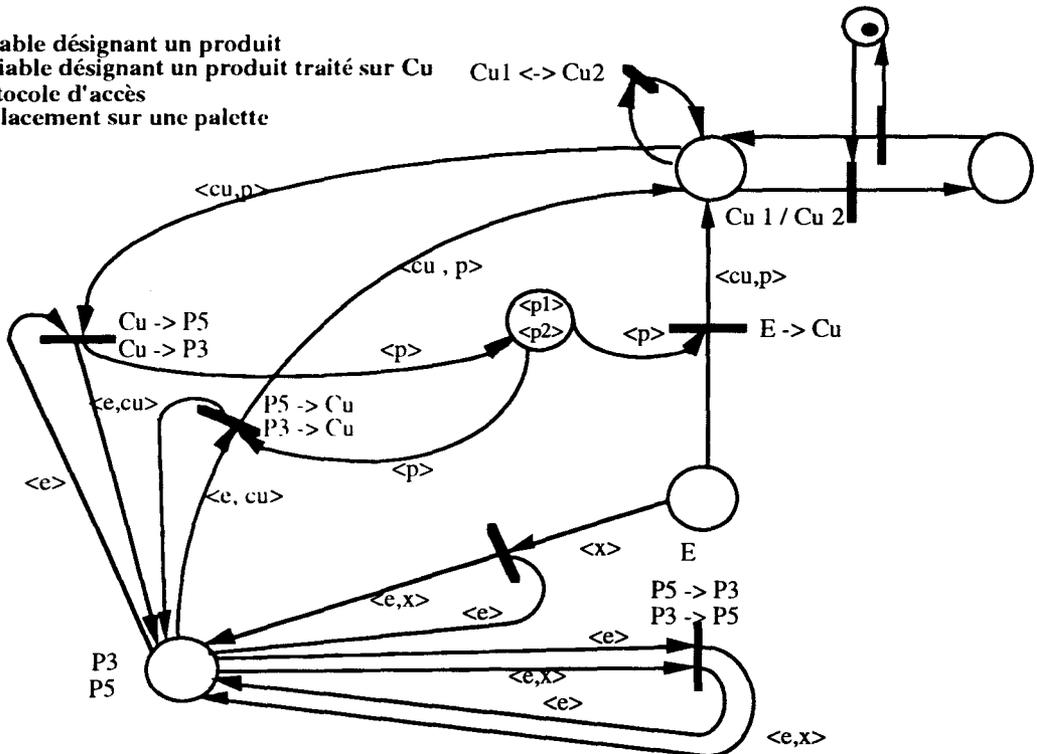
Comme l'indique la figure III.6, le regroupement des places P3 et P5 mène à des transitions qui ont les mêmes places an amont et en aval et dont les annotations sont identiques. Ces transitions sont énumérées ci dessous :

- \* Cu 1 / Cu 2 -> P5 et Cu 1 / Cu 2 -> P3
- \* P5 -> Cu 1 / Cu2 et P3 -> Cu 1 / Cu 2
- \* P5 -> P3 et P3 -> P5

En appliquant la règle d'unification des nœuds à ces transitions, nous pouvons supprimer une occurrence de chaque paire (figure III.7). Par ailleurs, l'élimination des transitions identiques fait partie des règles de réduction définies pour les réseaux de Petri ordinaires qui conservent les propriétés du modèle de départ [Berthelot 1983].

**Support:**

- <x> : variable désignant un produit
- <cu>: variable désignant un produit traité sur Cu     Cu1 <-> Cu2
- <p> : protocole d'accès
- <e> : emplacement sur une palette



**Figure III.7 : élimination des transitions identiques**

### Etape 5 : Elimination des Boucles

Une transition, qui boucle une place sur elle même, modélise un transfert élémentaire entre deux lieux physiques qui sont représentés par une seule place sur le modèle. Le modèle de la figure III.7 en donne trois exemples :

- \* la transition  $P5 \rightarrow P3 \vee P3 \rightarrow P5$ ,
- \* la transition  $Cu\ 1 \leftrightarrow Cu\ 2$  modélisant les transferts entre les deux tampons de Cu par R2.

En maintenant les propriétés du réseau, nous pouvons éliminer les transitions représentant de telles boucles, c'est-à-dire ayant les mêmes places en aval et en amont et dont les étiquettes sur les arcs sortants sont identiques aux étiquettes des arcs entrants. De plus, lorsqu'une place, bouclée sur elle-même par une telle transition, est marquée, la transition est toujours tirable. Nous enlevons donc les transitions bouclant les places  $P3 \vee P5$  et  $Cu\ 1 / Cu\ 2$  sur elles-mêmes. La figure III. 8 représente le résultat de cette dernière étape de réduction.

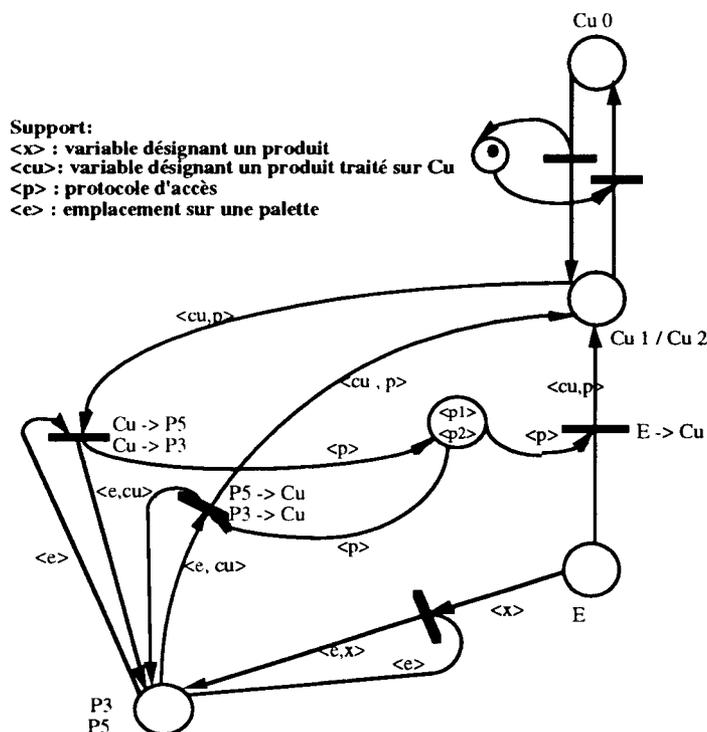


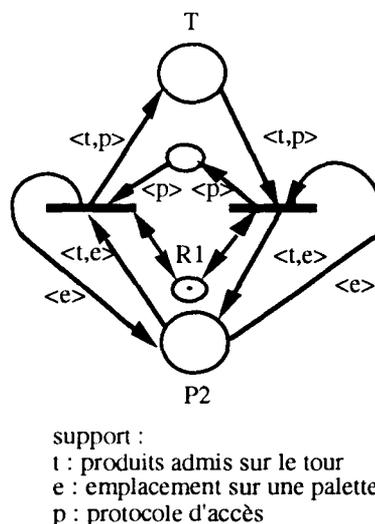
Figure III.8 : réduction du modèle de l'îlot 2 : modèle final

Les étapes de réduction du modèle d'un îlot conduisent à un modèle plus concis, qui

conserve les informations relatives à un blocage potentiel d'un îlot. Les relations d'accessibilité existant entre les différents lieux physiques de l'îlot sont conservées ainsi que leurs conditions d'accessibilité. Ces conditions, qui dépendent du marquage actuel, sont représentées soit par des conditions de tir soit par le marquage des places de type "protocole". Le regroupement de ces places permet enfin de conserver les informations quantitatives concernant la capacité des ressources de l'îlot.

#### 4.1.2.2. Réduction de l'îlot 1 de la Cellule

L'îlot 1 de la cellule est formé par le tour T (ressource de transformation avec chargement interne) et le robot R1. Son tampon d'entrée et de sortie correspond à la zone P2 sur le convoyeur central de l'installation. La figure III.9 représente le modèle de cet îlot résultant de la méthode de modélisation proposée dans le chapitre II.

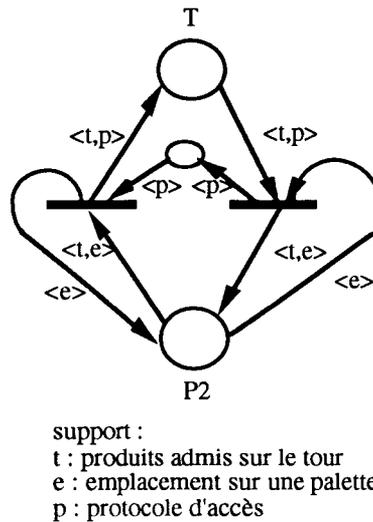


**Figure III. 9 : modèle de l'îlot 1**

La réduction de ce modèle se résume à l'élimination de la place implicite modélisant le robot R1 (étape 1 de la démarche de réduction). Le réseau de Petri de la figure III.9 ne comporte que deux places qui sont de type différent :

- \* la place T est une place de type 2 modélisant une ressource de transformation avec chargement interne,
- \* la place P2, correspondant au tampon de l'îlot, fait partie des places de type 1.

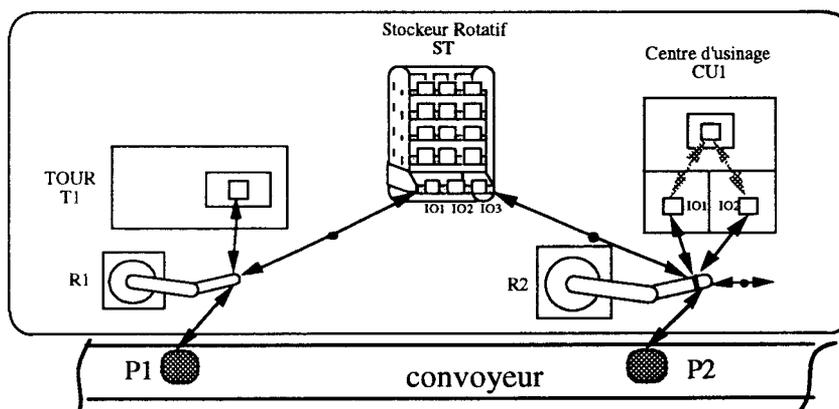
L'unification des deux places n'étant pas possible, le modèle de la figure III.10 correspond donc au modèle réduit de cet îlot.



**Figure III.10 : modèle réduit de l'îlot 1**

#### 4.1.2.3. Réduction d'un îlot comportant une ressource partagée entre deux îlots élémentaires

La figure III.11 schématise un îlot constitué par un ensemble de ressources dont les lieux de synchronisation forment deux composantes fortement connexes sur le graphe des accessibilités du système. L'intersection des nœuds de ces deux composantes n'étant pas vide, nous regroupons les ressources dans un seul îlot (cf paragraphe 4.3.3.2 du chapitre II). En effet, le stockeur rotatif ST est une ressource partagée entre les deux îlots élémentaires composés de CU1 et R2 d'une part et de T1 et R1 d'autre part.



**Figure III.11 : îlot avec ressource partagée**

Le modèle réseau de Petri de cet îlot est présenté sur la figure III.12. Pour conserver la

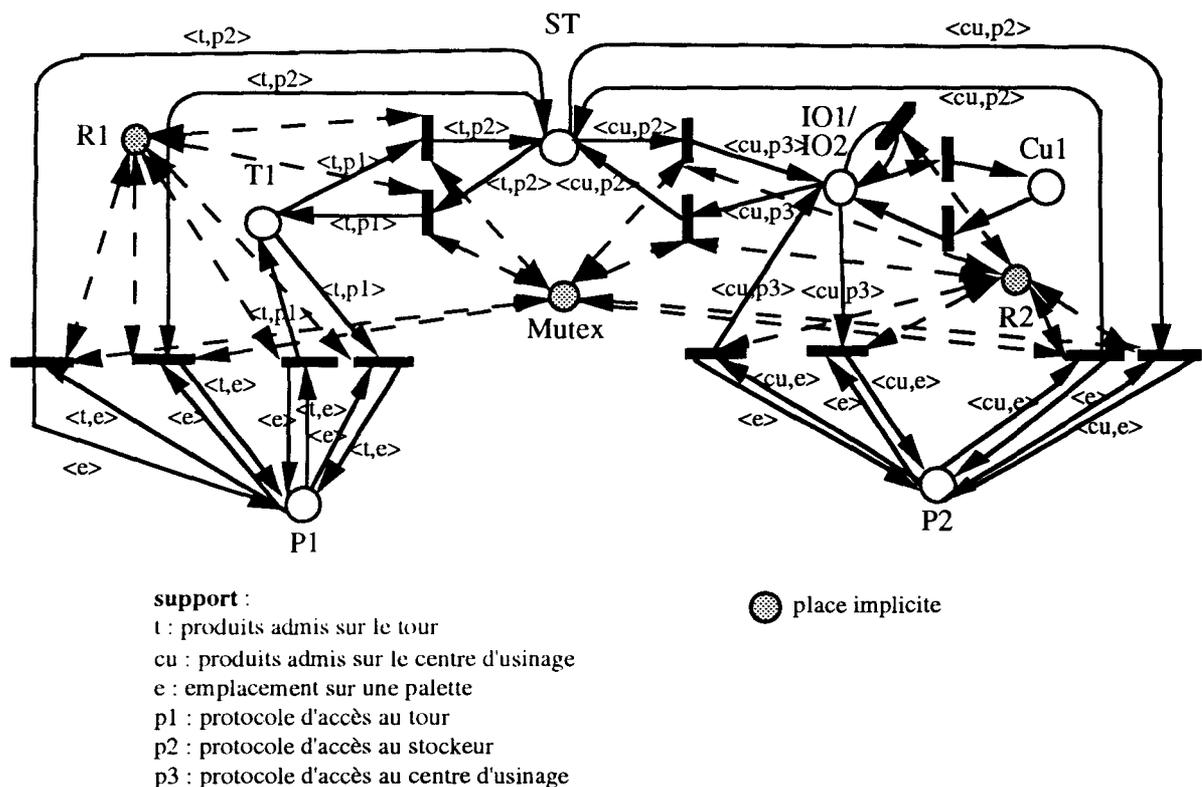
lisibilité du modèle, nous avons supprimé les places modélisant les protocoles d'accès aux ressources.

Le centre d'usinage Cu 1 de cet îlot est, comme pour celui de la cellule de l'Ecole Centrale de Lille, une ressource de transformation avec chargement externe. Ses lieux de synchronisation IO1 et IO2 sont représentés par une seule place sur le modèle réseaux de Petri (IO1 / IO2).

Le tour T1 appartient à la classe des ressources de transformation avec chargement interne, son lieu de synchronisation, qui est aussi son lieu de transformation, est modélisé par T1.

Le stockeur ST présente dix emplacements de stockage et les deux robots R1 et R2 peuvent y accéder.

La place "mutex" représente l'accès mutuellement exclusif à cette ressource partagée. Le stockeur ne peut être sollicité que par une seule commande de chargement ou de déchargement à un instant donné.

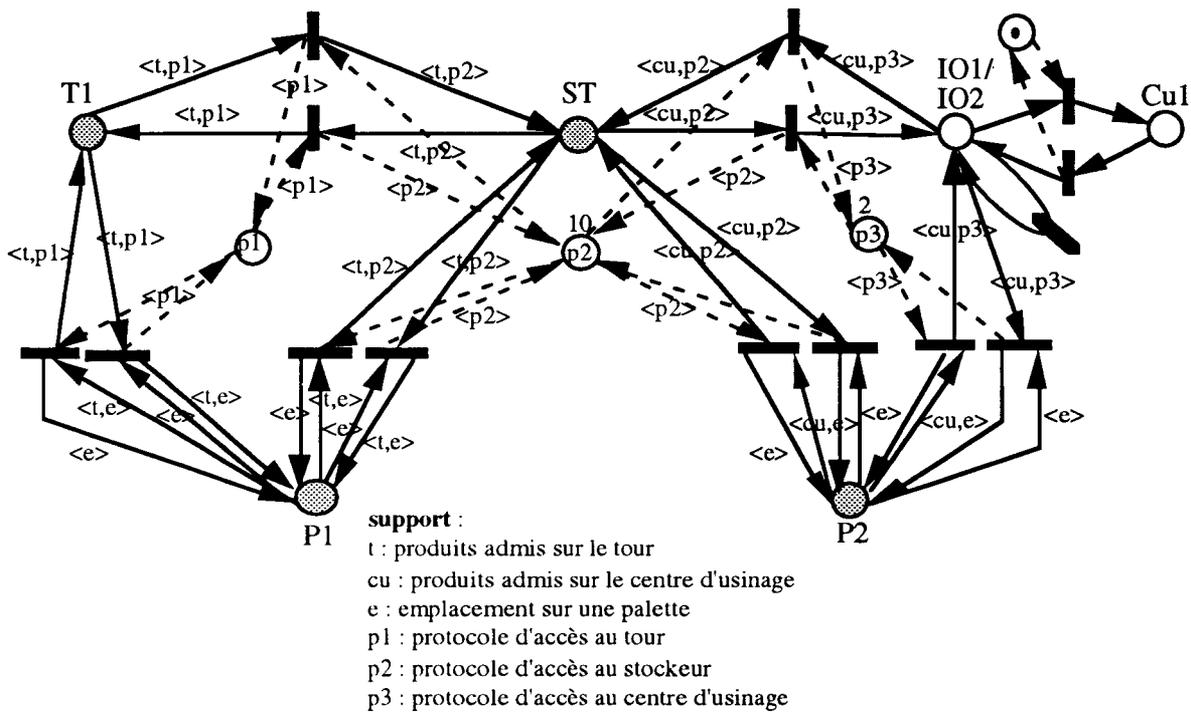


**Figure III.12 : modèle de l'îlot avec ressource partagée**

Nous exposons par la suite la réduction de cet îlot suivant les étapes proposées.

**Etape 1 : Elimination des Places Implicites**

Le modèle de la figure III.12 comporte trois places implicites : les places "mutex", R1 et R2. Leur suppression conduit au modèle de la figure III.13 , sur laquelle nous avons représenté les protocoles d'accès aux différentes ressources.



**Figure III.13 : élimination des places implicites**

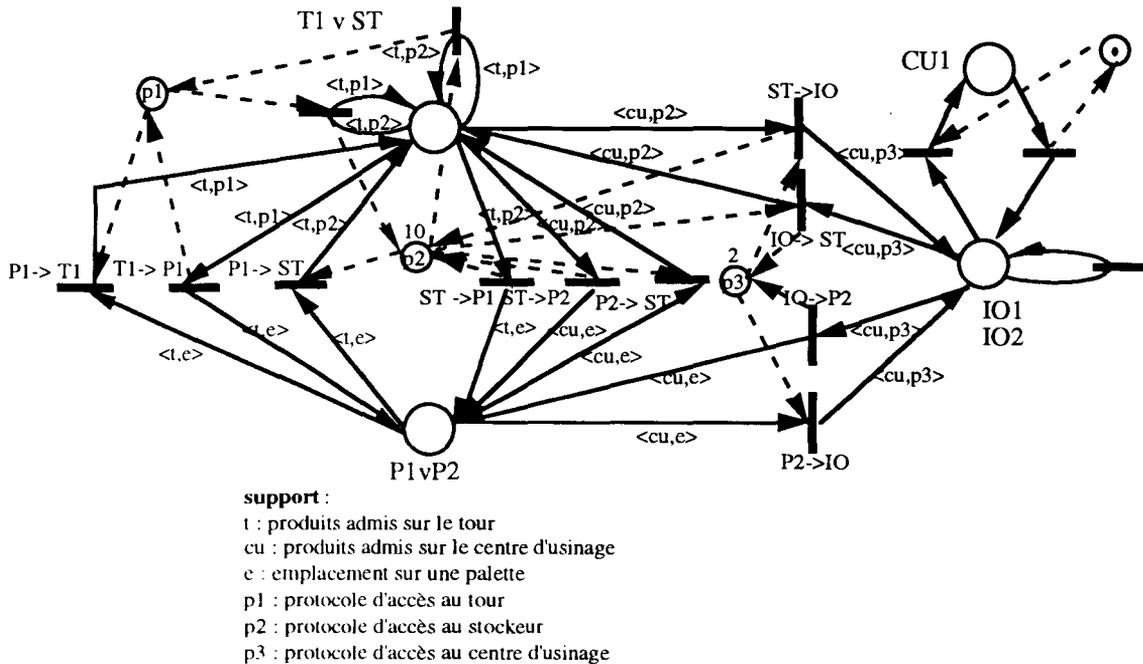
**Etape 2 : Fusion des Places**

Dans cette étape nous regroupons les places du modèle caractérisées par un comportement identique par rapport au flux des produits.

Nous pouvons établir les correspondances suivantes entre les quatre types de places définis et les places du modèle de la figure III.13 :

- \* les tampons de l'îlot (places de type 1) : P1 et P2,
- \* lieux de synchronisation des ressources de stockage et des ressources de transformation avec chargement interne (places de type 2) : ST et T1,
- \* lieux de synchronisation des ressources avec chargement externe (places de type 3) : IO1/IO2,
- \* lieux de transformation des ressources avec chargement externe (places de type 4) : Cu1.

D'après nos propositions, nous regroupons donc dans cette étape les places P1 et P2 dans une place et les places T1 et ST dans une autre. La figure III.14 représente le résultat de ce regroupement.



**Figure III.14 : regroupement des places**

**Remarque :** Nous avons supprimé les arcs étiquetés de  $\langle e \rangle$ , indiquant la libération d'une palette dans le cas d'un déchargement ou l'occupation d'un emplacement dans le cas d'un chargement d'un produit sur l'un des tampons de l'îlot.

La place P1 est directement accessible à partir du tour et du stockeur rotatif, la place P2 l'est à partir du centre d'usinage et de ST. Les transitions du modèle de la figure III.14 modélisent ces propriétés. Les conditions associées aux arcs représentent la pré-condition d'un chargement d'un produit sur une palette : la présence d'une palette vide dans le tampon correspondant. La fusion des deux places P1 et P2 dans la place P1 v P2 conserve donc les spécificités de chacune, nous obtenons une représentation équivalente.

Le regroupement de ST et de T1 conserve, lui aussi, les relations d'accessibilité de chacune des ressources et ses conditions d'accessibilité, représentées cette fois-ci par le marquage des places de type "protocole d'accès". La fusion de ces deux places est alors possible, car, d'une part, elles présentent les mêmes **types** de relation d'accessibilité, d'autre part, leurs **conditions** d'accessibilité sont représentées de manière identique. Les places de type 2 sont toujours et exclusivement en relation d'accessibilité avec d'autres lieux de synchronisation



- \* P1 -> ST, P1 -> T1 par une transition nommée P1 -> ST v P1 -> T1,
- \* T1 -> ST, ST-> T1 par une transition nommée ST -> T1 v T1 -> ST.

Nous remarquons, que les arcs adjacents à la transition P1 -> ST v P1 -> T1 ne montrent pas les mêmes annotations que les arcs de la transition P2 -> ST. De ce fait, les transitions ne sont pas identiques et ne peuvent donc pas être fusionnées. Les différents arcs indiquent deux types de produits qui sont admis chacun dans une partie de l'îlot, le premier type n'est admis que sur T1 et ST, le deuxième type ne l'est que sur ST et Cu 1.

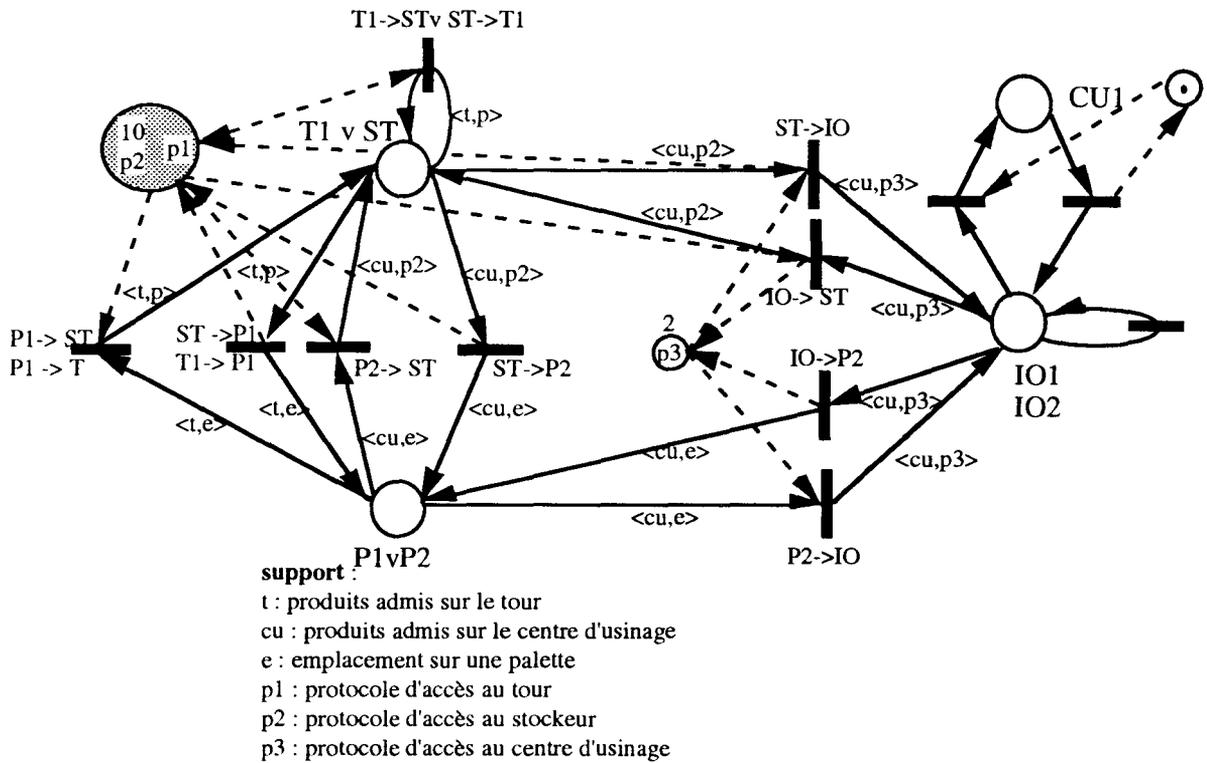
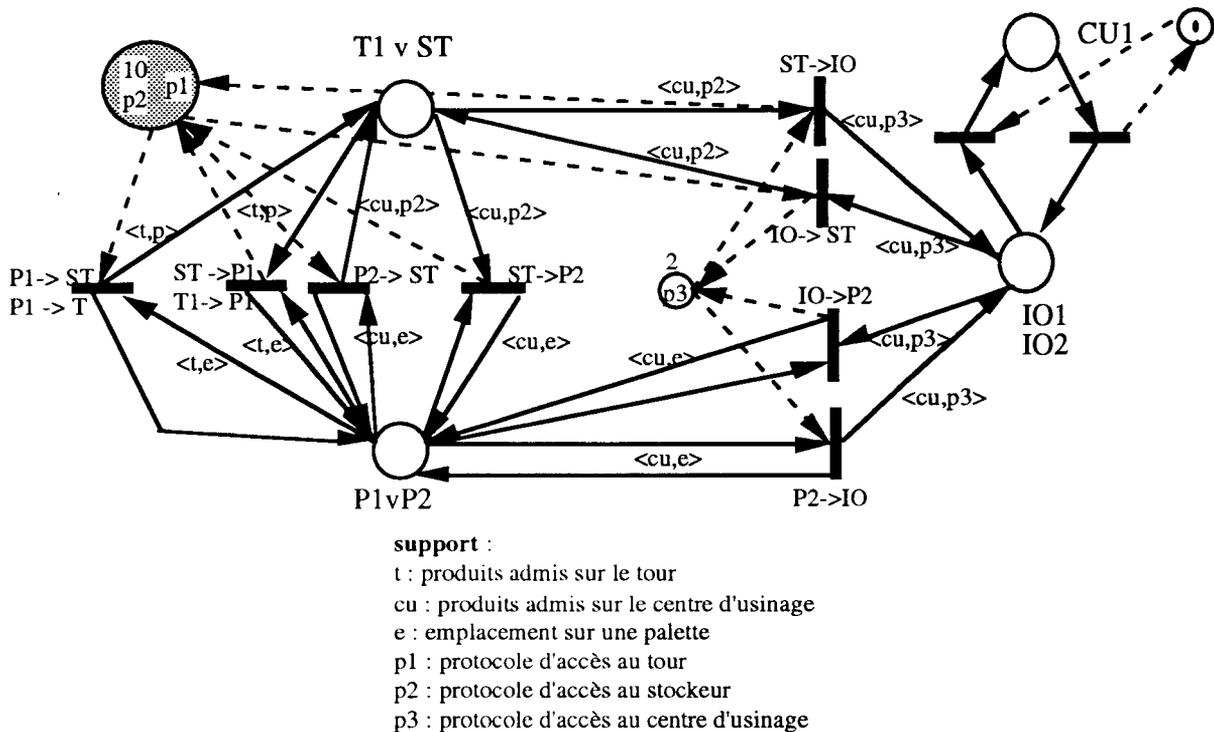


Figure III.16 : fusion des transitions identiques

**Etape 5 : Elimination des Boucles**

Nous éliminons dans cette dernière étape les transitions qui ont les mêmes places en amont et en aval. La transition T1 -> ST v ST -> T1 est de ce type. Elle a la place ST v T1 et la place de type protocole duale en amont et en aval. Nous pouvons donc la supprimer sans affecter les propriétés du réseau.

Nous éliminons également, dans cette étape, la transition qui boucle la place IO1/IO2 sur elle même. La figure III.17 montre le modèle réduit final obtenu par la démarche de réduction.



**Figure III.17 : modèle réduit de l'îlot avec ressource partagée**

#### 4.1.2.4. Conclusion

Nous avons illustré sur différents exemples significatifs l'application de la démarche de réduction des modèles des îlots. Il s'agit respectivement des deux îlots simples de la cellule de l'Ecole Centrale de Lille et d'un îlot comportant une ressource partagée entre deux îlots simples. Ce dernier exemple nous a permis de montrer l'intérêt d'une telle réduction. Le modèle est plus concis, il conserve d'une part le comportement réel des ressources modélisées, d'autre part les propriétés du modèle de départ relatives à son analyse.

### 4.1.3. Une Démarche pour l'Analyse du Modèle Réduit

#### 4.1.3.1. Introduction

Dans le paragraphe 4.1.1. nous avons indiqué, que le blocage du modèle d'un îlot, s'il est possible, ne peut être atteint que par la saturation de ses ressources. Afin de détecter ce blocage éventuel, nous analyserons le comportement du modèle réduit dans une situation extrême de ce type. Cette situation correspond à un mode opératoire du système qui ne comporte que des produits admis dans l'îlot, susceptibles d'assurer la saturation des ressources le composant.

Cependant, un îlot peut comporter plusieurs ressources qui ne sont pas forcément admises pour l'ensemble des produits visitant l'îlot. Dans le cas de l'îlot avec ressource partagée, par exemple, les produits de type <cu> ne sont admis que sur le centre d'usinage et le stockeur, tandis que les produits de type <t> ne le sont que sur le tour et le stockeur.

Puisque nous recherchons la condition de la vivacité du modèle sans contrainte sur les modes opératoires, nous devons analyser chaque îlot selon différents points de vue. Chaque point de vue correspondra alors à un mode opératoire de l'îlot. Comme chacun des modes opératoires peut conduire à un blocage partiel du modèle, nous examinons d'abord le comportement du modèle dans des modes opératoires ne comportant qu'un seul type de produit admis dans l'îlot. Ensuite nous analyserons toutes les combinaisons possibles de deux types de produits, puis de trois types de produits jusqu'à ce que nous arrivions à un mode opératoire qui comporte l'ensemble des types des produits admis dans l'îlot.

Un îlot échange des produits avec le reste du système de production par l'intermédiaire du moyen de transport central. Un îlot n'est donc pas autonome et son comportement dépend de la situation du système global. Pour l'analyse du comportement du modèle réduit nous intégrons une représentation du moyen de transport de façon agrégée par une seule place. Cette place contiendra lors de l'analyse l'ensemble des emplacements sur les supports de transport, qui sont admis pour les produits du mode opératoire examiné.

Nous proposons par la suite une démarche d'analyse du modèle réduit dans son environnement qui permettra de définir une condition de non-saturation de l'îlot en garantissant la vivacité du modèle. Les étapes de la démarche seront exposées sur l'exemple de l'îlot 2 de l'Ecole Centrale de Lille. Nous précisons, dans un deuxième temps, son application aux exemples de l'îlot 1 et de l'îlot avec ressource partagée.

#### ***4.1.3.2. L'Analyse de l'Îlot 2***

##### **Etape 1 : Intégration du Mode Opératoire extrême au Niveau du Modèle**

Nous recherchons, dans un premier temps un mode opératoire extrême qui conduit à une saturation des ressources de l'îlot.

Le modèle réduit de l'îlot 2 (figure III.8) distingue deux types de produits : les produits à usiner sur le centre d'usinage (<cu>) et les produits de type <x>, englobant l'ensemble des produits du système.

Comme les produits de type <cu> sont admis sur l'ensemble des ressources de l'îlot 2, le seul



transport interdits pour ce type de produit n'influencent évidemment pas le comportement de l'îlot.

Le convoyeur de la cellule de notre exemple comporte 17 palettes universelles de capacité unitaire. Pour saturer le système de transport par rapport à l'îlot analysé, nous chargeons chaque palette par un produit de type  $\langle cu \rangle$ . La place P1 du modèle de la figure III.17 est donc marquée par 17 jetons de type  $\langle e, cu \rangle$ .

Puisque la place E n'est pas accessible, son marquage n'influence pas le comportement du système.

Les tampons d'entrée du centre d'usinage (Cu1 et Cu2) sont occupés chacun par un produit de type  $\langle cu \rangle$ , le marquage de la place Cu1 / Cu2 correspond donc à 2 jetons de type  $\langle cu, p \rangle$ . Le centre d'usinage est saturé et la place "protocole" associée à Cu1 / Cu2 est vide.

La figure III.19 représente la saturation de la cellule par rapport à l'îlot 2. Le marquage de saturation comporte 19 produits de type  $\langle cu \rangle$  et mène à un blocage local du modèle réduit de l'îlot 2. Seules les transitions t1 et t9 sont tirables. La séquence t9, t10 modélise les allers et retours des produits entre le lieu de transformation de Cu et ses lieux de synchronisation, la séquence t1, t2 correspond au passage des palettes chargées devant l'îlot.

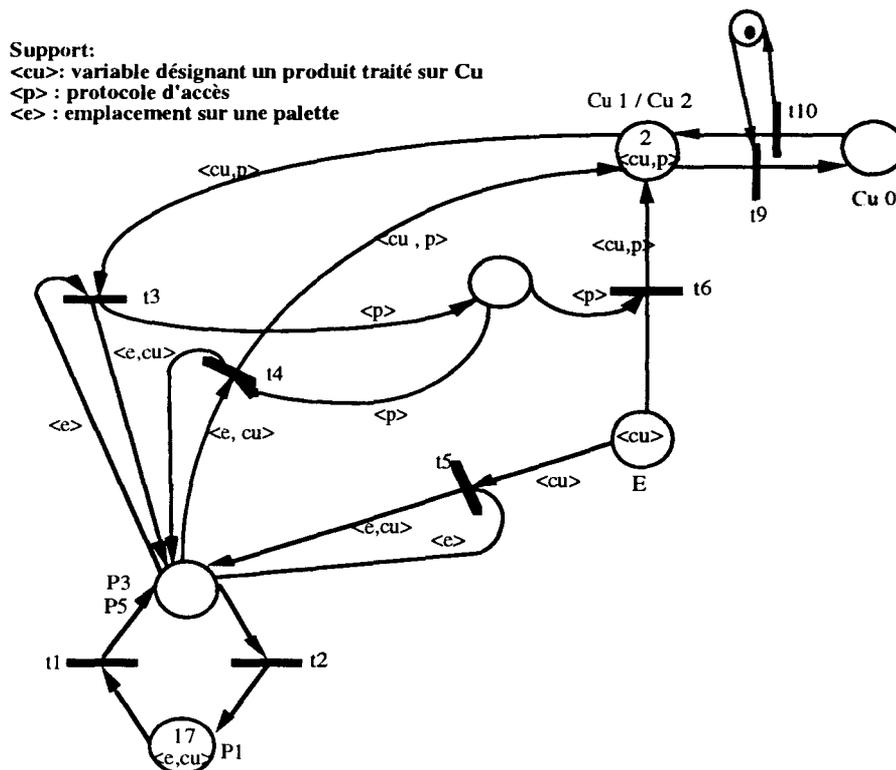


Figure III.19 : saturation de l'îlot 2

### **Etape 3 : Recherche de la Condition de Vivacité du Modèle**

Dans cette dernière étape, nous déterminons la condition de vivacité à charge maximale du modèle réduit de l'îlot et, par conséquent, du modèle de l'îlot par rapport au mode opératoire défini.

Reconsidérons l'exemple de l'îlot 2 de la figure III.19. Il suffit de libérer un seul emplacement dans l'îlot pour que le modèle devienne vivant. Si nous libérons, par exemple, un emplacement sur une palette, la place P1 sera marquée par 16 jetons de type  $\langle e, cu \rangle$  et un jeton de type  $\langle e \rangle$  qui correspond à l'emplacement libéré. Les transitions t3 et t5 sont alors sensibilisées. Nous pouvons franchir la transition t3, afin de décharger un produit du centre d'usinage. Les transitions t4 et t6 seront sensibilisées par la suite.

Si nous franchissons, dans ce cas, immédiatement la transition t6, nous obtenons de nouveau un marquage de saturation, le modèle se bloque de nouveau. Nous devons interdire le franchissement de cette transition afin d'assurer l'absence du blocage. Cela revient à limiter l'encours maximal dans l'îlot 2 à 18 produits. Nous déduisons donc, que le modèle de l'îlot 2 est vivant, si, à tout instant, il y a au maximum 18 produits de type  $\langle cu \rangle$  sur les ressources de l'îlot et du système de transport.

#### **4.1.3.3. L'Analyse de l'Îlot 1 de la Cellule**

Nous analysons maintenant le modèle réduit de l'îlot 1 de la cellule de l'Ecole Centrale de Lille.

### **Etape 1 : Intégration du Mode Opératoire Extrême au Niveau du Modèle**

Puisque l'îlot 1 de la cellule ne comporte qu'une seule ressource, tous les produits qui visitent l'îlot sont du type  $\langle t \rangle$ . Pour cette première étape d'analyse de l'îlot nous nous limitons donc à l'ajout de la place P1 modélisant le convoyeur central de la cellule (figure III.20).

### **Etape 2 : Recherche du Marquage de Saturation**

Nous remplissons, dans cette étape, l'îlot et le système de transport avec des produits du mode opératoire défini auparavant. Les 17 palettes universelles du convoyeur portent donc chacune un produit de type  $\langle t \rangle$ . La place P1 est marquée par 17 jetons du type  $\langle e, t \rangle$ . Le tour, l'unique ressource de cet îlot, dispose d'un emplacement de stockage. Pour sa saturation, il suffit de marquer la place T par un jeton  $\langle t, p \rangle$ .

La figure III.21 représente la saturation de l'îlot 1 comportant 18 produits. Elle conduit au blocage local du modèle, seule la séquence de tir des transitions t1, t2 est réalisable.

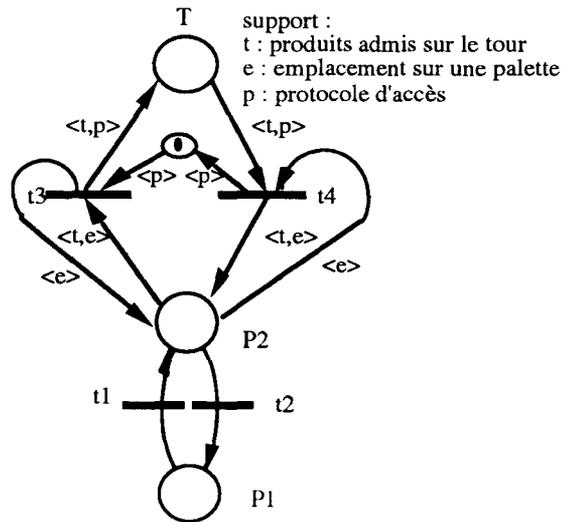


Figure III.20 : îlot 1 avec représentation du système de transport

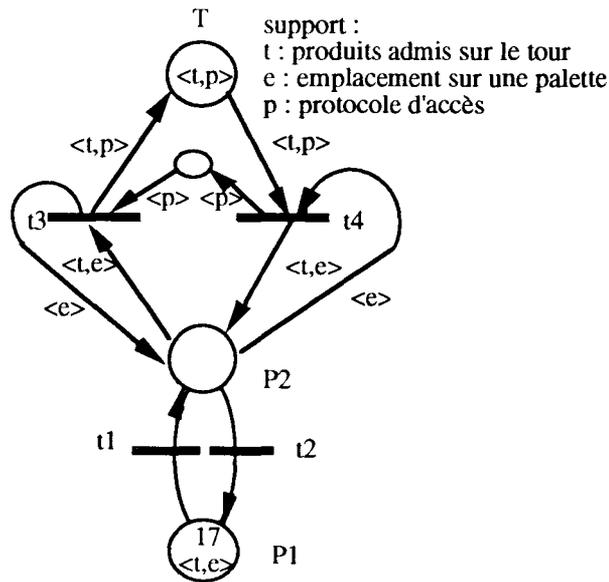


Figure III.21 : saturation de l'îlot 1

**Etape 3 : Recherche de la Condition de Vivacité du Modèle**

De la même manière que dans l'exemple précédant, la libération d'un seul emplacement de stockage, soit sur le tour, soit sur une palette, rend le modèle vivant. Le déchargement du tour nous permet de trouver une séquence de tir des transitions qui contient toutes les transitions et qui conduit dans le marquage initial (par exemple la séquence t3, t4, t2, t1). Nous proposons donc de limiter l'encours dans l'îlot 1 à tout instant à 17 produits.

#### 4.1.3.4. L'Analyse de l'Îlot avec Ressource Partagée

Cet îlot comporte deux ressources de transformation, le tour T1 et le centre d'usinage Cu1. De ce fait, le modèle réduit fait référence à deux types de produits différents selon que les produits sont admis sur le tour ou le centre d'usinage. Nous sommes donc amenés à analyser le comportement du modèle dans trois modes opératoires différents : deux modes opératoires ne comportant qu'un seul produit et un mode opératoire comportant les deux types de produits.

##### **Etape 1: Définition d'un Mode Opérateur Extrême**

Les trois modes opératoires pour l'analyse de cet îlot sont les suivants :

- \* le premier mode opératoire ne comporte que des produits de type <cu>, donc exclusivement admis sur le centre d'usinage et le stockeur,
- \* le deuxième mode opératoire ne comporte que des produits de type <t>,
- \* le troisième mode opératoire comporte des produits de type <x> := <t> v <cu> qui sont admis sur l'ensemble des ressources de l'îlot.

Les produits de type <x> vérifient donc à la fois la condition <cu> et la condition <t> exprimées sur les arcs du modèle réduit.

Concernant la représentation du système de transport, nous ajoutons la prise en charge de sa contrainte de capacité sous forme de la place TS (figure III.22). Nous supposons par la suite que le système de transport est constitué par un convoyeur sur lequel se déplacent 17 palettes universelles de capacité unitaire.

##### **Etape 2 : Recherche d'un Marquage de Saturation**

Comme pour les exemples précédents nous saturons les ressources de l'îlot avec des produits dont le type correspond au mode opératoire défini.

Dans le cas du premier mode opératoire (produits de type <cu>), nous chargeons le stockeur avec 10 produits, la place de type "protocole" associée à la place ST v T est alors marquée par un seul jeton de type <p1> représentant le protocole d'accès au tour.

Nous saturons également les 17 palettes par des jetons de type <cu,e> et le centre d'usinage par deux jetons de type <cu,p3>. La figure III.22a représente cette situation.

La saturation de l'îlot pour les produits de type <t> est obtenu de la même manière. La figure III.22b indique la saturation du modèle par rapport aux produits de type <x>. Nous remarquons le marquage de la place ST v T1 par 11 jetons de type <x,p>. La place protocole associée n'est donc plus marquée.

La saturation conduit dans les deux cas au blocage local de l'îlot. Seules les transitions t1 et t9 sont tirables. Le marquage de saturation comporte dans le premier cas 29 produits, dans le

deuxième cas 30 produits.

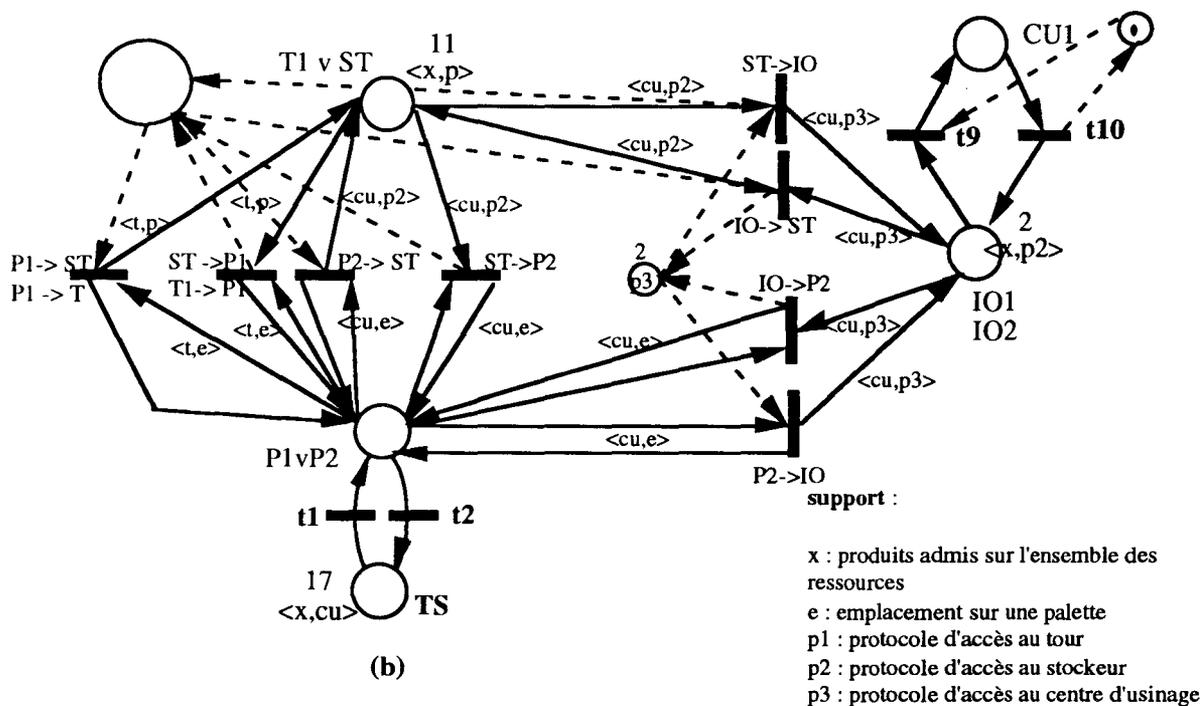
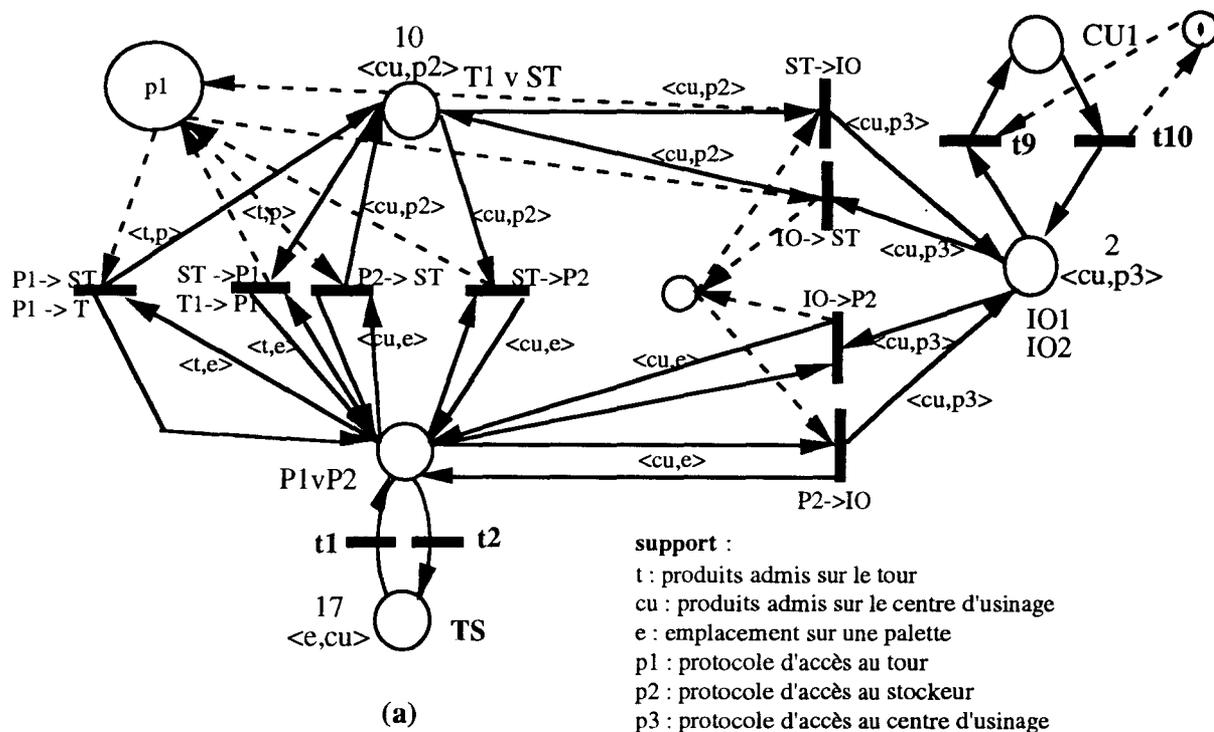


Figure III.22 : saturation de l'îlot dans deux modes opératoires



### **Etape 3 : Recherche de la Condition de Vivacité du Modèle**

Si nous libérons, comme dans les exemples précédents un seul emplacement de stockage, le modèle de la figure III.22b (mode opératoire 3) devient vivant.

Dans le cas du mode opératoire 1, en libérant un emplacement, la partie du modèle admise pour les produits de type <cu> devient vivante. Les transitions dont les arcs font référence à un produit de type <t> ne sont pas franchissables, car les produits de type <cu> ne vérifient pas leur pré-condition, ils appartiennent à un autre mode opératoire.

Nous proposons donc de limiter l'encours dans l'îlot à 28 produits de type <cu> et à 29 produits de type <x> afin de garantir la vivacité du modèle à charge maximale.

#### **4.1.4. La Généralisation des Résultats**

En conclusion il apparaît, que l'analyse des modèles des îlots nous a permis de définir une méthode d'analyse des conditions de leur vivacité. Partant de l'observation que chaque modèle est issu d'un graphe d'état enrichi par des protocoles d'accès, nous avons présenté une démarche qui permet d'exprimer une borne supérieure de l'encours garantissant un comportement vivant du modèle.

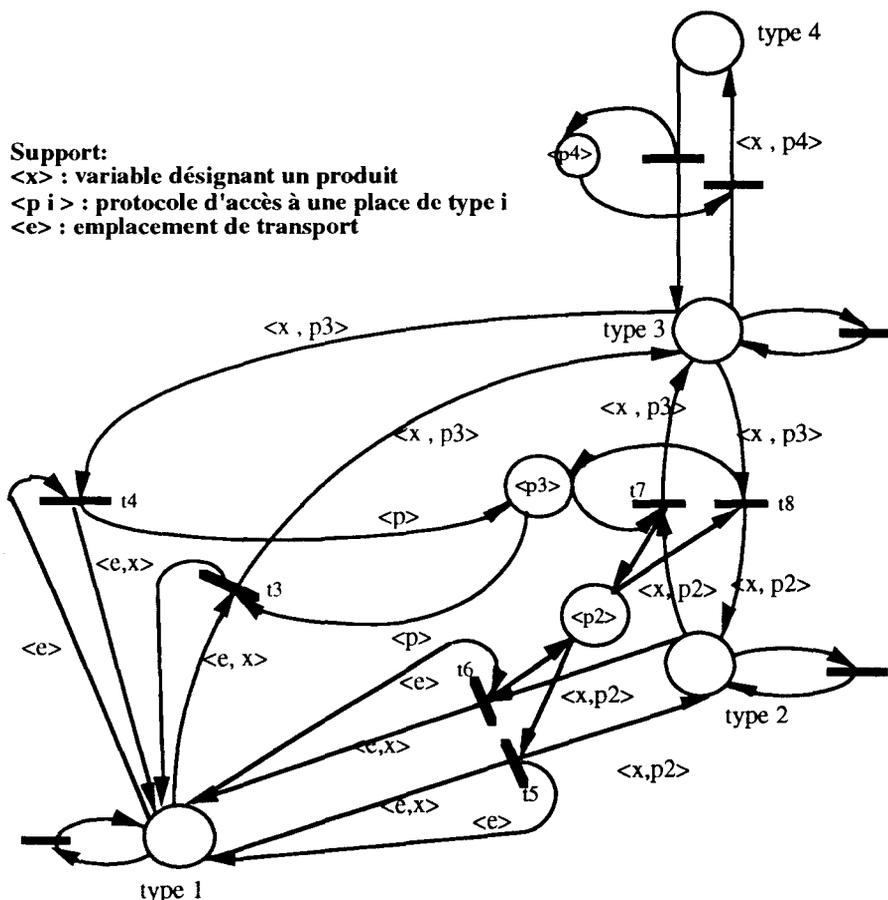
Les propriétés d'un îlot ont été transposées sur un modèle réduit, qui représente les localisations des produits dans l'îlot à un niveau plus abstrait. Ce modèle réduit est sémantiquement équivalent au modèle du départ.

Cette étude introduit l'idée d'un modèle abstrait d'un îlot du point de vue des propriétés analysées. Nous construisons ce modèle abstrait à partir des quatre types des places définies pour la réduction des modèles (figure III.23).

La place de type 1 regroupe l'ensemble des tampons d'entrée / sortie d'un îlot. La place de type 2 réunit les ressources de transformation avec chargement interne et les moyens de stockage présents dans un îlot. Les lieux de synchronisation des ressources avec chargement externe (de type centre d'usinage) sont modélisés par la place de type 3 et leurs lieux de transformation par la place de type 4.

Nous associons également des places de type "protocole d'accès" aux places de type 2, 3, et 4. Elles sont marquées par des jetons de type <p2>, <p3> et <p4> respectivement et leur marquage initial correspondra aux limites de capacité des emplacements de stockage disponibles dans l'îlot sur chaque type de ressource.

Les places de type 1, 2, et 3 modélisent des lieux de synchronisation. Du fait de la définition d'un îlot comme composante symétrique et complète du graphe des accessibilités du système, chacune de ces places est accessible à partir de toute autre. Les transitions du modèle générique représentent ces accessibilités. Nous supposons pour la définition du modèle générique que l'îlot comporte plusieurs lieux d'un même type mutuellement accessibles. Ces lieux sont donc regroupés en une seule place. L'accessibilité directe entre ces lieux se traduit par une transition qui boucle chacune des places de type 1, 2 ou 3 sur elle-même.



**Figure III.23 : modèle d'un îlot abstrait**

Le modèle de la figure III.23 inclut de fait toutes les configurations possibles d'un îlot. Il constitue dans ce sens un îlot générique. Dans le cas de l'îlot avec ressource partagée (paragraphe 4.1.2.3 de ce chapitre), certains lieux physiques, représentés par une même place ne sont pas directement accessibles entre eux (par exemple les tampons de l'îlot), le modèle réduit, déduit du modèle abstrait, ne contiendra donc pas de place de type 1 bouclée sur elle-même.

Le modèle générique contient aussi les modèles des îlots 1 et 2 de la cellule de l'Ecole

Centrale de Lille ainsi que le modèle réduit de l'îlot avec ressource partagée. Par rapport au modèle de l'îlot 2, par exemple, nous retrouvons les quatre types de places modélisant ses lieux caractéristiques. Ses transitions forment un sous-ensemble des transitions du modèle de l'îlot générique.

De manière générale, le modèle réduit d'un îlot d'un SFPM sera alors formé par un sous-ensemble des places et des transitions du modèle générique. Pour l'analyse des propriétés, l'identification d'un modèle générique permet de généraliser la démarche d'analyse et les résultats obtenus. En effet, nous pouvons appliquer la démarche d'analyse présentée pour les exemples au cas général du modèle abstrait.

A cette fin nous supprimons les transitions bouclant les places de type 1, type 2 et type 3 sur le modèle de la figure III.23. Nous intégrons une vue abstraite du système de transport par l'ajout d'une place qui exprime la limite de capacité de ce système, comme nous l'avons fait auparavant (place P1 de la figure III.24).

#### **Etape 1 : Définition d'un Mode Opérateur Extrême**

Nous partons par la suite de l'hypothèse qu'il existe un type de produit qui soit admis sur toutes les ressources de l'îlot abstrait.

Ainsi que nous l'avons montré lors de l'analyse de l'îlot avec ressource partagée, la non-vérification de cette hypothèse a seulement un effet quantitatif sur les résultats de l'analyse.

#### **Etape 2 : Recherche du Marquage de Saturation**

Nous supposons que le système dispose de  $n$  emplacements de transport admis pour les produits de type  $\langle x \rangle$ . Ce type englobe l'ensemble des produits traités dans l'îlot, la place P1 est donc marquée par  $n$  jetons de type  $\langle c, x \rangle$ .

Le marquage de saturation de la place type 2 correspond à  $l$  jetons de type  $\langle x, p2 \rangle$ , où  $l$  est la quantité d'emplacements de stockage sur des ressources avec chargement interne ou sur des stockeurs de l'îlot.

La place de type 3 modélise l'ensemble des lieux de synchronisation des ressources avec chargement externe. Nous supposons que l'îlot examiné comporte  $m$  emplacements de ce type. La place est donc marquée par  $m$  jetons de type  $\langle x, p3 \rangle$ . La variable  $p3$  représente un emplacement de stockage sur une des ressources. Puisque nous avons saturé les ressources, la place de type protocole d'accès associée n'est donc pas marquée.

La place de type 4 modélise les lieux de transformation des ressources avec chargement externe. Cette place n'est pas marquée, car tous les produits se trouvent sur les lieux de synchronisation de leurs ressources respectives. Sa place de type protocole contient donc les  $o$  jetons de type

$\langle p4 \rangle$  modélisant les  $o$  lieux de transformation de ce type dans l'îlot.

La situation représentée sur la figure III.24 correspond au marquage de saturation défini. Tous les lieux de l'îlot et du système de transport, capables de recevoir un produit de type  $\langle x \rangle$  sont occupés par un produit. Le modèle est localement bloqué, car seules les transitions  $t1$  et  $t9$  sont tirables. La séquence de tir  $t1, t2$  correspond au passage des palettes chargées devant l'îlot, la séquence de tir  $t9, t10$  aux allers et retours des produits sur les ressources avec chargement externe de type du centre d'usinage de l'exemple de l'Ecole Centrale de Lille.

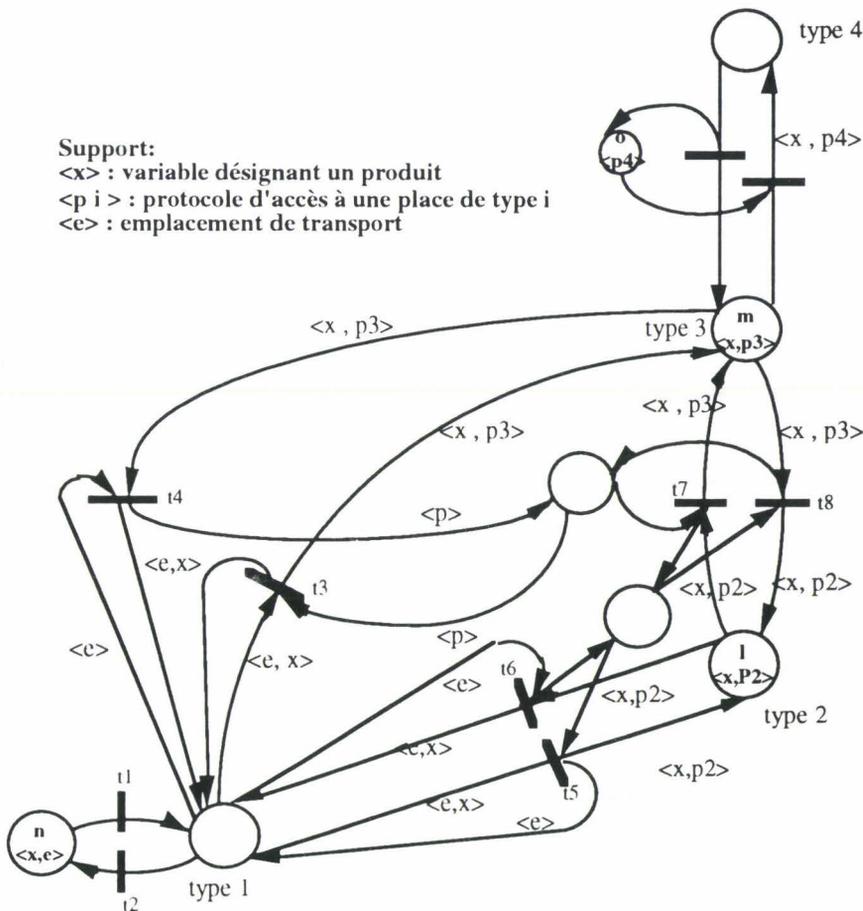


Figure III.24 : saturation du modèle abstrait

**Etape 3 : Recherche de la Condition de Vivacité**

Comme dans les cas traités précédemment, il suffit de libérer un seul emplacement, pour que le modèle devienne vivant.

A titre d'exemple nous libérons un emplacement de transport. La place P1 est donc marquée par  $(n-1)$  jetons de type  $\langle x,e \rangle$  et un jeton de type  $\langle e \rangle$ , représentant l'emplacement libéré. Les transitions  $t6$  et  $t4$  sont sensibilisées. Si nous tirons  $t4$ , nous libérons un emplacement sur les

ressources de type 3. Ceci mène à la sensibilisation de  $t_7$  et  $t_5$ . Nous retrouvons sur ce modèle alors des séquences de tirs de transitions qui contiennent toutes les transitions et qui mènent dans l'état initial. Le modèle est vivant.

Ces résultats nous permettent de déduire la règle suivante, pour déterminer le marquage maximal de non-saturation d'un îlot par rapport à un mode opératoire examiné :

Le marquage maximal de non-saturation d'un îlot par rapport à un type de produit est déduit du marquage de saturation de l'îlot par décrémentation d'une unité. Puisque le marquage de saturation est obtenu par la saturation de chaque ressource et la saturation des emplacements de transport admis pour le type de produit, il peut être calculé selon la formule suivante :

$$(ET_x + EI_x) - 1$$

avec :

- \*  $ET_x$  nombre des emplacements de transport, admis pour les produits de type  $\langle x \rangle$ ,
- \*  $EI_x$  nombre des emplacements sur les ressources de l'îlot, admis pour les produits de type  $\langle x \rangle$ .

La définition du modèle abstrait d'un îlot et son analyse permettent de ce fait de généraliser les résultats des démarches de réduction et d'analyse proposées. Nous pouvons donc affirmer que les modèles des îlots, conçus selon la méthode proposée dans le chapitre II, sont vivants sous réserve d'une limitation de l'encours par rapport à chaque îlot. Afin d'assurer le bon comportement du système et de sa commande, nous sommes amenés à représenter cette contrainte d'exploitation sur les modèles.

#### 4.1.5. L'Intégration des Résultats au Niveau des Modèles

La gestion de l'encours demande une vue globale du processus de fabrication. Supposons la situation suivante : la cellule de l'Ecole Centrale de Lille se trouve dans un mode opératoire qui ne comporte que des produits admis dans l'îlot 1 (produits de type  $\langle t \rangle$ ). Dans la cellule circulent 17 produits de ce type, le modèle de l'îlot 1 est alors vivant et dans le tampon d'entrée de la cellule E se présente un produit de type  $\langle t \rangle$ .

Si nous limitons l'encours de manière locale sur le modèle de l'îlot 1, rien n'interdit la prise en compte du produit  $\langle t \rangle$  présent dans le tampon E de la cellule. En effet, dès lors qu'une palette vide se présente dans un des tampons de l'îlot 2 (P3 ou P5), auquel E appartient, ce

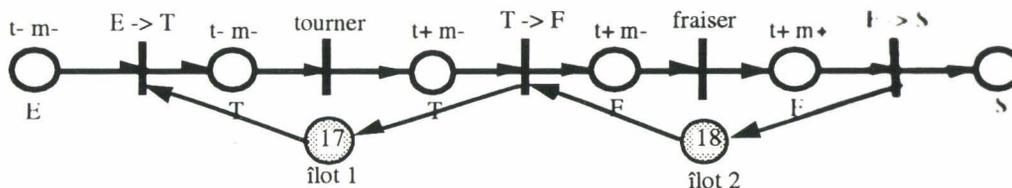
produit peut être chargé. Ceci conduit à la saturation de l'îlot 1 donc à son blocage.

Pour pallier ce problème, nous proposons une représentation de la contrainte de l'encours au niveau des gammes opératoires. La représentation d'un encours maximal de chaque îlot s'intègre de manière naturelle dans le modèle des gammes opératoires, car elles disposent d'une vue abstraite du processus de fabrication.

Le mécanisme de représentation choisi consiste en l'ajout de places de type "protocoles d'accès" au niveau de chaque gamme. A cette fin nous établissons dans un premier temps les équivalences entre un produit décrit par sa gamme et son type tel qu'il est représenté sur les modèles des îlots. Les produits décrits par la gamme opératoire P1 de l'exemple illustratif (figure III.25), sont de type  $\langle cu \rangle v \langle t \rangle$ . Ils sont d'abord transformés sur le tour de la cellule, donc des produits de type  $\langle t \rangle$  et ensuite usinés sur Cu comme produits de type  $\langle cu \rangle$ .

Par l'intermédiaire des places ajoutées, nous limitons l'encours des produits d'un type donné à la borne supérieure de l'encours d'un îlot issue de l'analyse précédente. De ce fait, les places de type "protocole" sont connectées en aval et en amont des transitions modélisant un transfert vers leurs îlots respectifs.

La figure III.25 indique l'intégration de la gestion de l'encours au niveau de la gamme P1 de l'exemple illustratif. Les places "îlot 1" et "îlot 2" modélisent la gestion de l'encours de leurs îlots respectifs. Leur marquage initial correspond au marquage maximal de non-saturation selon notre approche.



**Figure III.25 : gestion de l'encours sur les gammes opératoires**

Dans le cas de l'îlot 1, la place "îlot 1" limite l'encours à 17 produits de type  $\langle t \rangle$ . Ceci correspond au niveau du modèle de l'îlot (figure III. 20), par exemple, à un marquage de la place P1 par 16 jetons de type  $\langle e,t \rangle$  et un jeton de type  $\langle e \rangle$ . La place T sera marquée dans ce cas par un jeton. Si la place "îlot 1" au niveau des gammes opératoires n'est plus marquée, le franchissement des transitions E->T de la gamme P1 est alors interdit. Nous ne pouvons donc pas injecter un produit  $\langle t \rangle$  localisé sur le tampon d'entrée. La gestion de l'encours au niveau des gammes opératoires et la collaboration des deux modèles garantit de cette manière un

comportement sans blocage de l'îlot .

Si l'encours de non-saturation de l'îlot est dépendant du mode opératoire du système, c'est le cas pour l'exemple de l'îlot avec ressource partagée, nous sommes alors amenés à représenter cette contrainte selon les différents points de vue.

La figure III.26 représente trois gammes opératoires :

- \* les produits de la gamme opératoire A sont fraisés sur CU1
- \* les produits de la gamme opératoire B sont tournés sur T1
- \* les produits de la gamme C sont tournés sur T1 et fraisés sur Cu1

où T1 et Cu1 sont les ressources de transformation de notre îlot avec ressource partagée.

Les produits de la gamme A vérifient donc la condition <cu>, les produits de la gamme B sont de type <t> et ceux de la gamme C vérifient les deux conditions, si nous utilisons la notation introduite dans le paragraphe 4.1.3.4. L'encours maximal pour un type de produit seul est donc de :

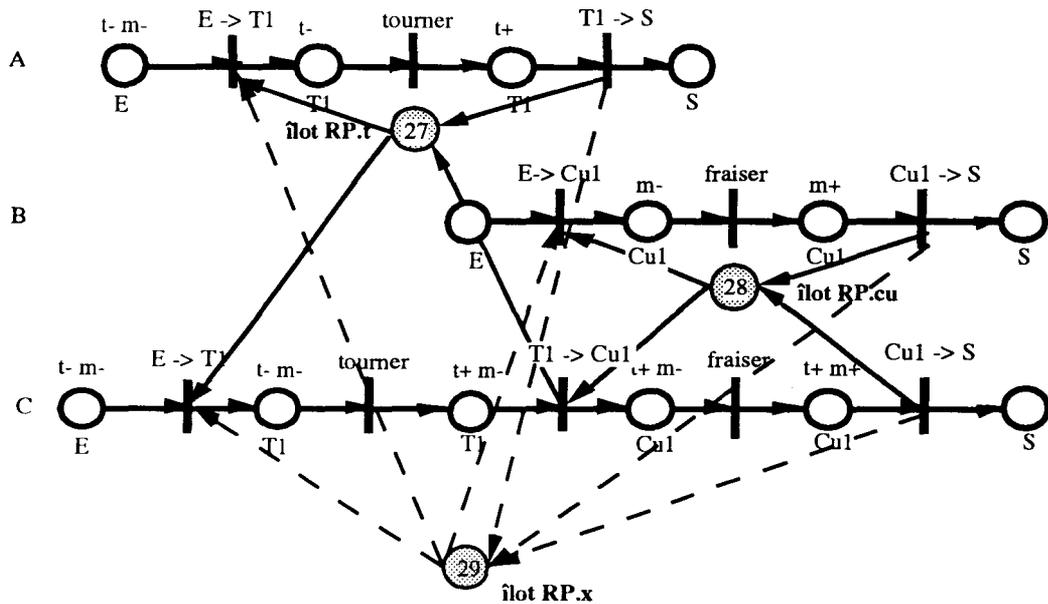
- \* 28 produits pour le type <cu>
- \* 27 produits pour le type <t> et
- \* 29 produits pour le type <x> (= <cu> v <t>).

Nous représentons, dans un premier temps, ces contraintes sur les gammes opératoires (figure III.26). Les produits de la gamme C sont d'abord de type <t> et ensuite de type <cu>. De ce fait nous les connectons aux deux places limitant l'encours par rapport à ce type de la manière précisée sur cette figure (cf places îlot RP.t pour le type <t>, et îlot RP.cu pour le type <cu>).

Si nous admettons par la suite une production mixte, c'est-à-dire un mode opératoire qui comporte plusieurs types de produits, les mécanismes introduits jusqu'alors n'empêchent pas un blocage de l'îlot. Les emplacements de stockage de l'îlot et du système de transport sont partagés entre les produits des différentes gammes.

Supposons que nous fabriquons des produits de la gamme A et de la gamme B en parallèle. Les places de type "protocole" permettent un encours de 55 produits (27 pour la gamme P2 et 28 pour la gamme P1) dans l'îlot, l'îlot ne comportant que 30 emplacements au total. Nous pouvons donc obtenir facilement un blocage de l'îlot en saturant les ressources avec des

produits des deux gammes.



**Figure III.26 : limitation de l'encours local dans le cas d'une production mixte**

Pour éviter ce blocage, nous intégrons les résultats obtenus lors de l'analyse de l'îlot par rapport aux produits de type <x>. Ce type de produits englobe les types <cu> et <t> et les produits de ce type sont, de ce fait, admis sur toutes les ressources de l'îlot. L'analyse de l'îlot dans un tel mode opératoire permet la définition d'une contrainte sur l'encours total dans l'îlot.

Nous intégrons donc cette contrainte au niveau des gammes opératoires de la manière précisée sur la figure III.26 (traits pointillés). Les produits de la gamme A vérifient la condition <cu>v <t> ainsi que les produits de la gamme B et C. De ce fait, toutes les gammes sont connectées à la place "îlot RP.x" représentant la limitation de l'encours total.

Les gammes opératoires représentent donc les contraintes relatives à la saturation d'un îlot. Cependant, la charge d'un îlot influence le comportement des autres, car les îlots partagent les emplacements de transport dans le système. Le franchissement de la transition T-> F au niveau de la figure III.25 restitue un jeton à la place îlot 1. Si une palette vide se présente maintenant devant le tampon de l'îlot 2, le produit localisé sur E peut être injecté dans le système. L'encours actuel dans le système correspond dans ce cas à 18 produits. Deux franchissements successifs de la transition T->F permettront ensuite l'injection de deux autres produits de type <t>. L'encours dans le système sera donc de 20 produits répartis de la manière suivante :

\* 2 produits sur le centre d'usinage

- \* 1 produit sur le tour
- \* 17 produits sur les palettes

Si les 17 palettes contiennent des produits qui ne sont pas encore finis, aucun produit ne pourra évoluer au niveau de sa gamme. Nous disons que le système est globalement bloqué. Une gestion globale de l'encours par rapport à un produit est donc nécessaire, elle tient compte des influences réciproques entre îlots. Une approche complémentaire pour la gestion de l'encours global dans le système sera présentée dans le paragraphe concernant la validation des gammes opératoires.

## 4.2. L'ANALYSE DU MODÈLE DE LA CELLULE

### 4.2.1. La Démarche de L'Étude des Propriétés

Le modèle de la cellule, deuxième composante du modèle du système de transport, est une traduction du système de transport central, qui représente l'évolution des ressources à transferts libres. A ce niveau de la hiérarchie du modèle du système de transport, les îlots ne sont représentés que par leurs tampons d'entrée et de sortie.

Chaque place du modèle du moyen de transport central modélise une localisation possible d'une ressource à transfert libre. Ces localisations, de capacité unitaire, ont été identifiées lors de l'analyse du système. La traduction du modèle issu de l'analyse (figure II.10 pour le convoyeur de l'exemple) en réseau de Petri mène, dans un premier temps, à un graphe d'état, où les transitions modélisent les transferts élémentaires réalisables sur cette ressource.

Rappelons qu'une cellule a été définie comme une composante fortement connexe du graphe des accessibilités du système, dont la forte connexité est établie par un seul moyen de transport central (chapitre II 4.3.2.3). Le moyen de transport vérifie donc une propriété de forte connexité pour l'ensemble du graphe reliant les lieux caractéristiques. L'ensemble de lieux caractéristiques, représenté par les nœuds du graphe des accessibilités, sont des lieux appartenant au moyen de transport. Ce sont, par exemple, les lieux de synchronisation formant les tampons d'entrée et de sortie des îlots. Ils sont représentés par des places sur le modèle réseaux de Petri. Le graphe d'état modélisant un moyen de transport central est déduit du graphe des accessibilités du système et hérite de ce fait de la propriété de forte connexité précédemment évoquée.

En phase de génération du modèle de la cellule, ce graphe d'état fortement connexe a été

enrichi par des places de type protocole limitant le marquage de chaque lieu à sa capacité unitaire. Le modèle final est construit à partir d'un modèle structurellement vivant, seule la saturation du moyen de transport peut alors conduire à un blocage.

Pour l'analyse des propriétés du modèle nous proposons une transformation du modèle de la cellule en un modèle qui conserve les informations quantitatives du modèle de départ. Cette transformation, appelée projection totale sur les places [Genrich 1986] (annexe 9), conduit à un réseau de Petri ordinaire.

Rappelons ici, que les jetons du modèle de la cellule caractérisent le même type d'objet : un jeton individuel correspond toujours à une ressource à transfert libre, la transformation des jetons individuels en jetons banalisés ne représente donc pas une perte d'information du point de vue de l'analyse des propriétés recherchées : tous les supports offrent en effet le même comportement vis-à-vis des transports.

Pour l'analyse du modèle obtenu, nous proposons d'utiliser la règle de tir de transition stricte, qui interdit le marquage d'une place par plus d'un jeton. Nous éliminons les protocoles d'accès et retrouvons un graphe d'état de type Condition / Événement. Ce graphe d'état est structurellement vivant. Le marquage maximal de non-saturation est alors égal au nombre des places du modèle moins un jeton, car le graphe d'état est fortement connexe et marqué.

#### 4.2.2. Un Exemple d'Application

La figure III.27 présente un extrait du convoyeur de la cellule de l'École Centrale de Lille.

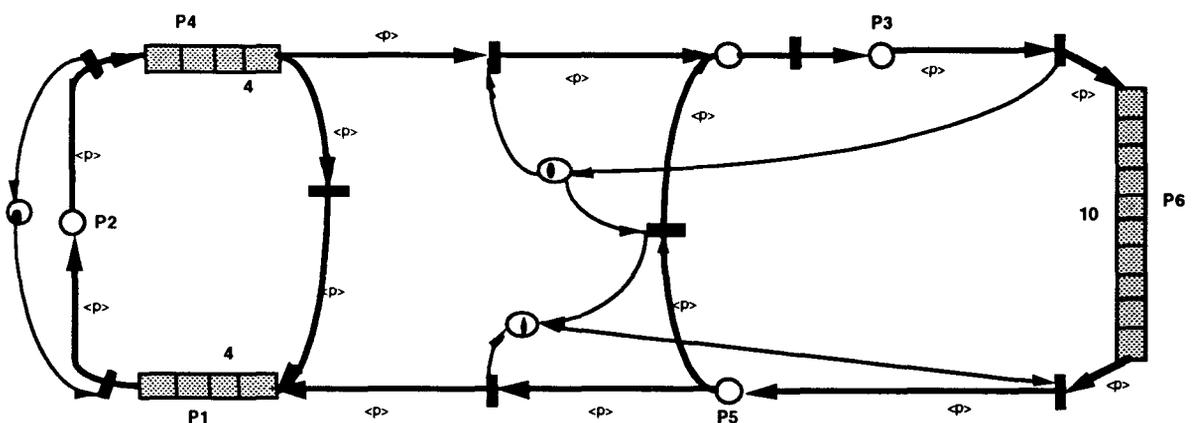


Figure III.27 : modèle du convoyeur (extrait)

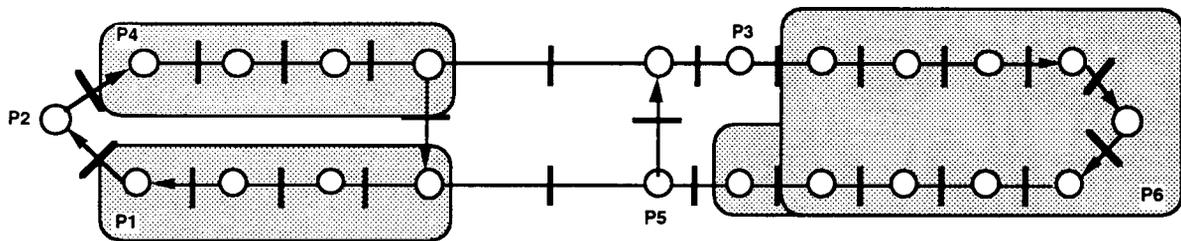
Les places P3 et P5 sont les tampons de l'îlot 2 et la place P2 est le tampon de l'îlot 1,

composé du robot R1 et du tour.

La projection totale substitue aux individus modélisant les palettes des jetons ordinaires. Le graphe réseau de Petri reste alors le même, seules les étiquettes des arcs sont enlevées. Les places sont alors marquées par des jetons banalisés.

Par construction, chaque place du modèle est de capacité unitaire. Cette capacité est représentée par les protocoles d'accès du modèle. Nous pouvons donc enlever ces protocoles et interdire le marquage d'une place par plus d'un jeton, comme l'exige la règle de tir stricte.

Le modèle résultant est un graphe d'état, fortement connexe (figure III.28). Ce graphe d'état est structurellement vivant (il existe un marquage pour lequel il est vivant).



**Figure III.28 : graphe d'état du convoyeur**

Si le marquage initial comporte autant de jetons que de places, nous obtenons la saturation du modèle, aucune transition ne peut être tirée. La libération d'une place quelconque permet de faire avancer un jeton. Du fait de la forte connexité du graphe on peut exécuter une séquence de tir qui contient toutes les transitions. Le modèle est donc vivant pour tout marquage strictement inférieur au marquage de saturation.

Le modèle de la figure III.28 comporte 22 places, son marquage de saturation est alors égal à 22 jetons. Nous pouvons donc admettre au maximum 21 palettes. Le convoyeur de la cellule de l'Ecole Centrale de Lille comporte 40 lieux de capacité unitaire. Le marquage de saturation correspondant est égal à 40 jetons. Puisque sur le convoyeur ne se déplacent que 17 palettes, le modèle est vivant.

### 4.3. CONCLUSION

L'analyse des deux composantes du modèle hiérarchique du système de transport a permis de définir les conditions de vivacité de chaque type de modèle. Puisque les flots manipulent des

produits unitaires, l'analyse a permis de définir un encours maximal pour chaque îlot, qui évite son blocage partiel.

Au niveau du modèle cellule, seuls les supports de transport sont vus par le modèle. Nous avons ainsi défini la quantité maximale de supports pour un système de transport évitant son encombrement total conduisant à un blocage du modèle. Les deux conditions proposées sont des conditions applicables d'une manière générale dans le contexte de la méthode de conception proposée. Le concepteur de la coordination peut donc définir à partir de l'analyse du système l'encours maximal pour chaque îlot et la quantité de ressources à transferts libres admissibles sur le moyen de transport central.

L'analyse des propriétés a été menée dans les deux cas d'une manière comparable. Partant du fait que les modèles étaient obtenus à partir d'un graphe d'état, nous avons transformé le modèle en un modèle équivalent permettant une analyse plus simple tout en conservant les propriétés comportementales du modèle d'origine.

## **5. L'ANALYSE DES GAMMES OPÉRATOIRES**

### **5.1. INTRODUCTION**

Les gammes opératoires sont construites à partir des graphes d'état pour les processus de fabrication manufacturière et des graphes d'événement pour les assemblages [Cruette 1991]. Les bonnes propriétés des réseaux de Petri appartenant à ces deux classes sont faciles à prouver et dans le cas des gammes opératoires, nous déduisons qu'elles sont à terminaison propre.

Les gammes opératoires ne sont pas des réseaux de Petri fortement connexes, la place finale n'a pas de successeur et la place initiale n'a pas de prédécesseur. Pour cette classe de réseaux de Petri, la propriété de terminaison propre correspond à la vivacité des réseaux fortement connexes. En fait, en reliant la place finale sur les places initiales d'un réseau à terminaison propre, le réseau résultant est vivant [Brams 1983].

La terminaison propre des gammes opératoires signifie, en outre, qu'un produit peut être fabriqué selon toutes les séquences représentées sur sa gamme (tous les états sont accessibles),

et qu'un jeton marquant la première place de sa gamme, finira par arriver dans la dernière, c'est-à-dire que'à partir d'un produit brut, nous obtenons, de manière sûre, un produit fini.

Cependant, les gammes opératoires ne comportent a priori aucune référence sur la capacité limitée d'un SFPM. L'intégration des protocoles d'accès aux îlots au niveau des gammes opératoires apporte une première solution à ce problème.

Cette solution n'est pas suffisante, car seul le point de vue local à chaque îlot y est pris en compte. Ainsi que nous l'avons présenté à l'aide de la gamme du produit P1 (4.1.5.), ce point de vue local ne peut pas empêcher un blocage global du système, d'autant plus qu'en règle générale notre système ne contient pas autant d'emplacements de stockage que l'ensemble des produits admis dans les îlots. Dans le cas de notre exemple, les places "îlot 1" et "îlot 2" permettent un encours de 35 produits (17 produits dans l'îlot 1 et 18 produits dans l'îlot 2), le système entier ne comportant que 27 emplacements pour tous les produits (17 palettes, 6 emplacements sur SA, 2 sur Cu, 1 sur T et le 1 sur Zo).

La validation des gammes opératoires porte donc sur deux aspects. Le premier aspect concerne la validation du modèle en tenant compte de la capacité limitée du système physique. Cette étape définit un mécanisme qui évite le blocage du système physique par saturation de ces ressources.

Le deuxième aspect porte sur la validation du modèle obtenu. En effet, l'ajout des places de type "protocole d'accès" modifie les caractéristiques du réseau. De ce fait, les bonnes propriétés du modèle ne peuvent plus être assurées. Nous présenterons une démarche de validation, qui permet d'assurer l'absence de tout blocage sur les gammes opératoires.

## **5.2. LA VALIDATION DES MODÈLES PAR RAPPORT À LA CAPACITÉ LIMITÉE DU SFPM**

Ainsi que nous l'avons énoncé précédemment, la saturation du système peut conduire à son blocage total. Nous recherchons, dans ce paragraphe, une borne supérieure de l'encours global du système qui évite cette situation. Nous proposons, dans ce but, une représentation du système qui permet la déduction d'un encours maximal. Nous indiquons, dans un deuxième temps, l'utilisation de cette représentation pour la définition d'un encours maximal.

### 5.2.1. Une Représentation du Système pour la Recherche des Situations Bloquantes

La capacité des ressources de l'atelier par rapport aux produits individuels est représentée de manière précise sur les modèles des îlots. Pour déterminer une borne supérieure de l'encours dans un système, nous proposons une analyse de l'ensemble des îlots le constituant.

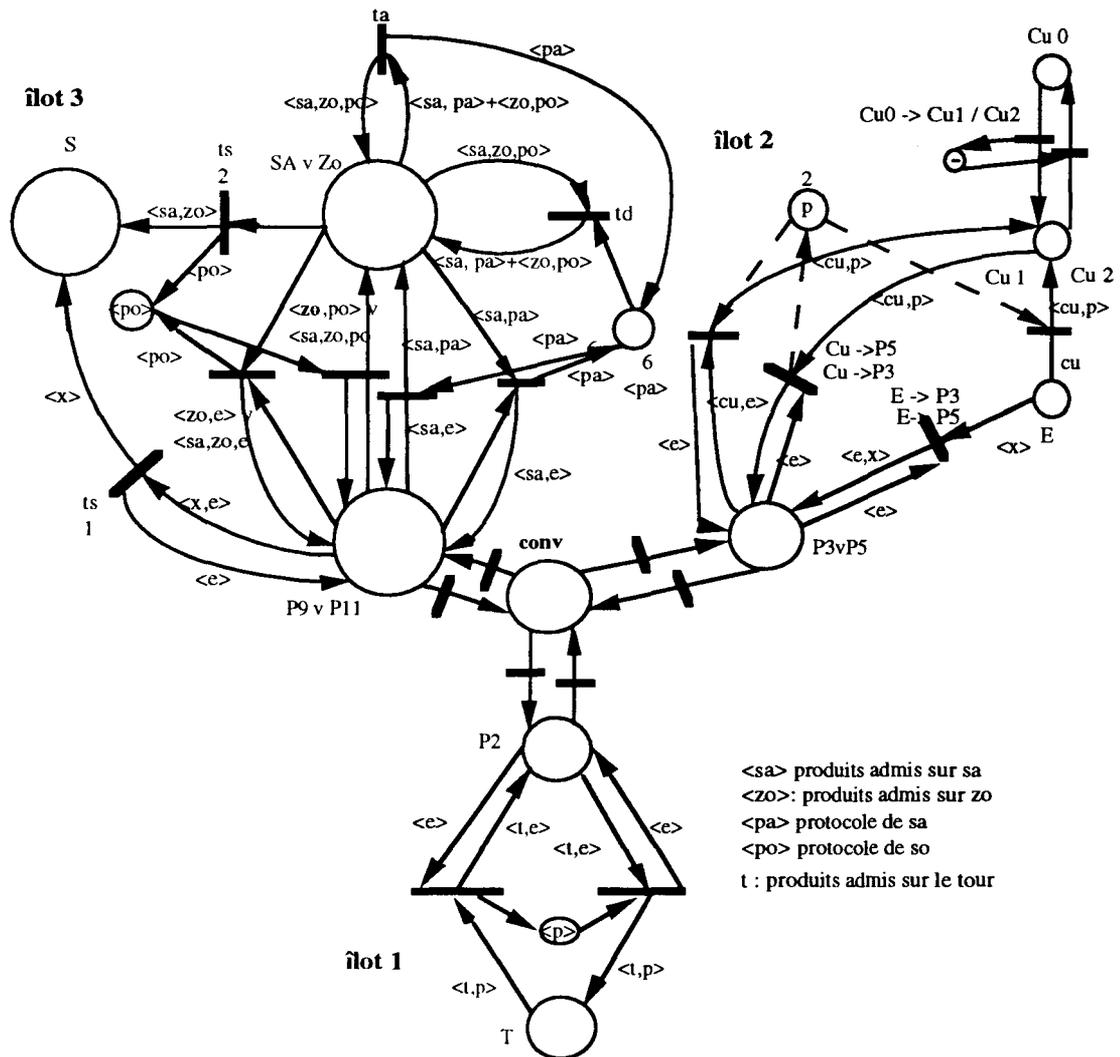


Figure III.29 : modèle de la cellule de l'Ecole Centrale de Lille

Afin d'obtenir une représentation concise qui comporte toute information relative à la capacité limitée du système nous connectons les modèles réduits des îlots par l'intermédiaire d'une place qui représente, de manière agrégée, la capacité limitée du système de transport. La figure III. 29 indique une telle représentation des îlots de la cellule de l'Ecole Centrale de Lille.

La place "conv" correspond au convoyeur central de cette installation.

La figure III.29 comporte le modèle réduit de l'îlot 3 de l'Ecole Centrale de Lille. La réduction du modèle de cet îlot est brièvement présentée en annexe 10. Rappelons toute fois les principales caractéristiques . L'îlot 3 est composé d'un stockeur d'assemblage (SA), d'une zone opératoire (ZO), d'un robot d'assemblage RA et du stock de sortie de la cellule S. Les tampons d'entrée et de sortie de l'îlot correspondent aux places P9 et P11. Les différentes ressources sont desservies par le robot portique (RP).

Le stockeur d'assemblage offre 6 emplacements de stockage pour les produits de type <sa>. Ce type de produit est posé lors de l'assemblage sur des produits de type <zo>. Les produits de type <zo> ne sont pas admis sur SA, et les produits de type <sa> ne peuvent atteindre la zone d'assemblage que s'il y a déjà un produit de type <zo>.

A partir de ces informations nous pouvons déterminer les conditions de vivacité du modèle suivant la règle de calcul énoncée en paragraphe 4.1.4. Nous appliquons cette règle aux modes opératoires suivants :

- \* mode opératoire 1 comportant des produits de type <sa> exclusivement,
- \* mode opératoire 2 comportant des produits de type <zo> et
- \* mode opératoire 3 comportant des produits de type <zo> v <sa>.

L'îlot dispose de 23 emplacements de stockage pour les produits de type <sa> répartis respectivement sur les 17 palettes du convoyeur et le stockeur d'assemblage (6 emplacements). L'encours de non-saturation est donc égal à 22 produits pour le premier mode opératoire.

Les 17 palettes sont également accessibles pour les produits de type <zo>, ainsi que l'unique emplacement de la zone opératoire. La borne supérieure qui assure la vivacité du modèle est alors égal à 17 produits de type <zo>.

Concernant le mode opératoire 3, l'ensemble des ressources est accessible, l'encours de non-saturation correspond dans ce cas à 23 produits.

La figure III.30 indique l'intégration de ces contraintes au niveau des gammes opératoires de l'exemple proposé. Nous remarquons que les produits de la gamme P21 vérifient la condition <sa> et les produits de la gamme P22 la condition <zo>. La contrainte issue de l'analyse du mode opératoire 3 pour l'îlot 3 est représentée par la place "îlot 3".

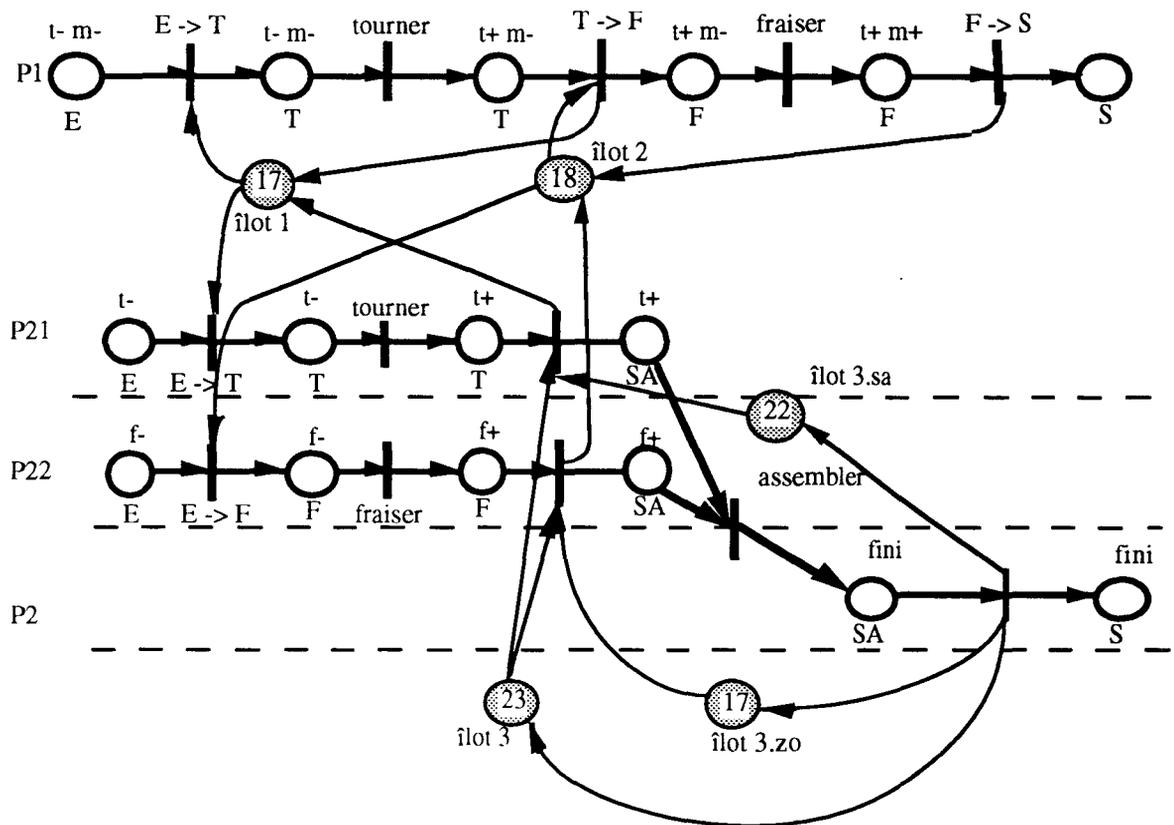


Figure III.30 : gammes opératoires avec représentation des contraintes de non blocage issues de l'analyse des îlots 1, 2 et 3

### 5.2.2. L'Analyse du Modèle agrégé d'une Cellule

Le modèle de la cellule de l'Ecole Centrale de Lille composé des modèles des îlots est du même type que le modèle d'un îlot. Les places représentent les lieux physiques capables de recevoir un produit, les transitions représentent des relations d'accessibilité entre ces lieux. De ce fait, nous pouvons appliquer la démarche proposée précédemment pour la validation des îlots à la validation globale du comportement du SFPM.

#### Etape 1 : Définition d'un Mode Opérateur Extrême

Le modèle de la figure III.29 fait référence à quatre types de produits différents <cu>, <t>, <zo>, <sa>. Chaque type de produit est admis sur un sous-ensemble des ressources de la cellule. Selon la démarche exposée, nous sommes amenés à analyser le modèle dans les modes opératoires suivants :

- \* quatre modes opératoires comportant chacun un type de produit exclusivement,
- \* six modes opératoires comportant chacun deux types de produits, comme par exemple les combinaisons des types  $\langle cu \rangle v \langle t \rangle$ ,  $\langle cu \rangle v \langle zo \rangle$ ,
- \* trois modes opératoires des combinaisons de trois types de produits comme, par exemple,  $\langle cu \rangle v \langle t \rangle v \langle zo \rangle$ ,
- \* un mode opératoire de l'ensemble des types.

Les modes opératoires ne comportant qu'un seul type de produit ont déjà été examinés, lors de l'analyse des îlots. A titre d'exemple, nous décrirons par la suite l'exemple de l'analyse du modèle dans un seul mode opératoire : le mode comportant des produits de type  $\langle y \rangle := \langle cu \rangle v \langle t \rangle$  uniquement. Les résultats obtenus permettront par la suite une généralisation.

### Etape 2 : Recherche du Marquage de Saturation

Nous remplissons, comme dans les exemples précédents le système avec des produits des modes opératoires définis.

Pour l'exemple du mode opératoire choisi, la place "conv" contiendra donc 17 jetons de type  $\langle e, y \rangle$ , la place Cu1 / Cu2 deux jetons du type  $\langle y, p \rangle$  et la place T un jeton de type  $\langle y, p \rangle$ . Dans cette situation, qui comporte 20 produits, le modèle comporte un blocage local, comme nous pouvons le vérifier sur la figure III.29.

Mais ce blocage ne concerne que des transitions dont les arcs font référence à des produits qui n'appartiennent pas au mode opératoire examiné. Les transitions franchissables par les produits de type  $\langle sa \rangle$  et  $\langle zo \rangle$  ne sont pas tirables, car les produits du mode opératoire défini ne vérifient pas la condition exprimée par ces arcs. Les produits de type  $\langle sa \rangle$  ou  $\langle zo \rangle$  appartiennent à un autre mode opératoire.

En ce qui concerne les autres transitions, elles sont vivantes. Si nous sortons un produit de l'atelier par le franchissement de la transition ts 1 ou de la transition ts 2, nous trouvons une séquence de tir qui contient toutes les transitions de cette partie du modèle et qui conduit au marquage du modèle obtenu **après** le franchissement de la transition ts. La partie du modèle, qui concerne les produits de type  $\langle y \rangle$  est alors vivante.

Cependant, le franchissement d'une des transitions ts 1 ou ts 2 a réduit la quantité des produits effectivement présents dans le système à 19. Nous en déduisons, que la partie examinée du modèle est vivante, si nous limitons l'encours à 19 produits de type  $\langle y \rangle$ .

Ce résultat est identique à celui que nous avons obtenu pour l'analyse des îlots : le modèle est vivant pour un mode opératoire donné, si nous limitons l'encours au marquage de saturation décrétementé d'un jeton.

Nous pouvons donc appliquer la règle de calcul énoncée pour la détermination de l'encours non bloquant dans un îlot au cas du système global. Ceci se justifie pour le fait que le modèle analysé est du même type que les modèles des îlots. Le marquage maximal de non-saturation du modèle d'un atelier composé des modèles réduits des îlots par rapport à un mode opératoire est donc égal au marquage de saturation du modèle décrétementé d'une unité. La saturation du modèle correspond à la saturation de toutes les ressources admises pour les produits composant le mode opératoire.

Cette règle permet de déterminer, pour notre exemple, les bornes suivantes :

- \*  $\langle cu \rangle v \langle zo \rangle$  : 19 produits
- \*  $\langle t \rangle v \langle sa \rangle$  : 23 produits
- \*  $\langle t \rangle v \langle sa \rangle v \langle zo \rangle v \langle cu \rangle$  : 26 produits.

### 5.2.3. La Représentation des Résultats sur les Gammes Opératoires

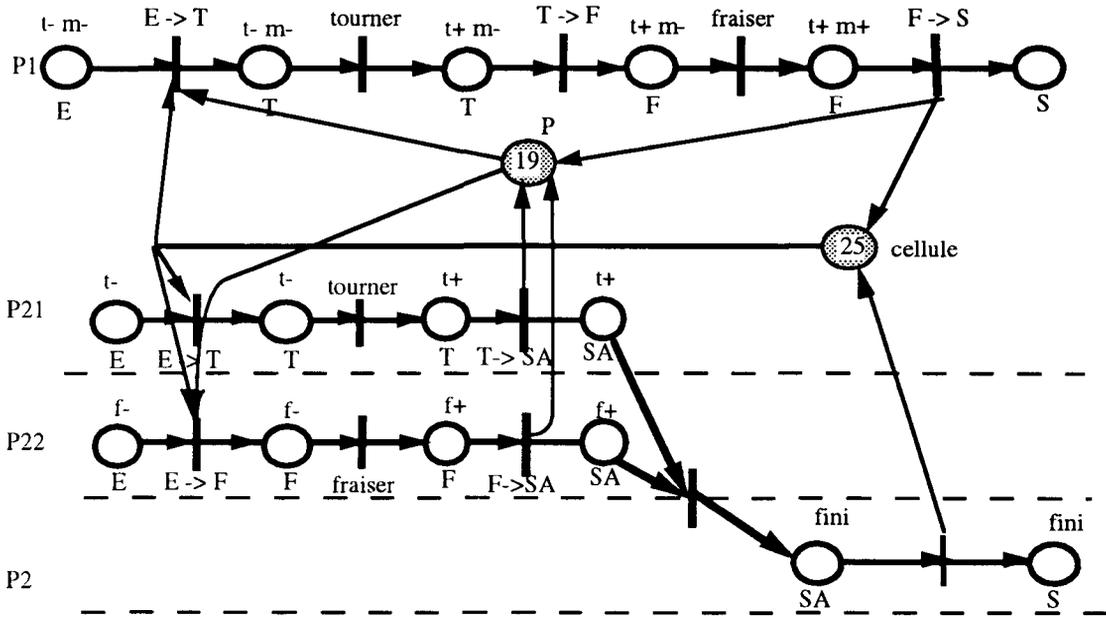
Afin d'intégrer les résultats de l'analyse précédente au niveau des gammes opératoires, nous établissons dans un premier temps les équivalences entre les types des produits et les produits représentés par leur gammes.

Nous représentons dans un deuxième temps les contraintes sur l'encours des produits par des places de type "protocole" connectées en amont et en aval des transitions de transfert.

La condition  $\langle cu \rangle v \langle t \rangle$  est vérifiée par exemple par les produits des gammes P1. Les produits de cette gamme sont d'abord de type  $\langle t \rangle$ . Nous connectons donc une place représentant l'encours maximal de 19 produits de type  $\langle cu \rangle v \langle t \rangle$  (place P de la figure III.31) en amont de la première transition de cette gamme. Les produits de cette gamme changent de type par le franchissement de la transition T->F. Ils deviennent des produits de type  $\langle cu \rangle$ . Puisque les produits de ce type vérifient toujours la condition  $\langle cu \rangle v \langle t \rangle$ , nous restituons les jetons à la place P1 seulement après le franchissement de la transition F->S modélisant la sortie d'un produit fini.

Les produits des gammes P 21 et P22 vérifient aussi la condition  $\langle cu \rangle v \langle t \rangle$ . Ils sont d'abord de type  $\langle t \rangle$  et de type  $\langle cu \rangle$  respectivement. Cependant, ils changent de type par le franchissement des transitions T->SA et F->SA. De ce fait, la place P est connectée en aval de ces transitions modélisant ainsi la restitution d'un jeton.

La place “cellule” représente la limitation de l’encours global dans le système. Les produits de chaque gamme doivent vérifier la condition  $\langle cu \rangle v \langle t \rangle v \langle sa \rangle v \langle zo \rangle$  à tout instant, cette place est donc connectée en amont des premières transitions et en aval de la dernière transition de chaque gamme.



**Figure III.31 limitation de l’encours sur les gammes opératoires**

Remarque : Nous avons supprimé les places limitant l’accès aux îlots sur cette figure.

Ces différentes places traduisent les contraintes sur la capacité limitée du système physique exprimée au niveau des gammes opératoires. Ces bornes garantissent le non-blocage du système et permettent un encours maximal indépendamment du mode opératoire choisi.

Cependant l’analyse du système selon les différents modes opératoires paraît fastidieuse et l’expression des contraintes issues de cette analyse sur les gammes opératoires est assez compliquée. Puisque la vivacité des modèles a été prouvée pour tout marquage inférieur au marquage de saturation, nous pouvons, par exemple, limiter l’encours global dans le système au minimum de l’encours admis par chacun des îlots.

Cette règle simplifiera la gestion de l’encours. Un encours faible augmente en plus les performances du système par rapport à des critères comme le temps de séjour des produits dans l’atelier. Cependant, un encours trop restreint peut nuire aux performances du système par rapport à des critères comme le taux d’engagement des machines [Bénassy 1987]. Une validation dynamique complémentaire devra permettre, dans ce cas, la définition d’un encours

optimal en fonction des productions demandées et des objectifs d'exploitation.

### 5.3. UNE DÉMARCHÉ POUR L'IDENTIFICATION DES BLOCAGES DES GAMMES OPÉRATOIRES

En introduction de ce chapitre, nous avons indiqué que l'ajout des places de type "protocole" peut modifier les bonnes propriétés des gammes opératoires issues de la phase de génération. Afin de valider les gammes opératoires (modifiées), nous nous référons à la méthode d'identification d'une catégorie particulière de blocages formulée dans [Cruette 1991]. Cette démarche consiste à rechercher les configurations de marquage des gammes opératoires qui induisent leur blocage.

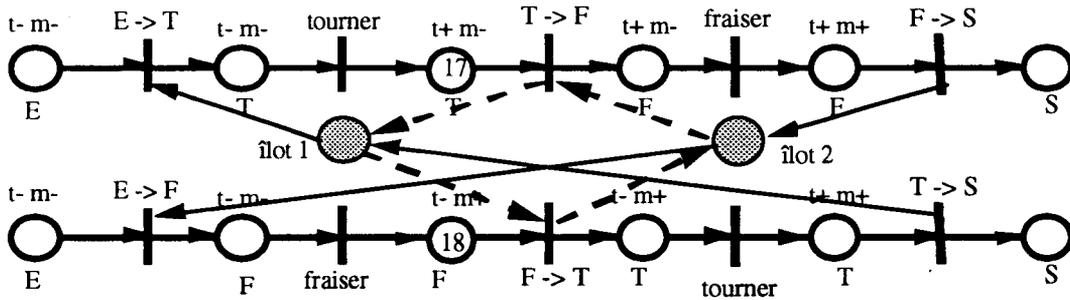
L'application des résultats proposés par CRUETTE se limite aux cas pour lesquels les places de type "protocole d'accès" vérifient une contrainte de modélisation. Détaillons la connexion des protocoles, que nous avons choisie, à l'aide de l'exemple de la figure III.32. Une marque qui franchit la transition  $T \rightarrow F$  restitue un jeton, modélisant un emplacement de stockage libéré dans l'îlot 1, et prend un jeton d'accès à l'îlot 2. Les protocoles d'accès à l'îlot suivant sont connectés sur la même transition qui restitue les jetons protocoles de l'îlot précédant.

En raison de cette contrainte de modélisation, les conditions suffisantes d'apparition d'un blocage local sont rappelées ci-après [Cruette 1991] :

- (a) les places modélisant les protocoles d'accès forment un circuit à travers les transitions des gammes opératoires. La principale caractéristique de ces circuits est qu'ils ne comportent aucune place appartenant à la gamme opératoire,
- (b) toutes les places modélisant les protocoles sont vides,
- (c) la restitution d'un jeton alloué se fait par une transition appartenant au circuit.

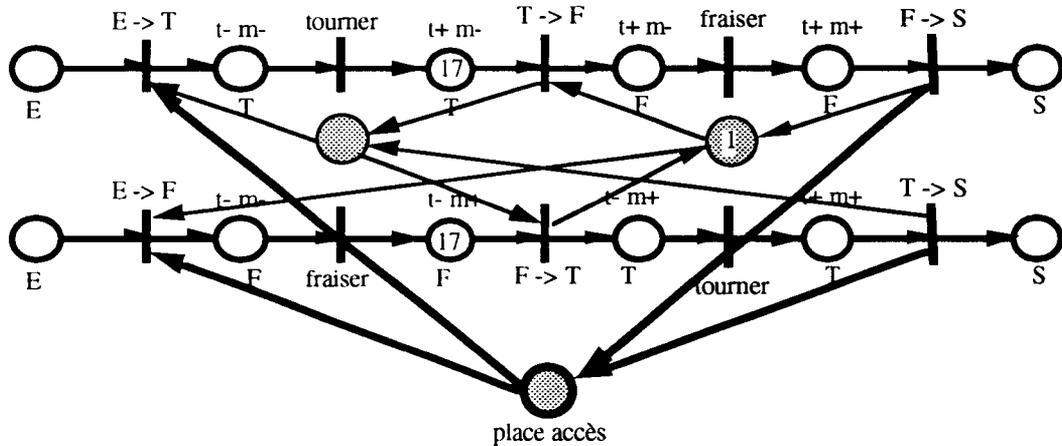
Sur les gammes opératoires de l'exemple proposé dans notre mémoire, il n'existe pas en fait de tels circuits. La figure III.32 précise la situation d'un blocage local sur un exemple d'illustration. Il s'agit, en effet d'un blocage de type "ressources imbriquées". La transition  $T \rightarrow F$  et la transition  $F \rightarrow T$  ne sont pas tirables.

Le circuit indiquant le blocage est mis en évidence par les traits pointillés. Les places "îlot1" et "îlot2" modélisant les protocoles sont vides, leur marquage initial comporte 17 et 18 jetons respectivement.



**Figure III.32 : blocage des gammes opératoires**

Afin d'interdire ce blocage local, nous limitons le marquage entre les transitions d'entrée (E->T et E->F) et de sortie (F->S et T->S) du circuit en rajoutant une place, qui gère l'accès à cette partie du modèle potentiellement bloquante. Le marquage initial de cette place contient la somme des jetons dans les places d'accès aux îlots moins un jeton. Une saturation partielle de cette partie du modèle est ainsi évitée. La figure III.33 intègre cette place dans la figure III.32.



**Figure III.33 : éviter le blocage**

La figure III.33 indique aussi le fonctionnement de cette place "accès". Elle est initialement marquée par 34 jetons et limite de ce fait le marquage dans la partie bloquante à 34 jetons. Dans la situation représentée ci-dessus, la transition T->F reste ainsi franchissable, le modèle évolue sans blocage.

Remarquons, de plus, que du fait de la connexion de ces places gérant l'accès aux parties potentiellement bloquantes du modèle à travers des transitions de transfert, les réseaux ainsi obtenus vérifient la contrainte de modélisation énoncée. Nous pouvons donc continuer la recherche des circuits et définir ainsi de manière récursive les contraintes de marquage maximal pour le modèle global.

La recherche de ces circuits permet la détection des blocages du type “ressources imbriquées” qui peuvent apparaître au niveau des gammes opératoires. La configuration des gammes opératoires conduisant à une telle situation est détectée par la recherche de ces circuits. Le mécanisme proposé par CRUETTE interdit le blocage du modèle, les gammes opératoires sont donc à terminaison propre.

## 6. CONCLUSION

Nous avons présenté dans ce chapitre une démarche de validation formelle des modèles de la commande de coordination. L’analyse a été effectuée de manière ascendante et nous avons exploité au mieux les propriétés de modularité et de structuration hiérarchique des modèles.

Les modèles de coordination des ressources, les modèles des ressources et les graphes de synchronisation sont des graphes d’état, vivants et saufs.

Le modèle du système de transport est lui-même un modèle hiérarchique. Pour l’analyse de la première composante, constituant les modèles des îlots, nous avons proposé une démarche de réduction des modèles initiaux en un modèle sémantiquement équivalent par transformations d’équivalence. A partir de ce modèle nous avons pu déterminer la condition de vivacité de chaque îlot.

Le regroupement des places conservant un comportement identique a permis de définir un modèle abstrait des îlots. Ce modèle générique comporte toutes les configurations possibles d’un îlot. L’application de la démarche de validation à ce modèle nous a conduits à la généralisation des résultats. Nous avons ainsi énoncé une règle d’évaluation de l’encours de non-saturation de chaque îlot.

Comme pour les modèles des îlots, le modèle du moyen de transport central est un graphe d’état, auquel ont été ajoutés des protocoles d’accès bornant le marquage de chaque place à la capacité de stockage du lieu physique qu’elle modélise. La transformation du modèle en graphe d’état nous a permis la définition d’un encours de non-saturation du point de vue transport.

Les gammes opératoires donnent le point de vue de plus haut niveau d’abstraction du processus de fabrication, mais elles ne comportent pas de référence sur les capacités limitées

des ressources du système. Nous leur avons donc confié la gestion de l'encours local et de l'encours global du système. Les résultats d'analyse des îlots sont alors transposables à ce niveau.

Cependant, l'intégration des places de type "protocole d'accès" modifie les caractéristiques des gammes opératoires et peut notamment mener à des blocages du modèle. Nous avons mis en évidence dans ce but un mécanisme évitant le blocage des gammes opératoires à partir de la recherche de circuits.

Concernant la gestion de l'encours global du système, nous avons proposé une représentation du système issue des modèles réduits des îlots. Nous avons déduit de cette représentation la condition de non-saturation du système garantissant un comportement sans blocage indépendamment du mode opératoire choisi.

Cependant, cet encours n'est pas forcément optimal en termes d'exploitation du système. Une démarche complémentaire consiste alors en la validation du comportement dynamique du système et de sa commande.

La validation du comportement dynamique, en général, correspond à l'évaluation des performances du système, face à différents scénarii, c'est-à-dire différents modes opératoires, points de fonctionnement ou modes transitoires [Castelain 1987].

Une étape d'évaluation de performances permettra alors la définition d'un encours en fonction du mode opératoire choisi, qui satisfasse au mieux les critères et objectifs de production du système. Cette évaluation peut être effectuée sur les modèles de coordination des produits.

Pour l'étude du comportement dynamique, la simulation est un outil couramment utilisé [Cavaille et Proth 1989]. Nous présentons dans l'annexe 11 les principales caractéristiques d'un simulateur adapté à nos modèles [Ausfelder et al 1992] et que nous avons développé dans une première phase de notre travail de recherche.

# CONCLUSION GÉNÉRALE :

## RÉSUMÉ ET PERSPECTIVES

Dans ce mémoire, nous avons développé une méthode de conception de la commande de coordination au sein d'un système de conduite pour les ateliers flexibles. Cette méthode mène à un ensemble de réseaux de Petri, qui ont par construction de bonnes propriétés.

Dans le premier chapitre, la modélisation de la conduite nous a permis de définir, d'une part, les différentes tâches de la commande de coordination, coordination par rapport aux produits et coordination par rapport aux ressources, et, d'autre part, les besoins de cette activité, issus d'une analyse des inter-relations et des inter-connexions avec les autres activités. Nous avons résumé ces besoins sous les termes d'adaptabilité et de sûreté de la commande de coordination.

Dans le deuxième chapitre nous avons développé des modèles adaptés à la coordination par rapport aux produits :

- \* les gammes opératoires pour la coordination globale du processus de fabrication par une approche modélisant l'évolution des produits comme une séquence de transformations et de transports,
- \* le modèle du système de transport qui représente le flux des produits dans l'atelier par un modèle réseau de Petri hiérarchisé,

ainsi que des modèles de la coordination par rapport aux ressources :

- \* les graphes de synchronisation
- \* et les modèles des ressources,

qui mettent en évidence le caractère réversible des opérations positionnelles.

La conception de ces quatre modèles s'effectue en parallèle et les méthodes présentées

permettent une automatisation du processus de conception. Les différents types des modèles sont liés dans une hiérarchie basée sur la décomposition de l'information. Ces liens sont exploités pour la communication entre les modèles, qui est de type client-serveur. La définition des modules génériques et leur insertion de manière méthodique permettent une automatisation de cette deuxième phase de la conception des modèles.

La modélisation du comportement commandé d'un atelier flexible par réseau de Petri et la déduction de la commande de coordination à partir de ce modèle est une démarche retenue par différents laboratoires de recherche. Les références dans l'introduction du chapitre II en sont le témoignage. L'avantage de la méthode de modélisation présentée réside, d'une part, dans sa rigueur dès la phase de spécification, qui définit les entrées de la génération et leur format, d'autre part dans ses résultats. Les modèles réseaux de Petri générés avec la méthode proposée sont vivants par construction ; leur vivacité peut être obtenue par des calculs simples qui conduisent à éviter le blocage par saturation d'une partie ou de la totalité du système.

Les modèles de la commande de coordination forment la base de données pour les activités de caractère décisionnel de la conduite. Leur utilisation pour le recouvrement et l'allocation des ressources a été exposé brièvement à la fin du chapitre II. Les modèles de la coordination correspondent ainsi à un "noyau informationnel" d'un système de contrôle / commande.

La conception de ces deux activités, basée sur les informations mises à disposition par les modèles de la coordination et compatible avec la représentation réseau de Petri des données, permettra la définition d'une démarche cohérente pour la conception de la conduite des ateliers flexibles. C'est dans cet esprit que nous comprenons le modèle de la conduite, présenté dans le chapitre I comme une spécification des besoins pour le développement d'un logiciel de conception assistée d'un système de contrôle/ commande.

# ANNEXES

## ANNEXE 1 : LES ACTIGRAMMES DE SADT

La conception d'un système complexe, tel qu'un système de contrôle / commande, demande des modèles de description afin de les comprendre [Ranky 1990]. La méthode SADT s'inscrit dans le cadre de la modélisation des systèmes et représente ce qu'un système doit faire en faisant abstraction du "comment" [I.G.L. 1989]. SADT est donc une méthode de spécification fonctionnelle, qui conduit à des modèles permettant la compréhension des interactions et des activités des différentes fonctionnalités présentes dans un système. Le formalisme graphique associé simplifie cette compréhension.

La méthode SADT analyse un système d'une manière imposée :

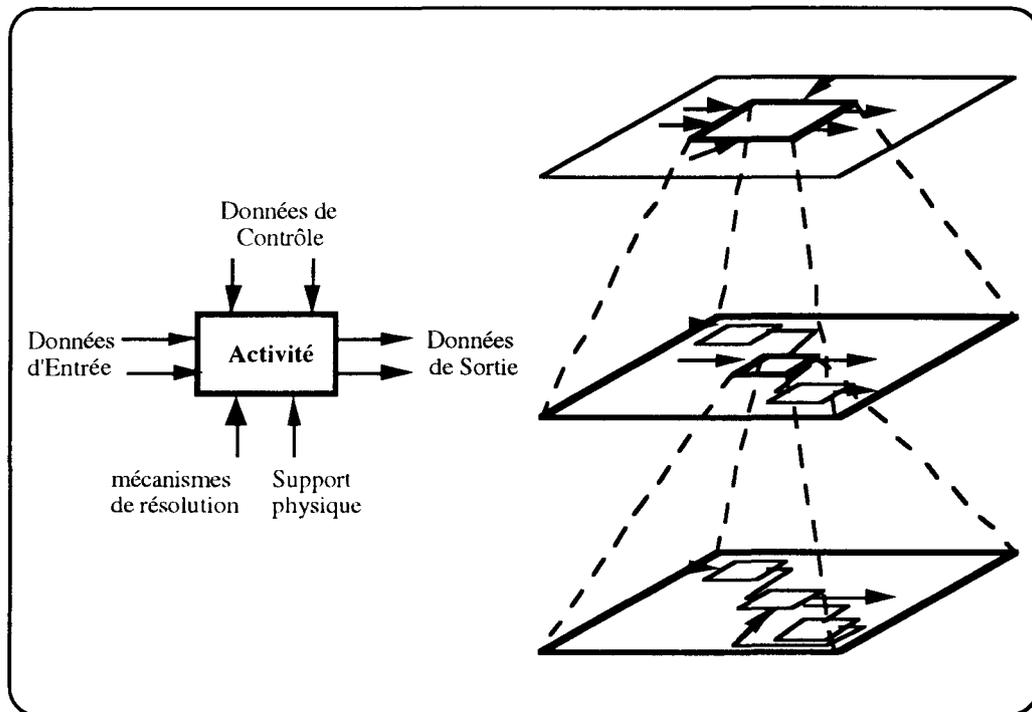
- \* descendante,
- \* modulaire,
- \* hiérarchique.

La description du système commence donc par le point de vue le plus global et le plus abstrait du système, et la méthode SADT le décompose en activités de moins en moins complexes. La méthode consiste à obtenir, à chaque phase de décomposition, des fragments maîtrisables par l'esprit humain. Cette démarche introduit graduellement des détails et trouve sa justification dans les concepts d'analyse (décomposition) et de synthèse (reconstitution) des systèmes.

SADT est particulièrement adapté au niveau conceptuel d'une spécification, et répond donc à nos besoins d'une description exhaustive des fonctions nécessaires à la conduite d'un atelier flexible. Comme nous nous intéressons davantage aux activités de ces fonctions, nous nous limitons à l'utilisation des ACTIGRAMMES du formalisme SADT.

Un ACTIGRAMME est la description d'une activité modélisée par une boîte. Cette boîte génère une ou plusieurs données en sortie. Elle transforme et modifie les données d'entrée avec des directives de contrôle qui représentent des contraintes d'évolution et qui ne sont pas modifiées. La figure A.1 montre ces primitives des ACTIGRAMMES et décrit la structure

hiérarchisée de la démarche. Au niveau des modèles de la conduite d'un SFPM, présentés dans le chapitre I, nous n'indiquons ni les mécanismes utilisés par chaque activité ni son support. La définition d'un support physique est spécifiée de plus en plus précisément en cours d'étude. Par exemple, une phase préliminaire peut décider qu'une fonction est réalisée par le pilote humain ou par un logiciel, et puis, en phase ultérieure de construction du système, un ordinateur sera spécifié comme support du logiciel réalisant cette activité.



**Figure A.1 : les primitives des ACTIGRAMMES**

De même les mécanismes de résolution dépendent de la solution adoptée pour la réalisation de la fonction. Au niveau de l'allocation des ressources, des algorithmes issus de la recherche opérationnelle ou des heuristiques peuvent être considérés comme mécanismes de résolution. Le modèle proposé peut être ainsi enrichi en cours de la conception de la conduite.

## ANNEXE 2 : PRÉSENTATION DE L'EXEMPLE

L'Exemple que nous avons choisi pour illustrer notre démarche est la cellule flexible de l'École Centrale de Lille (figure A2).

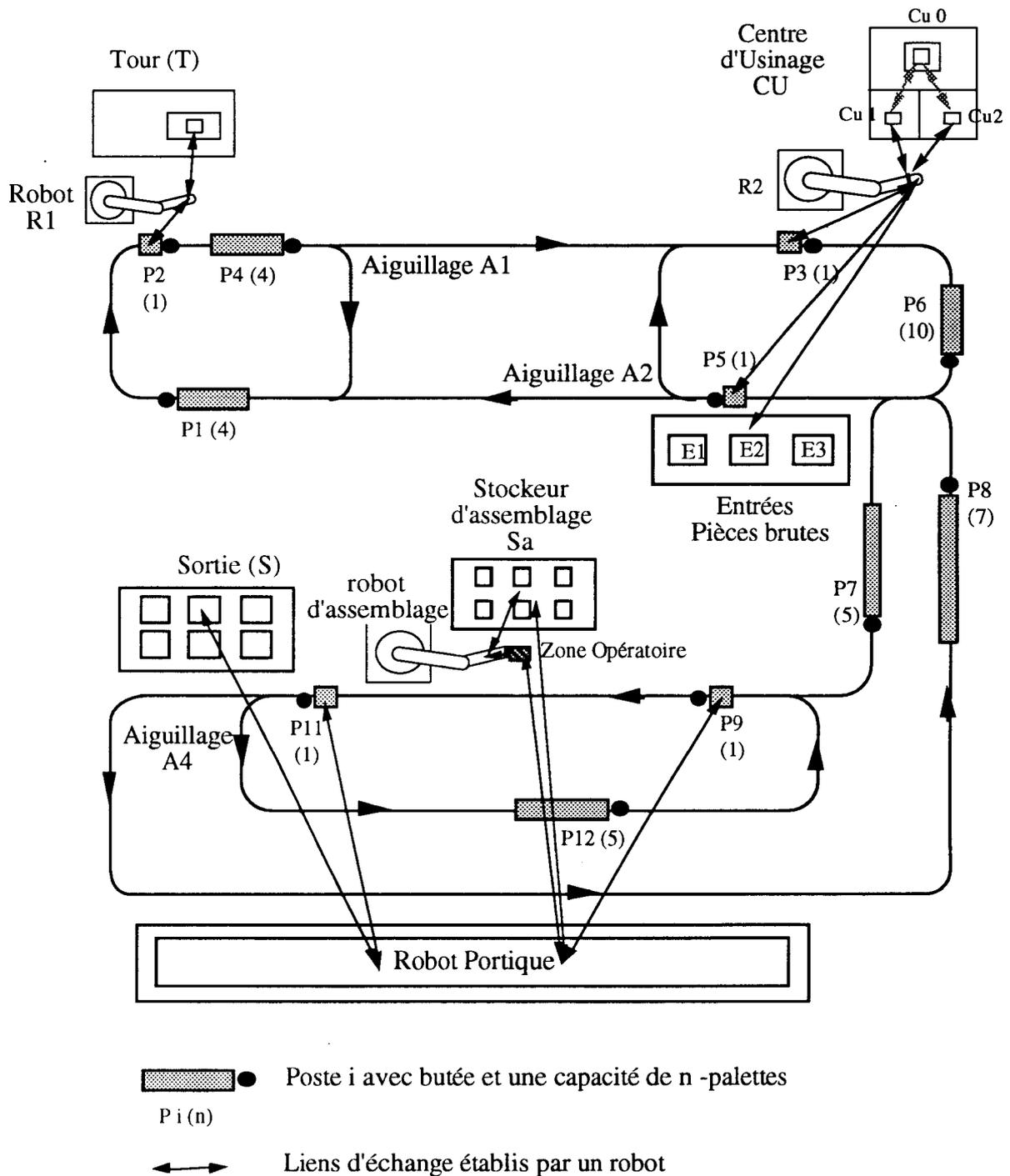
Autour d'un convoyeur central sont groupés un tour à commande numérique (T), un centre d'usinage (CU) une station d'assemblage ainsi que deux robots manipulateurs R1 et R2 et un robot portique (RP). Les pièces brutes arrivent dans le tampon d'entrée de la cellule E et les produits finis quittent l'installation sur le tampon de sortie S.

Sur le convoyeur central les produits se déplacent sur des palettes. Les palettes sont des palettes universelles, c'est-à-dire qu'elles transportent tout type de produit, mais un seul à la fois. Elles peuvent être arrêtées dans les sections P1 à P12 par des butées. Derrière chaque butée se forme une file d'attente de palettes dont la longueur est limitée. La taille maximale de la file dans chaque section est indiquée entre parenthèses pour chaque fifo.

Le centre d'usinage (CU) possède deux tampons d'entrée / sortie Cu 1 et Cu 2. Les pièces sont chargées sur ces tampons par R2 et manuellement bridées sur des palettes (chargement externe, sans synchronisation de R2). Les palettes transportent la pièce dans la zone opératoire de CU. Le chargement de Cu 1 et Cu 2 se fait à partir des zones P3 et P5 ou du tampon d'entrée de la cellule E, le déchargement sur des palettes qui sont arrêtées en P3 ou P5.

Outre l'alimentation de Cu, le manipulateur R2 sert également à transférer les pièces brutes du tampon d'entrée E vers P5 ou P3. Le tampon d'entrée est composé de plusieurs chargeurs, un pour chaque produit, sur lesquels les pièces brutes glissent automatiquement dans une position définie. A partir de cette position, R2 peut saisir les pièces.

Le robot R1 saisit les pièces arrêtées en P2 et les charge sur le tour T. Le tour est muni d'un système de sécurité d'accès, ce qui nécessite un chargement à l'intérieur du tour et une synchronisation des deux ressources. Les pièces usinées sont remises sur des palettes arrêtées en P2.



**Figure A2 : la cellule de l'École Centrale de Lille**

La station d'assemblage est formée par un robot d'assemblage (RA), son stockeur (Sa) et la zone opératoire (Zo). Le robot d'assemblage muni d'un préhenseur pneumatique prend des pièces stockées sur le stockeur d'assemblage (Sa) et les transporte sur des pièces arrivées dans sa zone opératoire (Zo). Ces pièces ont été posées auparavant sur la zone opératoire par le robot

portique (RP). Le robot portique peut accéder à des palettes arrêtées en P9 ou P11. Une pièce assemblée, comme tout autre produit fini, est transportée par le robot portique vers le tampon de sortie de la cellule (S).

Dans le chapitre II, nous proposons un regroupement des ressources en trois îlots :

- \* l'îlot 1, formé par le tour T et le robot R1 et son tampon d'entrée / sortie P2,
- \* l'îlot 2, formé par le centre d'usinage Cu, le tampon d'entrée de la cellule E et le robot R2. Les tampons de cet îlot sont les sections P3 et P5 du convoyeur,
- \* l'îlot 3, formé par la station d'assemblage, le tampon de sortie de la cellule S et le robot portique. L'îlot est accessible par ses tampons P9, P11.

Parmi les produits qui sont fabriqués dans cette installation nous nous intéressons essentiellement à deux types. Le produit 1 est d'abord tourné, puis fraisé et le produit 2 est composé de deux produits intermédiaires P21 et P22. Le produit P21 est tourné et le produit P22 est fraisé. Pour leur assemblage, le produit P21 est posé sur le produit P22.



### ANNEXE 3 : LES RÉSEAUX DE PETRI À PRÉDICATS / TRANSITIONS [GÉNRICH 1986]

**Définition** : Soit  $L$  un langage de premier ordre, et soit  $L_S$  les sous-langage, qui utilise seulement  $\Pi_S$ , les prédicats qui décrivent les relations statiques. La classe  $PRT_L$  est composée de réseaux marqués et annotés,  $MN = (N, A, M^0)$ , où  $N$  est le réseau sous-jacent,  $A$  son annotation dans  $L$ , et  $M^0$  sont marquage initial.

1.  $N$  est un réseau orienté ( $S, T, F$ ).
2.  $A$  est l'annotation de  $N = (A_n, A_s, A_t, A_f)$ , où
  - (a)  $A_n = R$  est une structure selon la logique de premier ordre pour  $L_S$ , qui est nommée support de  $MN$  (une sorte de légende);
  - (b)  $A_s$  est une bijection entre l'ensemble des places  $S$  et l'ensemble de prédicats variables  $\Pi_S$ ;
  - (c)  $A_t$  est une application de l'ensemble des transitions  $T$ , vers l'ensemble des formules (nommées sélecteurs des transitions), qui ne sont composées que d'opérateurs et de prédicats statiques, c'est-à-dire, qui sont dans  $L_S$ .
  - (d)  $A_f$  est une application de l'ensemble des arcs  $F$ , vers l'ensemble  $LC$  des sommes symboliques de uplets de termes de  $L$  : l'application,  $A_f(x,y)$ , qui associe à un arc  $(x,y)$  une place en amont ou en aval ( $x=s$  ou  $y=s$ ), est dans  $LC(n)$ ,  $n$  étant l'indice du prédicat annotant  $s$ .
3.  $M^0$  est le marquage initial des places : ce marquage est une application qui associe à chaque place  $s$  de  $S$  une somme symbolique d'uplets de constantes : si  $n$  est l'indice du prédicat annotant  $s$ , alors  $M^0(s)$  est contenu dans  $LC(n)$ , et la valeur de  $M^0(s)$ ,  $R(M^0(s))$ , est une combinaison linéaire de coefficients qui prennent la valeur 1 ou 0.



#### **ANNEXE 4 : LES HIÉRARCHIES DANS LES RÉSEAUX DE PETRI [HUBER ET AL 1990]**

Dans le domaine de la modélisation des systèmes, les inconvénients d'une représentation à un seul niveau du comportement du système sont bien connus :

- \* le manque de vue générale et concise du comportement global,
- \* la représentation du système qui est surchargée de détails,
- \* la structure du système qui ne se reflète pas sur le modèle.

La définition des hiérarchies pour les réseaux de Petri permet de combiner les avantages d'une approche hiérarchique de la représentation des systèmes, introduite déjà par exemple dans SADT, avec le pouvoir d'expression des réseaux de Petri et leur pouvoir de modéliser le comportement des systèmes à événements discrets.

La hiérarchie dans les réseaux de Petri est introduite par le remplacement d'un nœud (place ou transition) par un sous-réseau, nommé page, qui modélise le comportement du procédé de manière plus détaillée. Les mécanismes d'appel, inspirés par la programmation structurée, sont :

- \* transitions ou places de substitution
- \* transitions d'appel
- \* ensembles fusionnés de places ou de transitions.

Une place ou une transition de substitution remplacent un réseau de Petri sous-jacent. Chaque fois que la transition de substitution est tirée, elle est remplacée par sa page correspondante. Ceci correspond au remplacement de la transition par une macro-instruction dans la programmation structurée. Le marquage d'une place de substitution sollicite de la même manière l'exécution de la sous- page correspondante.

Les transitions d'appel sont l'équivalent d'un appel à un sous-programme. Plusieurs transitions peuvent faire appel en même temps à un même sous-réseau (même page). La page est insérée par cet appel et le passage des paramètres sous formes de jetons permet de déterminer un traitement spécifique.

Les ensembles fusionnés de nœuds permettent de replier de manière conceptuelle plusieurs

nœuds dans un seul. Cela correspond au pliage "classique" des réseaux de Petri de haut niveau.

## ANNEXE 5 : PROPRIÉTÉS DU GRAPHE DES ACCESSIBILITÉS DU SYSTÈME [KAUFMANN 1968]

**Définition Graphe Complet** : Soit  $G = (E, U)$  un graphe, il est complet si :

$$\forall X_i \in E, \forall X_j \in E (i \neq j) : (X_i, X_j) \notin U \Rightarrow (X_j, X_i) \in U$$

autrement dit, toute paire de sommets est moins reliée par un arc.

**Définition Graphe Symétrique** : Soit  $G = (E, U)$  un graphe, il est symétrique si :

$$\forall X_i \in E, \forall X_j \in E : (X_i, X_j) \in U \Rightarrow (X_j, X_i) \in U$$

autrement dit : si un arc existe dans un sens entre deux sommets, l'arc inverse existe aussi.

**Définition Sous-Graphe** : Soit  $G = (E, U)$  un graphe construit sur  $E$  et  $G_2 = (A, U_2)$  un autre graphe construit sur  $A$ , avec  $A \subseteq E$  tel que :

$$\forall X_i \in A \forall X_j \in A, (X_i, X_j) \in U \Rightarrow (X_j, X_i) \in U_2,$$

alors  $G_2$  est un sous-graphe. En termes moins mathématiques, on dira que le sous-graphe est construit avec un sous-ensemble de sommets et comprend tous les arcs relatifs à ce sous-ensemble.

**Définition Graphe fortement connexe** : Un graphe  $G = (E, U)$  est fortement connexe si de tout sommet  $X_i$  on peut atteindre tout autre sommet  $X_j$  en suivant un chemin du graphe.

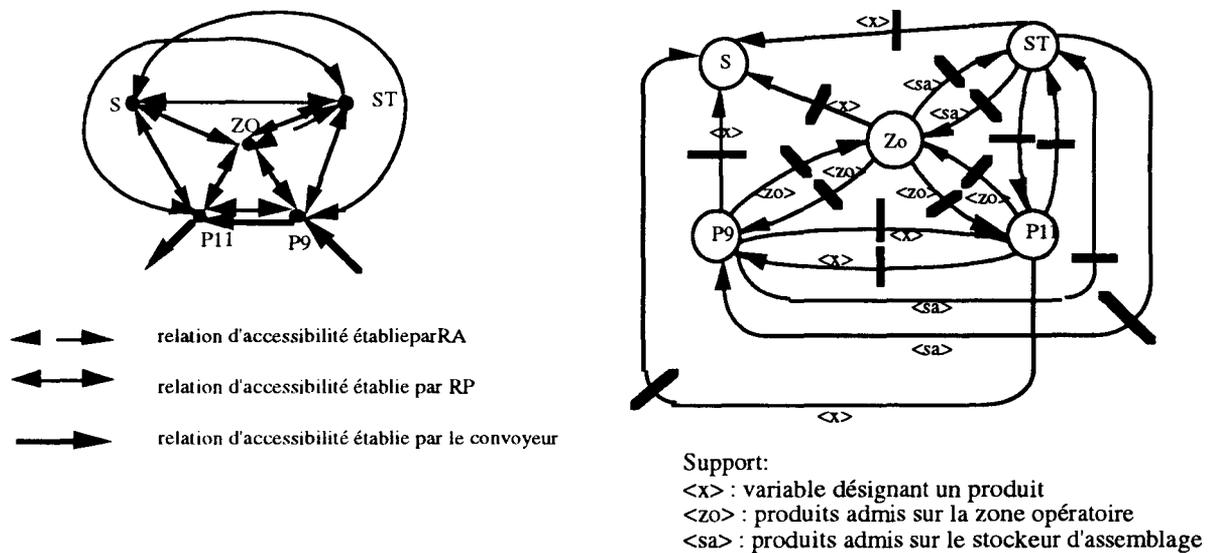


### ANNEXE 6 : GÉNÉRATION DU MODÈLE RÉSEAU DE PETRI DE L'ÎLOT 3 DE LA CELLULE

Afin de montrer l'application de la démarche proposée à un deuxième exemple, nous exposons par la suite la génération du modèle de l'îlot 3 de la Cellule de l'Ecole Centrale de Lille. Nous avons choisi cet exemple, car il permet d'illustrer la démarche dans le cas d'un îlot de type assemblage.

**Étape 1 : Traduction de la Composante du graphe correspondant à l'îlot 3 en réseau de Petri**

La figure A3 schématise cette première étape. Nous remarquons l'utilisation des variables testant la nature des produits. La variable <sa> modélise les produits, qui en cours d'assemblage sont posés sur les produits de type <zo> (cf. annexe 2). Seuls les produits <sa> sont admis sur le stockeur d'assemblage. Le robot d'assemblage RA réalise cet assemblage, qui consiste en un transfert élémentaire d'un produit <sa> sur un produit <zo>.



**Figure A.3 : génération de l'îlot 3 : étape 1**

Remarque : Nous avons supprimé les arcs sortants du tampon de sortie de la cellule.

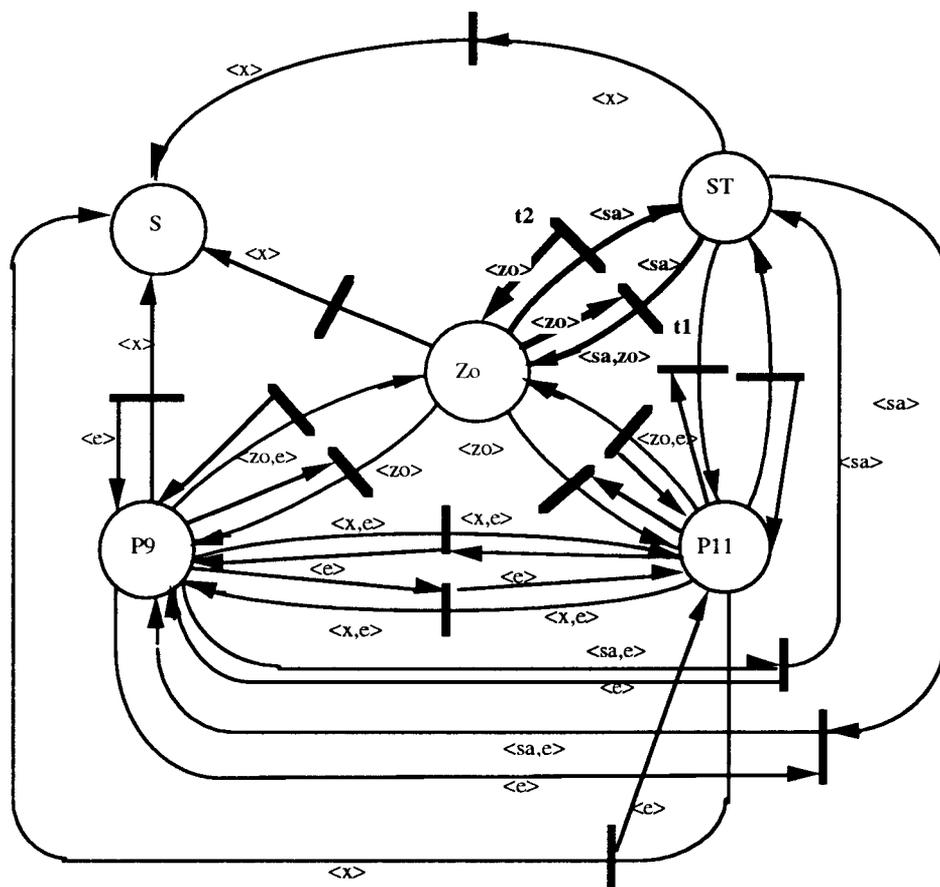
**Étape 2 : Intégration de la gestion des emplacements**

a) la gestion des emplacements sur les palettes

L'intégration de la gestion des emplacements sur les palettes s'effectue de la même manière que dans l'exemple de l'îlot 2 présenté dans le chapitre II. Le chargement d'une palette est modélisé par un arc supplémentaire étiqueté par <e> qui relie la place

représentant un tampon de l'îlot et sa transition amont indiquant le chargement (figure A.4).

L'assemblage des produits, c'est-à-dire le positionnement d'un produit <sa> sur un produit <zo>, est représenté de la même manière. La précondition d'un assemblage est la présence d'un produit <zo> sur la zone opératoire, l'assemblage génère un nouveau produit de type <sa,zo> (transition t1, figure A.4). Cette représentation permet de retracer les composants d'un produit assemblé et même inverser l'opération, (transition t2, figure A4).



Support:  
 <x> : variable désignant un produit  
 <zo> : produits admis sur la zone opératoire  
 <sa> : produits admis sur le stockeur d'assemblage  
 <e> : emplacement sur une palette

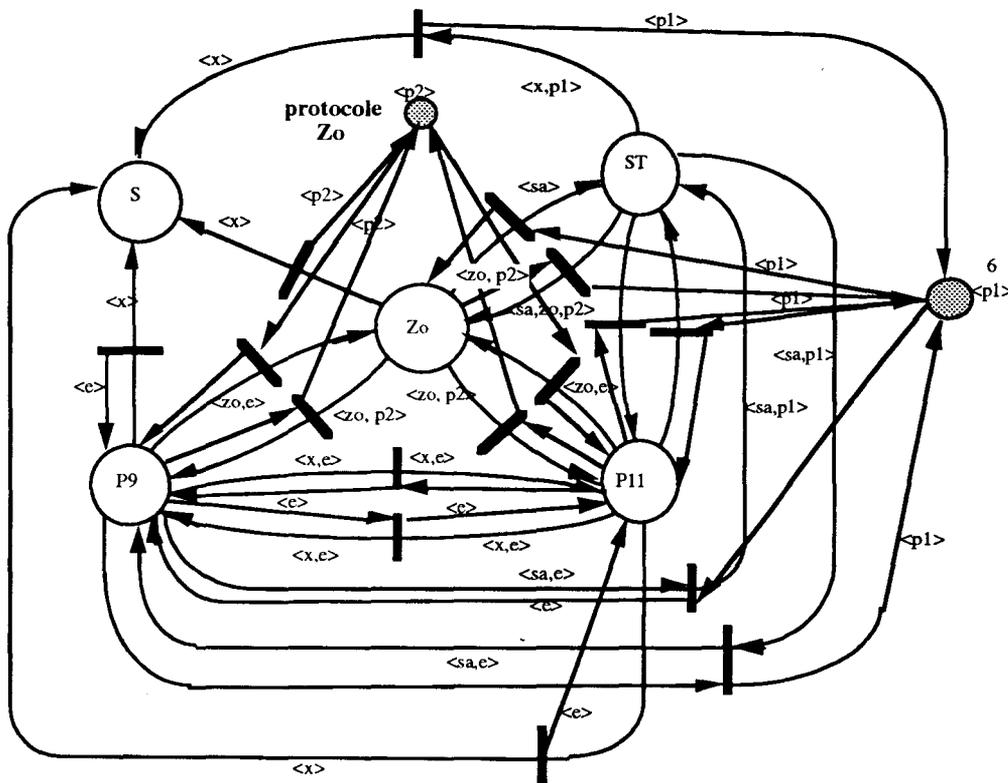
Figure A.4 : génération de l'îlot 3 : étape 2a

b) la gestion des emplacements des ressources de l'îlot

Le stockeur d'assemblage SA comporte 6 emplacements pour les produits de type <sa>.

Lors d'un assemblage un de ces emplacements est libéré, comme pour les autres transferts élémentaires modélisant un déchargement de la ressource.

La zone opératoire est de capacité unitaire. La gestion de son emplacement est indiquée sur la figure A.5 (place protocole Zo).



Support:  
 <x> : variable désignant un produit  
 <zo> : produits admis sur la zone opératoire  
 <sa> : produits admis sur le stockeur d'assemblage  
 <e> : emplacement sur une palette  
 <p1> : protocole d'accès à SA  
 <p2> : protocole d'accès à Zo

**Figure A.5 : génération de l'îlot 3 : étape 2b**

### Etape 3 : L'accès aux ressources partagées.

L'îlot 3 est un îlot complexe, qui comporte deux ressources partagées entre le robot RA et le robot RP : la zone opératoire Zo et le stockeur d'assemblage Sa. L'accès simultané à ces deux ressources par les deux robots doit donc être interdit. Nous représentons cette contrainte d'exploitation par deux places de type "mutex" une qui gère l'accès à Sa, l'autre qui gère l'accès à Zo. Les deux places sont connectées en amont et en aval de chaque transition modélisant un transfert vers ce deux ressources ou un transfert partant d'elles.

Le deuxième type de ressources partagées sont les deux robots. Eux aussi sont modélisés chacun par une place qui est connectée de la manière précisée dans le chapitre 2.



**ANNEXE 7 : DÉFINITIONS DES PROPRIÉTÉS RÉSEAU DE PETRI [BRAMS 1983]**

**Définition de la Vivacité :** Un réseau de Petri est vivant pour un marquage initial  $M_0$ , si toutes ses transitions sont vivantes. Une transition  $t_j$  est vivante, si pour tout marquage accessible du marquage initial, il existe une séquence de franchissement, qui contient la transition  $t_j$ .

**Définition de la Finitude :** Un réseau de Petri est borné pour un marquage initial  $M_0$ , si toutes les places sont bornées par  $M_0$ . Une place  $P_i$  est dite bornée par un marquage initial  $M_0$ , si pour tout marquage accessible à partir de  $M_0$  le nombre de marques dans  $P_i$  est fini.

**Définition de la Réinitialisabilité :** Un réseau de Petri a un état d'accueil  $M_a$  pour un marquage initial  $M_0$ , si pour tout marquage accessible à partir de  $M_0$ , il existe une séquence de franchissement, qui mène dans  $M_a$ . Un réseau de Petri est réinitialisable pour un marquage initial  $M_0$ , si  $M_0$  est un état d'accueil.



## ANNEXE 8 : DÉFINITIONS DES TRANSFORMATIONS D'ÉQUIVALENCE UTILISÉES [GENRICH 1989]

Les transformations d'équivalence permettent de transformer un modèle réseau de Petri à Prédicat / Transition dans un modèle sémantiquement équivalent. Elles sont réversibles et permettent de retrouver le modèle d'origine à partir du modèle transformé, car les deux réseaux appartiennent à la même classe abstraite des réseaux de Petri à Prédicat / Transition.

Parmi les dix-huit transformations définies par GENRICH nous ne nous servons que de deux d'entre elles pour obtenir le modèle réduit d'un îlot : le fusionnement des nœuds et la réécriture locale des variables.

**Définition** : Séparation / Unification des Nœuds :

- (a) Soit pour un nœud  $n$  le sélecteur  $r$  ( $r1 \vee r2$ ). Remplacer le sélecteur de  $n$  par  $r1$  et rajouter au réseau une copie de  $n$ , dont  $r2$  est le sélecteur, et de ses arcs adjacents.
- (b) Soit  $n1, n2$  des nœuds, qui, à l'exception de leur sélecteurs ( $r1$  et  $r2$ ), sont des copies de chacune, ainsi que leurs arcs adjacents. Enlever  $n2$  et remplacer le sélecteur de  $n1$  par ( $r1 \vee r2$ ).

Dans le cas des modèles des îlots, le sélecteur d'une place correspond à son nom, et il désigne un lieu caractéristique. Les annotations des transitions définissent les transferts élémentaires dans l'îlot.

**Définition** : Réécriture Locale des Variables :

Remplacer, d'une manière cohérente, une variable annotant un nœud par une autre.

Les variables, utilisées dans les modèles des îlots, désignent différents types de produits admis dans l'îlot. La réécriture locale nous permettra la représentation du mode opératoire examiné au niveau du modèle réseau de Petri.



**ANNEXE 9 : LA PROJECTION TOTALE SUR LES PLACES [GENRICH 1986]**

La projection totale sur les places conduit à une représentation quantitative de la structure dynamique d'un réseau de Petri à Prédicat / Transition. Cette opération transforme les individus d'un réseau à Prédicat / Transition en jetons d'un réseau à places /transition.

Soit  $\lambda = \{ \langle x1 \rangle, \langle x2 \rangle, \langle x3 \rangle \}$  le marquage d'une place. La projection totale sur cette place, notée  $I \lambda I$ , est alors  $I \lambda I = 3 \langle \rangle$ , où  $\langle \rangle$  symbolisent des jetons banalisés.

Comme le modèle Prédicat / Transition d'une cellule est uniforme, les arcs ont tous la même arité, la projection totale mène à un réseau de Petri ordinaire. L'arité d'un arc indique la quantité d'individus qui passent la transition lors d'un tir, elle est égale à un dans nos modèles.



## ANNEXE 10 : RÉDUCTION DE L'ÎLOT 3

### **Etape 1 : Elimination des Places Implicites .**

Le modèle de l'îlot 3 comporte quatre places implicites, les deux places modélisant l'accès exclusif à Zo et à Sa et les deux places représentant les deux robots. Leur élimination conduit au modèle de la figure A.5.

### **Etape 2 : Fusion des Places.**

Nous pouvons établir les correspondances suivantes entre les quatre types de places définies dans le chapitre III et le modèle de la figure A.5 :

- \* tampons de l'îlot (places de type 1) : P9 et P11
- \* lieux de synchronisation des ressources de stockage (places de type 2) : S, Sa, Zo.

Nous considérons la zone opératoire comme une place de type 2, car elle est en relation d'accessibilité exclusivement avec des lieux de synchronisation et sa condition d'accessibilité est représentée par une place de type capacité.

En utilisant cette équivalence, les autres étapes de la réduction, fusion des places de type protocole, fusion des transitions en parallèle et élimination des boucles sont applicables comme dans les autres exemples présentés.

Afin de mettre en évidence le chargement d'un produit sur le tampon de sortie (franchissement de la transition  $t_s$ ), nous avons dissocié pour l'analyse de la cellule (figure III.29) la place S de la place SA $\vee$ Zo $\vee$ S (figure A.6).

Nous avons représenté sur cette figure l'assemblage (transition  $t_a$ ) par la somme symbolique  $\langle sa, pa \rangle + \langle zo, p \rangle$ .

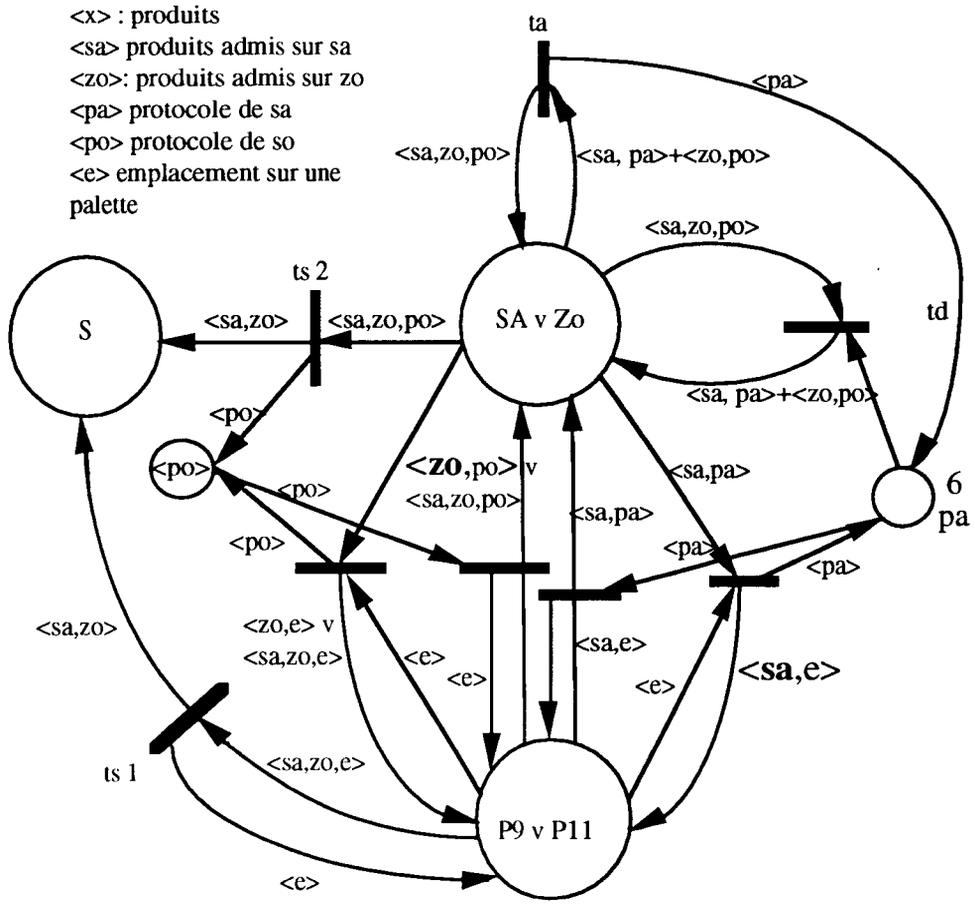


Figure A.6 : modèle réduit de l'îlot 3



## **ANNEXE 11 : PRÉSENTATION D'UN SIMULATEUR**

### **1. Introduction**

Le simulateur, présenté ici en annexe, a été initialement conçu pour la validation des modèles de la commande. Il nous a permis en particulier de mettre en évidence des situations de blocages par saturation ce qui nous a conduit à développer la validation formelle présentée au chapitre III.

Un deuxième champ d'application est la validation du comportement dynamique du système. En raison de la première vocation du simulateur, une approche interactive a été choisie au détriment de la rapidité de simulation.

Le simulateur utilisé comme outil pour la validation dynamique, s'inscrit dans le cadre des simulateurs conçus pour une application spécifique, où nous sommes intéressés par une étude détaillée du comportement dynamique du procédé. L'utilisation des modèles de la commande de coordination, nous permet une analyse fine du procédé face à des scénarii définis par l'utilisateur.

### **2. L'Intégration du Comportement Dynamique du Procédé dans les Modèles**

Si les modèles de la commande de coordination sont utilisés pour commander le procédé, l'interface reçoit des requêtes d'exécution et retourne des comptes rendus d'exécution ou des réponses signalant leur échec. Le délai entre les deux signaux, requête et réponse, est fonction du comportement dynamique de la ressource exécutant l'ordre requis. En simulation, nous ne disposons pas des signaux des capteurs et nous devons intégrer la dynamique du procédé dans les modèles.

Au niveau de la modélisation par réseaux de Petri, un délai temporel peut se traduire par une condition supplémentaire de franchissement des transitions [Sifakis 1977, Chrétienne 1983]. Dans tels réseaux (dits réseaux de Petri temporisés), les marques possèdent deux états (disponibles ou indisponibles). Le procédé est alors réduit à un ensemble de compteurs référencés par l'horloge commune.

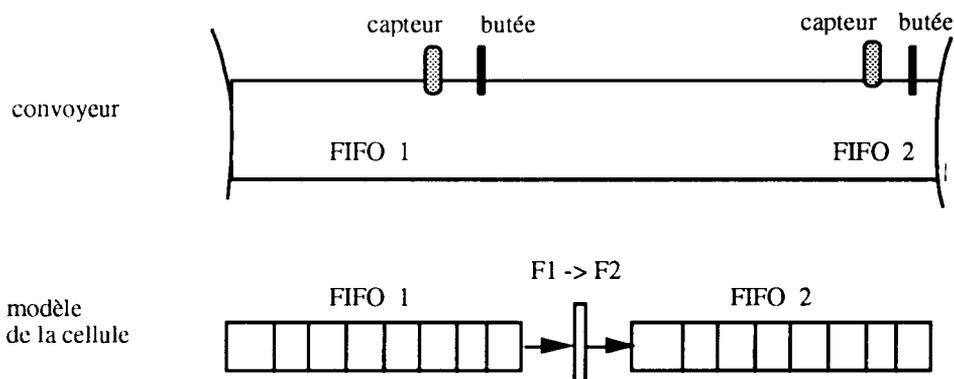
Il existe deux "écoles" selon que la temporisation est associée aux places ou aux transitions. Comme dans nos modèles, toute action est représentée par une transition, nous avons choisi

une temporisation des transitions. Le délai temporel de franchissement, induit par la temporisation, correspond alors à la durée de l'exécution d'une commande par le procédé.

Nous pouvons distinguer deux cas de temporisation des transitions selon que la modification de l'état du procédé résulte directement de l'application d'une commande - et dans ce cas, nous pouvons mesurer la durée de l'exécution de l'ordre -, ou selon que la durée de l'exécution dépend essentiellement de l'état courant d'une ressource. Toutes les actions élémentaires modélisées au niveau de la commande des ressources sont du premier type. Nous temporisons donc ces transitions avec la durée effective.

L'exemple type concernant la durée d'exécution d'un ordre, qui dépend de l'état courant de la ressource, correspond à un transfert élémentaire sur un convoyeur.

En général, la présence d'une palette est détectée par un capteur en tête de fifo. L'information concernant la réalisation d'un transfert entre deux fifos est alors seulement disponible, si la palette se présente en tête de la fifo d'arrivée. La durée de ce transfert dépend donc essentiellement des palettes déjà présentes dans la fifo d'arrivée.



**Figure A.7 : temporisation des transferts**

Un transfert de la fifo F1 vers la fifo F2 est alors seulement achevé, si le capteur de F2 détecte la présence de la palette (figure A.7). Le modèle du convoyeur représente le comportement d'une palette dans la fifo par la structure de la figure A.7. Nous temporisons donc la transition "F1->F2" avec la durée du transfert F1->F2 dans le cas d'une fifo F2 vide, ce qui correspond à la durée minimale du transfert. Si F2 contient déjà des palettes, la palette est insérée en queue et seulement disponible en tête de fifo si les autres palettes ont quitté F2. Une temporisation du modèle du convoyeur traduit, dans ce cas, le comportement du procédé de manière exacte.

La dynamique du procédé est ainsi intégrée dans les modèles de la commande de coordination à des endroits précis et le modèle de départ reste, mis à part des temporisations, inchangé.

### 3. La Description du Programme de Simulation

Le simulateur a été réalisé en Le\_Lisp [Le\_Lisp] afin de profiter du haut degré d'interactivité mis à disposition par ce langage. Nous avons choisi une programmation orientée objet, disponible dans Le\_Lisp pour la réalisation du programme de simulation. Cette démarche permet de traduire la structure des modèles, composés de places, de transitions et de jetons en structures de données, en leur associant un traitement qui modélise leur comportement [Meyer 1990].

Une analyse des modèles nous a conduits à définir des sous-classes de chaque type de primitives montrant un comportement différent. Pour la classe "place", nous avons ainsi défini les sous-classes "place\_request", "place\_acknowledge", "place\_en\_voyage" et "place\_fifo" (les places "nack" ont le même comportement que les places "ack"). Une place de type "fifo" traduit la représentation agrégée d'une file d'attente du modèle du convoyeur. Le comportement de cette structure physique, l'insertion d'une nouvelle palette à la fin et accessibilité de la première palette, sont traduits par des méthodes définies pour la classe "place\_fifo".

De la même manière ont été définis différents types de transitions, qui se distinguent selon qu'elles font partie d'une structure représentant un indéterminisme ou selon qu'elles peuvent être en conflit de tir, ainsi que quatre types de jetons modélisant les produits, les ressources à transferts libres, les protocoles d'accès ou les jetons des graphes de synchronisation.

Un modèle simulable est donc généré par une création des instances de ces classes, correspondant aux entités constituant le modèle. Dans la même étape les modèles des ressources sont instanciés à partir de leur modèles génériques.

Dans cette phase, l'utilisation de la programmation orientée objet et des modèles réseau de Petri de la commande de coordination permet un gain de temps considérable pour la traduction du comportement du procédé dans un modèle simulable, par rapport à la programmation d'un simulateur spécifique, visant les mêmes champs d'application.

Le cycle de simulation se déroule selon les étapes suivantes :

- \* évaluer les événements
- \* chercher les transitions tirables
- \* résoudre les indéterminismes
- \* tirer les transitions

Cette boucle générale du simulateur est composée, comme dans la plupart des simulateurs de réseaux de Petri [Castelain 1987, Tankoano et al 1991], d'une étape d'évolution par événement (évaluer les événements) et d'une étape d'évolution par activité (tirer les transitions). Les événements, mémorisés dans un échancier, sont générés par le marquage des places par des jetons à leur date de disponibilité. Rappelons, que la temporisation rend les jetons indisponibles pour une certaine durée.

Dans l'étape "chercher les transitions tirables", nous évaluons chaque place marquée afin de déterminer les transitions qui sont susceptibles d'être tirées. Suivant l'approche de la programmation orientée objet, nous avons défini des méthodes spécifiques à chaque type de place du modèle.

Le traitement des événements, dans une résolution simultanée des conflits et des indéterminismes, assure la détection de toutes les situations demandant une décision. Notons tout de même, que seules ces situations pour lesquelles aucune règle de décision n'a été spécifiée au préalable sont mémorisées.

Les objets modélisant les transitions, sujettes à des indéterminismes ou potentiellement en conflit, contiennent un champ particulier nommé "condition" dont le contenu est testé pendant l'étape d'évaluation du marquage. La condition peut être le résultat d'une évaluation d'une fonction `Le_Lisp` quelconque, comme, par exemple une fonction réalisant un calcul de priorité.

A l'heure actuelle, les indéterminismes et conflits non résolus sont répertoriés au niveau utilisateur. Cette démarche a été importante pour la validation des modèles, car elle permet d'explorer tous les chemins possibles des modèles réseaux de Petri.

Dans la dernière étape de chaque cycle les nouveaux événements sont générés par le tir des transitions. Ces événements sont affichés afin de pouvoir tracer l'évolution du marquage des modèles et ensuite les jetons, qui ont franchi les transitions, sont mémorisés dans l'échancier avec leur date de disponibilité.

La mémorisation de données qui permettent une exploitation statistique des résultats de la

simulation, s'effectue au niveau des jetons pour les informations relatives aux produits et aux palettes. Les données retenues au niveau des produits sont les temps d'attente, les temps de transfert et le temps passé dans le système. Les données sont mémorisées dans des champs définis pour cette classe de jetons et mises à jour par des méthodes associées au tir des transitions correspondantes.

Les données relatives aux ressources, en l'occurrence le taux d'utilisation, sont mémorisées au niveau de leur structure de données et mises à jour par le tir des transitions appartenant à leur graphe de commande.

#### **4. Conclusion**

Le simulateur, dont nous avons brièvement présenté les principales caractéristiques, a été initialement conçu pour la validation de la commande de coordination. La structure des données et le programme de simulation sont adaptés aux modèles issus de la démarche de conception. Du fait d'une application comme outil de validation, une simulation interactive de type "pas à pas" qui met en évidence l'évolution du marquage et qui permet une analyse des indéterminismes était le premier souci de la conception.

Mais le simulateur trouve son application aussi en phase d'évaluation des performances détaillée, si nous recherchons une adaptation fine des mécanismes de résolution des conflits et des indéterminismes. La convivialité du langage choisi permet dans ce cas une modification rapide des mécanismes définis.

Pour ces applications nous pouvons nous limiter à une simulation d'une partie du modèle de la commande de coordination. Comme l'ensemble des indéterminismes et des conflits est représenté sur les gammes opératoires et le modèle du système de transport, une simulation de ces deux modèles est suffisante.

Ceci permet une accélération considérable de la simulation par l'élimination de certains états possibles au niveau du modèle simulé. Une fois de plus nous profitons, dans ce cas, de la construction modulaire des modèles de la commande de coordination.

Le simulateur se prête aussi à une analyse du comportement du système en mode transitoire, car l'utilisateur peut définir des séquences d'entrée de produits sans contraintes et peut ainsi composer librement ses modes opératoires et ses points de fonctionnement.



## RÉFÉRENCES BIBLIOGRAPHIQUES

- [ADEPA 1979] ADEPA : “Le GEMMA guide des modes de marche et d’arrêt”, 1979.
- [Alla et al 1984] H. Alla, P. Ladet, J. Martinez, M. Silva-Suarez : “Modelling and validation of complex systems by coloured Petri nets : application to a flexible manufacturing system” ; Proceedings of fifth european workshop on application and theory of Petri nets, Aarhus, Danemark, juin 1984.
- [Amar 1994] S. Amar : “Systèmes automatisés et flexibles de production : méthode de conception du système de coordination par prototypage orienté objet de la partie procédé” ; Thèse de Doctorat, Université de Lille I, soutenance prévue.
- [Amar et al 1992] S. Amar, E. Craye, J.C. Gentina : “Une méthode hiérarchique de spécification et de prototypage des systèmes de production flexibles” ; APII, vol 26, no 5-6, 1992.
- [André et al 1993] C. André, S. Elkhatabi, J.C. Gentina, M.A. Péraldi : “Esterel programming in FMS” ; Proceedings of IEEE 1993 international conference on systems, man and cybernetics, LeTouquet, France, 1993.
- [Archimède 1991] B. Archimède : “Conception d’une architecture réactive, distribuée, hiérarchisée pour le pilotage des systèmes de production” ; Thèse de Doctorat, Université de Bordeaux, 1991.
- [Ausfelder et al 1992] C. Ausfelder, E. Castelain, J.C. Gentina : “An Object Oriented Simulation Tool to Validate the Dynamic Behaviour of FMS” ; Proceedings of European Simulation Multiconference, York, United Kingdom ; June, 1-3, 1992.

- [Ausfelder et al 1993a] L. Kermad, C. Ausfelder, J.P. Bourey, E. Castelain : “An integrative approach for a functional specification of FMS control” ; Computer Integrated Manufacturing Systems, vol 6, no 4, november 1993.
- [Ausfelder et al 1993b] C. Ausfelder, L. Kermad, J.P. Maik, J.C. Gentina, D. Delfieu, R. Moisand, A.E.K. Saharoui : “Integration of operating modes in the control of FMS combining synchronous and asynchronous approaches” ; Proceedings of IEEE 1993 international conference on systems, man and cybernetics, Le Touquet, France, 1993.
- [Ausfelder et al 1993c] C. Ausfelder, E. Castelain, J.-C. Gentina : “A hierachical modular model of FMS” ; Proceedings of IEEE 1993 international conference on systems, man and cybernetics, Le Touquet, France, 1993.
- [Ausfelder et al 1994] C. Ausfelder, E. Castelain, J.C. Gentina : “A method for hierarchical modelling of the command of FMS” ; IEEE transactions on systems, man and cybernetics, à paraître.
- [Baptiste et Chappe 1992] P. Baptiste et D. Chappe : “Interêt des RdP à terminaison propre pour le pilotage temps réel d’ateliers flexibles ;” APII, vol 26, no 5-6, 1992.
- [Bastide 1991] R. Bastide : “Objets coopératifs : un formalisme pour la modélisation des systèmes concurrents” ; Thèse de Doctorat, Université Paul Sabatier Toulouse, 1991.
- [Bénassy 1987] J. Bénassy : “La gestion de production“ ; éd Hermès, Paris 1987.
- [Berry et al 1987] G. Berry, P. Couronné, G. Gouthier : “Programmation synchrone des systèmes réactifs : le langage ESTEREL” ; TSI, vol 6, no 4, 1987.
- [Berthelot 1983] G ; Berthelot : “Transformation et analyse de réseaux de Petri, applications aux protocoles” ; Thèse d’état, Université P. et M. Curie, Paris 1983.
- [Bonetto 1985] R. Bonetto : “Les ateliers flexibles de production” ; éd Hermès, Paris 1985.

- [Booch 1988] G. Booch : “Ingénierie du logiciel avec ADA” ; Addison Wesley Europe, InterEditions, Paris 1988.
- [Booch 1991] G. Booch : “Object oriented design, with applications” ; the Benjamin / Cuning Publishing Company, 1991.
- [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 Lille I, 1991.
- [Bourey 1988] J.P. Bourey : “Structuration de la partie procédurale du système de commande de cellules de production flexibles dans l’industrie manufacturière” ; Thèse de Doctorat, Université de Lille I, 1988.
- [Bourey 1993] J.P. Bourey : “Méthode de conception de la commande des systèmes flexibles de production manufacturière” ; Habilitation à diriger des recherches, Université de Lille I, 1993.
- [Bourey et al 1989] J.P. Bourey, E. Castelain, J.-C. Gentina, M. Kapusta : “CASPAIM : a computer aided design of the control system of FMS” ; IMACS annuals on computing and applied mathematics, 1989.
- [Bourey et al 1992] J.P. Bourey, E. Castelain, E. Craye, J.C. Gentina : “Petri nets in discrete production systems” ; Systems and Control Encyclopaedia, Pergamon Press, 2nd vol, 1992.
- [Bourjault 1986] A. Bourjault : “Modélisation d’un processus d’assemblage” ; APII, vol.20, no 2, 1986.
- [Bourrières et Chevillard 1988] J.P. Bourrières, A. Chevillard : “Modélisation des convoyeurs à transfert libre à l’aide des réseaux de Petri” ; RAPA, no 4, 1988.
- [Brams 1983] G.W. Brams : “Réseaux de Petri : Théorie et Pratique” ; 2 tomes, éd. Massons, Paris 1983.

- [**Brandimarte et al 1992**] P. Brandimarte, W. Ukovich, A. Villa : "Factory level scheduling : a basis for a hierarchical approach" ; Proceedings of third international conference on computer integrated manufacturing, Troy, New York, May 1992.
- [**Bruns et Applerath 1991**] R. Bruns et H.J. Appelrath : "Ein universelles Modell für Ablaufplanungsprobleme" ; Wirtschaftsinformatik, 33. Jhg., Heft 6, 1991.
- [**Cao et Ho 1990**] Xi-Ren Cao et Yu-Hhi Ho : "Models of discrete event dynamic systems" ; IEEE Control Systems, Vol 1, No 4, juin 1990.
- [**Carlier et Chrétienne 1988**] J. Carlier et P. Chrétienne : "Problèmes d'ordonnement : modélisation/ complexité/ algorithmes" ; éd Masson, Paris, 1988.
- [**Castelain 1987**] E. Castelain : "Modélisation et simulation interactive de cellules de production flexibles dans l'industrie manufacturière" ; Thèse de Doctorat, Université de Lille I, 1987.
- [**Castelain et Gentina 1988**] E. Castelain et J.C. Gentina : "Petri nets and artificial intelligence in the context of simulation and modelling of manufacturing systems" ; Proceedings of IMACS/ ICOM, Paris, 1988.
- [**Cavaille et Proth 1989**] J.B. Cavaille et J.M. Proth : "SIPRODIS Pratique de la simulation en production discontinue" ; Collection Novotique EC2, Nanterre, 1989.
- [**Chassang et Tron 1983**] G. Chassang et H. Tron : "Gérer la production avec l'ordinateur" ; éd Dunod, Paris 1983.
- [**Chrétienne 1983**] P. Chrétienne : "Les réseaux de Petri temporisés" ; Thèse d'état, Université P. et M. Curie, Paris, 1983.
- [**Combacau 1991**] M. Combacau : "Commande et surveillance des systèmes à événements discrets complexes : application aux ateliers flexibles" ; Thèse de Doctorat, Université de Paul Sabatier, Toulouse 1991.

- [Craye 1989] E. Craye : “De la modélisation à l’implantation automatisée de la commande hiérarchisée des cellules de production flexibles dans l’industrie manufacturière” ; Thèse de Doctorat, Université de Lille I, 1989.
- [Cruette 1991] D. Cruette : “Méthodologie de conception des systèmes complexes à événement discrets : application à la conception et la validation hiérarchisée de la commande de cellules flexibles de production dans l’industrie manufacturière” ; Thèse de Doctorat, Université de Lille I, 1991.
- [Daenzer 1988] G. Daenzer : “Systems Engineering” ; Verlag Industrielle Organisation, Zürich 1988.
- [David et Allah 1989] R. David et H. Allah : “Du Grafset aux réseaux de Petri” ; éd. Hermès, Paris, 1989.
- [Dungern et Schmidt 1990] O.v. Dungern et G. Schmidt : “Vorbereitende und begleitende Ablaufplanung für flexible Montagezellen in industrieller Umgebung” ; Robotersysteme 6, 1990, pp. 225-235.
- [Elkhatabi 1993] S. Elkhatabi : “Intégration de la surveillance de bas niveau dans la conception des systèmes à événements discrets : application aux systèmes de production flexibles” ; Thèse de Doctorat, Université de Lille I, 1993.
- [Elkhatabi et al 1992] S. Elkhatabi, D. Corbeel, J.-C. Gentina : “Integration of Dependability in the Conception of FMS” ; Proceedings of IFAC - ICOM ; Toronto, Canada ; May 25 - 28, 1992.
- [Fandel et al 1990] G. Fandel, H. Dyckhoff, J. Reese : “Industrielle Produktionsentwicklung” ; Springer-Verlag, Berlin, Heidelberg, New York, 1990.
- [Favrel et Campagne 1984] J. Favrel et J.P. Campagne : “Génération automatique des gammes d’assemblage” ; Revue française de gestion industrielle, vol. 3, no 1-2, 1984.
- [Ferreira et Pradin 1991] J.F. Ferreira-Ribeiro et B. Pradin : “Selection of manufacturing cells for group technology”, Proceedings of the 13th IMACS world congress, Lille, mai 1991.

- [**Fu et Liu 1992**] Li - Chen Fu et Pei - Sen Liu : "Hierarchical dynamical scheduling for a flexible manufacturing system" ; Proceedings of 3rd international conference on CIM, Troy, N.Y. May 20 -22, 1992.
- [**Gasnier et Vercauter 1988**] B. Gasnier et C. Vercauter : "An interactive tool for computer aided manufacturing sequence specification" ; Proceedings of IFAC 1988, Swansea 1988.
- [**Gentina et al 1992**] J.C. Gentina, E. Craye, J.P. Bourey, E. Castelain, S. Dimster, M. Kapusta : "Approche de prototypage de systèmes flexibles de production discrète" ; Rapport de fin de Contrat MRT-Productique, Avril 1992.
- [**Genrich 1986**] H.J. Genrich : "Predicate / Transition Nets" ; [Jensen et Rozenberg 1991].
- [**Genrich 1989**] H.J. Genrich : "Equivalence transformations of Predicat / Transition -Nets" ; [Jensen et Rozenberg 1991].
- [**Genrich et Lautenbach 1981**] H.J. Genrich et K. Lautenbach : "System modelling with high-level Petri nets" ; Theoretical computer science 13, 1981.
- [**Jensen et Rozenberg 1991**] K. Jensen et G. Rozenberg (ed) : " High-level Petri nets - theory and application" ; Springer-Verlag, Berlin, Heidelberg, New York, 1991.
- [**Hammadi 1991**] S. Hammadi : "Une méthode d'ordonnancement minimisant les temps d'attente et de transit dans les systèmes de production flexibles de type job-shop" ; Thèse de Doctorat, Université de Lille I, 1991.
- [**Haralakis et al 1992**] G. Haralakis, R. Nagi, J-M. Proth, F. Vernadat : Design of Manufacturing Systems : a bottom-up approach based on Petri nets " ; Proceedings of IFAC - ICOM ; Toronto, Canada ; May 25 - 28, 1992.
- [**Hébrard 1992**] A. Hébrard : "Etude et réalisation d'un module de pilotage et d'un outil de représentation graphique appliquées à la conception d'un système de production flexible" ; Thèse de Doctorat, Université de Lille I, 1992.

- [Huber et al 1990] P. Huber, K. Jensen, R.M.Shapiro : “Hierarchies in coloured Petri nets” ; [Jensen et Rozenberg 1991]
- [Huvenoit et al 1994] B. Huvenoit, E. Craye, J.-P Bourey : “Design and implementation methodology based on petri net formalism of flexible manufacturing systems control” ; Production planning and control, à paraître.
- [I.G.L. 1989] I.G.L. Technology : “SADT : un langage pour communiquer” ; Ed Eyrolles, Paris, 1988.
- [Kapusta 1988] M. Kapusta : “Génération assistée d’un graphe fonctionnel destiné à l’élaboration structurée de la partie commande pour des cellules de production flexible dans l’industrie manufacturière” ; Thèse de Doctorat, Université de Lille I, 1988.
- [Kaufmann 1968] A. Kaufmann : “Introduction à la combinatoire en vue des applications” ; Ed Dunot, Paris 1968.
- [Kermad 1992] L. Kermad : “Modélisation des systèmes de production en vue de l’intégration des modes de marche dans CASPAIM II” ; DEA de Productique, Ecole Centrale de Lille ; Lille 1992.
- [Le\_Lisp] Le\_Lisp de l’INRIA, Manuel de Référence, version 15.24.
- [Le Moigne 1990] J.-L. Le Moigne : “La modélisation des systèmes complexes” ; Afcet systèmes, éd Dunod, Paris, 1990.
- [Meffert 1986] H. Meffert : “Marketing - Grundlagen der Absatzpolitik” ; 7. Aufl., Wiesbaden 1986.
- [Meyer 1990] B. Meyer : “Object-oriented software construction” ; Prentice Hall, 1990.
- [Milot 1988] P. Milot : “Supervision des procédés automatisés et ergonomie” ; éd. Hermès, Paris, 1988.

- [Morel 1992] G. Morel : “Contribution à l’automatisation et à l’ingénierie des systèmes intégrés de production “ ; mémoire d’habilitation à diriger les recherches, Nancy 1992.
- [Murata 1989] T. Murata : “Petri-Nets : properties, analysis and applications” ; Proceedings of the IEEE, vol 77, no 4, avril 1989.
- [Panetto et al 1991] H. Panetto, P. Lhotse, G. Morel, M. Roesch : “SPEX : du génie logiciel pour le génie automatique” ; Génie logiciel et Systèmes experts, no 25, 1991.
- [Panwalker et Iskander 1977] S. Panwalker et W. Iskander : “A survey of scheduling rules“ ; Operations Research, Vol 25, no 1, 1977.
- [Parayre 1992] : T. Parayre : “Le MESAP : vers une méthodologie d’exploitation des systèmes automatisés de production” ; Thèse de Doctorat, Université de Valenciennes, 1992.
- [Péraldi 1993] M.A. Péraldi : “Conception et réalisation de systèmes temps réel par une approche synchrone” ; Thèse de Doctorat, Université de Nice - Sophia Antipolis, 1993.
- [Portmann 1988] M.C. Portmann : “Méthodes de décomposition spatiale et temporelle en ordonnancement de la production” ; APII, vol 22, 1988.
- [Ramdage et Wonham 1989] P. Ramdage et W.M. Wonham : “The control of discrete event systems” ; Proceedings of the IEEE, vol 77, no 1, janvier 1989.
- [Ranky 1990] P. Ranky : “Flexible Manufacturing Cells in CIM” ; CIMware Ltd., Guildford, UK, 1990.
- [Reisig 1982] W. Reisig : “Petri-Netze, eine Einführung” ; Springer-Verlag, Berlin, 1982.
- [Roboam et al 1989] M. Roboam, M. Zanettin, L. Pun : “GRAI-IDEF0-Merise (GIM) : Integrated methodology to analyse and design manufacturing systems” ; computer integrated manufacturing systems, vol 2, no 2, may 1989, pp.82-98.

- [**Ross 1985**] D.T. Ross : “Applications and extensions of SADT” ; Computer IEEE, April 1985.
- [**Ross et al 1975**] D.T. Ross, J.B. Goodenough, C.A. Irvine : “Software engineering : process, principles and goals” ; Computer IEEE, no 65, may 1975.
- [**Roubellat et Thomas 1988**] F. Roubellat et V. Thomas : “Une méthode et un logiciel pour l’ordonnancement en temps réel des ateliers” ; APII, vol. 22, no 5, 1988.
- [**Saharoui 1987**] A.E.K. Saharoui : “Contribution à la surveillance et à la commande d’atelier” ; Thèse de Doctorat, Université Paul Sabatier, Toulouse, 1987.
- [**Saharoui 1992**] A.E.K. Saharoui : “Vers une approche globale de surveillance des systèmes à événements discrets” ; APII, vol 26, no 2, 1992.
- [**Schmidt 1989**] G. Schmidt : “CAM : Algorithmen und Decision Support für die Fertigungsplanung” ; Springer-Verlag, Berlin 1989.
- [**Schmidt et Glüer 1988**] G. Schmidt et D. Glüer : “Die Anwendung von Petri-Netzen zu Modellbildung, Simulation und Steuerungsentwurf bei flexiblen Fertigungssystemen” ; Automatisierungstechnik 36, 1988.
- [**Sibertin-Blanc 1985**] C. Sibertin-Blanc : “High level Petri-nets with data structures “, 6th European workshop on theory and application of Petri-nets “, Espoo, Finlande, 1985.
- [**Sifakis 1977**] J. Sifakis : “Use of Petri nets for performance evaluation” ; Measuring, Modelling and Evaluating computer systems, North Holland, 1977.
- [**Silva et Valette 1990**] M. Silva et R. Valette : “Flexible manufacturing systems and Petri-nets” ; Proceedings of eleventh international conference on application and theory of petri-nets, Petri nets tutorial, Paris, juin 1990.
- [**Slack 1988**] Slack, N. : “Manufacturing systems flexibility - an assessment procedure” ; Computer Integrated Manufacturing Systems ; vol 1, n° 1, 1988.

- [**Tahon et Dindeleux 1992**] C. Tahon et E. Dindeleux : "Système d'aide à la décision en gestion de production" ; Bulletin de liaison n°1, GR Automatique du CNRS, Pôle SED GT3 Ordonnancement, 1992.
- [**Tankoano et al 1991**] J. Tankoano, D. Boudebous, J.C. Derniame : "PETRI-S : Un simulateur de systèmes de production automatisés décrits à l'aide de réseaux de Petri interprétés colorés" ; APII, vol 25, no 1, 1991.
- [**Tardieu et al 1983**] H. Tardieu, A. Rochfeld, R. Coletti : "La méthode Merise" ; Edition des Organisations, Paris 1983.
- [**Tawegoum et al 1992**] R. Tawegoum, E. Castelain, J.C. Gentina : "Real time piloting of FMS" ; Proceedings of third international workshop on project management and scheduling, Como, Italy, 1992.
- [**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, 1992.
- [**Valette 1986**] R. Valette : "Petri nets in production systems" ; Advances in Petri nets 1986, Springer-Verlag, Berlin 1987.
- [**Valette et al 1988**] R. Valette, M. Paludetto, B.P. Labreuille, P. Farail : "Approche orientée objet HOOD et réseaux de Petri pour la conception de logiciel temps réel" ; proceedings of first international workshop on software engineering and its applications, EC2 Toulouse, december 1988.
- [**Vielcanet 1990**] : P. Vielcanet : "HOOD design method and control/command techniques for the development of real-time software" ; CISI Ingenerie-Aerospace Branch 1990.

# TABLE DES MATIÈRES

## INTRODUCTION GÉNÉRALE :

LES PRINCIPALES CARACTÉRISTIQUES DES SYSTÈMES FLEXIBLES DE PRODUCTION MANUFACTURIÈRE.....	7
--	---

## CHAPITRE I :

### UN MODÈLE DE LA CONDUITE DES SFPM

1. Introduction.....	13
2. Le Besoin d'une Approche Intégrée pour la Conception de la Conduite.....	14
2.1. La Problématique de la Conduite.....	14
2.2. Une Approche Systémique pour la Conduite.....	15
3. Le Modèle de La Conduite (Ausfelder et al 1993a).....	19
3.1. La Conduite et son Environnement.....	19
3.1.1. Gérer la Production.....	19
3.1.2. Fabriquer.....	20
3.1.3. Conduire.....	21
3.2. La Modélisation de La Conduite.....	22
3.2.1. Surveiller.....	23
3.2.2. Mise à Jour des Modèles.....	29
3.3. l'Activité "Commander".....	32
3.3.1. Allouer les Ressources.....	32
3.3.2. Coordonner.....	35
3.3.3. Commander les Actionneurs.....	36
3.3.4. Conclusion.....	37

4. Un Cahier des Charges pour la Coordination.....	38
4.1. Introduction.....	38
4.2. Les Contraintes de Qualité d'un Système de Coordination du Point de Vue Génie Logiciel.....	38
4.3. Les Besoins de la Coordination issus du Modèle de la Conduite .....	39
4.3.1. La Sûreté de la Coordination .....	39
4.3.2. L'Adaptabilité de la Coordination.....	40
5. Conclusion.....	41

## CHAPITRE II :

### UNE MÉTHODE POUR LA CONCEPTION DE LA COMMANDE DE COORDINATION DES SFPM

1. Introduction.....	43
2. La Démarche Proposée .....	44
2.1. Introduction.....	44
2.2. La Spécification des Objectifs et l'Analyse de l'Atelier.....	46
2.3. La Génération des Modèles de la Coordination .....	46
2.3.1. Les Gammes Opératoires .....	46
2.3.2. Le Modèle du Système de Transport.....	47
2.3.3. Les Graphes de Synchronisation .....	48
2.3.4. Les Modèles des Commandes des Ressources.....	48
2.4. La Connexion des Modèles .....	49
2.5. Conclusion.....	49
3. La Spécification des Objectifs de Production et l'Analyse du Système.....	50
3.1. La Spécification des Produits : Les Gammes Logiques .....	50
3.2. L'Analyse du Procédé.....	54
3.2.1. La Décomposition structuro-fonctionnelle des Ressources [Amar et al 1992].....	55
3.2.2. La Synthèse du Graphe des Accessibilités du Système.....	58
3.2.3. Les Contraintes de Fonctionnement.....	61
3.3. Conclusion.....	61

4. La Génération des Modèles de la Coordination .....	62
4.1. Introduction .....	62
4.2. La Génération des Gammes Opératoires.....	62
4.2.1. Introduction .....	62
4.2.2. Les Données d'Entrée.....	63
4.2.3. Un Algorithme pour la Génération des Gammes Opératoires.....	63
4.2.4. La Hiérarchisation des Gammes Opératoires.....	66
4.2.5. Conclusion .....	69
4.3. La Génération du Modèle du Système de Transport.....	70
4.3.1. Introduction .....	70
4.3.2. La Structuration d'un SFPM [Ausfelder et al 1993c] .....	72
4.3.2.1. Introduction.....	72
4.3.2.2. L'Identification des Îlots .....	73
4.3.2.3. L'Identification d'une Cellule.....	75
4.3.2.4. L'Identification d'un Atelier .....	76
4.3.2.5. Conclusion.....	76
4.3.3. La Génération des Modèles Réseau de Petri .....	77
4.3.3.1. La Hiérarchie Proposée.....	77
4.3.3.2. La Modélisation des Îlots.....	77
4.3.3.3. La Modélisation d'une Cellule.....	82
4.3.4. Conclusion .....	85
4.4. Les Graphes de Synchronisation .....	87
4.4.1. Introduction .....	87
4.4.2. La Génération des Graphes Inter-Ressources.....	88
4.4.2.1. Les Données d'Entrée .....	88
4.4.2.2. La Conception des Graphes de Synchronisation Inter-Ressources .....	89
4.4.3. Les Graphes de Synchronisation Intra-Ressource.....	91
4.4.4. Conclusion .....	92
4.5. Les Modèles des Ressources Simples.....	93
4.5.1. Introduction .....	93
4.5.2. Deux Exemples : Modèles d'un Manipulateur et d'un Tour .....	94
4.5.3. Conclusion .....	95
4.6. Conclusion .....	96

5. La Communication entre les Modèles.....	97
5.1. Introduction.....	97
5.2. La Communication entre Graphes de Synchronisation et Modèles des Ressources .....	99
5.3. La Communication entre le Modèle du Système de Transport et les Graphes de Synchronisation .....	101
5.4. La Communication entre Gammes Opératoires et Modèle de Transport et Gammes Opératoires et Modèles des Ressources.....	104
5.5. Conclusion.....	106
6. La Coordination au Sein d'un Système de Conduite.....	108
6.1. Une Coordination Sûre et Adaptative.....	109
6.2. La Connexion de la Coordination à l'Interface du Procédé .....	110
6.3. Les Modèles de la Coordination comme Modèles du Niveau Hiérarchique....	112
7. Conclusion.....	113

### CHAPITRE III :

#### La Validation des Modèles de la Commande de Coordination des SFPM

1. Introduction.....	115
2. L'Analyse des Modèles des Ressources Simples .....	116
3. L'Analyse des Graphes de Synchronisation .....	117
4. L'Analyse du Modèle du Système de Transport.....	118
4.1. L'Analyse des Modèles des Îlots.....	118
4.1.1 La Démarche de l'Analyse.....	118
4.1.2. La Réduction du Modèle.....	120
4.1.2.1 La Réduction du Modèle de l'îlot 2.....	121
4.1.2.2. Réduction de l'îlot 1 de la Cellule .....	127
4.1.2.3. Réduction d'un îlot comportant une ressource partagée entre deux îlots élémentaires.....	128
4.1.2.4. Conclusion .....	134
4.1.3. Une Démarche pour l'Analyse du Modèle Réduit .....	134
4.1.3.1. Introduction .....	134
4.1.3.2. L'Analyse de l'Îlot 2 .....	135
4.1.3.3. L'Analyse de l'Îlot 1 de la Cellule.....	138
4.1.3.4. L'Analyse de l'Îlot avec Ressource Partagée.....	140
4.1.4. La Généralisation des Résultats.....	142

4.1.5. L'Intégration des Résultats au Niveau des Modèles.....	146
4.2. L'Analyse du Modèle de la Cellule .....	150
4.2.1. La Démarche de L'Étude des Propriétés .....	150
4.2.2. Un Exemple d'Application .....	151
4.3. Conclusion .....	152
5. L'Analyse des Gammes Opératoires.....	153
5.1. Introduction .....	153
5.2. La Validation des Modèles par Rapport à la capacité limitée du SFPM.....	154
5.2.1. Une Représentation du Système pour la Recherche des Situations Bloquantes .....	155
5.2.2. L'Analyse du Modèle agrégé d'une Cellule .....	157
5.2.3. La Représentation des Résultats sur les Gammes Opératoires .....	159
5.3. Une Démarche pour l'Identification des Blocages des Gammes Opératoires .....	161
6. Conclusion.....	163

**CONCLUSION GÉNÉRALE :**

<b>RÉSUMÉ ET PERSPECTIVES.....</b>	<b>165</b>
------------------------------------	------------

**ANNEXES**

Annexe 1 : Les ACTIGRAMMES de SADT .....	167
Annexe 2 : Présentation de l'Exemple .....	169
Annexe 3 : Les Réseaux de Petri à Prédicats / Transitions .....	173
[Genrich 1986] .....	173
Annexe 4 : Les Hiérarchies dans les Réseaux de Petri [Huber et al 1990] ....	175
Annexe 5 : Propriétés du Graphe des Accessibilités du Système [Kaufmann 1968].....	177
Annexe 6 : Génération du Modèle Réseau de Petri de l'Îlot 3 de la Cellule .....	179
Annexe 7 : Définitions des Propriétés Réseau de Petri [Brams 1983] .....	183
Annexe 8 : Définitions des Transformations d'Equivalence Utilisées [Genrich 1989].....	185
Annexe 9 : La Projection totale sur les Places [Genrich 1986].....	187
Annexe 10 : Réduction de l'Îlot 3 .....	189

Annexe 11 : Présentation d'un Simulateur.....	191
1. Introduction .....	191
2. L'Intégration du Comportement Dynamique du Procédé dans les Modèles .....	191
3. La Description du Programme de Simulation .....	193
4. Conclusion .....	195
 <b>RÉFÉRENCES BIBLIOGRAPHIQUES.....</b>	 197

# LISTE DES FIGURES

## CHAPITRE I :

### UN MODÈLE DE LA CONDUITE DES SFPM

Figure I.1 : les fonctions de la conduite .....	15
Figure I.2 : modèle O.I.D. d'un système complexe [Lemoigne 1990] .....	16
Figure I.3 : la conduite et son environnement .....	19
Figure I.4 : l'activité "conduire" .....	22
Figure I.5 : l'activité "surveiller" .....	23
Figure I.6 : chargement automatique d'un tour .....	26
Figure I.7 : l'activité "commander" .....	32

## CHAPITRE II :

### UNE MÉTHODE POUR LA CONCEPTION DE LA COMMANDE DE COORDINATION DES SFPM

Figure II.1 : les quatre types de modèles .....	45
Figure II.2 : la démarche de conception .....	50
Figure II.3 : gamme élémentaire .....	52
Figure II.4 : gamme d'assemblage .....	52
Figure II.5 : traitement de lot .....	53
Figure II.6 : choix des séquences .....	53
Figure II.7 : exemple de gammes logiques .....	54
Figure II.8 : bibliothèque fonctionnelle .....	55
Figure II.9 : décomposition du tour [Amar et al 1992] .....	56
Figure II.10 : décomposition du convoyeur .....	57
Figure II.11 : graphe des accessibilités du système .....	60
Figure II.12 : la génération des gammes opératoires .....	64
Figure II.13 : les gammes opératoires de l'exemple .....	66
Figure II.14 : gamme agrégée .....	67
Figure II.15 : gammes hiérarchisées .....	69
Figure II.16 : la démarche de génération du modèle .....	71

Figure II.17 : ressource partagée .....	75
Figure II.18 : modèle structurel de l'exemple .....	77
Figure II.19 : la traduction du graphe des accessibilités en réseaux de Petri .....	78
Figure II.20 : îlot 2 de la cellule .....	80
Figure II.21 : accès aux ressources partagées .....	81
Figure II.22 : primitives de connexion .....	83
Figure II.23 : génération du modèle du convoyeur.....	84
Figure II.24 : hiérarchie entre le modèle de la cellule et le modèle de l'îlot 1.....	85
Figure II.25 : les modèles de la commande .....	87
Figure II.26 : synchronisation d'un chargement interne.....	90
Figure II.27 : graphe de synchronisation modélisant un chargement interne.....	91
Figure II.28 : synchronisation des transferts intra-ressources.....	92
Figure II.29 : modèles d'un manipulateur (a) et d'un tour (b).....	94
Figure II.30 : modules de communication .....	98
Figure II.31 : communication ressources / graphes de synchronisation .....	100
Figure II.32 : communication modèles des îlots / graphes de synchronisation .....	102
Figure II.33 : communication modèle de la cellule / graphe de synchronisation .....	103
Figure II.34 : modèle d'un aiguillage .....	104
Figure II.35 : module de connexion .....	105
Figure II.36 : communication gammes opératoires / modèle du système de transport .....	106
Figure II.37 : schéma de la commande de coordination de l'exemple .....	107
Figure II.38 : la coordination au sein de la commande .....	108
Figure II.39 : communication coordination / interface du procédé .....	111

### CHAPITRE III :

#### LA VALIDATION DES MODÈLES DE LA COMMANDE DE COORDINATION

Figure III.1 : modèles des ressources .....	117
Figure III.2 : graphe de synchronisation .....	118
Figure III.3 : modèle intermédiaire de l'îlot 2 .....	119
Figure III.4 : modèle final de l'îlot 2 .....	121
Figure III.5 : réduction du modèle de l'îlot 2 : étape 1 .....	122
Figure III.6 : réduction du modèle de l'îlot 2 : étape 2 .....	124
Figure III.7 : élimination des transitions identiques .....	125
Figure III.8 : réduction du modèle de l'îlot 2 : modèle final .....	126
Figure III.9 : modèle de l'îlot 1.....	127

Figure III.10 : modèle réduit de l'îlot 1 .....	128
Figure III.11 : îlot avec ressource partagée .....	128
Figure III.12 : modèle de l'îlot avec ressource partagée.....	129
Figure III.13 : élimination des places implicites .....	130
Figure III.14 : regroupement des places .....	131
Figure III.15 : fusion des places de type "protocole".....	132
Figure III.16 : fusion des transitions identiques .....	133
Figure III.17 : modèle réduit de l'îlot avec ressource partagée .....	134
Figure III.18 : intégration d'un mode opératoire extrême .....	136
Figure III.19 : saturation de l'îlot 2 .....	137
Figure III.20 : îlot 1 avec représentation du système de transport.....	139
Figure III.21 : saturation de l'îlot 1 .....	139
Figure III.22 : saturation de l'îlot dans deux modes opératoires .....	141
Figure III.23 : modèle d'un îlot abstrait.....	143
Figure III.24 : saturation du modèle abstrait .....	145
Figure III.25 : gestion de l'encours sur les gammes opératoires .....	147
Figure III.26 : limitation de l'encours local dans le cas d'une production mixte .....	149
Figure III.27 : modèle du convoyeur (extrait).....	151
Figure III.28 : graphe d'état du convoyeur .....	152
Figure III.29 : modèle de la cellule de l'Ecole Centrale de Lille .....	155
Figure III.30 : gammes opératoires avec représentation des contraintes de non blocage issues de l'analyse des îlots 1, 2 et 3.....	157
Figure III.31 : limitation de l'encours sur les gammes opératoires .....	160
Figure III.32 : blocage des gammes opératoires .....	162
Figure III.33 : éviter le blocage .....	162

## ANNEXES

Figure A.1 : les primitives de ACTIGRAMMES.....	168
Figure A.2 : la cellule de l'Ecole Centrale de Lille .....	170
Figure A.3 : génération de l'îlot 3 : étape 1 .....	179
Figure A.4 : génération de l'îlot 3 : étape 2a .....	180
Figure A.5 : génération de l'îlot 3 : étape 2b .....	181
Figure A.6 : modèle de l'îlot 3 utilisé pour l'analyse de la cellule.....	190
Figure A.7 : temporisation des transferts .....	192