

N° d'ordre : 291

50376  
1988  
203

50376  
1988  
203

# THESE

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE FLANDRES ARTOIS

pour obtenir le grade de

**DOCTEUR DE L'UNIVERSITE**

en

**PRODUCTIQUE, AUTOMATIQUE ET INFORMATIQUE  
INDUSTRIELLE**

par

**Michel KAPUSTA**  
Ingénieur IDN

**GENERATION ASSISTEE D'UN GRAPHE FONCTIONNEL**  
**DESTINE A L'ELABORATION STRUCTUREE**  
**DU MODELE DE LA PARTIE COMMANDE**  
**POUR LES CELLULES DE PRODUCTION FLEXIBLES**  
**DANS L'INDUSTRIE MANUFACTURIERE**

soutenu le 5 Décembre 1988 devant la Commission d'Examen

Membres du Jury :

M. PRUNET	Rapporteur
M. STAROSWIECKI	Rapporteur
M. CASTELAIN	Examineur
M. CORBEEL	Examineur
M. GENTINA	Examineur, Directeur de Thèse
M. BOUREY	Invité
M. BRARD	Invité
M. THUREL	Invité

**AVANT - PROPOS**

Le travail présenté dans ce mémoire a été effectué au Laboratoire d'Automatique et d'Informatique Industrielle de l'INSTITUT INDUSTRIEL DU NORD sous la direction de Monsieur GENTINA, Professeur à l'IDN. Je tiens à le remercier de m'avoir accueilli dans son équipe de recherche.

Je tiens également à remercier

Monsieur PRUNET, Professeur à l'Université de MONTPELLIER,

Monsieur STAROSWIECKI, Professeur à l'Université des Sciences et Techniques de LILLE, d'avoir accepté d'être les rapporteurs de cette thèse,

et

Monsieur BOUREY, Enseignant-chercheur à l'IDN,

Monsieur CASTELAIN, Maître de Conférences à l'IDN,

Monsieur CORBEEL, Maître de Conférences à l'IDN,

pour l'honneur qu'ils me font en participant à mon Jury de Thèse.

Je suis en outre très heureux de la présence à ce Jury de Monsieur BRARD, Ingénieur de recherches à la TELEMECANIQUE, et de Monsieur THUREL, Directeur de la branche Automatique et Informatique Industrielle de la Division Recherche et Développement de cette entreprise.

Je voudrais ici remercier tous les membres du Laboratoire d'Automatique et d'Informatique Industrielle de l'IDN, et en particulier Messieurs Jean-Pierre BOUREY, Emmanuel CASTELAIN et Etienne CRAYE pour l'aide précieuse qu'ils m'ont apportée sur le plan scientifique et sur le plan humain.

Enfin, je remercie les personnes qui m'ont aidé lors de la réalisation matérielle de ce mémoire : Madame VERIN pour la dactylographie et Monsieur VANGREVENINGE pour la reprographie.

**SOMMAIRE**

INTRODUCTION GENERALE	page 19
CHAPITRE I	page 31
CHAPITRE II	page 87
CHAPITRE III	page 299
CONCLUSION GENERALE	page 389
BIBLIOGRAPHIE	page 395
ANNEXES	page 413

## **CHAPITRE I LE PROJET CASPAIM**

	pages
INTRODUCTION .....	33
<b>I - LES MODELES .....</b>	<b>35</b>
I.1 - Introduction .....	35
I.2 - La Partie Commande .....	37
I.2.1 Un modèle orienté vers l'implantation .....	37
I.2.2 Présentation informelle .....	38
a) Exemple .....	38
b) Modèle de base .....	39
c) Les Réseaux de Petri structurés (RdPS) .....	42
d) Les Réseaux de Petri structurés et colorés (RdPSC) .....	48
d.1) présentation simplifiée de la coloration .....	48
d.2) les réseaux à prédicats .....	51
e) Les Réseaux de Petri structurés adaptatifs et colorés (RdPSAC) ....	55
I.3 - Le Niveau Hiérarchique .....	59
I.4 - Le Procédé .....	65
I.5 - Conclusion sur les modèles .....	67
<b>II - LA SIMULATION .....</b>	<b>69</b>
II.1 - Introduction .....	69
II.2 - Le simulateur .....	70
II.3 - Conclusion .....	72
<b>III - UNE DEMARCHE COMPLETE DE CONCEPTION ASSISTEE DEPUIS LES   SPECIFICATIONS JUSQU'A L'IMPLANTATION .....</b>	<b>73</b>
III.1 - Les questions posées .....	73
III.1.1 Le besoin d'une méthodologie de conception assistée des modèles .....	73
III.1.2 L'interprétation difficile des résultats de simulation .....	74
III.1.3 La réalisation de systèmes de commande implantés : l'objectif industriel de la démarche .....	74
III.2 La chaîne de C.A.O. ....	78
III.2.1 La démarche de spécification .....	78
III.2.2 La modélisation .....	80
III.2.3 La simulation .....	81
III.2.4 L'implantation .....	82
<b>CONCLUSION .....</b>	<b>83</b>

**CHAPITRE II**      PREMIERE ETAPE DE CONCEPTION ASSISTEE  
DE LA PARTIE COMMANDE :  
OBTENTION DU PREGRAPHE

	pages
INTRODUCTION .....	89
I - LE FORMALISME DE DESCRIPTION DES GAMMES OPERATOIRES .....	91
II - L'ANALYSE DE COHERENCE ET DE COMPLETUDE	
DE L'ENCHAINEMENT LOGIQUE DES REGLES OPERATOIRES .....	119
II.1 - Présentation générale .....	119
II.2 - Exemple .....	130
II.3 - Plan du moteur d'inférence. Survol de quelques fonctionnalités utiles de l'analyseur .....	221
II.3.1 Le corps principal .....	221
II.3.2 Compléments de l'analyse principale .....	239
II.4 - Remarque sur le problème du contrôle de cohérence dans les bases de connaissances .....	240
III - LA TRADUCTION ET L'AGREGATION DES REGLES OPERATOIRES	
SOUS FORME D'UN GRAPHE RdP COLORE : LE PREGRAPHE .....	243
III.1 - Conditions d'agrégation .....	243
III.2 - Méthode d'agrégation .....	245
III.3 - Un exemple d'utilisation .....	247
III.4 - Les variables libres locales du Prégraphe .....	248
IV - LES ETAPES DE MODULATION .....	275
IV.1 - La multiplication de sites .....	275
a) A partir de la description sous forme de Prégraphe .....	275
b) A partir de la description sous forme de règles opératoires .....	277
IV.2 - La multiplication de blocs ou ensembles de sites .....	277
a) A partir de la description sous forme de Prégraphe .....	277
b) A partir de la description sous forme de règles opératoires .....	278
IV.3 - L'ajout de lieux de stockage intermédiaires .....	279
IV.4 - L'agrégation de sites .....	280
IV.5 - Un mot sur la flexibilité .....	281
V - L'ETUDE SIMPLIFIEE DE LA COMBINATOIRE DES GAMMES .....	285
V.1 - La méthode présentée par Alain BOURJALUT .....	287
V.2 - Adaptation .....	290
a) Obtention des contraintes d'antériorité .....	290
b) Etude combinatoire en fonction des contraintes .....	290
CONCLUSION .....	295

**CHAPITRE III**      UTILISATION DANS LA CHAINE DE CONCEPTION  
DU PROJET CASPAIM DE LA BASE DE TRAVAIL  
FOURNIE PAR LE PREGRAPHE

	pages
INTRODUCTION .....	301
<b>I - STRUCTURATION, A PARTIR DU PREGRAPHE, DU MODELE RdPSAC</b>	
<b>DE LA PARTIE COMMANDE .....</b>	<b>303</b>
I.1 - L'idée générale .....	303
I.2 - Les modules associés aux places du Prégraphe .....	304
I.3 - Les modules de liaison .....	320
I.4 - Réduction du parallélisme de commande .....	325
I.4.1 Problèmes d'exclusion mutuelle .....	325
I.4.2 Problèmes de synchronisation .....	327
I.5 - Niveaux de structuration terminaux .....	330
I.6 - Conclusion sur la structuration .....	336
<b>II - LA QUESTION DE L'EXTENSION DU MODELE DE LA PARTIE COMMANDE .....</b>	<b>341</b>
II.1 - Le problème posé .....	341
II.2 - Présentation d'un exemple .....	343
II.3 - Problèmes rencontrés pour la modélisation de la partie commande .....	346
II.4 - Prégraphe .....	348
II.5 - Graphe développé .....	352
II.6 - Discussion d'un petit exemple pouvant concerner éventuellement les réseaux à prédicats .....	353
II.7 - Remarque sur le maintien de la cohérence entre les différentes images du procédé .....	354
II.8 - Conclusions .....	355
<b>III - LES PROBLEMES DE PREDIMENSIONNEMENT DE L'ARCHITECTURE DES UNITES DE PRODUCTION ET DE PREETUDE SIMPLIFIEE DE LEUR COMPORTEMENT : LE ROLE POSSIBLE DU PREGRAPHE .....</b>	<b>359</b>
III.1 - Démarches analytiques et simulation : présentation générale du problème.	359
III.2 - Les problèmes liés au projet CASPAIM : ébauche d'une démarche de prédimensionnement .....	363
III.2.1 Introduction .....	363
III.2.2 Les essais actuels .....	364
a) Etude du régime permanent .....	365
a.1) calcul des débits moyens en régime stationnaire sans files d'attente .....	365
a.2) les files d'attente .....	375
a.3) étude des stratégies de production en régime permanent .....	377
b) Etude des régimes non permanents .....	378
b.1) affinement par simulation de la détermination des limites d'un régime permanent .....	378
b.2) les unités de production flexibles .....	379
b.3) les outils de simulation .....	381
<b>CONCLUSION .....</b>	<b>385</b>

**ANNEXES**

	pages
ANNEXE 1 : Informations complémentaires relatives au choix du modèle Prégraphe du paragraphe II.4, Chapitre III Partie II .....	4 1 5
ANNEXE 2 : Exemples de fonctions du Niveau Hiérarchique pour l'exemple du Chapitre III Partie II .....	4 2 1
ANNEXE 3 : Discussion d'un petit exemple industriel .....	4 3 1
ANNEXE 4 : Notes sur le projet PTA .....	4 5 3
ANNEXE 5 : Echantillon des règles de transposition RdPSAC->GRAFCET .....	4 5 7

**INTRODUCTION GENERALE**

## INTRODUCTION GENERALE

La fin des années 70 et le début des années 80 ont vu un ralentissement très important du taux de croissance des grands pays industrialisés dont les marchés se sont approchés de la saturation. Ces limites ont poussé les entreprises à exacerber une stratégie mondiale dans laquelle se sont intégrés également des pays nouvellement développés à faible coût de main d'oeuvre (ISTO 79).

Retenons simplement pour ce qui nous intéresse ici que :

- i) La demande qui était jusqu'alors relativement insensible à la conjoncture est devenue au début des années 80 sujette à de brusques variations cycliques liées aux cycles de l'économie mondiale.
- ii) L'augmentation de la productivité est devenue un enjeu essentiel qui a déclenché une série d'innovations technologiques spectaculaires liées à l'automatisation des processus de production.

C'est dans ce contexte que doivent se placer les notions plus techniques qui vont suivre.

## 1. LA NOTION DE FLEXIBILITE :

- La diversification de la production et la multiplication des variantes autour d'un produit de base qui sont des tendances de marché récentes ont progressivement amené les concepteurs d'unités de production à abandonner le concept de machine spécifique.
- En outre, la rotation de plus en plus rapide des produits fait qu'il est devenu impossible d'amortir le prix d'une machine spécifique pendant la durée de vie économique du produit.

La nécessité est alors apparue de concevoir des machines polyvalentes et des architectures de commande rapidement reconfigurables, et à plus forte raison encore si la production se fait en petite et moyenne série. (Rappelons qu'environ 75 % des pièces mécaniques fabriquées le sont à moins de 50 exemplaires (IFRO 84)).

- Par ailleurs, la recherche de la diminution des délais de production afin :
  - i) de créer la valeur ajoutée au dernier moment
  - ii) d'épouser le plus rapidement possible une variation de la demande a fait apparaître la nécessité
    - d'accroître de façon très forte le taux d'engagement des machines par la continuité du flux de leur approvisionnement.
    - de prévoir tous les cheminements possibles d'une pièce dans un atelier (avec délestages et choix multiples d'itinéraires).
- Enfin, la diminution des en-cours et des stocks intermédiaires qui coûtent très cher et dont la gestion technique et financière accroît notablement le prix de revient du produit (parfois de 40 % (BON 85)) a créé la nécessité d'assurer à tout moment l'orientation et l'engagement d'une pièce vers un processus de transformation et d'éviter son immobilisation dans un stock d'attente.

Ces problèmes d'amélioration importante de productivité dans un contexte d'offre et de demande à évolution très rapide ont donc amené les industriels au concept d'atelier flexible automatisé.

Les notions économiques et technologiques de flexibilité sont souvent employées de façon très floue actuellement dans le domaine industriel car

les techniques de pointe qu'elles recouvrent sont l'objet d'opérations de prestige. On peut trouver toutefois un cadre d'analyse et de classification économique et technologique dans un nombre important d'études. Citons seulement les travaux des économistes français CORIAT et BOYER ([COR 86]) et américain GERWIN ([GER 83]). Un article paru en février 1988 dans la revue anglaise CIMS ([SLA 88]) présente un cadre d'identification des différents types de flexibilité de production. Une classification qui a souvent servi de référence a été introduite par GERWIN et LEUNG en 1980. Elle distingue cinq notions :

- 1) **Design Change Flexibility** : qui se réfère à la possibilité de modifier le process en fonction de caractéristiques particulières qui peuvent être introduites dans le même produit de base.
- 2) **Mix Flexibility** : qui a trait à la possibilité de produire simultanément sur la même ligne différents produits alternatifs qui ont en commun certains éléments de base.
- 3) **Part Flexibility** : qui concerne le fait que l'ensemble du process peut être simplifié et compliqué par retrait ou ajout d'opérations productives, à des coûts nuls ou faibles.
- 4) **Volum Flexibility** : qui renvoie à la possibilité que possède le process de faire face à des fluctuations quantitatives correspondant à des variations de la demande.
- 5) **Routing Flexibility** : qui se réfère à la capacité de convoier le produit de façon souple à travers des segments du processus libres ou sous-engagés.

La réalisation et la gestion de tels ateliers soulèvent immédiatement des problèmes techniques très complexes. En effet, sont apparues avec l'étude de ces ateliers deux notions encore mal maîtrisées tant au niveau des méthodologies de conception qu'au niveau des technologies de réalisation. Il s'agit des notions d'intégration et de structuration des systèmes de commande.

## 2. L'INTEGRATION ET LA STRUCTURATION DES SYSTEMES DE COMMANDE

### a) Intégration

La recherche d'efficacité dans les techniques de production se traduit surtout actuellement par l'échange, le partage et la mise à jour automatiques des informations entre les machines et entre les différents services de l'entreprise. Les différents intervenants du cycle de production (et de conception) n'ont plus un fonctionnement autonome mais coordonné. C'est la notion d'usine intégrée. Les techniques liées aux réseaux locaux permettent maintenant de faire communiquer des matériels informatiques différents : calculateurs, automates programmables et commandes numériques DNC (Direct Numerical Control) au travers de réseaux de communication hiérarchisés. Signalons au passage que la construction de bases de données partageables pour des utilisations très différentes (par ex. : conception, production, maintenance, gestion commerciale) est encore très problématique (c'est un des problèmes abordé dans le cadre du projet BASEPTA - Base Application et Standard d'Echange du Poste de Travail de l'Automaticien - voir Annexe 4). Retenons seulement pour l'utilisation qui va en être faite ici que l'intégration du système de production se manifeste par l'apparition d'un (ou plusieurs) niveaux de supervision paramétrant le système de pilotage de l'appareil de production en lui imposant des stratégies ou des modes de marche dépendant d'une vision d'ensemble de l'état du système de pilotage, de l'état du procédé ou de données économiques ou sociales provenant des autres services de l'entreprise ou du monde extérieur.

La nécessité de systèmes de coordination permettant d'intégrer de tels flux d'information est apparue avec l'évolution des méthodes de gestion de production - planification, ordonnancement, gestion de stocks - (méthodes MRP, JIT, ou OPT (1)) vers une informatisation complète et vers un suivi continu de la gestion. Cette notion de suivi en "temps réel" aboutit actuellement au concept d'usine intégrée (ce que les anglo-saxons appellent CIM ou Computer Integrated Manufacturing) (BEN 86 | ING 88).

L'intégration des systèmes de commande pose dès la phase de conception le problème du choix des outils de modélisation. Ils ont pour but, en effet, de permettre une implantation finale propre après une étape d'évaluation (en général par simulation). La diversité des paramètres à intégrer à la représentation a mis en évidence les limites d'outils de conception et d'implantation aussi perfectionnés et aussi puissants que le GRAFCET.

NB : (1) MRP      MRP-1      Material Requirement Planning  
             MRP-2      Manufacturing Resource Planning  
JIT =      Just In Time (par ex. : méthode KANBAN du flux tendu)  
OPT =      Optimized Production Technology  
voir |BEN 87| |LAM 87|

b) Structuration

La notion d'usine intégrée s'accompagne en outre d'une augmentation spectaculaire de la taille et de la complexité des systèmes de pilotage dont la maîtrise fait intervenir des problèmes liés à la structuration et à la modularité de ces systèmes. On retrouve ici au sein des automatismes industriels un problème qui s'est posé en termes toutefois moins complexes pour les langages de programmation informatiques à la fin des années 60 et en particulier avec l'apparition du langage PASCAL.

La structuration des systèmes de commande est un problème actuel et crucial. C'est un des points importants sur lequel s'est concentré le groupe de travail "systèmes logiques" de l'AF CET lors de récentes réunions. Rappelons que ce sont les travaux de ce groupe associés à ceux de l'ADEPA qui ont donné naissance à la norme GRAFCET, norme française depuis novembre 1982 et internationale depuis novembre 1987 (|BLA 86| |BRA 79| |GRE 85| |THE 82a| |THE 82b|) (L'outil de spécification et de programmation que constitue le GRAFCET, enrichi d'un nouvel outil de spécification appelé GEMMA (Guide d'Etude des Modes de Marche et d'Arrêt), permet de décrire des automatismes séquentiels complexes à l'aide d'une démarche structurée et descendante (|BOU 87|). En particulier :

- i) la macro-représentation ou la représentation de macro-étapes répond au besoin d'approche structurée et progressive des applications industrielles complexes.
- ii) le GEMMA et la notion de forçage hiérarchisé, qui permet à un graphe de niveau supérieur de forcer la situation d'un graphe inférieur, ont rendu naturelle l'existence d'une partie commande hiérarchisée. Le GRAFCET complet comprend en effet trois niveaux que sont le GRAFCET de Sécurité, le GRAFCET de Conduite et le GRAFCET de Production Normale).

Quels que soient les outils de conception et de réalisation, la hiérarchisation du système de commande est un des points sur lesquels tous les intervenants semblent s'entendre actuellement, sans s'accorder toutefois sur la façon dont elle doit être organisée.

La difficulté soulevée par le problème de la structuration doit être certainement rattachée aux trois points suivants :

1°) Les méthodes de construction modulaire peuvent être descendantes (à complexité croissante) ou montantes (intégrantes). L'approche la plus reconnue est l'approche descendante, mais elle exige une vision globale du problème et de sa solution, ce qui est très délicat vu la taille du problème posé par les systèmes de production actuels. Les approches les plus fructueuses ont consisté jusqu'à maintenant à mêler, au sein de démarches essai-erreur itératives, des phases descendantes et montantes.

2°) Le choix des limites et des interfaces de chaque module est complexe. En effet, les caractéristiques de flexibilité et d'intégration rendent certains problèmes très imbriqués. Par exemple, la modélisation et la réalisation du système de commande exigent parfois une image très fine de la partie opérative et du procédé réel. On constate ainsi des différences dans la façon dont est appréhendée la décomposition modulaire au sein des différents travaux qui se sont attaqués au problème. Citons par exemple les projets

SECOIA et CONTROLE DE PRODUCTION du LAAS à TOULOUSE  
|SAH 87| |ATA 87a| |ATA 87b| |VAL 88|

PIASTRE du LAM à MONTPELLIER |PRU 87a| |PRU 87b|

CASPAIM du LAII-IDN à LILLE (voir chapitre I)

et les travaux

du LACN à NANCY |LHO 88|

du LAG à GRENOBLE |MER 88|

et du département d'Ingénierie Electrique et Informatique de  
l'Université de SARAGOSSE |MAR 87| |MAR 88| |VIL 88|.

3°) La démarche de structuration doit intervenir non seulement pour la spécification de la solution mais aussi pour la spécification du problème et ces deux démarches doivent être intimement liées. En effet :

i) La taille des problèmes devient telle qu'il est indispensable de savoir dès le départ à quel niveau doivent se poser certaines questions. C'est le sens de l'apparition des guides concepteurs qui proposent, au travers d'une interface utilisateur puissante, des outils de structuration du cahier des charges (|AUG 87|, |BEN 86|).

ii) La spécification du problème ne peut être séparée de la spécification progressive de sa solution. C'est là que réside le point faible des guides concepteurs et des techniques classiques de gestion de projet qui construisent un cahier des charges précisant des points qui ne peuvent en fait intervenir qu'après un certain degré d'étude de la solution. Cette

"précipitation" envisageable sur des projets classiques dont une certaine expérience est déjà acquise est inadaptée à l'étude de questions nouvelles qui sont encore l'objet de recherches.

En résumé, disons que les problèmes posés par les notions d'intégration et de structuration sont relatifs à la représentation des systèmes de commande intégrant des exigences très diverses et à la façon de construire proprement cette représentation.

En face de toutes ces questions mal résolues, la construction de cellules et d'ateliers flexibles de production est restée très problématique :

- i) Les machines-outils sont choisies essentiellement pour les opérations qu'elles savent faire sur les pièces et non pas pour la souplesse et l'ouverture de leur système de pilotage. L'hétérogénéité et l'absence actuelle de standardisation des systèmes de commande locaux des machines ont conduit à une résolution empirique des problèmes de coordination inter-machines dont l'analyse, confiée aux fournisseurs différents, a souvent eu des résultats aux performances décevantes.
- ii) L'absence de méthodes de conception, d'évaluation et d'implantation, et l'évolution rapide des matériels ont rendu trop limitées, trop locales, de nombreuses réalisations. L'étude et la mise au point des systèmes de commande ont nécessité des investissements trop spécifiques qu'il faut réitérer à chaque nouvelle entreprise, d'autant plus que la part logicielle devient exorbitante en comparaison des investissements consentis pour la réalisation mécanique (parfois jusqu'à 50 % (IDEF 85)). En effet, un des points critiques des ateliers flexibles est encore pour longtemps certainement l'absence d'outil théorique d'évaluation. Les techniques de validation se reportent alors sur des simulations qui nécessitent évidemment des modèles de simulation. Tout ceci constitue des investissements de réalisation indirects très importants.
- iii) Enfin, l'absence d'une méthodologie de construction constituant un standard de communication pose des problèmes énormes de concertation. Les différents corps de métier ont besoin d'un standard d'échanges de plus en plus pointu. Ce problème a constitué un des obstacles connus de l'atelier CITROEN de MEUDON et il justifie pleinement des projets tels que PTA (Poste Travail Automaticien). Actuellement, c'est aussi un des sujets de réflexion essentiels de la commission "Systèmes Logiques" de l'AFCEP.

C'est dans ce contexte que sont nés un certain nombre de projets issus de la recherche universitaire et de sa concertation avec le monde industriel, visant à construire, par l'intermédiaire de méthodes de conception rigoureuses et systématiques les systèmes de commande des ateliers de production modernes. Citons PIASTRE du LAM à MONTPELLIER (|PRU 87a| |PRU 87b|), SECOIA du LAAS à Toulouse (|SAH 87|) et enfin le projet CASPAIM développé au Laboratoire d'Automatique et d'Informatique Industrielle de l'I.D.N. et dont il va être question ici.

Le projet CASPAIM (Conception Assistée de Systèmes de Production pour l'Industrie Manufacturière) s'intéresse à l'élaboration d'une chaîne modulaire de C.A.O. prenant en charge la conception, l'évaluation et la réalisation de systèmes de pilotage de cellules flexibles de production depuis la phase de spécification du cahier des charges jusqu'à l'implantation finale. Le but ultime du projet est d'amener l'utilisateur à définir progressivement les spécifications à l'endroit où elles doivent intervenir, en langage accessible et clair, au fur et à mesure que le logiciel élabore une solution d'une façon aussi transparente que possible.

Ce projet est bien sûr ambitieux. Il est le fruit du travail d'une équipe et son état d'avancement actuel donne des résultats encourageants. Plusieurs composantes du projet ont déjà fait l'objet de soutenances de thèses (|CAS 87|, |KAR 87|, |BOU 88|). Le travail qui va être présenté dans ce mémoire est une nouvelle composante du projet. Il s'intéresse à une première étape de la conception assistée du système de commande et permet, à partir de données issues du cahier des charges, de construire de façon assistée le schéma de montage de la partie du système de commande qui représente l'ensemble des commandes applicables au procédé.

**Ce mémoire va être composé de trois chapîtres :**

- Le chapitre I présentera le projet CASPAIM et la situation au sein de ce projet du travail spécifique qui fait l'objet de ce mémoire. En particulier, la décomposition du système de commande en deux niveaux principaux (Partie commande Réseau de Petri, Niveau Décisionnel) sera précisée.

- **Le chapitre II** permettra de décrire ce travail qui constitue une première étape de conception assistée du modèle Réseau de Petri de la Partie Commande. On montrera comment à partir d'une étude des gammes opératoires peut être obtenu le graphe Réseau de Petri préliminaire appelé Prégraphe récapitulant le rôle fondamental de l'unité de production et constituant le plan de l'assemblage ultérieur automatique du modèle de la Partie Commande.
  
- **Le chapitre III** sera consacré au rôle du Prégraphe et aux développements en cours. Il y sera montré comment le modèle de la Partie Commande est obtenu à partir du Prégraphe par une étape de développement structuré. Les problèmes limites de modélisation seront présentés ensuite. Seront également discutés des projets d'extension de l'utilisation de ce prégraphe à des fins de prédimensionnement de l'architecture matérielle et de préétude du comportement des unités de production flexibles étudiées.
  
- **Une annexe** apportera un complément d'information à tous les points qui n'auront pas pu être développés au sein du mémoire.

## CHAPITRE I

### LE PROJET CASPAIM

## INTRODUCTION

Le travail qui est l'objet de ce mémoire concerne une première étape de conception assistée de la partie commande d'unités flexibles de production. Il fait partie d'un projet plus vaste, le projet CASPAIM. Ce projet s'intéresse à la Conception Assistée de Systèmes de Production Automatisés pour l'Industrie Manufacturière. Il est plus particulièrement orienté actuellement vers la conception de cellules flexibles pour des productions en petite et moyenne série, voire unitaires (par exemple du type de celles de l'atelier CITROEN de MEUDON pour la fabrication de prototypes d'étude de moteurs).

Ce projet est le fruit d'une petite équipe de recherche. Il est l'objet d'un débat collectif permanent. Chaque chercheur a pris sous sa responsabilité une composante du projet qui ne peut être comprise que dans la structure d'ensemble. C'est pour cette raison qu'il est nécessaire de présenter le projet CASPAIM dans son état d'avancement actuel. Ceci permettra :

- i) d'une part, de situer correctement l'apport spécifique du mémoire au sein du projet.
- ii) d'autre part, de situer le projet CASPAIM lui-même dans le contexte des travaux en cours autour des notions présentées dans l'introduction générale.

**Le chapitre I est constitué de trois parties :**

- La première partie présente les modèles qui ont été choisis pour représenter de façon complémentaire le système de commande et le procédé des unités de production étudiées.
- La deuxième partie présente un outil de simulation qui a été développé (ICAS 87b) autour de ces modèles afin de les évaluer.
- La troisième partie montre comment une démarche complète de conception assistée depuis les spécifications jusqu'à l'implantation du système de commande s'est développée autour du noyau originel. En particulier, le processus de conception assistée de la partie commande, aujourd'hui achevé, est introduit. C'est dans ce processus que s'inscrit la contribution de ce mémoire. L'état d'avancement actuel et les objectifs du projet CASPAIM sont également présentés.

## I. LES MODELES

### L1 INTRODUCTION

Le choix et l'élaboration des modèles de représentation du système de pilotage pour unité flexible de production s'est fait progressivement au L.A.I.I. depuis 1979. (\*)

Encouragée par le renouveau des modèles à événements discrets (Réseaux de PETRI, GRAFCET), l'idée est née à cette époque de perfectionner les modèles Réseaux de PETRI existants afin :

- i) d'approcher au plus près la réalité des systèmes de contrôle des processus industriels dont l'exigence de polyvalence s'accroissait progressivement.
- ii) de donner à ces modèles une structure syntaxique permettant d'étendre à ces outils de description les caractéristiques de la programmation structurée, facilitant par là même leur construction et leur validation.

Les modèles utilisés au L.A.I.I. n'ont, depuis lors, cessé d'évoluer en s'enrichissant des notions de flexibilité, d'intégration ou paramétrisation, et de structuration qui ont été présentées dans l'introduction générale.

Le modèle général actuel de représentation des systèmes de production flexibles utilisé pour le projet CASPAIM est présenté ci-dessous.

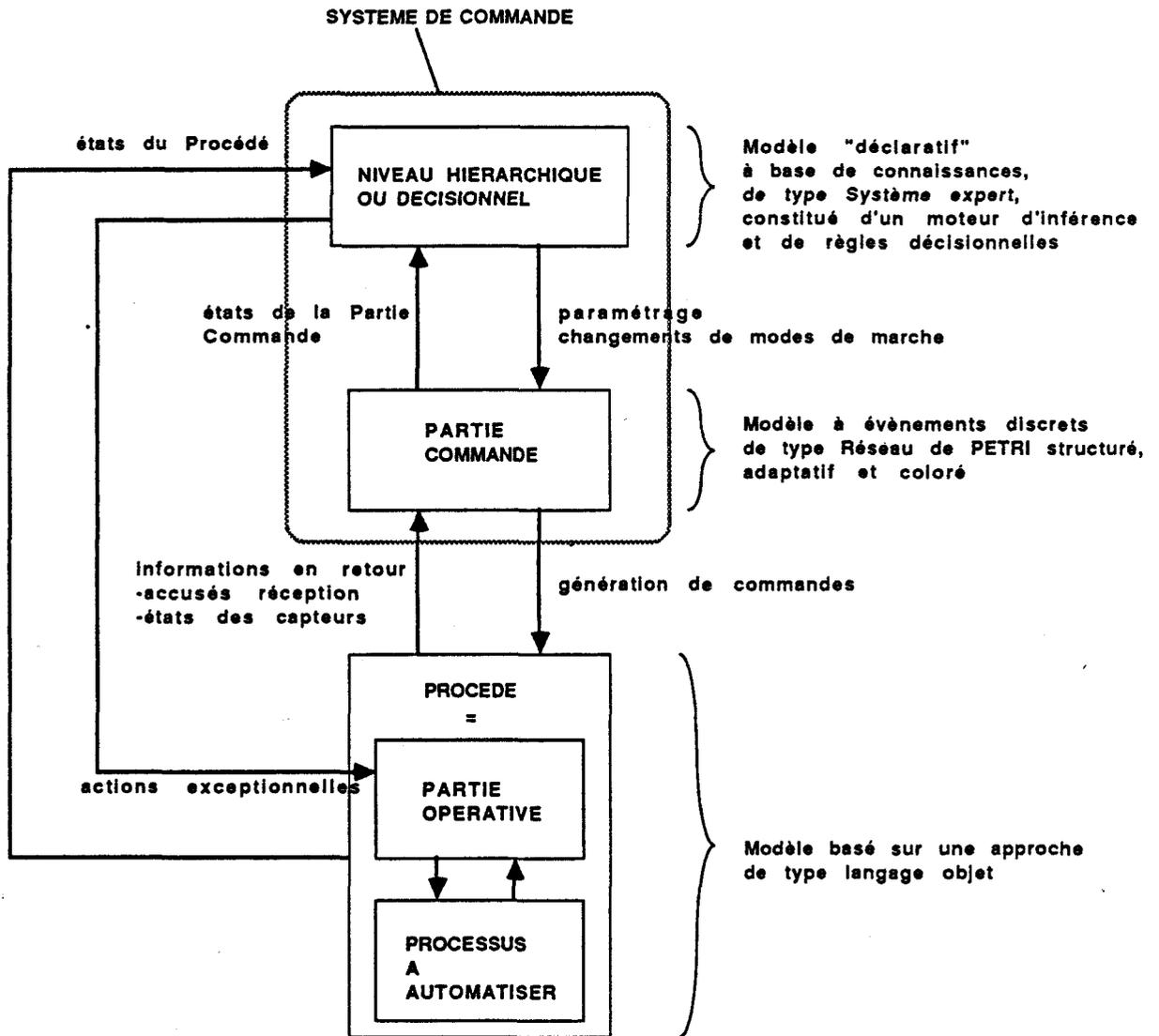


Figure I.1

**PARTIE COMMANDE :** Son rôle est d'assurer la coordination et le séquençage des commandes applicables à la Partie Opérative.

**NIVEAU DECISIONNEL :** Son rôle est de construire les décisions et les choix destinés à résoudre les indéterminations de la Partie Commande.

**PROCEDE :** Le Procédé est constitué de la Partie Opérative (actionneurs, capteurs) et de l'ensemble des équipements et matériels intervenant dans le processus de production (machines, pièces, outils, et éventuellement automates, programmes,...)

NB (\*) Pour toute l'évolution des travaux du LAII sur la commande discrète des processus industriels, voir :  
COR 79		COR 80		COR 81		COR 83a		COR 83b		COR 83c		COR 84		COR 85a
COR 85b		VER 82		CAS 85		CAS 87a		CAS 87b		CAS 87c		GIR 84		GEN 87a
GEN 87b		BOU 86		BOU 87b		BOU 87c		BOU 88a		BOU 88b		BOU 88c		KAR 85
KAR 86		KAR 87		KAP 87	.									

## 1.2 LA PARTIE COMMANDE

### 1.2.1. Un modèle orienté vers l'implantation

Nés au début des années 60, les Réseaux de PETRI ont connu un regain d'intérêt dans les années 70 pour la modélisation des systèmes de contrôle de processus industriels discrets, des réseaux de communication et des bases de données réparties. La maîtrise de la complexité de ces systèmes exige leur décomposition en sous-systèmes qui peuvent ou qui doivent fonctionner en parallèle, de façon asynchrone. Ces modèles permettent d'aborder, dès la phase de spécification, les problèmes de synchronisation caractéristiques de la coopération et de la coordination des matériels, de la composition des services et de l'optimisation de l'emploi des ressources. Malgré l'apparition de modèles récents liés aux outils de l'Intelligence Artificielle (langages acteurs (|TAH 87|, et |CAV 87| pour le langage OASYS), pour le langage systèmes experts (|PEC 87| |BEL 86| |BON 86|)) les Réseaux de PETRI restent sur le plan formel les modèles les plus avancés pour la représentation et la description logiques des structures de contrôle des processus discrets (|BRA 83a| |BRA 83b| |COU 86|).

Par ailleurs, la représentation par un modèle et la simulation informatique ne constituent pas une fin en soi, bien qu'elles permettent d'imaginer et de valider le comportement du système modélisé. Le but ultime proposé par le projet CASPAIM est l'implantation du modèle sur dispositifs programmés industriels et plus spécialement sur automates programmables.

Or, le GRAFCET (Graphe Fonctionnel de Commande Etapes-Transitions), outil graphique de spécification d'automatismes séquentiels est un modèle de type graphe à événements discrets issu des Réseaux de PETRI, et ce modèle est devenu un outil de programmation des automatismes séquentiels sur automates programmables. Bénéficiant depuis novembre 82 d'une norme française et depuis novembre 87 d'une norme internationale, le GRAFCET est devenu une référence industrielle pour les automatismes séquentiels (|BLA 86| |BOU 87a| |BRA 79| |GRE 85| |THE 82a| |THE 82b|). Bien que le modèle Réseau de PETRI développé pour représenter les cellules flexibles de production au sein du projet CASPAIM soit plus puissant que le GRAFCET, la transposition reste envisageable à l'aide de quelques primitives de transposition. Ce passage a d'ailleurs été réalisé dans le cadre du projet, pour plusieurs exemples de tailles intéressantes, sur la gamme des automates TSX 67 TELEMECANIQUE (|BEA 87| |CRU 87| |BOI 88| |TEL 87|).

Le modèle Réseau de PETRI Structuré Adaptatif Coloré (RdPSAC) choisi ici et dont les propriétés vont être brièvement rappelées maintenant constitue donc un outil unique depuis la conception jusqu'à la réalisation de la partie Commande en passant par les phases de modélisation, de validation par simulation et enfin d'implantation.

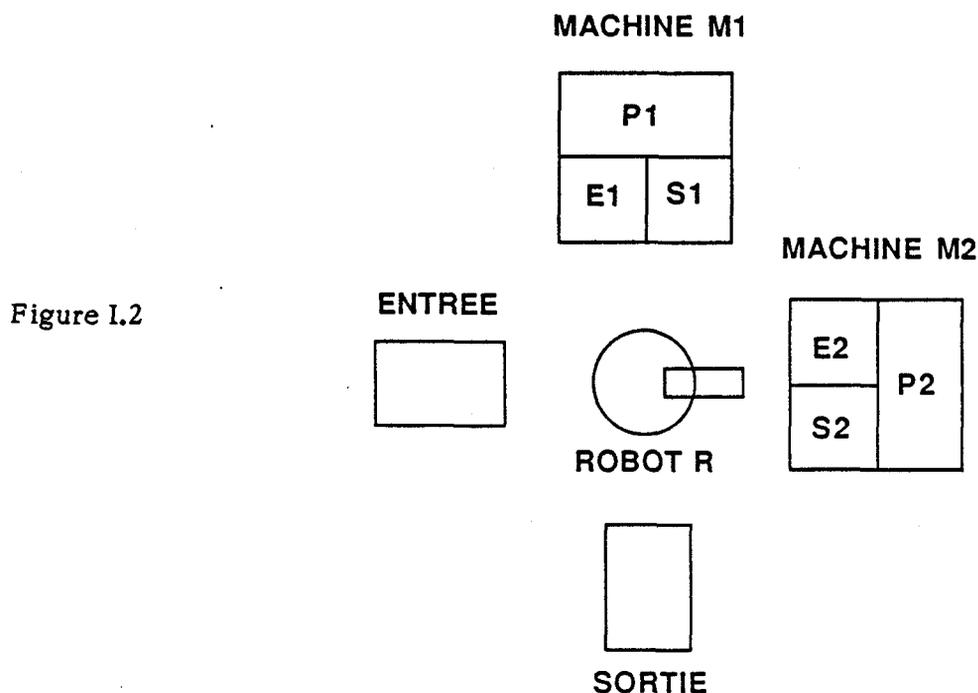
### 1.2.2. Présentation informelle

Certaines des notions qui vont être données maintenant ont été développées en détail dans deux thèses déjà présentées autour du projet CASPAIM (|CAS 87b| |BOU 88b|). Les rappels qui seront alors faits ne constituent qu'un résumé non formel. Seuls les éléments nouveaux feront l'objet de plus de précision. Afin de rendre l'exposé plus concret, nous l'illustrerons par un exemple.

#### a) exemple

Il s'agit d'une petite cellule flexible de production constituée :

- 1) d'un banc d'entrée "ENTREE" }
- 2) d'un banc de sortie "SORTIE" } dont on ne décrira pas le fonctionnement
- 3) de 2 machines M1 et M2 :
  - M1 et M2 sont autonomes et disposent chacune
    - i) d'un poste d'usinage P
    - ii) d'un tampon d'entrée E pour le chargement (FIFO 2 places)
    - iii) d'un tampon de sortie S pour le déchargement (FIFO 2 places)
- 4) d'un robot qui assure le transfert des pièces entre les divers organes constitutifs du système.



Dans cet atelier circulent plusieurs types de pièces ;  
celles-ci sont usinées :

- soit sur la station M1
- soit sur la station M2
- soit sur M1 et sur M2 dans un ordre indifférent.

La partie graphique du modèle RdPSAC de la partie Commande est donnée à la figure I.3. et pour le moment, volontairement, sans explication détaillée sur la façon dont elle est obtenue.

#### b) modèle de base

Il s'agit d'abord d'un modèle Réseau de PETRI

- \* Ce modèle est représentable graphiquement.
- \* Il représente le comportement dynamique d'un système de commande discret.
- \* Les places sont associées au concept d'état du système.

Comme il s'agit d'un système qui est une partie commande, une grande partie de ces états seront attachés à un ou plusieurs ordres que la partie commande réelle devra envoyer à la partie opérative.

Ainsi sur la figure I.3.

- + Pour la procédure de transfert ENTREE → M1 (en haut à gauche)
  - la place ENTREE → M1/NOP (No Operation) correspond à un état de non activation de cette procédure de transfert.
  - la place ENTREE → M1/TRSF-START (Transfert Start) correspond à l'ordre donné par la partie commande au robot R de commencer l'opération de transfert d'une pièce depuis le banc d'ENTREE sur le tampon d'entrée de la machine M1.
  - la place ENTREE → M1/TRSF-END (Transfert End) correspond à l'ordre donné par la partie commande au robot R de terminer l'opération de transfert.
  
- + Pour la procédure de conditionnement d'une pièce sur la machine M1 (à gauche)
  - la place M1/Nop correspond à un état de non-activation.
  - la place M1/E→P correspond à l'ordre donné par la partie commande à la machine M1 d'effectuer le transfert d'une pièce depuis son tampon d'entrée E sur le poste d'usinage P.

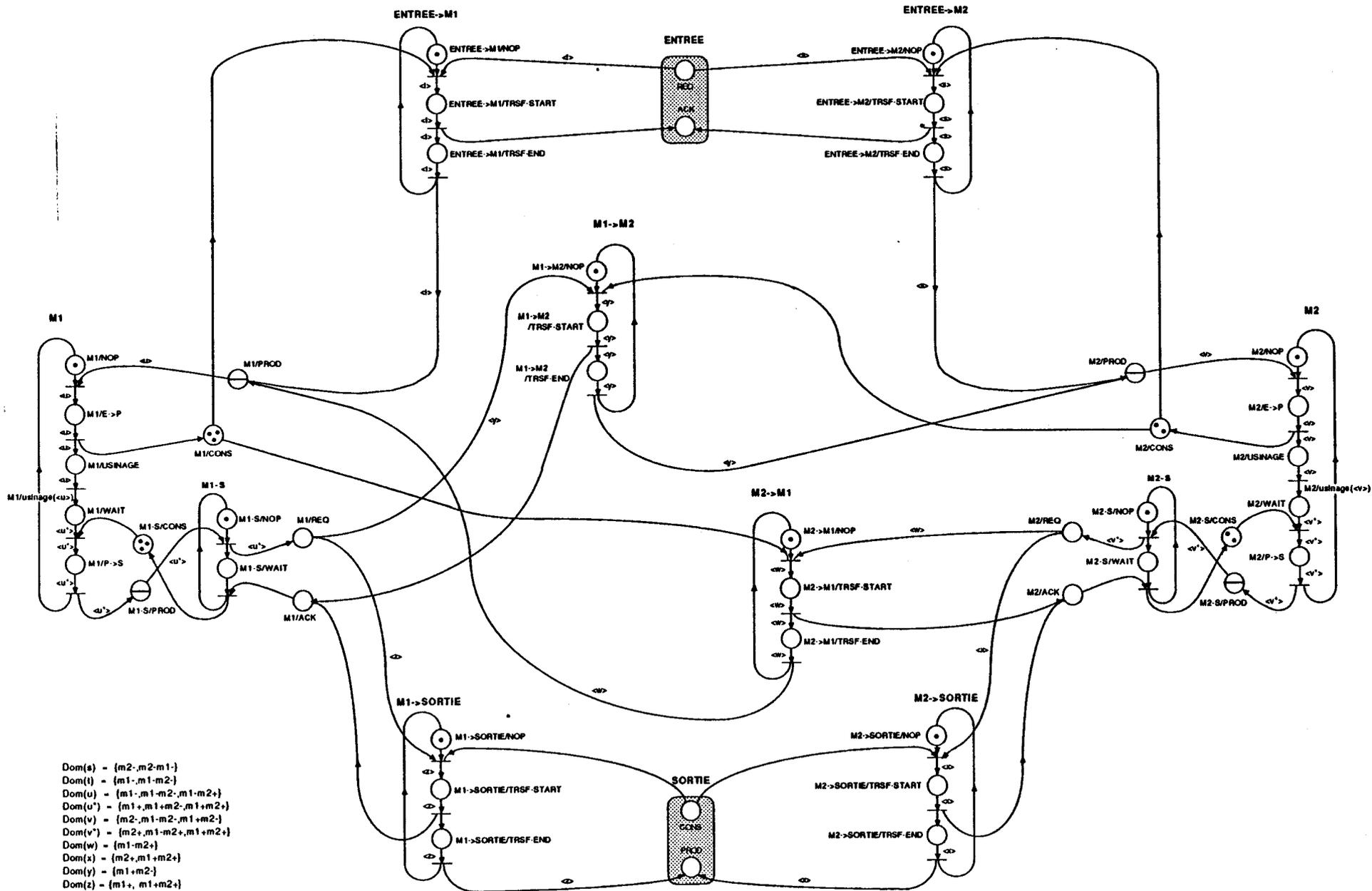
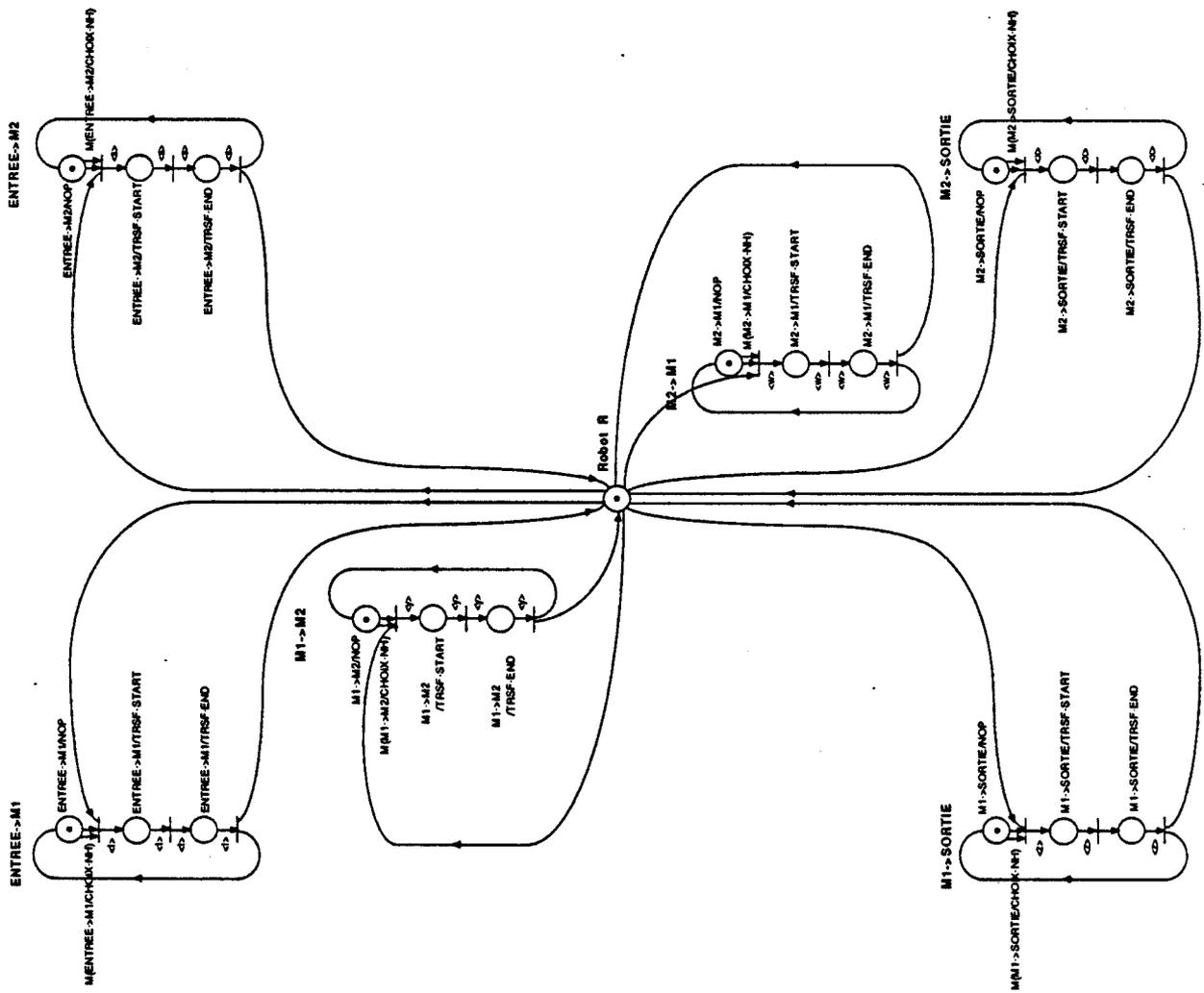


Figure I.3



INTERFACE PARTIE COMMANDE / NIVEAU HIERARCHIQUE

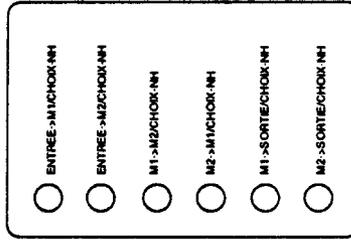


Figure I.3 bis

- Dom(e) = {m2, m2-m1}
- Dom(f) = {m1, m1-m2}
- Dom(u') = {m1, m1-m2, m1-m2}
- Dom(v') = {m2, m1-m2, m1-m2}
- Dom(w') = {m2, m1-m2, m1-m2}
- Dom(x) = {m1-m2}
- Dom(y) = {m2, m1-m2}
- Dom(z) = {m1, m1-m2}

- la place M1/USINAGE correspond à l'ordre donné à M1 d'effectuer une opération d'usinage.
- la place M1/WAIT correspond à un état d'attente.
- la place M1/P→S correspond à l'ordre donné à M1 d'effectuer le transfert d'une pièce depuis son poste d'usinage P sur son tampon de sortie S

On en déduit le sens des notations homologues sur l'ensemble du graphe.

- Les transitions sont associées à la notion d'évènements dont l'occurrence modifie l'état du système.
- L'identification d'un état du système et, par là même, des ordres que la partie commande doit activer se fait par le marquage de certaines places et par le déplacement et l'évolution de ce marquage dans le graphe.

Les règles de validation et de tir des transitions, c'est-à-dire les règles d'évolution du marquage du graphe ne seront pas rappelées ici (voir [BRA 83a] [BRA 83b] [THE 82a] [CAS 87b] [BOU 88b]).

### c) Les Réseaux de PETRI Structurés (RdPS)

Cette limitation du modèle a été introduite pour :

- i) d'une part, attribuer à l'outil RdP utilisé les caractéristiques syntaxiques d'un langage de programmation structurée. L'accroissement de la complexité des problèmes d'automatisation a été à l'origine de l'apparition d'automatismes programmables dont la puissance de programmation ne cesse de s'accroître depuis la fin des années 70. Simultanément est apparu le besoin de langages de spécification et de programmation d'automatismes présentant les qualités de la programmation structurée. Avec une quinzaine d'années d'écart, les problèmes de cohérence et de rigueur qui se sont posés pour les langages de programmation des calculateurs se sont déplacés vers les automatismes industriels. La création du GRAFCET a été dans ce sens. Le modèle RdP utilisé bénéficie lui aussi d'une syntaxe de structure.
- ii) d'autre part, permettre une implantation répartie du modèle facilitée par une décomposition syntaxique correspondant en même temps à une décomposition fonctionnelle.

La structure du graphe est construite autour de l'idée de décomposition modulaire. On distingue :

- 1) les graphes de processus apparentés à la notion de procédure ou application monoprocesseur informatique. Ils permettent le regroupement fonctionnel de tâches séquentielles mais non parallèles.
- 2) les graphes de liaison assurant les liens de coordination entre processus.

Nous allons rappeler brièvement ci-dessous la syntaxe utilisée :

1) **Les graphes de processus**

Sous forme BACKUS-NAUR

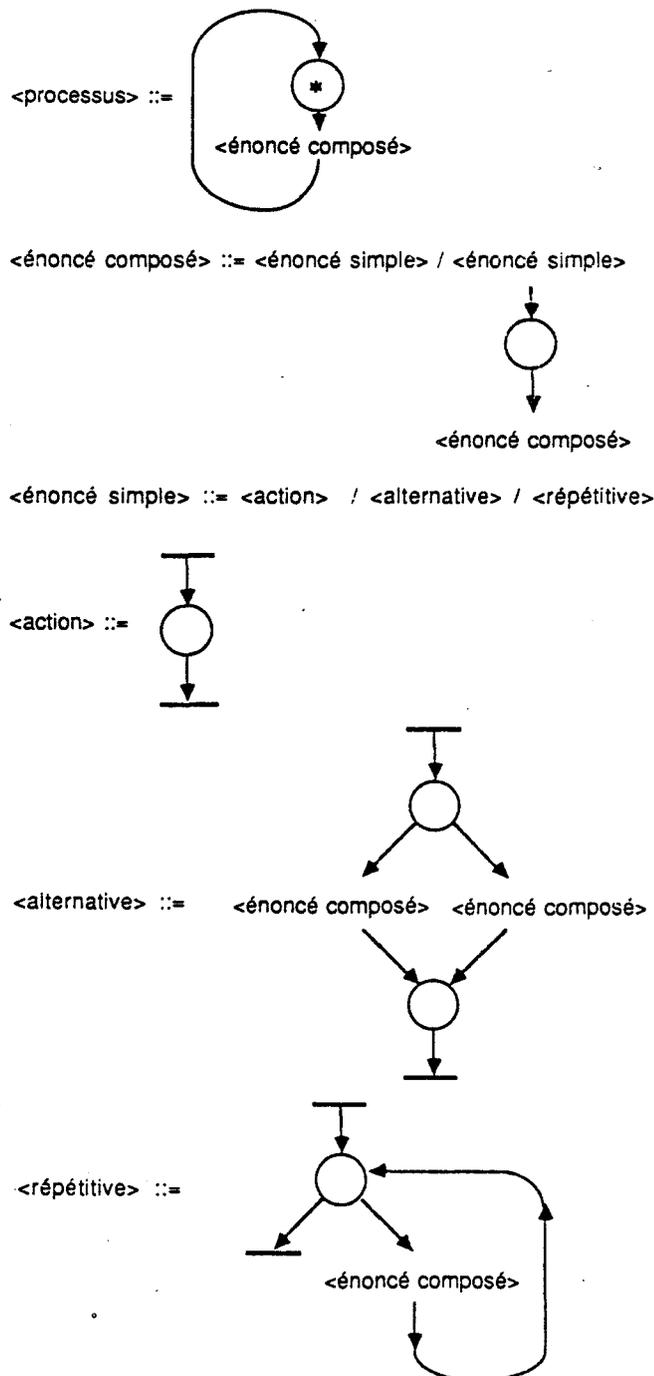


Figure I.4

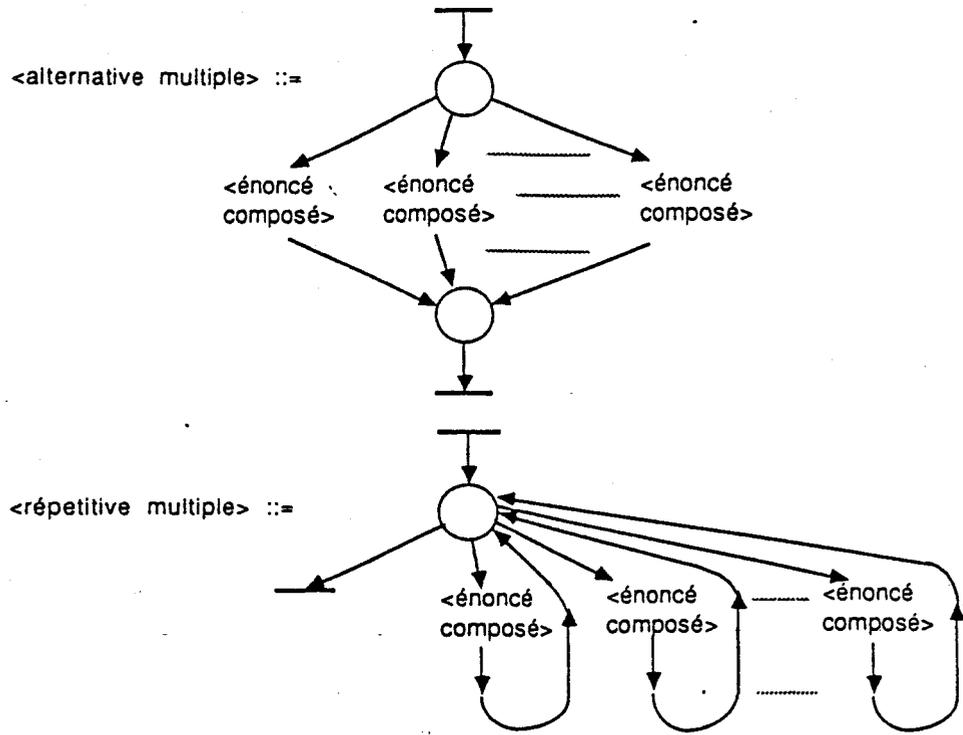


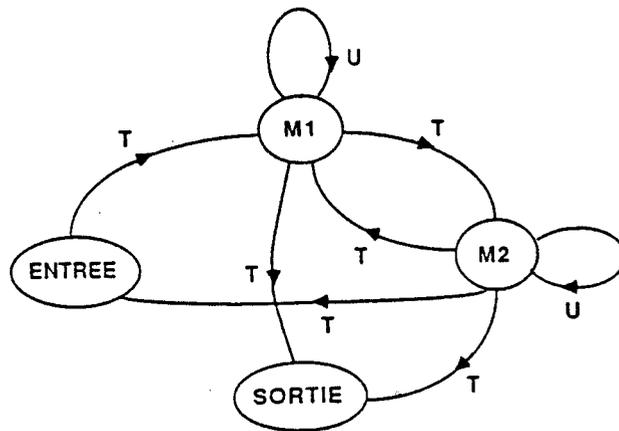
Figure I.4

Prenons l'exemple de la figure I.3. On peut répertorier :

- des processus de transformation de pièces tels que M1 et M2
- des processus de transfert de pièces tels que

ENTREE → M1  
 ENTREE → M2  
 M1 → SORTIE  
 M2 → SORTIE  
 M1 → M2  
 M2 → M1

Ces processus correspondent aux grandes étapes fonctionnelles décrivant le fonctionnement de la cellule.



T=Transfert=Etape positionnelle  
 U=Usinage=Etape transformationnelle

Figure I.5

La façon dont a été construit le graphe de la figure I.3 peut apparaître maintenant plus clairement. Il reste à rattacher les processus entre eux. Ce rôle est dévolu aux graphes de liaison.

## 2) Les graphes de liaison

### \* L'exclusion mutuelle

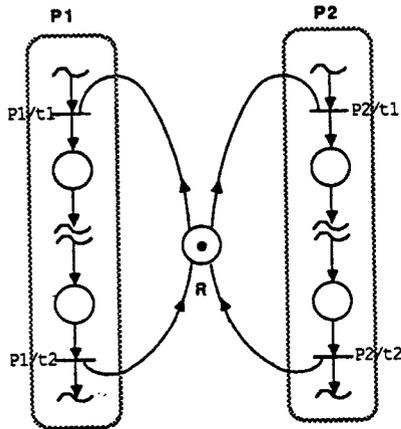


Figure I.6

R est une ressource critique partagée par 2 processus exclusifs P1 et P2. C'est par exemple le cas d'un Robot utilisé en commun par 2 machines. Cette structure appelée aussi sémaphore permet d'assurer l'unicité de la commande de la ressource.

ex : Un exemple de cette structure est donné par la commande du robot de transfert de la figure I.3 bis.

### \* La synchronisation avec accusé de réception

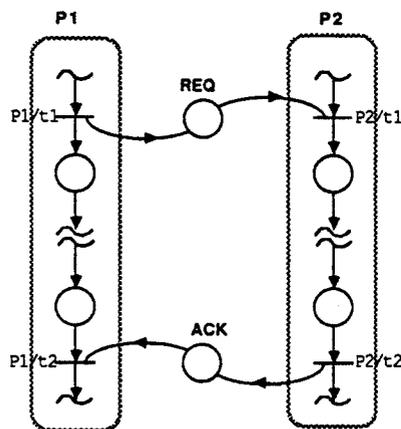


Figure I.7

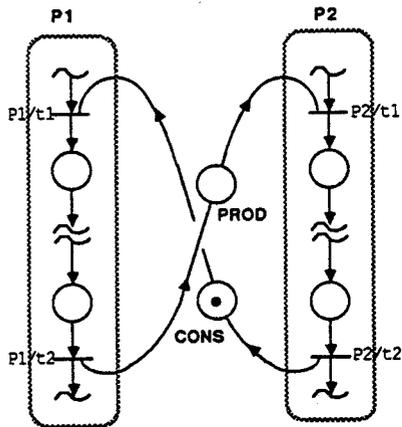
Le processus P1 synchronise le processus P2. P1 envoie à P2 une marque qui permet à ce dernier d'évoluer et attend après une certaine phase d'évolution propre un accusé de réception de la part de P2 pour achever son cycle. Ce schéma sert à représenter des protocoles de communication de type émission - réception.

REQ = requête (Request)

ACK = accusé de réception (Acknowledgment)

ex : Sur le schéma de la figure I.3, ces structures permettent au banc d'ENTREE et aux machines M1 et M2 d'effectuer une requête auprès du Robot R pour réaliser le transfert d'une pièce.

\* Le producteur/consommateur



Le producteur P1 produit des marques dans la place PROD et P2 consomme ces marques.

La place CONS permet à P2 de relancer la production d'une marque par P1.

Figure I.8

Pour la représentation d'un mécanisme de producteur-consommateur avec une zone tampon à plusieurs places entre P1 et P2, il est nécessaire d'étendre la structure précédente.

Le nombre de marques de la place CONS représente le nombre de places vides du tampon.

L'extension de la place PROD doit représenter la gestion des places occupées du tampon.

Par exemple, considérons un tampon à deux places géré en file d'attente (FIFO) entre deux sites S1 et S2

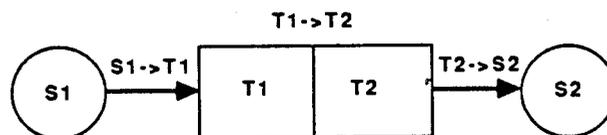


Figure I.9

Le mécanisme de gestion du tampon peut être représenté par le graphe ci-dessous

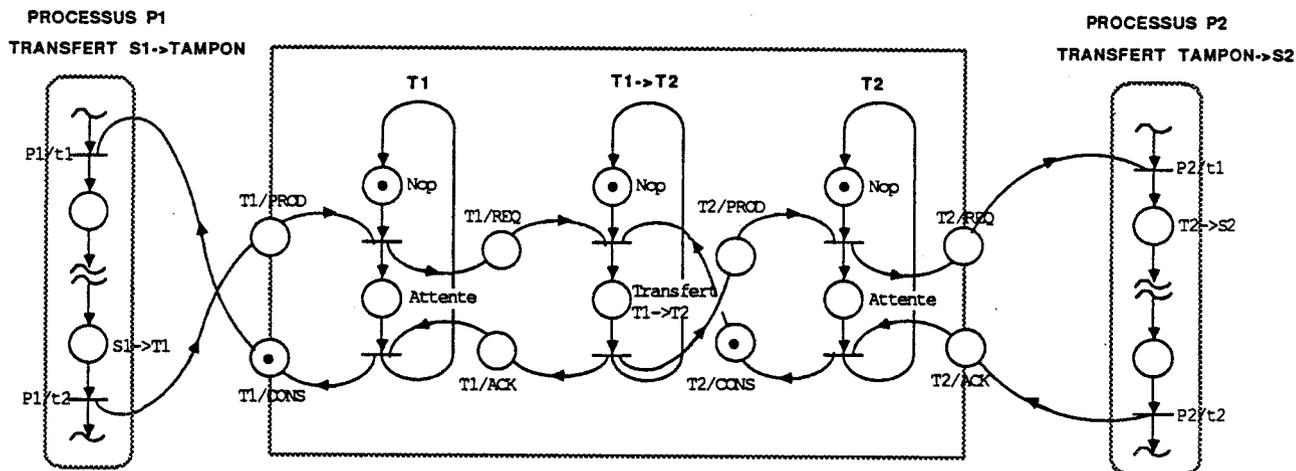


Figure I.10

Pour des raisons de simplicité et de lisibilité, on représentera de telles structures par une notation symbolique utilisant la primitive graphique " $\ominus$ " appelée "macro place de type FIFO". Le schéma précédent sera donc représenté sous la forme :

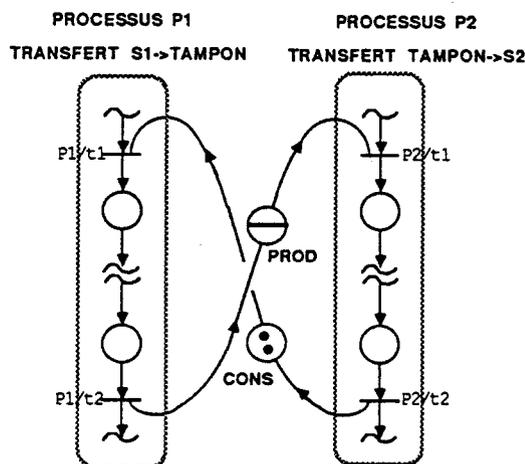


Figure I.11

On retrouve cette notation sur la figure I.3 pour la gestion des tampons d'entrée et de sortie des machines M1 et M2. D'autres structures de gestion du tampon (LIFO et anneau) peuvent être proposées.

\* Localisation des liaisons

Les interactions entre processus ne se font que par l'intermédiaire des transitions début et fin des structures de blocs définies pour les graphes de processus.

d) Les réseaux de PETRI structurés et colorés (RdPSC)

d.1) Présentation simplifiée de la coloration

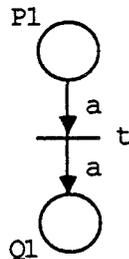
La coloration permet de faire évoluer au sein d'un même graphe un ensemble fini de marques différentes ("colorées").

Dans le cas des ateliers flexibles, la polyvalence des installations, la grande variété des pièces et leur souplesse de circulation dans l'atelier nécessiteraient, en présence d'un seul type de marques, de construire au moins un graphe de commande pour chaque type de produit fini. La possibilité d'effectuer des permutations dans le séquençement des opérations permettant l'élaboration d'un produit fini augmenterait encore cette décomposition.

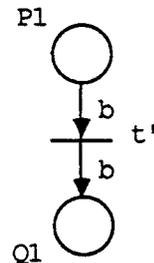
L'utilisation de marques colorées permet une agrégation de ces graphes en un seul modèle plus concis.

Le cheminement des marques d'une couleur se fera en suivant les arcs qui autorisent le passage de cette couleur.

par ex. :



Sur ce schéma, seules les marques de type a pourront transiter entre P1 et Q1 suivant les arcs (P1,t) et (t,Q1)



Ici, seules les marques de type b pourront transiter entre P1 et Q1 suivant les arcs (P1,t') et (t',Q1)

Figure I.12

L'agrégation de ces deux schémas donne le graphe suivant :

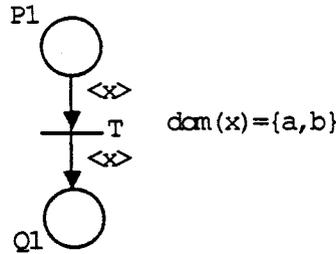


Fig. I.13

où chacune des couleurs a et b peut transiter en suivant les arcs (P1,T) et (T,Q1).

La définition formelle de la coloration utilisée est donnée dans (|CAS 87b|). Elle est, avec toutes ses extensions, bien entendu plus complexe que la simple approche intuitive donnée ici.

Précisons toutefois que l'agrégation du modèle de la partie commande n'est pas le but final recherché.

Des regroupements fonctionnels sont effectués, tels que les commandes d'usinages divers effectués sur la même machine sous le même système de contrôle.

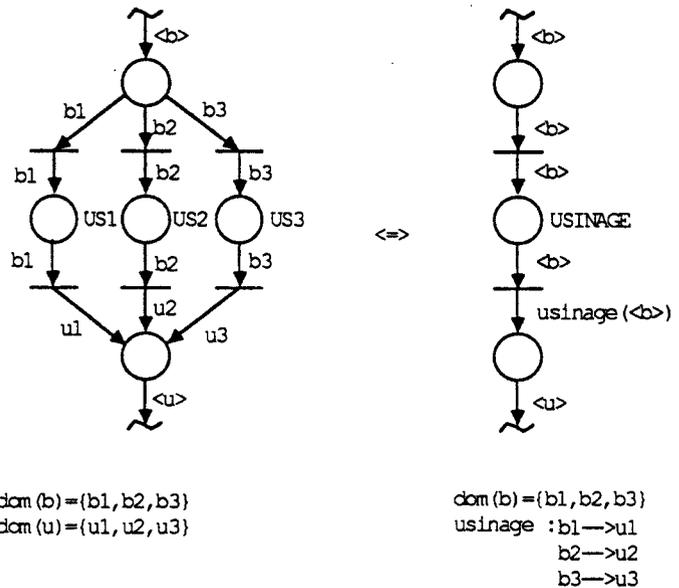


Fig. I.14

Mais pour des raisons d'implantation répartie future du modèle sur les dispositifs industriels actuels ou prévisibles dans un proche avenir, les regroupements structurels ne seront pas effectués. Ainsi, les processus de commande de deux machines identiques destinées à faire strictement les mêmes opérations seront laissés distincts.

Les marques qui seront utilisées permettront d'identifier l'évolution d'un produit pour fournir à la partie commande une image du processus automatisé.

Prenons l'exemple de la figure I.3

\* On répertorie trois gammes opératoires (une gamme opératoire est un ensemble d'opérations de transformation à accomplir sur un groupe de produits afin d'obtenir un produit final)

- la gamme des pièces qui doivent être usinées sur M1 repérée par la marque  $m1^-$
- la gamme des pièces qui doivent être usinées sur M2 repérée par la marque  $m2^-$
- la gamme des pièces qui doivent être usinées sur M1 et M2 dans un ordre indifférent et qu'on repèrera par la marque  $m1^-m2^-$

\* Le degré d'avancement de la transformation des pièces est repéré grâce aux symboles "-" et "+".

$m1^-m2^+$  indique | que l'usinage sur M1 n'a pas encore été effectué.  
que l'usinage sur M2 est terminé.

Autre exemple :

Cas où les transformations concernent des solidarités d'assemblage :

soit un assemblage ABCD

désignons par 11 la solidarisation sur l'interface A-B

12 la solidarisation sur l'interface B-C

13 la solidarisation sur l'interface C-D

$11^-12^-$  désignera l'ensemble des interfaces A-B et B-C dont les composants sont seulement positionnés.(par rapport aux pièces A, B, C ou CD libres)

$11^+12^-$  signifiera que la solidarisation de l'assemblage A-B est réalisée mais que les composants de l'interface B-C ont seulement été positionnés sans être solidarisés pour le moment.

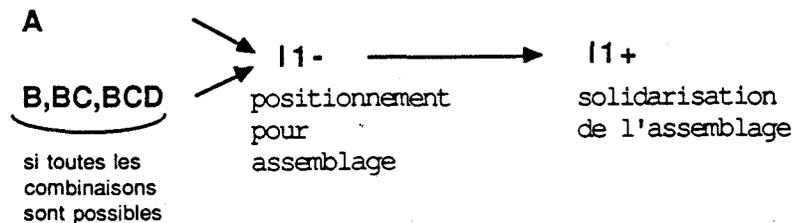
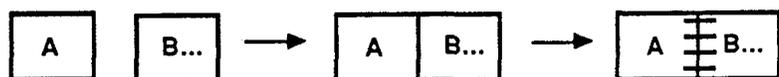


Fig. I.15



Les exemples précédents ne donnent, bien sûr, que des propositions de notation à titre indicatif.

Nous allons brièvement maintenant citer une extension des réseaux colorés que nous pensons faire intervenir à l'avenir mais que nous considérons avec prudence pour le moment (voir chapitre III, partie 2).

#### d.2) Les réseaux à prédicats

L'évolution de la complexité des systèmes de commande, au niveau de la recherche, se fait de plus en plus dans le sens de la reconnaissance individuelle des pièces et de l'adaptation de la commande à des situations unitaires très diverses.

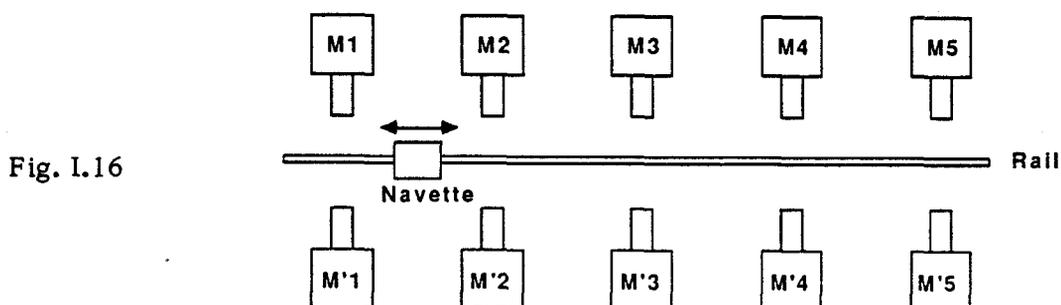
La représentation exhaustive et à l'avance de toutes les commandes possibles et de tous les séquencements du système de commande est donc de moins en moins envisageable, voire même impossible dans le cas où une infinité de situations peuvent se présenter. On cherche alors à réduire la taille des modèles à une dimension finie et exploitable et en conséquence à augmenter leur pouvoir de représentation. C'est cette situation qui se présente en particulier dans le cas des ateliers flexibles.

La coloration constitue une première étape d'abréviation du modèle RdP. La couleur des marques permet d'identifier différentes gammes opératoires et le degré d'avancement de l'évolution des produits au sein de ces gammes. Mais une identification encore plus fine de la typologie des pièces, image du procédé dans le modèle du système de commande, exigerait une multiplication des marques et des arcs correspondants. Afin de transporter avec chaque marque plusieurs informations sans accroître le nombre d'arcs et donc la taille du graphe, une extension de la coloration existe : ce sont les Réseaux à prédicats (IGEN 79). Une marque devient susceptible de représenter un n-uple de valeurs. Des prédicats relatifs à une ou plusieurs des composantes du n-uple sont associés à chacun des arcs. Les marques ne peuvent transiter que suivant les arcs dont elles vérifient les prédicats. La coloration simple n'autorise qu'un ensemble fini de marques différentes. Les réseaux à prédicats autorisent l'évolution de marques provenant d'un ensemble infini. Les prédicats permettent une définition de l'ensemble des marques autorisées à emprunter un arc en compréhension et non plus en extension. On dit que l'on augmente la puissance algorithmique du modèle.

On dispose ainsi d'un outil de modélisation extrêmement puissant dont les possibilités de représentation et d'utilisation des connaissances deviennent comparables à celles d'un langage "déclaratif" (par opposition à "procédural") et travaillant en logique des prédicats.

Le modèle RdP coloré utilisé dans le cadre du projet CASPAIM devra probablement être étendu aux réseaux à prédicats. Le cas de systèmes de commande dont certains ordres ou certains comportements dépendent de conditions se rapportant à des informations portées par les marques et dont les valeurs peuvent provenir d'un ensemble infini (par ex. : le temps, la date d'entrée, le numéro d'entrée) en donne l'exemple. Le comportement du système de commande nécessite dans ces cas une image très fine du procédé proche de la reconnaissance individuelle des pièces (exemple de temporisations de places, de pondérations d'arcs dépendant de calculs effectués à partir de la date d'entrée, du temps écoulé, du numéro d'entrée, etc...). Toutefois, dans les cas courants abordés, une image aussi fine du procédé dans le modèle de commande n'est exigée que pour des évaluations de performances (statistiques) ou pour des dimensionnements (temporisations) qu'il est possible de rejeter sur le modèle du procédé. Comme nous le verrons, ce dernier, greffé sur le modèle du système de commande, utilise des outils de représentation autres que les RdP (modèle de type langage objet). Ceux-ci permettent d'aborder les problèmes relatifs à la reconnaissance individuelle des pièces sans compliquer le modèle de la partie commande avec lequel le modèle du procédé dialogue. Pour la représentation de la partie commande, le modèle RdP coloré est alors tout à fait suffisant. Ceci est d'une grande importance si l'on garde l'idée que le modèle devra être implanté sur des dispositifs industriels et que celui-ci devra donc rester compatible avec le pouvoir de description des outils d'implantation afin de faciliter la transposition.

Il reste cependant une raison importante pour laquelle l'extension du modèle Réseau de PETRI coloré aux Réseaux à prédicats mériterait d'être effectuée, c'est le risque d'explosion de la taille du modèle dû à la multiplication des itinéraires possibles. Prenons l'exemple ci-dessous

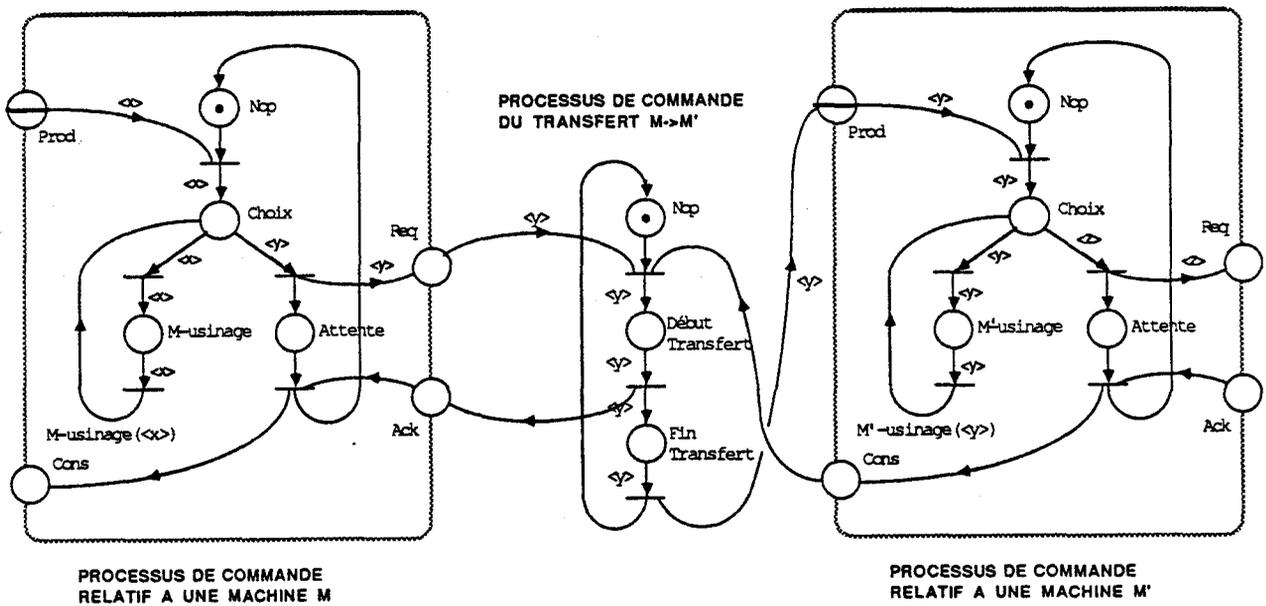


Nous avons ici deux rangées de machines de type M et M'. Les pièces peuvent passer d'un type de machine à l'autre par l'intermédiaire d'une navette. Sur le modèle RdP structuré et coloré, 25 liaisons doivent être prévues entre les cinq processus des machines M et les cinq processus des machines M' (ces liaisons étant en outre placées en exclusion mutuelle du fait de l'unicité de la navette). On s'aperçoit ici immédiatement du risque d'explosion du graphe de commande. C'est pour cette raison que le modèle est parfois qualifié de procédural par abus de langage. Cet argument opposé par des groupes de recherche qui travaillent sur des modèles plus déclaratifs (systèmes experts, langages orientés objet) peut être levé par les Réseaux à prédicats ([GEN 79 | BRA 83b]) dont la puissance ne cède en rien à celle d'un langage de programmation non séquentiel, non déterministe. De la même façon que la coloration constitue un niveau d'abréviation relatif à la typologie des pièces, les Réseaux à prédicats représenteraient un niveau d'abréviation supérieur prenant ici en compte les machines. Les marques circulant seraient alors de la forme (<x>, <X>)

- x     prenant ses valeurs dans les types de pièces
  - X     prenant ses valeurs dans un ensemble de machines.
- (\*) (voir figure I.17).

Cependant, pour le moment, la taille des exemples industriels réalistes envisagés n'exige pas cette extension. Afin de minimiser les problèmes d'implantation du modèle sur dispositifs industriels programmables existants, le choix a été fait de limiter l'agrégation à la coloration. Nous reparlerons d'ailleurs au chapitre III, partie 2 des problèmes posés autour de l'extension discutable de la complexité du modèle de la partie commande.

figure I.17



A titre d'exemple , considérons les 2 processus de commande des machines M et M' reliés par le processus de commande du transfert M->M'.

<x>, <y> et <z> sont des variables qui permettent d'identifier au regard de la Partie Commande les types de pièces transitant sur M et M'.

Supposons que l'on ait 5 machines de type M : M1, M2, M3, M4, M5

5 machines de type M' : M'1, M'2, M'3, M'4, M'5 .

Pour ne pas créer un processus de commande par machine  $M_i$  et par machine  $M'_j$  et un processus de commande de transfert pour chaque couple  $(M_i, M'_j)$  , les réseaux à prédicats permettent de conserver le schéma ci-dessus en remplaçant les variables <x>, <y>, <z> par des n-uplets:

<x> par  $\langle x, M \rangle$  où  $\text{dom}(\langle M \rangle) = \{M1, M2, M3, M4, M5\}$ ,

<y> par  $\langle y, M, M' \rangle$  où  $\text{dom}(\langle M' \rangle) = \{M'1, M'2, M'3, M'4, M'5\}$

et <z> par  $\langle z, M' \rangle$ .

Il est également nécessaire de placer 5 marques colorées différentes (une par machine) dans chacune des places Nop. Les arcs "CONS" (consommateurs) et les arcs de retour à la place Nop des processus de chacune des machines doivent être étiquetés pour permettre le transit de ces marques.

Connaissant M , le choix de M' serait alors effectué par un niveau supérieur de commande : le Niveau Hiérarchique ou Décisionnel dont on reparlera au paragraphe I.3, et au Chapitre III ,Partie II.

e) Les réseaux de PETRI structurés adaptatifs et colorés (RdPSAC)

Reprenons l'exemple de la figure I.3.

Une pièce devant être usinée sur M1 et sur M2, dans un ordre indifférent, va pouvoir être orientée par la partie opérative sur l'une ou l'autre des deux machines en premier lieu. Il existe donc ici un indéterminisme que la partie commande va devoir lever à chaque fois que cette situation va se présenter. On dit qu'il y a indéterminisme directionnel des pièces. Sur le modèle structuré et coloré de la partie commande, on voit que la marque  $m1\text{-}m2$  placée dans la place ENTREE/REQ peut aller simultanément vers les processus ENTREE  $\rightarrow$  M1 et ENTREE  $\rightarrow$  M2.

Prenons maintenant la partie du graphe relative à la commande du robot. Ce dernier peut être sollicité simultanément (ou du moins pendant sa période d'enregistrement des requêtes) par plusieurs demandes de transfert de pièces (ex. : M1  $\rightarrow$  M2 et M2  $\rightarrow$  SORTIE). Il s'agit encore d'un indéterminisme que la partie commande doit résoudre. La marque placée dans la place Robot R peut être attribuée a priori à n'importe lequel des processus de transfert appelants. On dit qu'il y a conflit d'accès à une ressource commune exclusive.

Le modèle RdP structuré et coloré ne suffit plus pour résoudre ces indéterminismes. Il est alors nécessaire de faire intervenir une extension du modèle qui va permettre l'aiguillage des marques en fonction, cette fois-ci, non plus d'informations portées par les marques comme dans les réseaux seulement colorés, mais en fonction de l'état du marquage de la partie commande et de critères extérieurs à celle-ci. Cette extension constitue le caractère Adaptatif du modèle. La définition formelle des Réseaux Adaptatifs est donnée dans (ICAS 87b)). Introduisons seulement la notion intuitive.

Considérons un arc unitaire

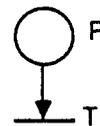


Figure I.18

Suivant cet arc peut circuler une marque éventuelle colorée. L'aspect adaptatif va consister à faire dépendre la pondération de l'arc, pour la marque concernée, du marquage d'une place Q. (Il va éventuellement consister à supprimer cet arc si la pondération devient nulle).

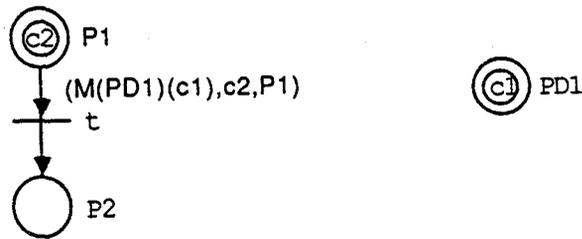


Fig. I.19

Le poids de l'arc amont est pour la couleur C2 égal au nombre  $M(PD1)(C1)$  de marques de couleur C1 de la place décisionnelle PD1.

Cette extension se situe dans la lignée des extensions concernant

- les arcs inhibiteurs [HACK 75a]
- les réseaux à priorités [HACK 75a,b]
- les réseaux automodifiants [VALK 77], [VALK 78]

Notons toutefois la remarque importante suivante :

La place PD1 peut être une place du graphe de la partie commande. Mais elle peut être aussi une place d'interface de la partie commande avec un niveau de commande supérieur qu'on appelle ici niveau décisionnel ou niveau hiérarchique et qui force le marquage de cette place. On arrive ainsi à paramétrer la partie commande par un niveau décisionnel, ce qui rend par ailleurs le réseau non autonome. Le niveau hiérarchique permet d'intégrer non seulement la connaissance de l'état de la partie commande à l'instant où il en a besoin, mais aussi la connaissance de l'état du procédé, ainsi qu'un certain nombre de critères extérieurs relevant des stratégies de pilotage (gestion de production, contraintes socio-économiques, etc...) et des modes de marche (interventions, mise en route, etc...). La flexibilité de cheminement des pièces dans l'unité de production ainsi que la polyvalence des machines se traduisent par un accroissement des indéterminismes directionnels.

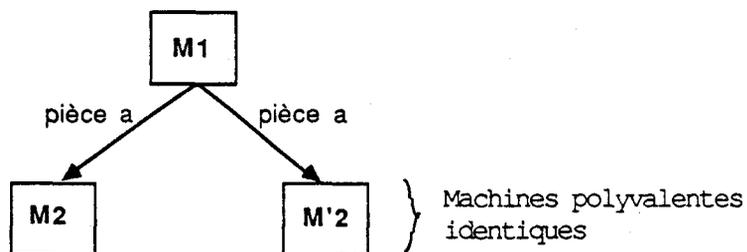


Fig. 1.20

De même, pour des raisons de rentabilité, l'accroissement des possibilités de cheminements va se traduire par la mise en commun accrue de certaines ressources (bras de manutention, chariots filoguidés) donc par une augmentation du nombre des conflits possibles.

**Exemples d'utilisation :**

**1. Résolution d'indéterminismes**

**a) indéterminismes directionnels**

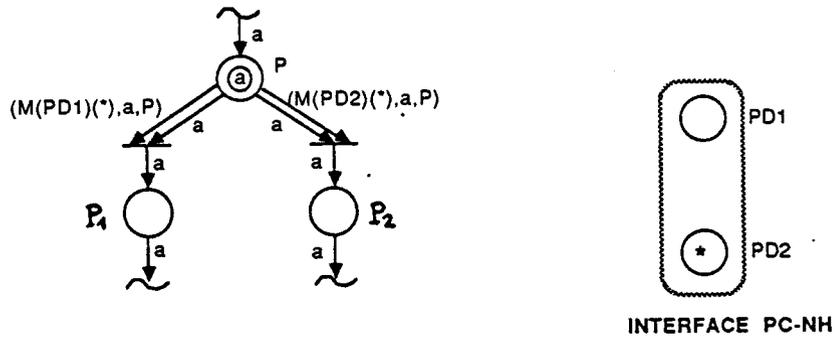


Figure I.21

Le chemin  $P \rightarrow P_2$  est bloqué pour la configuration de marquage.

**b) conflits**

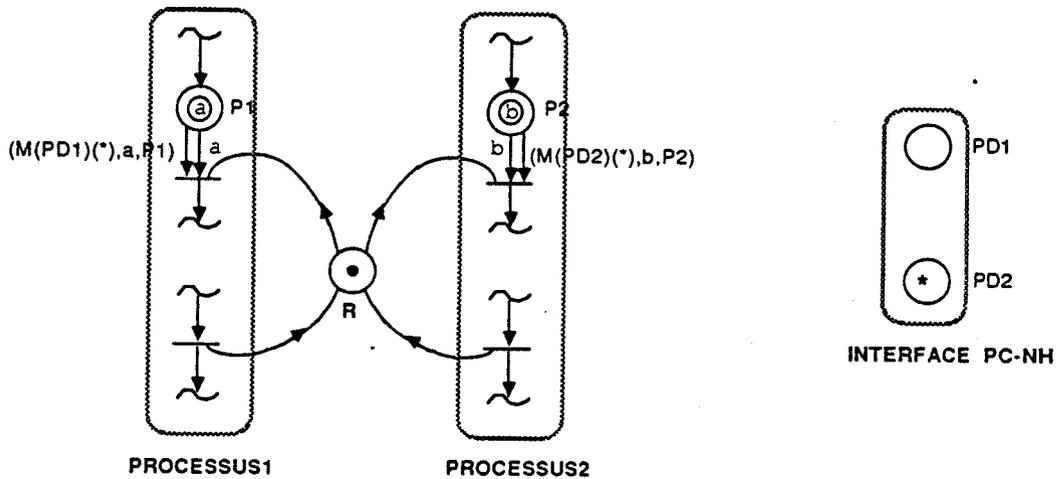


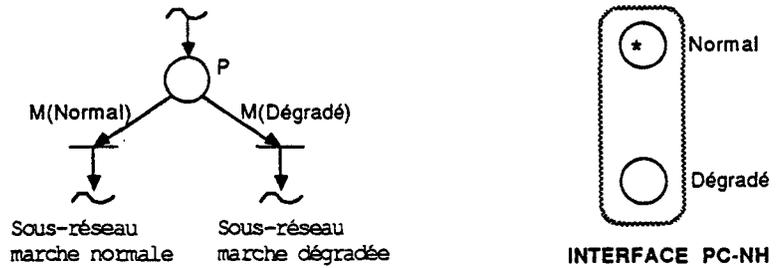
Figure I.22

Etant donné la configuration du marquage, l'allocation de la ressource R est faite au processus P1 pour la marque a.

## 2. Changement de mode de marche

Le marquage des places Normal et Dégradé connecte ou déconnecte de la place P les sous-réseaux respectifs.

Figure I.23



## 3) Gel de processus ou gel sélectif d'actions

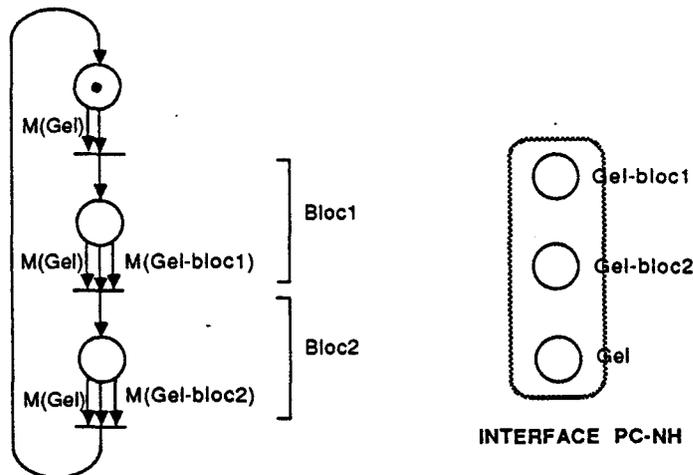


Figure I.24

Reprenons l'exemple de la figure I.3.

Des arcs adaptatifs sont placés en tête de chacun des processus de transfert afin de contrôler l'accès à la ressource que constitue le robot R.

Aucun arc relatif au gel de processus ou au changement de mode de marche n'est représenté sur cette figure pour ne pas compliquer les explications.

### Remarque sur les réseaux adaptatifs et les réseaux interprétés :

Le caractère adaptatif apporte, par rapport à l'interprétation des transitions (Réseaux de Petri interprétés, GRAFCET - cf [MOA 85]) une notion supplémentaire. Les arcs adaptatifs, étiquetés par des placês, peuvent non seulement être bloqués par le marquage de ces places mais la pondération de ces arcs est en plus variable. Cette propriété est en fait très rarement utilisée, ce qui rend le plus souvent l'utilisation des réseaux adaptatifs tout à fait équivalente à celle des réseaux interprétés, et leur transposition est immédiate.

### I.3 LE NIVEAU HIERARCHIQUE

La partie commande dont nous venons de décrire le modèle donne l'ensemble des états du système de commande, et l'ensemble des commandes applicables à la partie opérative.

Les caractéristiques liées à la flexibilité et à la sûreté de fonctionnement multiplient énormément les possibilités de comportement de la partie commandée et donc les choix à faire pour assurer le déterminisme de circulation des marques du modèle. Ainsi, comme nous l'avons vu précédemment, le modèle de la partie commande est non autonome. On a alors recours à un niveau décisionnel pour effectuer son paramétrage. C'est précisément par l'intermédiaire de ce niveau décisionnel que le système de pilotage va pouvoir tenir compte de stratégies de production issues de niveaux de planification et de gestion plus élevés. C'est par ce niveau décisionnel que le système de pilotage va pouvoir s'inscrire dans un système intégré de production (CIMS Computer Integrated Manufacturing System).

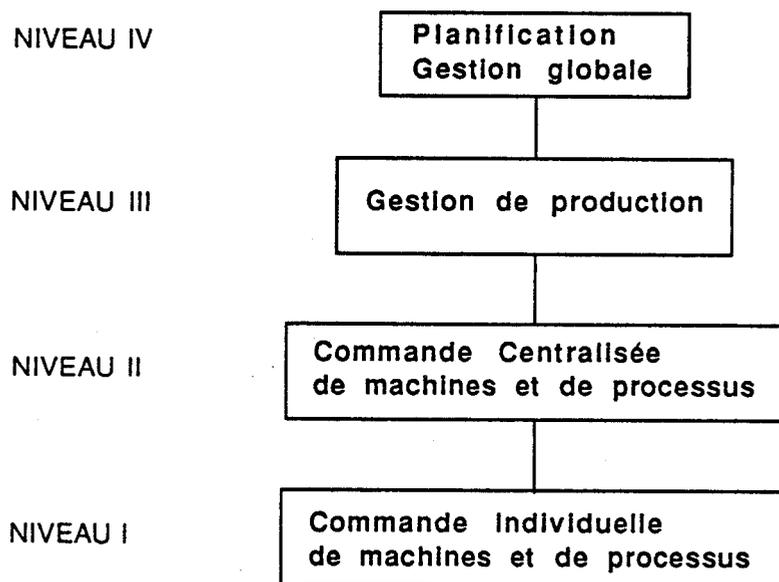
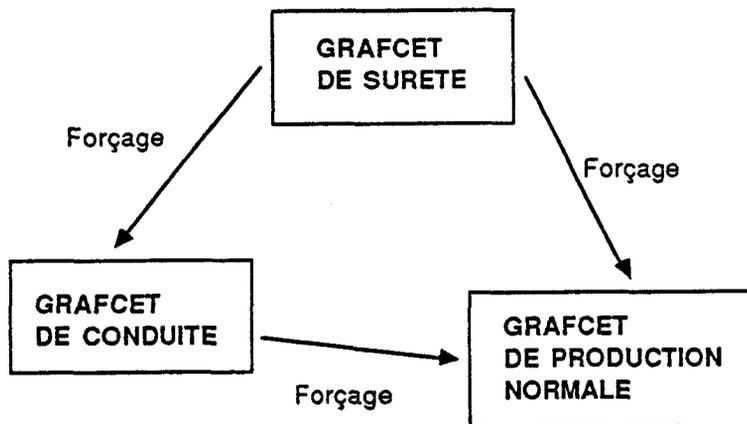


Figure I.25

Les premières études ont porté sur un modèle Réseau de PETRI du niveau décisionnel. Le but recherché était alors d'avoir un outil homogène pour la Partie Commande et le Niveau Hiérarchique. C'est ce qui est réalisé par exemple par la hiérarchie des GRAFCET qui intègre les dispositions précisées par le GEMMA (Guide d'Etude des Modes de Marche et d'Arrêt).

figure I.26



- Le GRAFCET de SURETE est un GRAFCET maître consacré en général à la surveillance de la sûreté de fonctionnement.
  - Le GRAFCET de CONDUITE permet d'assurer la commutation entre différents modes de marche du système automatisé.
  - Le GRAFCET de PRODUCTION NORMALE regroupe les graphes de fonctionnement indispensables à la production. Notons qu'on ne produit pas forcément dans tous ces modes. Ils peuvent être préparatoires, servir aux réglages, aux tests, etc...
- Les actions d'un GRAFCET sur un autre peuvent être représentées par des ordres de forçage. (voir [BOU 87a]).

Figure I.26

Cependant le choix s'est porté sur un modèle déclaratif du niveau hiérarchique.

- En effet, en fonction de faits qui sont liés à l'état du marquage de la partie commande, à certaines informations venant du modèle de la partie opérative et du procédé, et aussi à d'autres éléments de décision, le niveau hiérarchique élabore un marquage des places d'interface Partie-Commande/Niveau Hiérarchique en utilisant un ensemble de règles de décision.
- L'intégration de notions parfois diverses pour les règles de décision (gestion de production, contraintes de distance, critères socio-économiques), la modularité et la possibilité de reconfigurer ou d'enrichir ces structures de contrôle ont créé la nécessité de séparer les connaissances indispensables à la résolution des problèmes de la partie commande, du programme qui permet de les utiliser et qui constitue un moteur d'inférences.

- Par ailleurs, la grande expertise des règles décisionnelles fait que le niveau hiérarchique se situe dans une catégorie proche des systèmes experts.
- Enfin, la représentation procédurale et séquentielle du niveau décisionnel exigerait, même avec des outils d'abréviation permettant de limiter la taille de la représentation, la prévision exhaustive du séquençement d'un nombre à la fois énorme et limité de réponses spécifiques à des problèmes recensés de la Partie Commande.

La représentation déclarative des connaissances du Niveau Hiérarchique permet de fournir uniquement à ce dernier les outils de son raisonnement sans en préciser la chronologie ou le séquençement d'utilisation.

Cette définition de la réponse en compréhension et non plus en extension permet d'aborder théoriquement une infinité de successions de cas de figures.

C'est donc sur des outils de modélisation différents de ceux de la Partie Commande que s'est arrêtée la représentation du Niveau Décisionnel.

Reprenons l'exemple de la figure I.3.

Les interventions du niveau hiérarchique sont de trois types. Elles concernent :

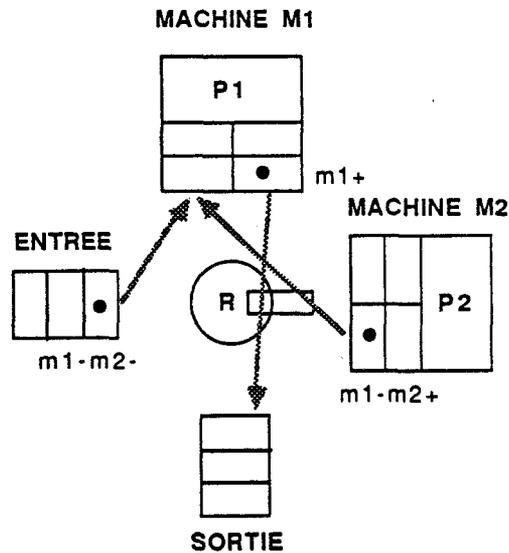
- la résolution des indéterminismes de la partie commande.
- la prévention des blocages.
- les changements de mode de marche : arrêt  
défaillance  
fonctionnement dégradé  
mise en route  
etc..

tels qu'ils sont spécifiés dans le GEMMA (voir [BOU 87a]).

a) Les indéterminismes :

Supposons que la situation de la cellule soit la suivante, au moment de prise en compte des requêtes de transfert.

Figure 1.27



On voit ici apparaître deux types d'indéterminismes :

- Les conflits d'accès à une ressource exclusive sont liés au fait que des ressources doivent être partagées, souvent pour des raisons de rentabilité. C'est le cas ici du robot R et du tampon d'entrée de la machine M1. Une zone d'attente associée à chaque ressource partagée doit être créée pour prendre en charge les requêtes et doit être gérée suivant des lois de priorité déterminées par le Niveau Hiérarchique.
- Les indéterminismes directionnels sont liés à la recherche d'une très grande souplesse de circulation des pièces dans l'atelier afin de diminuer considérablement leur temps de transit par rapport au temps pendant lequel elles subissent des opérations de transformation. C'est la levée de ces indéterminismes qui pose les problèmes les plus délicats au niveau décisionnel.

Un indéterminisme décisionnel apparaît sur le cas de figure étudié. Une pièce de type  $m1-m2^-$  est présente sur le banc d'ENTREE. Les usinages  $m1$  et  $m2$  pouvant être permutés, un choix va devoir être fait pour diriger la pièce sur M1 ou sur M2 afin d'effectuer un des deux usinages en premier.

Ces problèmes de choix provoquent l'apparition d'une arborescence d'états du système de commande et du procédé atteignables à partir de l'état courant. La recherche d'un chemin dans l'arborescence des états permettant d'optimiser un critère évoque des problèmes de recherche opérationnelle. Mais l'explosion

combinatoire de l'arbre des états est telle qu'un horizon devrait être fixé dans la profondeur d'exploration avec les effets indésirables que provoque cette méthode. On ne peut s'empêcher de faire l'analogie avec les problèmes de jeux (dames, échecs) (|LAU 86| |FAL 85|) à ceci près qu'il ne s'agit pas ici de situations compétitives où la décision est décentralisée sur deux intervenants, et à ceci près que les fonctions d'évaluation devraient ici tenir compte de l'apparition aléatoire de certains événements extérieurs.

La question posée est véritablement complexe.

On recherche alors classiquement à résoudre le problème par l'intermédiaire d'heuristiques. Il existe alors deux façons de procéder :

- i) La première consiste en une analyse indéterministe des choix. On réduit l'exploration de l'arborescence à celle d'une famille de sous-arbres. Cette technique porte le nom de programmation heuristique. Le développement d'algorithmes parallèles fait parler aujourd'hui de programmation heuristique parallèle (|SCH 84|).
- ii) La deuxième consiste en l'utilisation des systèmes experts. Cette technique construit le choix d'une solution à partir de règles arbitraires qui sont du ressort de l'expérience ou de l'expertise. Cette expertise est bien sûr à acquérir avant de construire ce type de système.

La résolution par le Niveau Hiérarchique des indéterminismes directionnels s'oriente actuellement vers l'utilisation des systèmes experts, qu'il serait peut être bon de coupler avec une technique d'analyse partielle de sous-arbres et de recherche opérationnelle, ou de simulation prédictive simplifiée. Pour le moment, l'étude s'est limitée à l'utilisation de règles très simples permettant d'éviter certains engorgements.

Ce sujet est en phase de développement.

b) Les blocages :

Supposons que le banc de sortie soit bloqué pendant une période suffisante et que durant ce laps de temps arrivent en entrée une série de pièces de type  $m1m2$ . Un cas de figure présenté ci-dessous peut se produire.

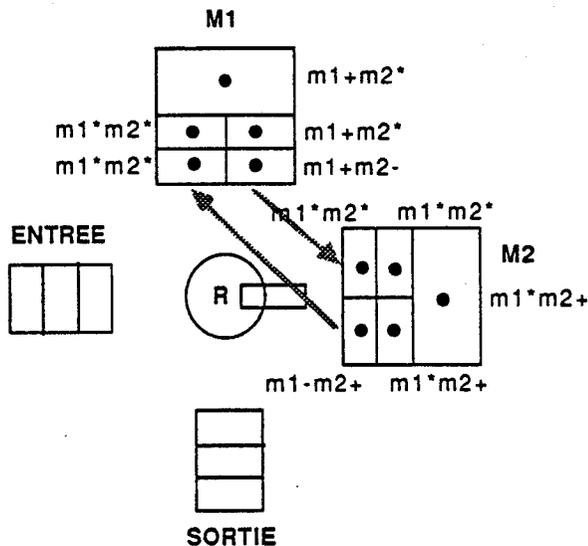


Figure I.28

La pièce  $m1^+m2^-$  placée en sortie de M1 ne peut accéder à M2 du fait de l'encombrement de cette dernière et bloque ainsi la machine M1.

Le même phénomène se produit pour la pièce  $m1-m2^+$  sur la machine M2.

Si le banc de sortie se libère à ce moment, on se trouve devant une situation de blocage qui ne peut se résoudre sans intervention du Niveau Hiérarchique.

Les règles du Niveau Hiérarchique relatives aux blocages doivent avoir pour mission de prévenir les blocages ou de les résoudre.

c) Les changements de mode de marche

De la même manière qu'un GRAFCET de CONDUITE règle les modes de marche d'un GRAFCET de PRODUCTION NORMALE, le Niveau Hiérarchique dispose de règles qui permettent d'effectuer le FORCAGE d'un marquage et, grâce aux arcs adaptatifs et aux places d'interface PC-NH, des opérations de GEL de marquage, de CONNEXION et DECONNEXION de sous-graphes.

De cette façon peuvent être prévus les modes dégradés du procédé automatisé.

#### L4 LE PROCEDE

Le terme "Procédé" recouvre l'ensemble des objets matériels ou logiciels auxquels s'adressent les ordres de la PARTIE COMMANDE et qui envoient en retour à celle-ci un certain nombre d'informations.

- On appelle en général PARTIE OPERATIVE la partie du procédé qui opère sur la matière d'oeuvre et le produit. Elle comporte en général :
  - i) des outillages et moyens divers mettant en oeuvre le processus d'élaboration
  - ii) des actionneurs destinés à mouvoir ou mettre en oeuvre ces moyens.
- L'INTERFACE PARTIE OPERATIVE/PARTIE COMMANDE est en général constituée des pré-actionneurs et des capteurs. Cette interface peut être très complexe et très hiérarchisée (voir la thèse de Madame ANAKOK (ANA 87I)).

Elle peut comporter des pré-actionneurs et des capteurs physiques et logiciels, avec en supplément un niveau d'arbitrage et de contrôle des échanges. Il peut s'agir véritablement de pré et post processeurs entre Partie Opérative et Partie Commande.

- Enfin, le PRODUIT manufacturé fait partie du PROCEDE. La reconnaissance individuelle des pièces manufacturées peut être, par exemple, une caractéristique de sa représentation.

Les frontières entre les diverses dénominations employées ci-dessus sont en général très sujettes à débat.

La représentation même simplifiée du procédé est très importante. Le procédé est l'interlocuteur du système de commande.

- i) D'une part, la connaissance ou la représentation de son évolution conditionne l'évolution de la partie commande.
- ii) D'autre part, c'est sur le comportement du procédé que l'on juge effectivement l'efficacité de l'unité de production.

Une image du procédé est déjà donnée à la partie commande par la coloration.

Une première représentation approximative pourra alors s'arrêter à la temporisation des places de la partie commande.

Mais pour plusieurs raisons telles que :

- i) L'élaboration de statistiques de fonctionnement nécessitant entre autres la reconnaissance individuelle des pièces.
- ii) La vérification de la bonne conception du système de commande au regard du comportement fin du procédé.
- iii) Et enfin la simulation des défaillances et des anomalies du procédé,

il s'avère nécessaire de disposer d'un modèle spécifique du procédé, en dehors de l'image opérative dont dispose déjà le système de commande.

La construction d'un modèle RdP dual de la Partie Commande pour modéliser la partie opérative peut être trop lourde et insuffisante. C'est pour cette raison qu'une représentation par langage objet, plus souple et plus modulaire, a été choisie (ICAS 87b).

Le procédé est ainsi représenté par des entités, ou objets, caractérisés par des champs descriptifs et des méthodes de comportement. Ces objets représentent les pièces, les organes actifs (robots, machines), les organes passifs (convoyeurs, stocks) et sont organisés en arborescence générique. Ils communiquent avec le système de commande et entre eux par envoi de messages qui modifient les champs et activent les méthodes de comportement.

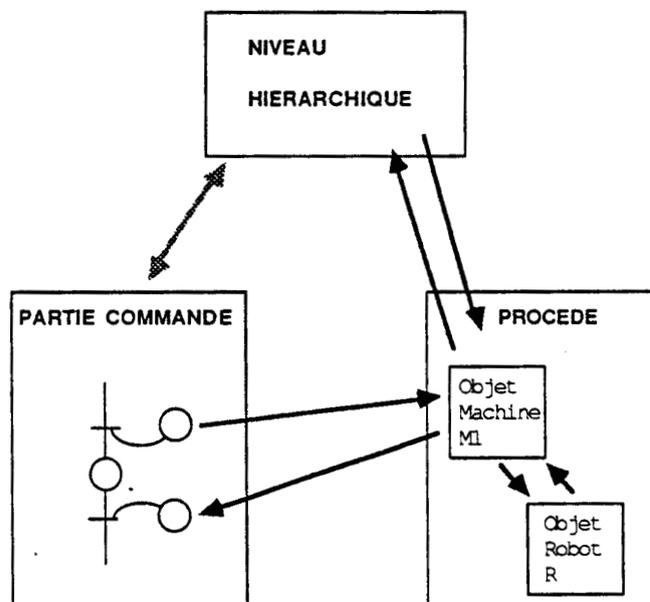


Figure I.29

## **L5 CONCLUSION SUR LES MODELES**

Les trois composantes essentielles du modèle utilisé viennent d'être présentées. Elles résultent, rappelons le, d'un choix effectué dans un but pratique industriel. D'autres types de modèles rencontrés dans les publications les plus récentes existent. Ils ont essentiellement un but d'évaluation de configurations matérielles d'atelier et ne s'intéressent pas en général à la conception du système de commande. L'outil retenu ici dans le projet CASPAIM s'adresse essentiellement à la réalisation et l'implantation du système de commande de l'unité de production. Il participe en outre à toutes les phases de la conception jusqu'à l'implantation.

Nous allons à ce propos parler maintenant de la phase d'évaluation par simulation du modèle modulaire. Elle a donné lieu à la création d'un logiciel original et qui occupe une place centrale dans la structure du projet.

## IL LA SIMULATION

### II.1 INTRODUCTION

Il est indispensable de connaître exactement le fonctionnement d'un système automatique, de pouvoir formuler et vérifier un certain nombre de propriétés.

D'autre part, l'évaluation qualitative est destinée à assurer une évolution correcte du système. C'est en effet une question essentielle que de savoir si un système est capable de réaliser l'ensemble des fonctions pour laquelle il a été conçu. L'évaluation qualitative doit permettre d'analyser et de résoudre les blocages et les indéterminismes induits par une mauvaise conception ou par une définition insuffisante.

D'autre part, l'évaluation quantitative consiste à pouvoir étudier l'aspect temporel des divers modèles utilisés afin d'analyser le comportement dynamique et les performances du système et de les optimiser en dimensionnant éventuellement certains paramètres ou en modifiant des stratégies.

L'augmentation de la puissance des modèles de représentation des systèmes de production automatisés a rendu très délicat l'étude de leurs propriétés.

Les modèles de description logique que sont les Réseaux de PETRI sont aujourd'hui sur le plan formel les modèles les plus avancés. C'est pour ces modèles qu'ont été étudiées le plus grand nombre de méthodes mathématiques de validation théorique. Ce sont essentiellement des techniques axées sur la recherche d'invariants liées à l'algèbre linéaire, à l'étude des grammaires formelles ou même des méthodes d'énumération des états (graphe de couverture). Des techniques de réduction ou de limitation de structure permettent d'abaisser la complexité de l'analyse, qui s'accroît avec l'augmentation des parallélismes de fonctionnement et avec la croissance exponentielle consécutive du nombre des états du système. Les outils théoriques de preuve formelle directe restent très limités et impuissants en face de la complexité des modèles les plus récents.

Par ailleurs, l'augmentation de la puissance et le caractère non autonome de certains modèles RdP ont rendu de nombreuses propriétés indécidables.

Enfin, des techniques associées à des langages non séquentiels et non déterministes permettent de représenter beaucoup plus facilement certains problèmes (ex. : Niveau décisionnel et Procédé) que les RdP. On a ainsi recours à des modèles hybrides pour lesquels plus aucun outil de preuve formelle n'est disponible. C'est le cas des modèles proposés dans le cadre du projet CASPAIM.

On a alors recours à des techniques de simulation.

Restée, il y a encore quelques années, l'apanage des bureaux d'études dans les secteurs de pointe, la simulation fait actuellement une entrée en force dans l'industrie. Alors que cette technique était essentiellement réservée à l'étude des processus continus, la simulation a fait plus récemment son entrée dans le domaine de la Productique et des systèmes de production discrets.

De récentes conférences consacrées uniquement à la simulation pour les systèmes de production discrets ont permis de faire le point sur l'état de l'art et ont montré l'énorme intérêt qui est porté à cette technique aussi bien dans la recherche universitaire que dans l'industrie (Conférences SIPPRODIS sur la pratique de la simulation en production discontinue, PARIS, 2 et 3 Juin 1987 avec le concours de la FNEGE, sous le patronage du MRES et de l'INRIA (ICAV 87)). A cette occasion a eu lieu un recensement des travaux des équipes de recherche en FRANCE, et le L.A.I.I. (Laboratoire d'Automatique et d'Informatique Industrielle de l'IDN) a été mentionné à propos du simulateur dont nous allons dire quelques mots maintenant.

## **IL2 LE SIMULATEUR**

Ce simulateur a été réalisé en 1986/1987 en langage Le-Lisp (ICAS 87b). Il est issu d'une expérience acquise sur plusieurs versions progressives construites depuis 1984 qui ont amené en particulier au choix du langage le supportant.

- Il a été installé sur mini-ordinateur VAX 11/750, micro-ordinateurs IBM PC et compatibles, et APPLE Mac Intosh. Des versions interprétées mais aussi compilées ont été testées.

- Il intègre les trois niveaux de modélisation décrits précédemment, c'est-à-dire
  - La Partie Commande
  - Le Niveau Hiérarchique
  - La Partie Opérative et le Procédé.

- Les possibilités d'utilisation modulaire du simulateur permettent de limiter l'étude à une simulation partielle concernant par exemple uniquement la Partie Commande pour l'évaluation du comportement ou concernant aussi le Procédé pour effectuer une évaluation statistique de performances. L'emploi modulaire d'outils de simulation de plus en plus fins permet une étude progressive et structurée du système de production.

La configuration du simulateur est ainsi adaptée à la complexité du modèle.

- Le simulateur peut travailler individuellement ou simultanément sur trois représentations qui forment un ensemble hétérogène :
  - \* Une représentation déclarative sous forme de règles de production activées par un moteur d'inférence pour le Niveau Décisionnel.
  - \* Une représentation de type objet avec transmission de messages pour le procédé.
  - \* Une représentation "procédurale" de type réseau de PETRI pour la Partie Commande.

Le simulateur peut en fait supporter un grand nombre de modèles RdP autres que les RdP structurés colorés adaptatifs. Il peut attaquer des réseaux à prédicats, des réseaux temporisés sur les places, interprétés sur les transitions, comprenant des time-out, des fonctions définies par l'utilisateur pour réaliser des générations de couleur, des temporisations, des interprétations fonctions de divers paramètres.

- Le simulateur combine deux approches classiques de la simulation, mais qui sont rarement présentées simultanément : un cycle par activité et un cycle par évènement. Dans le cycle par activité, c'est l'étude de l'état du graphe de commande qui permet de déterminer l'évolution du graphe. Si l'état résultant le permet, on réitère cette recherche.

Dans le cycle par évènement, l'évolution du système est régie par l'évaluation ou la présence d'évènements dans les divers échéanciers.

- Une des caractéristiques principales du simulateur est enfin l'existence d'une interruption programmée "Intervention Opérateur" permettant à tout instant de suspendre la simulation, de revenir au menu principal tout en assurant la reprise ultérieure de la simulation en cours. Cette interruption permet un gain considérable dans l'interactivité du système qui dépend essentiellement de l'utilisation du langage fonctionnel `Le_Lisp`, du principe de son évaluateur, de la souplesse d'utilisation qui autorise différentes approches de programmation dont l'approche déclarative. Que ce soit dans sa version interprétée ou compilée, le simulateur écrit en `Le_Lisp` permet à tout instant d'accéder à l'intégralité de l'univers de travail, c'est-à-dire à la fois les données relatives au modèle et la façon de les exploiter (fonctions utilisateur du programme). Le temps d'exécution est un peu plus important que pour certains systèmes de validation et simulation classiques écrits dans des langages algorithmiques impératifs et compilés (OVIDE en Fortran, SLAM en Fortran, QNAP2 en Fortran, etc...). Cependant, l'interactivité obtenue donne à l'utilisateur plus qu'un simple outil de simulation. Il s'agit véritablement d'un système de conception assistée.

### **II.3 CONCLUSION**

Le simulateur représente un des points clefs du projet CASPAIM. Il s'agit du premier logiciel qui a été écrit dans la chaîne modulaire qui constitue actuellement le corps du projet. Avec les modèles, il a constitué le point de départ de la démarche complète de conception assistée que nous allons présenter maintenant.

### III.

## UNE DEMARCHE COMPLETE DE CONCEPTION ASSISTEE DEPUIS LES SPECIFICATIONS JUSQU'A L'IMPLANTATION

### III.1 LES QUESTIONS POSEES

Avec la construction des outils de modélisation et d'évaluation d'unités de production flexibles décrits précédemment, de nouvelles questions se sont rapidement posées :

#### III.1.1. Le besoin d'une méthodologie de conception assistée des modèles

La première question a été de savoir comment réaliser concrètement et le plus facilement possible, à l'aide des techniques exposées, un modèle de l'unité de production à réaliser, c'est-à-dire une représentation adaptée de sa Partie Commande par un Réseau de PETRI structuré, coloré et adaptatif, de son Niveau Décisionnel par des règles de décision et de l'ensemble Partie Opérative-Procédé par des entités objet convenables.

Bien que les outils de modélisation soient bien définis, la fabrication en elle-même du modèle de l'atelier étudié reste délicate. Par exemple, comment à partir de la définition de la cellule décrite fig. I.2 obtenir simplement le graphe de la Partie Commande fig. I.3. Ce problème de recherche de méthodes de conception assistée du modèle est actuellement très important. Les réunions récentes de la Commission Système Logique de l'AFCEC ont montré comment ce point préoccupait les industriels présents confrontés à des problèmes de conception d'automatismes programmés de très grande taille. Ce problème appelle une démarche de structuration des systèmes de commande dont nous avons déjà parlé dans l'introduction générale et une démarche essentiellement assistée par ordinateur étant donné la taille des problèmes à traiter. Des systèmes informatiques de spécification et de production automatique de programmes de commande et/ou de simulation basés sur le modèle GRAFCET ont déjà été proposés (PIASTRE développé au LAMM par C. CAZALOT, F. PRUNET et J.L. STURLESE, AMI 100 développé par le LAMM et l'entreprise Automatique Industrielle S.A. (IPRU 87a| IPRU 87b)). Le Projet CASPAIM propose une démarche analogue, mais relative à un outil de

modélisation plus sophistiqué que le GRAFCET et que nous avons décrit précédemment. L'interaction avec l'utilisateur doit laisser transparentes les étapes techniques de modélisation et ne l'interroger que pour des spécifications de haut niveau. Par ailleurs, un des buts recherchés est de ne pas enfermer l'utilisateur dans un cadre défini par la structure de l'interface de communication.

### III.1.2. L'interprétation difficile des résultats de simulation

La deuxième question est de savoir comment interpréter les résultats de simulation, c'est-à-dire comment ajuster le modèle en l'absence d'outils hypothétiques théoriques d'optimisation pour obtenir plus modestement un bon comportement. Une démarche utilisant les techniques des systèmes experts est à envisager.

### III.1.3. La réalisation de systèmes de commande implantés : L'objectif industriel et opérationnel de la démarche.

La troisième question est relative à l'implantation effective des modèles sur des dispositifs technologiques industriels.

L'émulation de modèles Réseau de PETRI par des automates programmables serait idéale dans l'optique qui nous intéresse. C'est d'ailleurs un point qui serait souhaité par de nombreux laboratoires qui travaillent sur ces modèles dans un but industriel (LAAS, INPG-LAG).

Toutefois, il faut avoir à l'esprit les points suivants :

- i) Pour des raisons pratiques, il semble hors de question de solliciter les milieux industriels avec la connaissance de modèles trop techniques issus de la recherche,
- ii) Ces modèles sont en outre en constante évolution.

Par ailleurs

- iii) Il existe un modèle à événements discrets normalisé (norme française nov. 82, internationale nov. 87) qui bénéficie actuellement d'une large diffusion industrielle grâce à la collaboration efficace des milieux universitaires et industriels qui l'ont fait naître depuis 1975. Il s'agit du GRAFCET. Cet outil graphique est non seulement un outil de

spécification mais est en outre devenu un outil de programmation séquentiel implanté sur de nombreux automates industriels de grande diffusion.

Rappelons que ce modèle autorise la description d'un système au moyen de réseaux composés d'une part d'actions à exécuter et d'autre part de synchronisations, fonctions des variables d'état du processus.

Le projet CASPAIM propose donc, après une phase de traduction des modèles du système de commande grâce à des règles de transposition, une implantation effective et distribuée sur des matériels existants et avec des outils de programmation normalisés.

La spécification, la modélisation, l'évaluation et l'implantation du système de commande de cellules flexibles de production dans le contexte des questions qu'on vient de décrire constitue ainsi l'ensemble du projet CASPAIM.

Les figures qui suivent récapitulent la structure actuelle du projet, les étapes qui ont été réalisées, celles qui sont en développement et celles qui sont prévues.

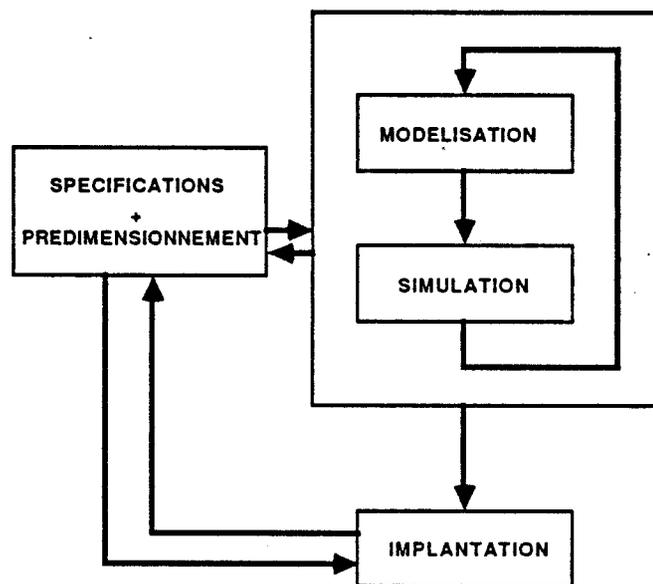


Figure I.30

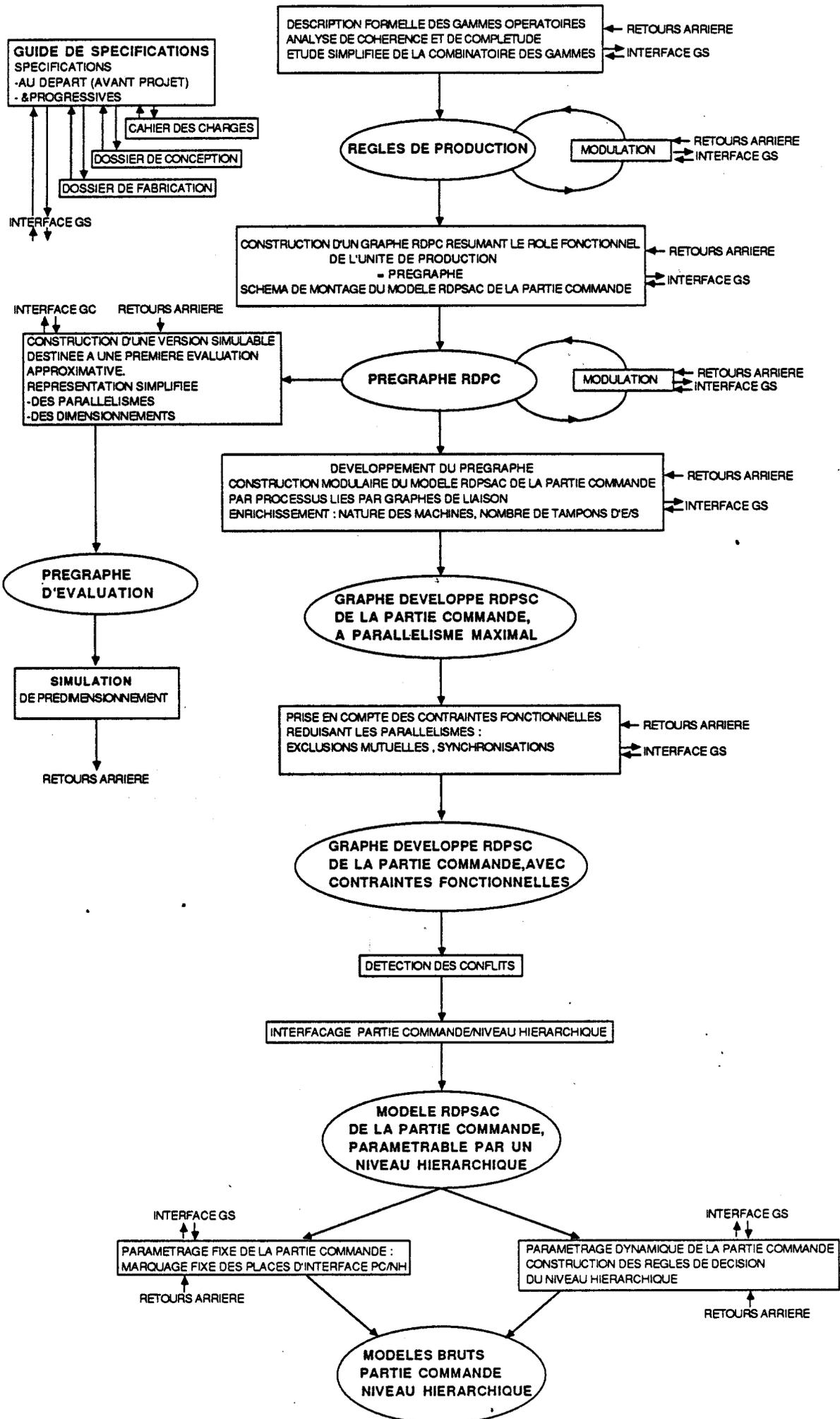
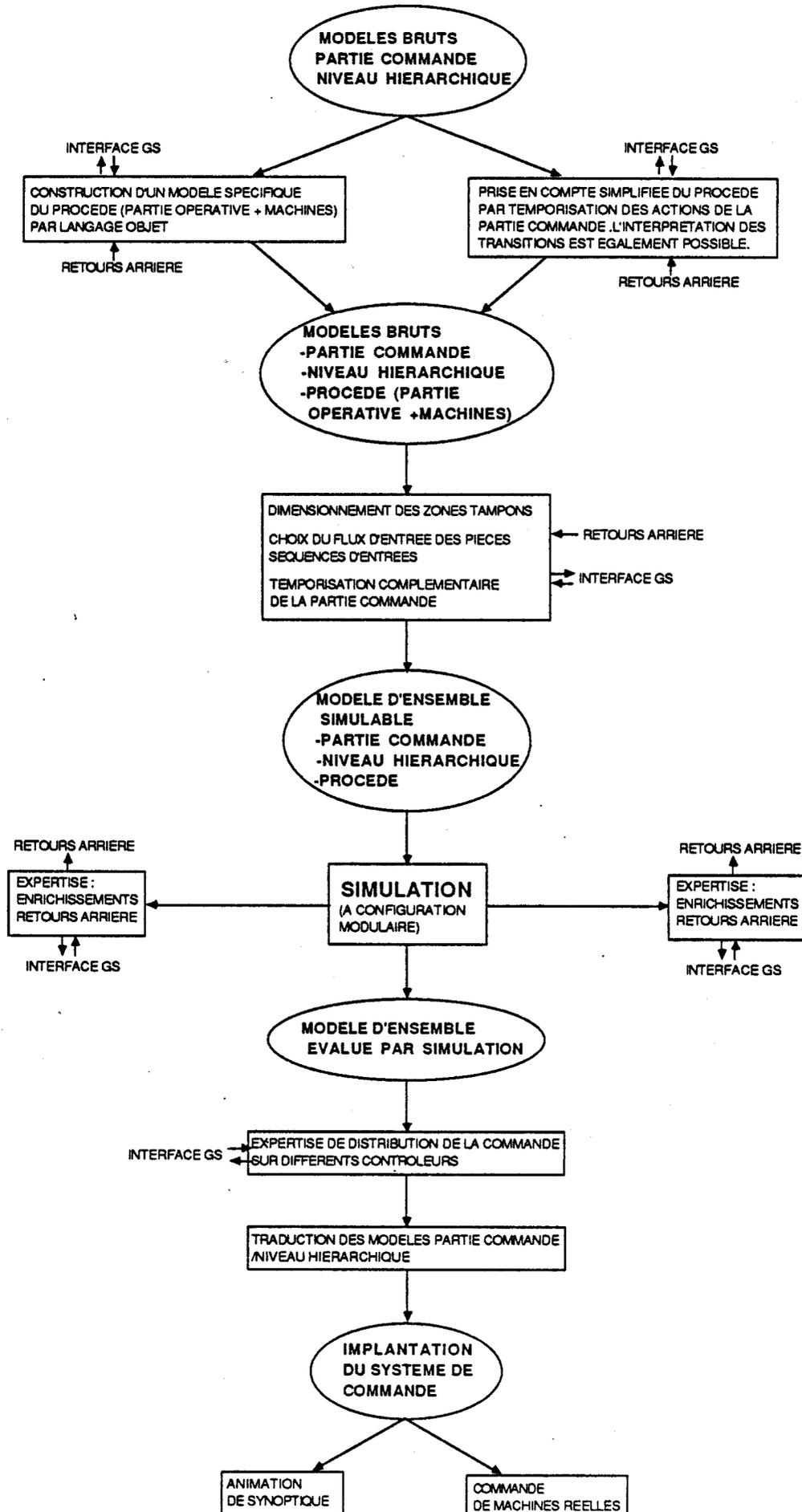


Figure I.31

(SUITE)



Donnons ici quelques commentaires afin de bien préciser ce que fait le projet, et afin de situer l'objet du travail présenté dans ce mémoire au sein du contexte plus général dans lequel il s'inscrit.

### III.2 LA CHAÎNE DE C.A.O.

#### III.2.1. La démarche de spécification

On distingue habituellement trois étapes dans la définition d'un produit industriel :

- le cahier des charges qui définit les besoins et les contraintes du client souvent exprimés en termes économiques. Ce document est établi en général par les services technico-commerciaux et par les services économiques.
- le dossier de conception. Etabli à partir du cahier des charges par les bureaux d'études, il spécifie les caractéristiques fonctionnelles du produit à réaliser pour répondre aux exigences du cahier des charges.
- le dossier de fabrication. Construit à partir du dossier de conception par les bureaux des méthodes, il définit les moyens de production à mettre en oeuvre pour réaliser le produit.

On imagine alors aisément la complexité de la définition du produit quand celui-ci est un atelier de production, et à fortiori quand cet atelier réunit des caractéristiques de flexibilité et d'intégration. Le nombre des corps de métier intervenant dans la définition est alors important et l'organisation des différentes interventions doit faire l'objet d'une démarche de gestion de projet. Cette démarche doit assurer l'absence de contradictions dans les spécifications et garantir la prise en compte de tous les aspects d'un ensemble de spécifications. Elle doit faciliter leur écriture et leur documentation. Ces exigences sont à la base d'un certain nombre de guides concepteurs. En particulier, un travail de thèse de docteur ingénieur (LAUG 87), mené en 1987 en liaison avec le LAII-IDN dans une équipe de RENAULT AUTOMATION centrée sur le projet ESPRIT 623 a consisté à recenser des outils d'aide à la définition et à la documentation de projets (HIPO, SADT, PERSPECTIVE, SARS et EPOS), à les évaluer précisément, et à adapter celui de ces outils de spécification de systèmes informationnels qui se prêterait le mieux, à la

spécification des systèmes de fabrication. Ceux-ci présentent en effet à la fois des aspects logiciels et matériels. Il s'agissait ensuite d'intégrer cet outil dans une chaîne de conception. Ce type de guide sert à assurer la cohérence et la complétude des définitions et des actions des différents intervenants du projet, mais à notre avis reste trop du domaine des avant-projets industriels où l'étape de spécification est réalisée une fois pour toute au départ de la chaîne de conception.

L'idée d'un guide de spécification est conservée pour le projet CASPAIM mais un certain nombre de problèmes seront spécifiés au fur et à mesure durant toute la démarche de conception, de simulation, d'évaluation et d'implantation des modèles. La définition des spécifications se fera donc à la fois au départ et durant toute les phases d'utilisation de l'outil CASPAIM (par l'intermédiaire de l'Interface GS (Guide de Spécifications) - voir figure I.31 -). Ceci permet de poser les problèmes à l'endroit où ils doivent intervenir et de retourner en fin de démarche un récapitulatif structuré et hiérarchisé des spécifications de l'utilisateur, constamment remises en question durant toute la démarche. On évite ainsi à la fois le désordre de certains cahiers des charges où tout est mélangé et le caractère artificiel des avant-projets trop spécifiés avant toute tentative de réalisation d'un modèle de l'atelier. L'interface GS représente actuellement les points de dialogue interactif avec l'utilisateur. Une seule étape d'un Guide de Spécifications a été abordée pour le moment (IGAS 88) pour le projet CASPAIM. Ce guide reste pour le moment un sujet de développement futur.

Rappelons que ce type de guide est actuellement un centre d'intérêt très important dans l'industrie. Il a pour tâche d'organiser les informations durant toute une chaîne de C.A.O. Citons le projet A.I.P. (Automatisation Intégrée de Production) mené par un consortium regroupant les Sociétés TELEMECANIQUE ELECTRIQUE, 3IP et le Laboratoire d'Automatique Industrielle de l'ISMCM (IFRA 87) qui va très loin dans ce sens. Développé dans l'idée de la réalisation d'un Poste de Travail pour Automaticien, il introduit l'idée d'un modèle virtuel "partagé" d'un équipement automatisé, consultable et modifiable par l'intermédiaire de différents filtres destinés à différents intervenants (aspect modèle de commande, aspect modèle électrique, aspect modèle pneumatique, aspect maintenance, etc...).

### III.2.2. La modélisation

Le but de la modélisation est d'obtenir les modèles de la PARTIE COMMANDE, du NIVEAU HIERARCHIQUE et du PROCEDE (PARTIE OPERATIVE + MACHINES). Ces modèles sont simulables et permettent en même temps une implantation facile sur dispositifs technologiques industriels programmables (micro-ordinateurs, automates). C'est dans ce but qu'ont été choisis les outils de modélisation. D'autres types de modèles récents, sophistiqués, ne bénéficient malheureusement pas d'un support technologique d'implantation industrielle efficace.

L'objectif du projet CASPAIM est d'assurer une assistance complète de la modélisation par logiciel, le but ultime étant de ne solliciter l'utilisateur que pour des spécifications de haut niveau ne nécessitant aucune connaissance des modèles. L'assistance logicielle doit donc rendre à terme la modélisation transparente à l'utilisateur qui ne désire pas y accéder. On estime qu'il est toutefois nécessaire de laisser une possibilité d'accès au modèle pour un utilisateur initié désirant avoir un contrôle fin de son modèle.

La construction du modèle du Niveau Hiérarchique et du modèle spécifique du Procédé est encore faite manuellement.

En ce qui concerne la construction de la partie commande, un premier projet de recherche (IKAR 871) s'est concentré sur l'édition assistée de Réseaux de Petri structurés à partir d'un système à base de connaissances travaillant sur une base de modules élémentaires Réseau de PETRI et vérifiant par analyse syntaxique la stricte conformité de la construction. Toutefois, cette démarche d'assistance à la construction exigeait de l'utilisateur une connaissance trop fine du modèle. Une démarche complète de réalisation assistée et transparente du modèle a alors été progressivement élaborée. Elle permet de partir des gammes opératoires pour aboutir au modèle Réseau de PETRI Structuré Adaptatif Coloré de la PARTIE COMMANDE. Cette démarche s'est décomposée en deux étapes.

- La première permet, à partir de l'étude des gammes opératoires, d'aboutir à un schéma de montage très propre (complet, cohérent et agrégé) de la PARTIE COMMANDE RdPSAC. Ce schéma de montage est un graphe RdP Coloré qu'on appelle PREGRAPHE. Cette étape qui fournit la base d'assemblage de la PARTIE COMMANDE constitue l'objet du mémoire présenté ici. Elle sera décrite en détail dans les chapitres qui suivent.

- La deuxième étape constitue le développement du PREGRAPHE pour obtenir le modèle RdPSAC de la PARTIE COMMANDE. Cette étape qu'on appelle STRUCTURATION ressemble à une compilation d'un code source fourni par le PREGRAPHE pour obtenir un code objet qui est le modèle de la partie commande. Toutefois, la différence réside dans le fait que la STRUCTURATION apporte un enrichissement par apport interactif de nouvelles informations. Il ne s'agit donc pas seulement d'un simple assemblage de modules RdP génériques suivant un schéma d'assemblage imposé par le Prégraphe mais surtout d'un enrichissement important de la description. Cette étape a fait l'objet d'une thèse récente (IBOU 88b).

L'ensemble de ces deux étapes est aujourd'hui complètement supporté par un logiciel écrit en langage LE\_LISP.

La première représente un logiciel d'environ 250 K octets.

La seconde un logiciel d'environ 350 K octets.

Une adaptation du PREGRAPHE permet par ailleurs d'effectuer de premières évaluations par calcul analytique et par simulation.

Cette application qui est encore en développement sera abordée au chapitre III, Partie III.

### III.2.3. La simulation

Le simulateur écrit en LE\_LISP a déjà été décrit.

Les développements en cours concernent une amélioration des performances du simulateur à l'aide d'un découpage des modèles sur une architecture multiprocesseurs. La Partie Commande, le Niveau Hiérarchique et le Procédé auraient ainsi une évolution parfaitement asynchrone.\* En outre la décomposition de la simulation de la Partie Commande sur plusieurs processeurs permettrait de diminuer notablement la taille des graphes RdP à simuler. Enfin, d'autres améliorations techniques relatives à la scrutation de l'état du graphe de commande sont prévues, destinées à diminuer la durée du pas de simulation, afin de répondre véritablement aux exigences industrielles concernant les temps de simulation.

\* La décomposition du Niveau Hiérarchique en plusieurs niveaux asynchrones est également envisagée à plus longue échéance.

Un point important concerne l'interprétation des résultats de simulation et le choix des retours-arrière. Aucune méthode n'a été mise au point pour le moment et ce sujet reste un projet de développement futur. Il s'agira d'une méthode d'expertise qui nécessitera donc l'accumulation d'un grand nombre d'expériences de simulation et d'études de cas industriels.

Ce problème étant très délicat, nous travaillons par ailleurs actuellement à l'élaboration d'une méthode de prédimensionnement progressif et de préétude de comportement de l'architecture matérielle et du système de commande à partir de modèles dérivés du PREGRAPHE (cf chapitre III, Partie III) afin de limiter les simulations fines aux tests terminaux.

#### III.2.4. L'implantation

L'implantation du modèle du système de commande a été réalisée sur quelques exemples, en particulier l'exemple présenté au début du chapitre et un exemple présenté dans (IBOU 88b).

- Le modèle RdPSAC de la PARTIE COMMANDE a été implanté sur des automates TELEMECANIQUE de type TSX 47 et TSX 67 reliés par Réseau TELWAY 7. La traduction sous forme GRAFCET est systématique grâce à la structuration des Réseaux de PETRI en processus. Le caractère adaptatif et les graphes de liaison sont traduits sous forme de réceptivité de transitions du GRAFCET. Le codage numérique des marques colorées se fait par l'intermédiaire de mots internes des automates. Des astuces de programmation permettent de représenter les liaisons entre processus implantés sur automates différents par échanges de mots communs (réseau TELWAY 7) ou par échanges de blocs-texte (liaisons SCM - Serial Communication Module - et réseau). (ICRU 87 | BEA 87 | TEL 87 | BOI 88). La distribution et la traduction du modèle de la PARTIE COMMANDE se révèle être systématique et donc facilement automatisable.

L'idéal serait alors de disposer sur les automates d'un code GRAFCET source intermédiaire compilable sur la machine cible par l'intermédiaire d'un compilateur spécialisé. Le code GRAFCET pourrait alors être écrit sur n'importe quel ordinateur tel un langage de programmation évolué. La traduction serait complètement automatique, alors qu'elle reste manuelle pour le moment.

projet CASPAIM sont l'objet de contrats de recherche avec des partenaires industriels, ce qui est peut-être le reflet de l'approche pragmatique qui a été choisie pour le projet.

Deux objections ont été formulées toutefois à l'encontre du projet :

La première objection reproche à la démarche de descendre encore pour le moment à un niveau trop fin de modélisation pour faire l'étude du comportement de configurations matérielles de cellules flexibles.

Cette objection qui a été émise par un responsable des méthodes et automatismes chez PEUGEOT Automobiles se réfère à la méthode de conception actuelle des chaînes de production et des systèmes de transport (BEN 85) :

- Une première approximation de type régime stationnaire et de type statistique (méthodes de calcul de files d'attente) est habituellement faite par des méthodes analytiques à partir de la détermination des flux, des durées et des vitesses. Plusieurs méthodes sont utilisées (DOL 83|CAV 87|, telles que l'algorithme CAN-Q (Commuter Analysis of Network of Queues) développé à partir de la méthode de BUZEN (CER 82) par SOLBERG (SOL 80)), les algorithmes de type MVA (Mean Value Analysis) avec les heuristiques de REISER ou HILDEBRANT.
- A cause des approximations faites, ces méthodes analytiques peuvent conduire à des erreurs de conception mais elles permettent un dégrossissage du problème par un premier dimensionnement. Les services méthodes des grandes entreprises utilisent alors des méthodes de simulation de configurations matérielles qui, employées seules, constitueraient d'après les utilisateurs une approche trop lourde. Des simulateurs dédiés ou des langages généraux de simulation statistique sont alors mis en jeu depuis le début des années 80. Parmi ces derniers, citons SLAM (SLAM I et II) simulateur général statistique à événements discrets basé sur un réseau à files d'attente et activités (PRI 79) ; RESQ (Research Queue Package) développé par le centre de recherche IBM à Yorktown, NY, USA issu d'une refonte de GPSS ; QNAP2 (Queuing Network Analysis Package 2).

(N.B. : L'évolution et le recensement de ces langages sont faits de façon précise et complète dans |CAV 87| - Conférence SIPRODIS).

Des sociétés de service en informatique industrielle telles que SYGRAL s'intéressent de très près à ce type de problème. L'entreprise TELEMECANIQUE ELECTRIQUE a dans ses projets la réalisation d'un compilateur C pour automate programmable.

- Le Niveau Hiérarchique tourne sur un IBM AT relié à un des TSX 67 par prise console

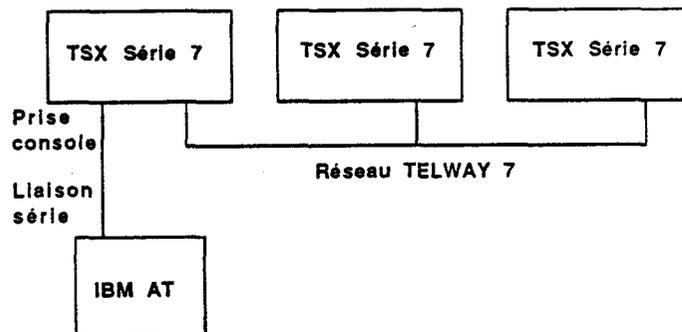


Figure I.32

Les règles de décision sont activées par un moteur d'inférence écrit en LE\_LISP sur IBM AT et interfacé avec des routines de communication (via la ligne de communication série) écrites en langage C qui permettent de scruter et modifier l'état des variables automates et en particulier l'état des étapes (ICRA 88).

### CONCLUSION

Le projet CASPAIM qui vient d'être décrit rapidement ici a déjà été présenté dans son état d'avancement actuel en quelques occasions (Congrès IMACS 88 à PARIS (BOU 88c), Conférence IBM à BRUXELLES (GEN 87), Sessions Systèmes Logiques de L'AFCEC en décembre 87 et mars 88, Présentations à des équipes de la DRD TELEMECANIQUE). Ce projet, ainsi que d'autres projets de chaînes de CAO de Productique qui sortent depuis quelques temps de la recherche universitaire, suscite un intérêt industriel à la fois chez des constructeurs d'automates industriels et de matériels informatiques (TELEMECANIQUE, IBM) et chez certains utilisateurs potentiels. C'est la preuve d'un certain rapprochement dans ce domaine entre les objectifs de la recherche universitaire, plus attentive aux réalités industrielles, et ceux du monde industriel confronté à des problèmes plus complexes. Quelques applications du

- 2) Un reproche a été fait quant au caractère "presque procédural" du modèle de la Partie Commande, qui nécessiterait la description exhaustive de trop de commandes par rapport aux modèles déclaratifs.

Répondons deux choses à cela :

- i) Les modèles à base de description langage objet (OASYS |CAV 87|), Travaux de VALENCIENNES (|TAH 87|) sont très intéressants. Il sera d'ailleurs possible de transformer les graphes de liaison de la partie procédurale en transmission de messages et les processus en objets dont le comportement sera de type RdP. C'est un peu cette démarche qui est utilisée au LACN (NANCY) avec méthodes comportementales écrites en GRAFCET. Nous avons retenu dans CASPAIM la solution qui a été la plus facile à installer sur matériels industriels.
- ii) La taille du modèle n'est pas un obstacle véritable lors de la conception puisque la construction du modèle de la Partie Commande est obtenu par une méthodologie maintenant supportée par des logiciels.

C'est d'ailleurs la première étape de cette conception assistée du modèle de la Partie Commande qui va être maintenant exposée.

La combinaison des deux méthodes, du type "prédiction-correction", permet actuellement d'approcher la solution par un prédimensionnement, puis d'affiner et de vérifier les résultats.

La réponse à cette objection est que :

i) Une phase de prédimensionnement par calcul analytique et par simulation est prévue dans le projet CASPAIM sur un modèle intermédiaire qui est un enrichissement du Prégraphe et qui pour le moment n'est pas encore stabilisé.

ii) Les techniques industrielles du moment s'intéressent à des chaînes transfert, éventuellement polyvalentes, mais linéaires. L'outil CASPAIM s'adresse à des unités de production flexibles au sens total donné à ce terme dans l'introduction générale. Il s'agit ainsi d'un problème nouveau et différent. On rappelle que ces unités ont pour but d'absorber une charge non cadencée et variée. La conception du système de commande et de l'architecture matérielle ne peut être scindée sur ce type d'unités.

iii) Les techniques industrielles courantes permettent de définir une architecture matérielle mais tout le système de commande reste à concevoir et est laissé à l'imagination des automaticiens.

**La deuxième objection** qui découle en fait de la précédente concerne la taille du modèle de la partie Commande. Cette objection se décompose en deux alinéas :

1) L'automatisation de la modélisation de la partie Commande génère trop de code Réseau de PETRI. Il s'agit là d'un problème classique connu pour les compilateurs et les langages d'assemblage (MIC 88). Notons que des solutions efficaces ont été trouvées dans ce domaine. Le caractère actuel un peu trop systématique de la construction automatisée sera affiné par la suite.

En outre, la décomposition multi-processeurs de la simulation permettra de travailler sur de gros graphes.

## **CHAPITRE II**

### **PREMIERE ETAPE DE CONCEPTION ASSISTEE DE LA PARTIE COMMANDE OBTENTION DU PREGRAPHE**

## INTRODUCTION

Le contexte et l'état du projet CASPAIM ont été présentés dans le chapitre précédent.

Le travail qui est l'objet de ce mémoire concerne plus particulièrement une première étape de conception assistée du système de commande des unités flexibles de production.

Au cours des chapîtres II et III nous nous proposons de présenter la chaîne de conception assistée qui permet, à partir du niveau des spécifications des gammes opératoires, d'obtenir le modèle Réseau de Petri Structuré, Adaptatif et Coloré de la Partie Commande d'une unité flexible.

Etant donné l'importance et la complexité de la démarche, la chaîne d'analyse et de conception de la Partie Commande conduisant jusqu'à l'implantation informatique a été scindée en deux étapes qui font chacune l'objet d'un travail de thèse :

- C'est la première étape qui est le point central de ce mémoire.

Elle concerne l'obtention, à partir du niveau de spécification des gammes, d'un graphe préliminaire appelé PREGRAPHE qui constitue l'ossature de montage de la Partie Commande. Cette démarche fait la matière du chapitre II dans sa totalité.

Le PREGRAPHE est en outre sujet à des évolutions constantes et s'avère être progressivement un outil intéressant pour d'autres développements en cours. Ces perspectives d'extension de l'utilisation du PREGRAPHE seront abordées au chapitre III.

- La deuxième étape concerne la construction de la Partie Commande à partir du PREGRAPHE par l'intermédiaire d'une phase qu'on appelle STRUCTURATION. Cette phase qui a donné lieu à une soutenance de thèse récente (IBOU 881) sera expliquée brièvement au début du chapitre III afin d'assurer une autonomie de lecture du mémoire.

Le chapitre II qui suit maintenant va être décomposé en cinq parties principales :

- I) Le formalisme de description des gammes opératoires.
- II) L'analyse de cohérence et de complétude d'enchaînement logique des gammes opératoires.
- III) La traduction et l'agrégation des règles opératoires sous forme d'un graphe RdP coloré : le PREGRAPHE.
- IV) Les étapes de modulation.
- V) L'étude simplifiée de la combinatoire des gammes.

## I. LE FORMALISME DE DESCRIPTION DES GAMMES OPERATOIRES

Il existe plusieurs degrés possibles de définition des notions de gammes opératoires, d'opérations et de moyens de production faisant référence à des bases de données techniques de différents niveaux destinés à des niveaux de supervision différents. Cette décomposition a été particulièrement abordée par plusieurs travaux et publications provenant du Laboratoire d'Informatique Appliquée de l'INSA de LYON (|CAM 85|) qui se sont concentrés sur la structuration de bases de données techniques autour de nomenclatures et gammes-mères en vue de l'élaboration automatique de gammes (|FAV 84| |CAM 84| |TEM 85| |CAM 87|).

Afin de ne pas surcharger les notions qui vont intervenir plus loin et qui sont plutôt orientées vers la construction du système de commande des cellules de production, les travaux mentionnés ci-dessus ne sont que cités ici. Sachons simplement qu'ils existent et qu'ils représentent dans le domaine de la PRODUCTIQUE des recherches tout à fait complémentaires des travaux du LAII (IDN) en amont du projet CASPAIM.

Ainsi, de façon simple, pour fixer les idées :

- Une gamme opératoire est un ensemble d'opérations élémentaires permettant de décrire un cycle de fabrication.

La gamme logique ne s'intéresse ni à l'identification des lieux autour desquels sont exécutées les différentes opérations ni aux opérations qui n'ont pas un rôle direct dans la transformation du produit. La gamme physique, elle, s'intéresse à l'identification des lieux (éventuellement limitée au type du lieu sans plus de décomposition) et à la succession des opérations réalisées autour de ces lieux pour permettre l'évolution pratique du produit manufacturé. Les lieux constituent les points du cycle de fabrication où une séquence d'ordres doit être donnée par la partie commande au procédé, où une décision doit être prise quant au traitement et quant à l'évolution des objets qui transitent par ce lieu.

Le niveau de détail abordé par ce qu'on appelle ici la gamme physique n'est pas profond. La gamme physique étudiée est en fait proche d'une gamme d'ordonnancement.

C'est dans le contexte de ces notions que vont être construites les gammes physiques opératoires pour chaque produit manufacturé.

Chacune des gammes va être décomposée en opérations élémentaires.

Chacune des opérations va être représentée par une règle opératoire de la forme

PREMISSES  $\rightarrow$  CONSEQUENTS

Les prémisses et les conséquents sont constitués d'une conjonction de prédicats qui sont de la forme :

(présence objet lieu)

qui signifie la présence de l'objet sur le lieu.

La forme générale des règles de production associées aux opérations de la gamme opératoire va donc être la suivante :

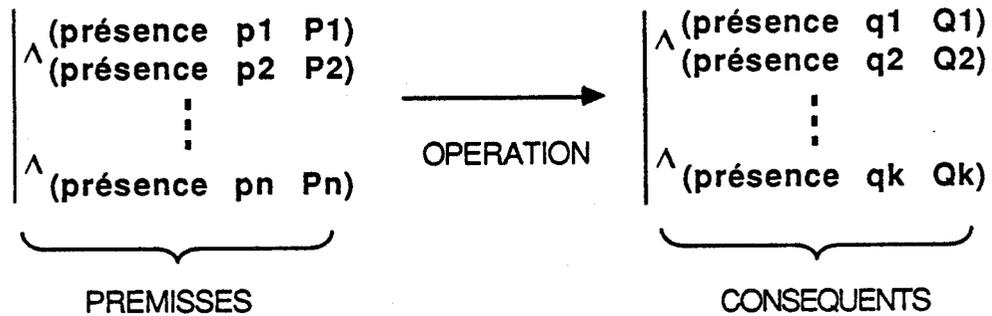


Figure II.1

La disjonction de prédicats est traduite par la disjonction des règles qui les utilisent.

Si l'on considère un ensemble de prédicats (ou base de faits) décrivant une situation atteinte par des objets manipulés par la gamme, les règles de production vont permettre par un jeu de réécriture de faire évoluer ces faits et donc de suivre les opérations de chacune des gammes.

Sur la figure précédente l'OPERATION ne va pouvoir être effectuée qu'en présence de l'ensemble des PREMISSES. Elle va consommer ces PREMISSES pour former des résultats ou CONSEQUENTS.

Bien que le formalisme réellement utilisé par les outils logiciels soit littéral, nous utiliserons souvent dans l'exposé une correspondance graphique construite à partir des Réseaux de PETRI colorés. Par exemple la règle précédente sera représentée par la transition suivante :

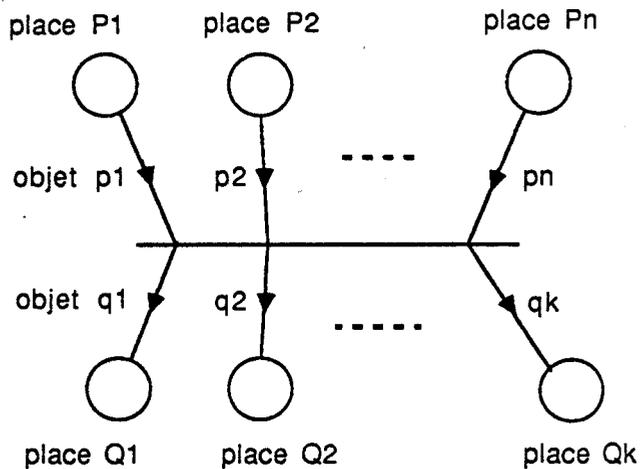


Figure II.2

On décrit ainsi la structure des règles opératoires.

Des informations supplémentaires sont en outre ajoutées à la description de chacune des règles pour classer les opérations qu'elles représentent.

On distingue trois types d'actions élémentaires :

1) Les actions transformationnelles (ou fonctionnelles) :

Ces opérations modifient l'état matériel du produit, c'est-à-dire ses caractéristiques intrinsèques (ex. : composition, forme, propriétés physiques, ...).

Ces actions désignent par exemple :

- les usinages (tournage, fraisage, alésage, ...)
- les conditionnements (traitement thermique, mise en peinture, ...)
- les solidarisations et désolidarisations d'un assemblage (soudage, vissage, frettage, ...)

**Ex. 1** : Considérons une pièce de type  $o1+o2-o3+t-o4-$ . Les deux opérations  $o1$  et  $o3$  ont déjà été effectuées sur cette pièce. Les opérations  $o2$ ,  $o4$  ainsi que l'opération de tournage  $t$  sont encore à faire. On représentera l'opération de tournage sous la forme suivante :

Littérale :

règle : tournage  $_o1+o2-o3+t-o4-/o1+o2-o3+t+o4-$

prémises : (présence  $o1+o2-o3+t-o4-$  tour)

conséquents : (présence  $o1+o2-o3+t+o4-$  tour)

type : fonctionnelle

détail : tournage.

RdPC

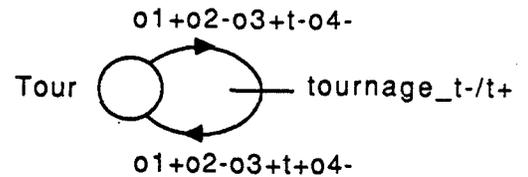


Figure II.3

**Ex. 2** : Considérons maintenant l'ensemble  $(ab)^-$  des deux pièces  $a$  et  $b$  positionnées pour être assemblées par soudage. L'opération de soudage sera représentée sous la forme suivante :

Littérale :

règle : soudage  $_(ab)^-/(ab)^+$

prémises : (présence  $(ab)^-$  poste  $_S$ )

conséquents : (présence  $(ab)^+$  poste  $_S$ )

type : fonctionnelle

détail : soudage

RdPC

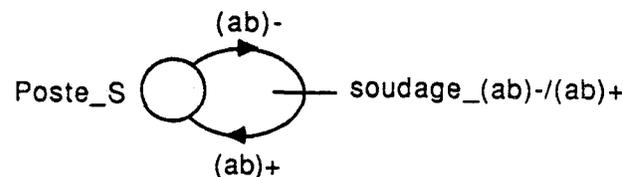


Figure II.4

## 2) Les actions positionnelles

Ce sont des opérations élémentaires qui ne servent qu'à situer les objets dans l'environnement de production, en vue de la réalisation des opérations précédentes. Elles ne répondent à aucune exigence inhérente du produit, mais dépendent de facteurs extrinsèques ou contingents (ex. : modes de fonctionnement).

Ces actions désignent :

- des transferts entre postes (chariots, navettes, convoyeurs, ...)
- des repositionnements sur un poste (retournements, ...)
- des positionnements avant assemblage et après désassemblage

**Ex. 1 :** Considérons une pièce a. On représentera le transfert de la pièce a d'un poste P1 vers un poste P2 sous la forme suivante :

Littérale :

règle :  $P1 \rightarrow P2\_a$

prémises : (présence a P1)

conséquents : (présence a P2)

type : positionnelle

détail : transfert simple

RdPC

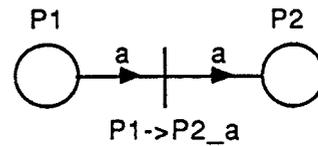


Figure II.5

**Ex. 2 :** Considérons une opération r de retournement d'une pièce  $o1+r-o2-$  effectuée sur le poste P. On a les représentations suivantes :

Littérale :

règle : retournement  $o1+r-o2-/o1+r+o2-$

prémises : (présence  $o1+r-o2-$  P)

conséquents : (présence  $o1+r+o2-$  P)

type positionnelle

détail : repositionnement

RdPC

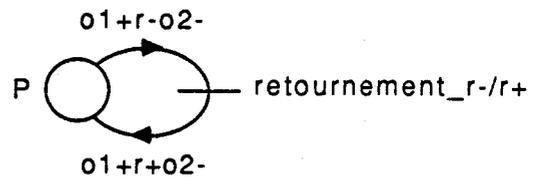


Figure II.6

**Ex. 3 :** Soit une opération de palettisation d'une pièce a provenant d'un poste A sur une palette p en attente au poste P.

Représentation littérale :

règle :  $A \rightarrow P\_a \& p/pa$

prémises : (présence a A)

(présence p P)

conséquents : (présence pa P)

type : positionnelle

détail : assemblage

RdPC

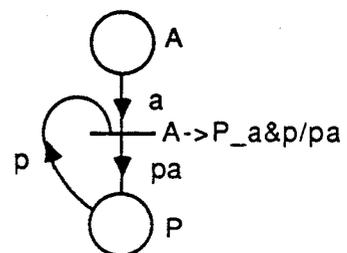


Figure II.7

**Ex. 4 :** De même pour la dépalettisation

Représentation littérale

règle :  $P \rightarrow A\_pa/a\&p$

prémises : (présence pa P)

conséquents : (présence a A)

(présence p P)

type : positionnelle

détail : désassemblage

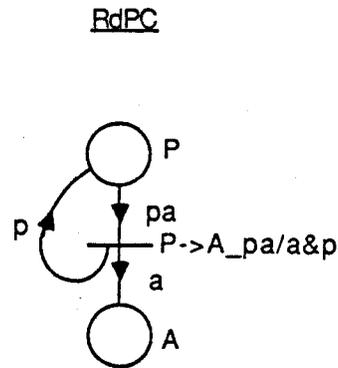


Figure II.8

**Ex. 5 :** Regardons maintenant le positionnement avant assemblage d'une pièce a venant d'un poste A et d'une pièce b venant d'un poste B.

Supposons que la mise en correspondance des deux pièces se fasse sur un poste P. On a les représentations ci-dessous.

Littérale :

règle : positionnement-assemblage1\_a&b/(ab)^-

prémises : (présence a A)

(présence b B)

conséquents : (présence (ab)^- P)

type : positionnelle

détail : assemblage

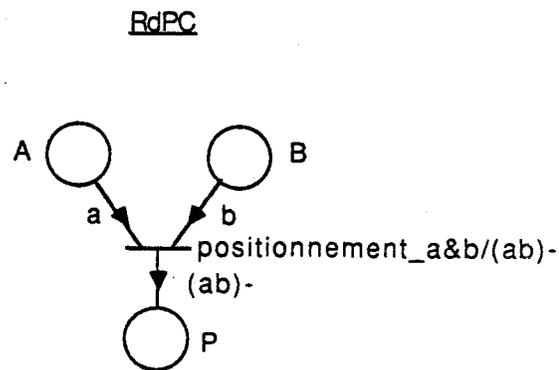


Figure II.9

Supposons maintenant que la mise en correspondance exige la présence en premier lieu de b sur le poste P. On a :

Représentation littérale

règle : positionnement-assemblage2\_a&b/(ab)^-

prémises : (présence a A)

(présence b P)

conséquents : (présence (ab)^- P)

type : positionnelle

détail : assemblage

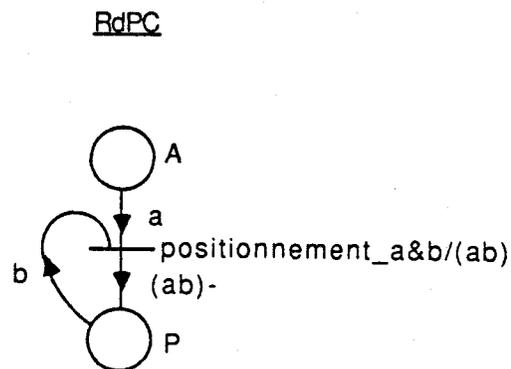


Figure II.10

3) Les actions informationnelles :

Ces actions permettent de garantir la conformité des états matériels effectivement obtenus à partir des interventions précédentes.

Ces actions désignent :

- des opérations de métrologie
- des inspections qualitatives.

**ex.** : Considérons une pièce  $f+$  ayant subi une opération de fraisage. A la sortie du poste de métrologie  $M$ , cette pièce va être classée dans trois intervalles  $f+1$ ,  $f+2$ ,  $f+3$  en fonction des résultats de mesures. On représente par exemple la classification de  $f+1$  par la description donnée ci-dessous.

représentation littérale :

règle : mesure  $f+/f+1$

prémises : (présence  $f+ M$ )

conséquents : (présence  $f+1 M$ )

type : informationnelle

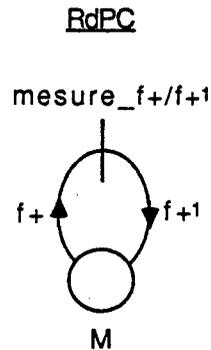


Figure II.11

Remarques :

**R1)** La classification des actions élémentaires est très importante pour la phase de développement et de construction ultérieure de la partie commande. Bien que dans un premier temps, seule la structure des règles opératoires est considérée pour examiner la cohérence et la complétude de l'enchaînement logique des actions élémentaires, les phases ultérieures menant à la réalisation de la partie commande utilisent le supplément d'information introduit par la classification. La classification précédente constitue une première classification qui a déjà évolué et qui sera amenée à évoluer en fonction des besoins du système de structuration de la partie commande dont on reparlera au chapitre III. Pour le moment, seule est utilisée la décomposition résumée ci-dessous.

Type de l'action	Détail
1 - <u>transformationnelle</u> (appelées aussi fonctionnelles dans les bureaux des méthodes)	dénomination quelconque (sans importance pour le moment)

2 - positionnelle

choix entre :

- i) transfert-simple
- ii) repositionnement
- iii) assemblage
- iv) désassemblage
- v) désassemblage-assemblage  
(ex. transfert entre  
une platine de bridage  
et une palette)

3 - informationnelle

dénomination quelconque  
(sans importance pour le  
moment).

**R2)** La dénomination des règles et des objets peut être quelconque et laissée au choix de l'utilisateur. Il est conseillé d'utiliser une dénomination précise et explicite.

Cependant certaines possibilités de génération automatique des noms de transitions et de variables de coloration, dans la phase ultérieure d'agrégation et d'obtention du Prégraphe (voir chapitre II partie III), imposeront à l'utilisateur une construction normalisée des noms de règles, de la forme :

nom général de l'action \_\_ identification plus précise en fonction  
de la pièce, de paramètres divers  
précisant l'action.

**ex. :** P1→P2\_t+f-l-m-

transfert de P1 à P2 de la pièce

t+f-l-m-

tournage fraisage lavage mesure

fraisage \_\_f170-/f170+

Séquence de fraisage de type 170 sur la pièce  
f170-

Pour la dénomination des objets, le choix est libre.

Par exemple :

- Prenons une pièce de type f12-t7+.  
Cette pièce doit subir une séquence de fraisage de type f12.  
La séquence de tournage de type t7 a déjà été effectuée.
  
- Prenons une pièce de type ((ab)-t+(de)+)-  
Cette pièce comporte trois groupes non encore solidarisés :
  - l'assemblage "ab" non solidarisé.
  - le composant "t" usiné.
  - l'assemblage "de" solidarisé.

Le premier problème qui se pose maintenant est de savoir comment décomposer une gamme en opérations élémentaires.

Cette décomposition va servir à construire ultérieurement la partie commande à partir de l'assemblage de modules RdPSAC génériques. Un certain nombre de conventions sont dès lors imposées par cette phase qui est appelée structuration.

Essentiellement, nous dirons que chacun des lieux (places RdP) qui interviennent dans les actions élémentaires correspond à un endroit où une séquence de commandes doit être appliquée par la partie commande au procédé lors du transit d'un objet sur ce lieu. A ces lieux physiques sont attachés de façon implicite une FIFO d'entrée et une zone opératoire à une place où la séquence d'ordres est appliquée (cf figure II.12). La phase de structuration amènera à associer un processus à chacun de ces lieux et à compléter les caractéristiques implicites par d'autres caractéristiques. Nous y reviendrons.

Les opérations élémentaires seront menées sur et entre ces lieux. Ce sont elles qui détermineront avec précision l'architecture du processus de commande des places auxquelles elles se réfèrent. Là aussi, nous y reviendrons.

Afin de familiariser le lecteur avec la décomposition de l'évolution des pièces en actions élémentaires, nous allons maintenant présenter un certain nombre de petits exemples d'illustration.

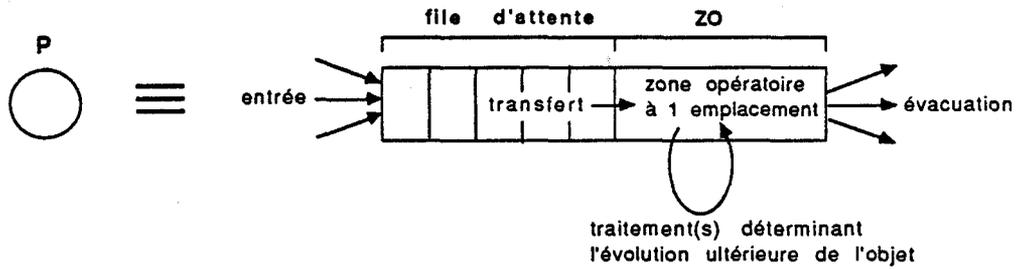


Figure II.12

Les caractéristiques brutes d'une place sont :

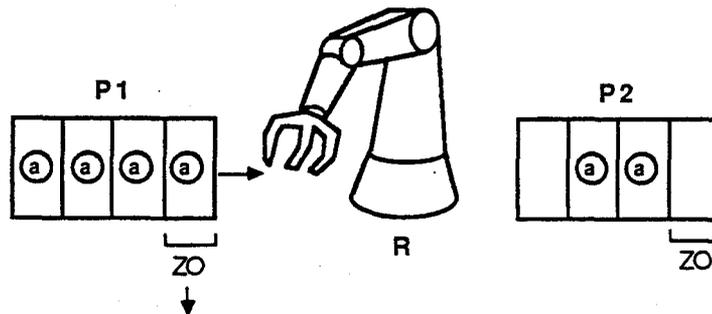
- une FIFO d'entrée
- une zone opératoire à une place où la séquence d'ordres du processus de commande associé est appliquée au procédé.

Ces caractéristiques brutes implicites sont modulées et complétées par la phase ultérieure de développement ou structuration. cf chapitre III partie I.

**ex. 1 :**

Examinons le cas d'un transfert simple de pièces a entre deux postes P1 et P2.

- a) Supposons que le transfert soit effectué de façon unitaire par un bras de manutention.



Ordre de transfert  
sur le bras manipulateur R  
si :  
-la ZO de P1 contient une pièce  
à transférer vers P2  
-la FIFO de P2 a une place libre

Figure II.13

Cette opération est représentée simplement par la règle ci-dessous

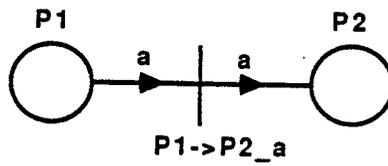
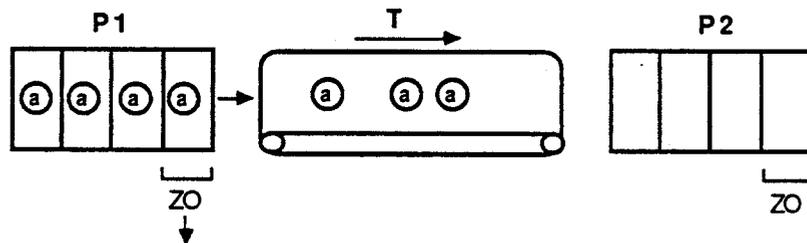


Figure II.14

- b) Supposons que le transfert soit effectué par un tapis roulant T en fonctionnement continu



Ordre de transfert  
sur le tapis roulant T

si :

-la ZO de P1 contient une pièce  
à transférer vers P2

-la FIFO de P2 comporte une place libre  
compte tenu des pièces "a" qui sont sur T

Figure II.15

On aura encore :

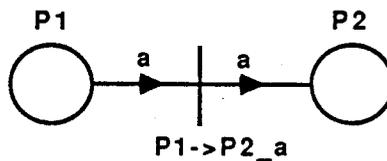
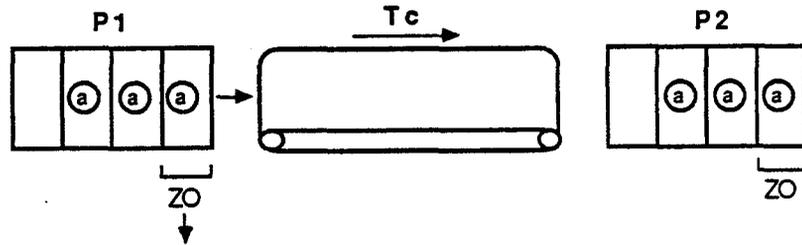


Figure II.16

- c) Supposons que le transfert soit effectué par un tapis roulant Tc commandé par le transfert d'une pièce a depuis la ZO de P1 et s'arrêtant lorsque toutes les pièces transportées ont été évacuées du tapis.



Ordre de transfert  
sur le tapis roulant Tc  
et envoi d'un ordre de marche  
(éventuellement redondant) à Tc  
si :  
-la ZO de P1 contient une pièce  
à transférer vers P2  
-la FIFO de P2 comporte une place libre  
compte tenu des pièces "a" qui sont sur Tc

Figure II.17

Là encore on aura :

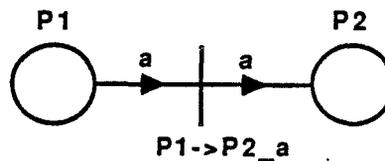


Figure II.18

Dans ces trois cas de figure, les places P1 et P2 et la règle P1→P2\_a cachent des réalités différentes dont les nuances seront appréhendées par "paramétrage" des modules génériques de construction automatique des processus de commande. La structure de base proviendra toutefois de la règle P1→P2\_a aussi simple soit-elle !

Terminons cette description succincte du transfert par quelques cas de figure un peu plus délicats.

- d) On suppose ici que le tapis roulant s'arrête automatiquement lorsque la FIFO de P2 est pleine et/ou qu'il peut bloquer les pièces a pendant un certain temps opératoire propre.

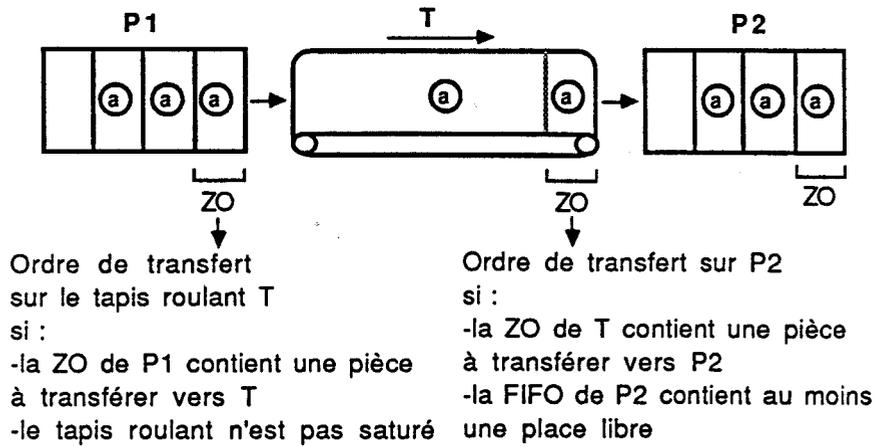


Figure II.19

Le tapis roulant a ici une faculté de commande sur les pièces a.

Alors de façon claire on a la représentation suivante sous forme de règles :

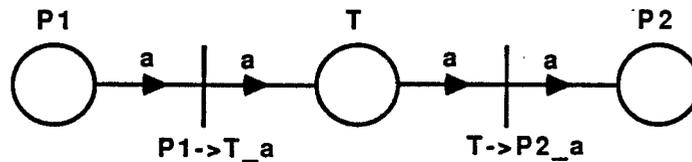


Figure II.20

- e) Considérons la figure ci-dessous

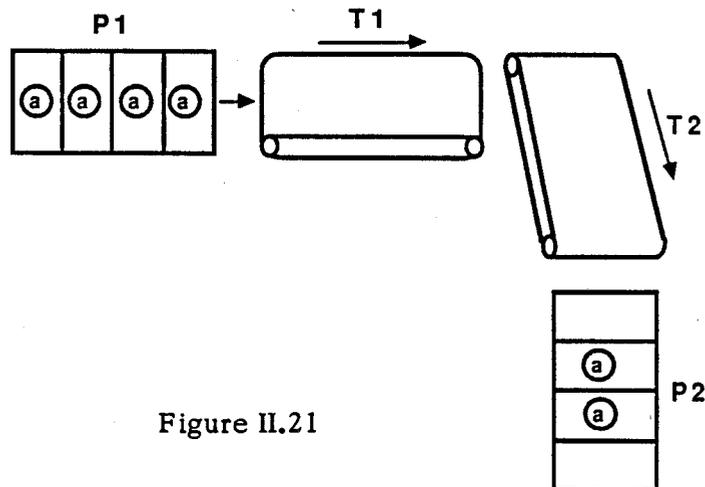
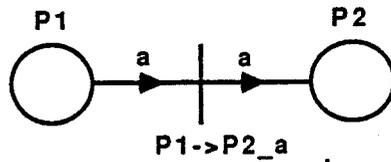


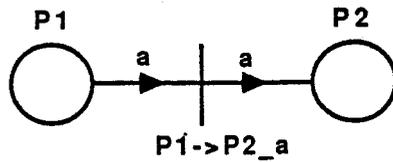
Figure II.21

Alors :

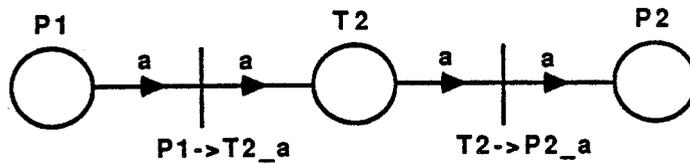
- si T1 et T2 fonctionnent suivant le mode b) on a :



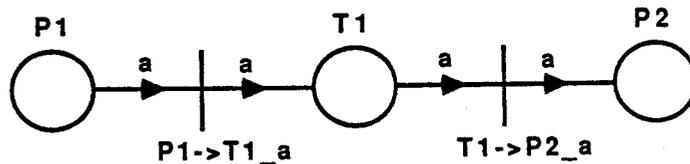
- si T1 et T2 fonctionnent suivant le mode c) on a :



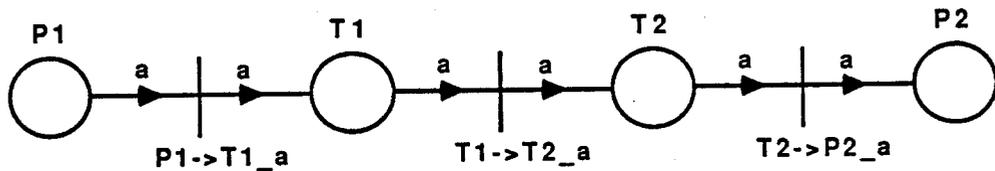
- si T2 fonctionne suivant le mode d) on a :



- Si T1 fonctionne suivant le mode d), alors



- Enfin si T1 et T2 fonctionnent suivant le mode d), alors on a :



Le problème, à chaque fois, est de bien déterminer les opérations élémentaires et les lieux sur lesquels les pièces sont placées en file d'attente d'une opération.

f) Enfin, prenons l'exemple ci-dessous

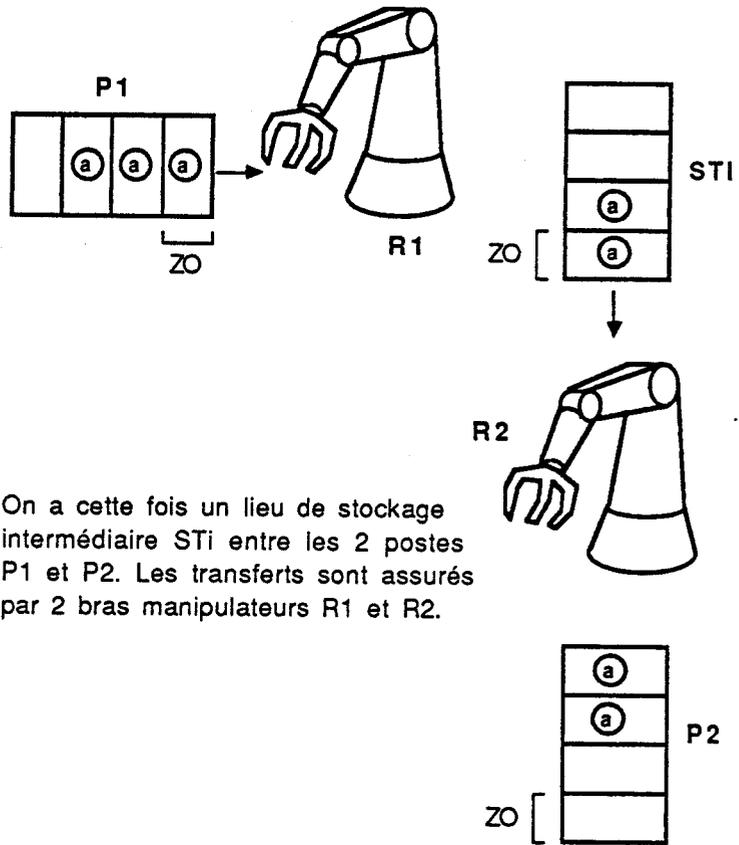


Figure II.27

On a cette fois un lieu de stockage intermédiaire STi entre les 2 postes P1 et P2. Les transferts sont assurés par 2 bras manipulateurs R1 et R2.

La représentation sous forme de règles opératoires est donnée ci-dessous :

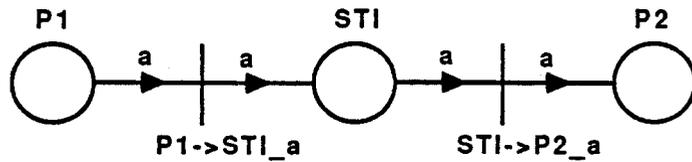


Figure II.28

ex. 2 :

Considérons le cas d'un poste de tournage T situé en guichet par rapport à la ligne de convoyage L

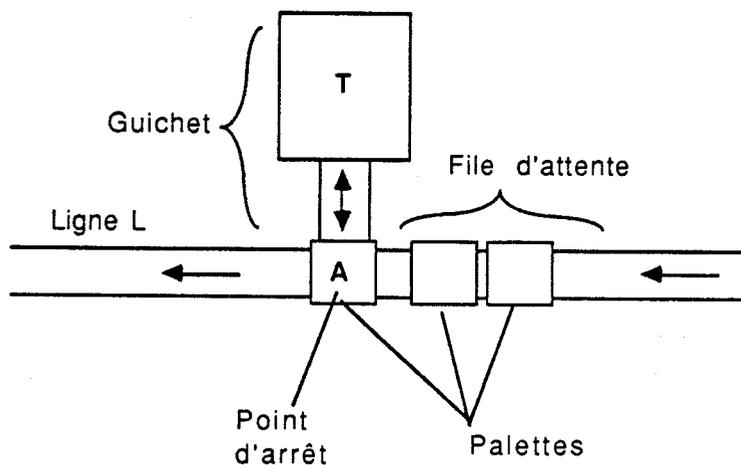


Figure II.29

La ligne L assure le déplacement de palettes p qui transportent chacune une pièce. On notera t- les pièces qui doivent être tournées, t+ les pièces qui ont été tournées.

La ligne L comporte un point d'arrêt A devant le poste T et une file d'attente devant ce point.

- a) On suppose que la pièce t- est transférée seule sur le tour et que la palette p reste en attente en A. On a la représentation suivante :

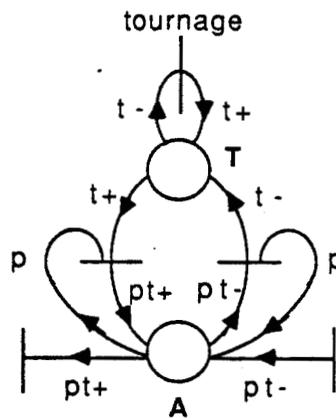


Figure II.30

- b) On suppose que la pièce t- est transférée avec sa palette p sur le tour et que le point d'arrêt A reste vide dans l'attente du transfert inverse T→A (cas de nombreux tours verticaux). On a alors la représentation ci-dessous :

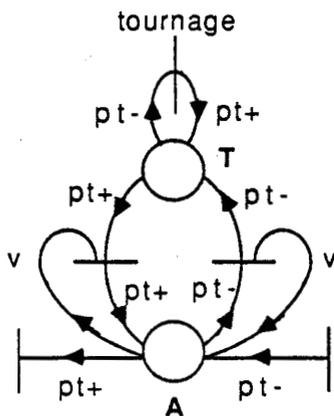


Figure II.31

la marque v étant introduite pour imposer que le point d'arrêt reste vide pendant l'évolution de l'ensemble pièce/palette dans le guichet.

**ex. 3 :**

- a) Considérons le cas d'une chaîne transfert linéaire sur laquelle on dispose de trois postes de travail P1, P2, P3 en ligne où sont effectuées les opérations d'usinage respectives o1, o2 et o3.

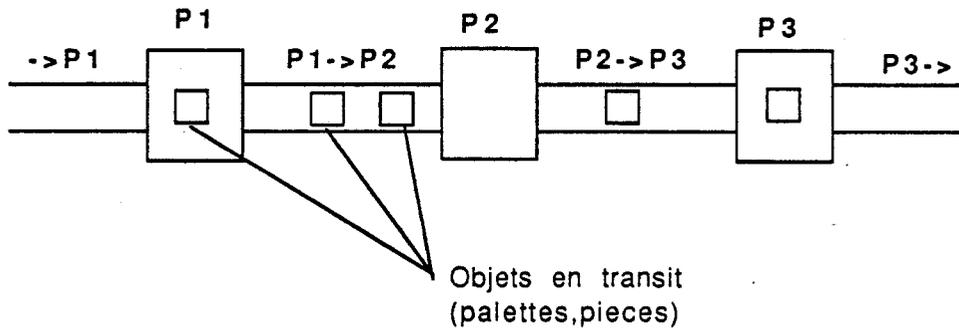


Figure II.32

Les parties  $\left\{ \begin{array}{l} \rightarrow P1 \\ P1 \rightarrow P2 \\ P2 \rightarrow P3 \\ P3 \rightarrow \end{array} \right.$  du système de transport peuvent être libres

(indépendantes) ou liées (synchronisées). (On dit alors que le transfert est libre ou lié).

On suppose que les systèmes de transfert n'ont aucune possibilité de commande sur les pièces en transit.

La décomposition en actions élémentaires ne s'intéresse qu'au cheminement des pièces et à leur évolution. On aura le schéma suivant :

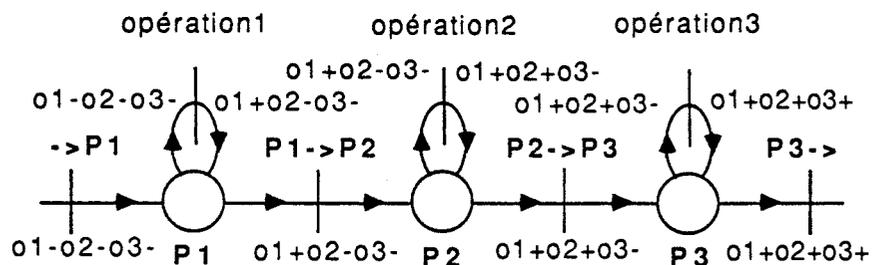


Figure II.33

- b) Supposons maintenant que deux lieux de stockage intermédiaire STi1 et STi2 soient placés respectivement entre P1 et P2 et P2 et P3. Ces lieux constituent des zones tampons entre postes, éventuellement dimensionnées de façons différentes en fonction du temps de cycle et de la fiabilité de chacun des postes.

Comme on l'a déjà vu, ces lieux immobilisent et gèrent éventuellement le stockage intermédiaire des pièces. Ils imposent une séquence de commande au procédé. On a alors la représentation suivante :

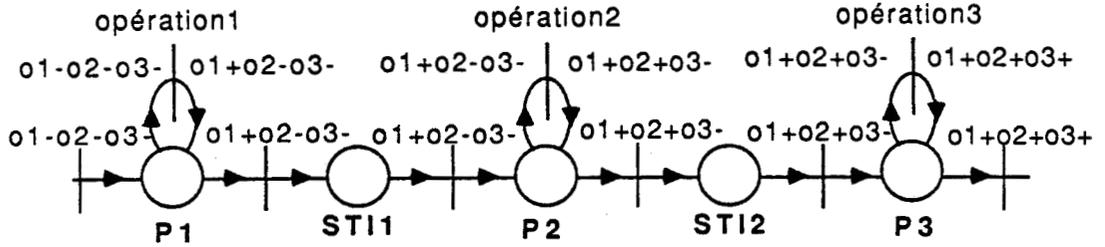


Figure II.34

**ex. 4 :**

Envisageons maintenant le cas d'une chaîne transfert linéaire polyvalente sur laquelle on dispose de trois postes de travail P1, P2 et P3 en ligne où sont effectuées les opérations d'usinage suivantes :

- o11 sur P1
- o12 sur P1
- o2 sur P2
- o3 sur P3

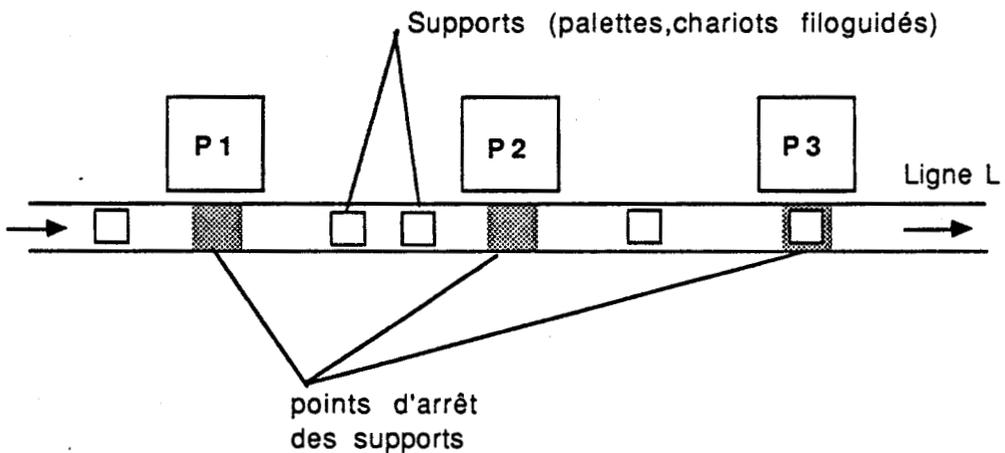


Figure II.35

Plusieurs supports s parcourent la ligne L dans le même sens.

On distingue deux types s1 et s2 de supports.

Ces supports ont des mouvements indépendants. C'est le cas par exemple de chariots filoguidés ou de palettes sur un convoyeur à butées débrayables (la palette peut être immobilisée sans immobiliser le convoyeur). On dit que le transfert est libre.

Des pièces o1o2 et o1o3 respectivement associées aux supports s1 et s2 peuvent circuler sur la ligne L. La panne d'une machine n'immobilisera pas la pièce qui, inachevée, pourra poursuivre son chemin vers un poste de remplacement ou une zone de traitement des rebus.

On aura la représentation suivante :

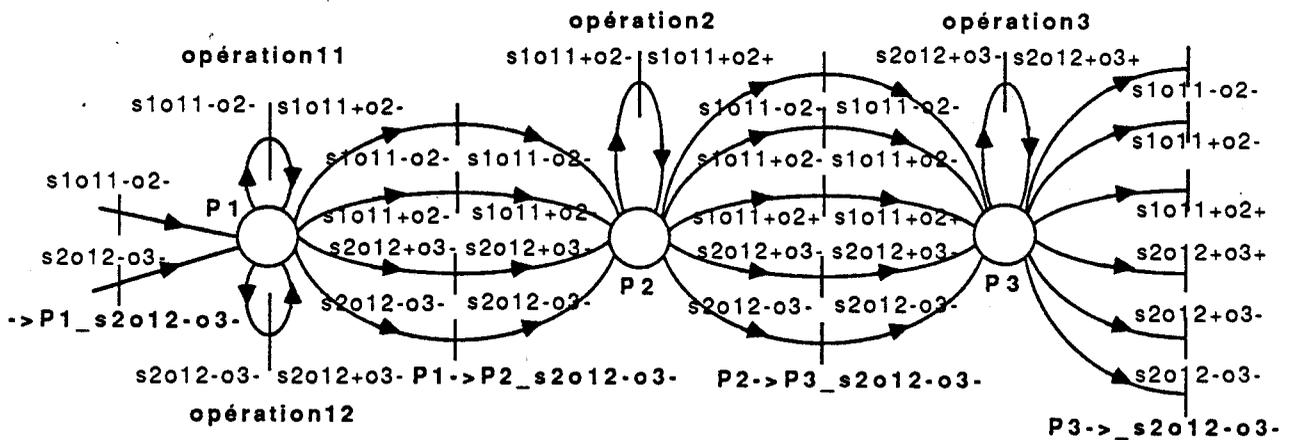


Figure II.36

**NB :** Pour des raisons de clarté, seuls quelques noms de règles élémentaires sont mentionnés.

**ex. 5 :**

- a) Considérons l'exemple d'un convoyeur Q circulaire utilisé comme tampon de chargement et déchargement d'une ligne de convoyage principale L par rapport à laquelle il est placé en épi.

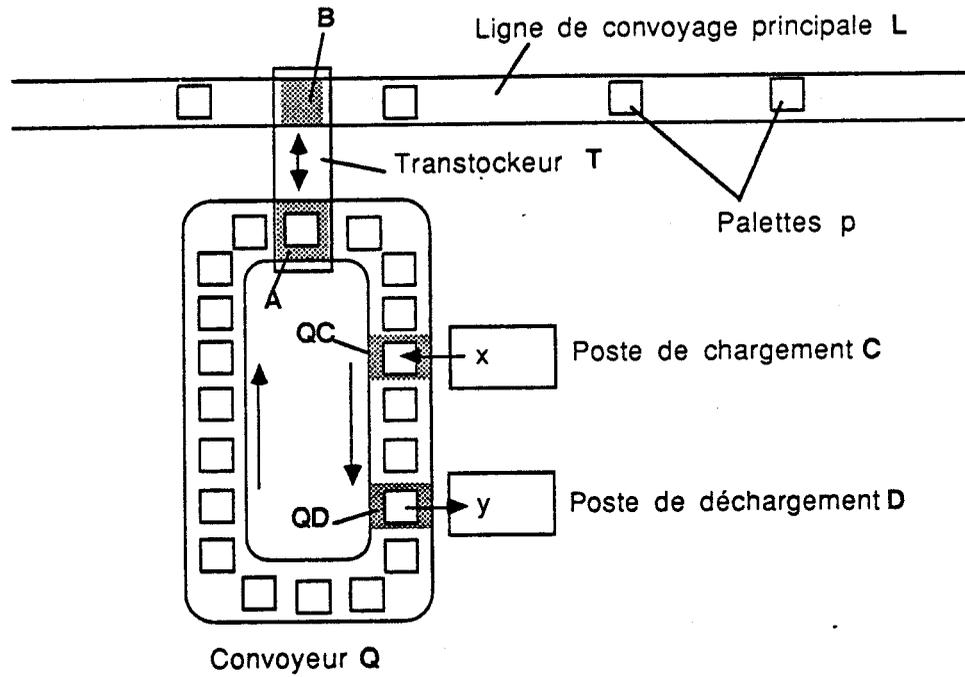


Figure II.37

Soit QC et QD les points d'arrêt du convoyeur Q correspondant aux postes C et D de chargement et déchargement de pièces diverses qu'on désignera par les identificateurs globaux x (chargement) et y (déchargement).

Soit A et B les points d'arrêt correspondant au transtockeur T respectivement sur le convoyeur Q et sur la ligne de convoyage L.

La représentation sous forme de règles élémentaires de ce cas de figure est donnée ci-dessous :

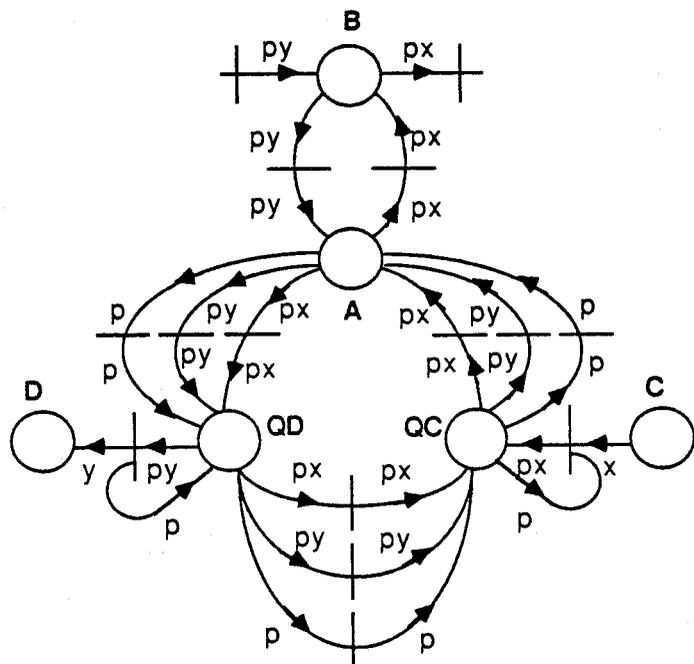


Figure II.38

- b) Supposons qu'à la place des palettes  $p$ , on ait des places libres d'un convoyeur à bande.

Le schéma devient :

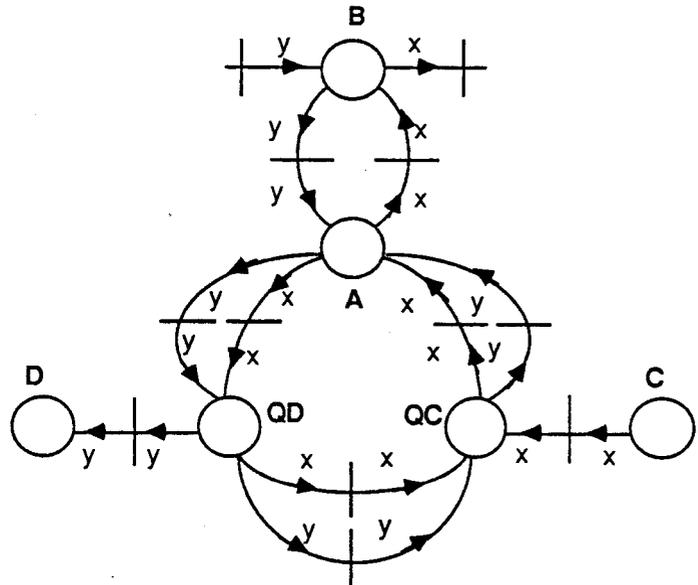


Figure II.39

- c) Supposons que la ligne de convoyage principale  $L$  utilise des palettes  $p_1$  différentes des palettes du convoyeur  $Q$ . On a entre A et B :

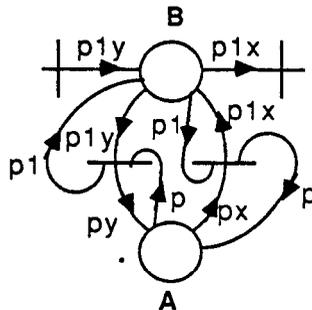


Figure II.40

- d) Si les palettes vides ne sont pas nécessairement immobilisées en B, on a le schéma ci-dessous sur lequel apparaît un indéterminisme directionnel pour les palettes  $P_1$ .

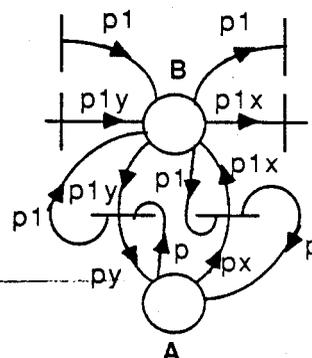


Figure II.41

e) Supposons que le convoyeur principal L soit un convoyeur à bande sans palettes. On a alors entre A et B les deux schémas suivants qui sont comparables aux fonctionnements c) et d) précédents vis à vis des pièces x et y.

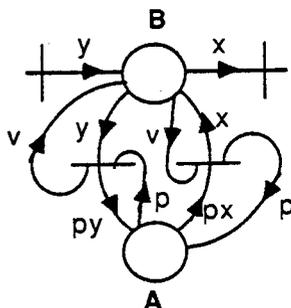


Figure II.42a

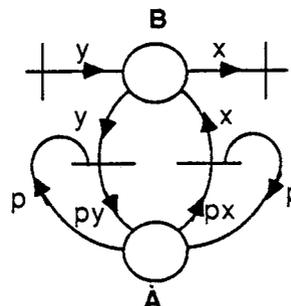


Figure II.42b

**ex. 6 :**

a) Nous allons prendre maintenant comme exemple la structure des modules standard de dérivation qui existe en général dans les machines modulaires flexibles d'assemblage et de contrôle.

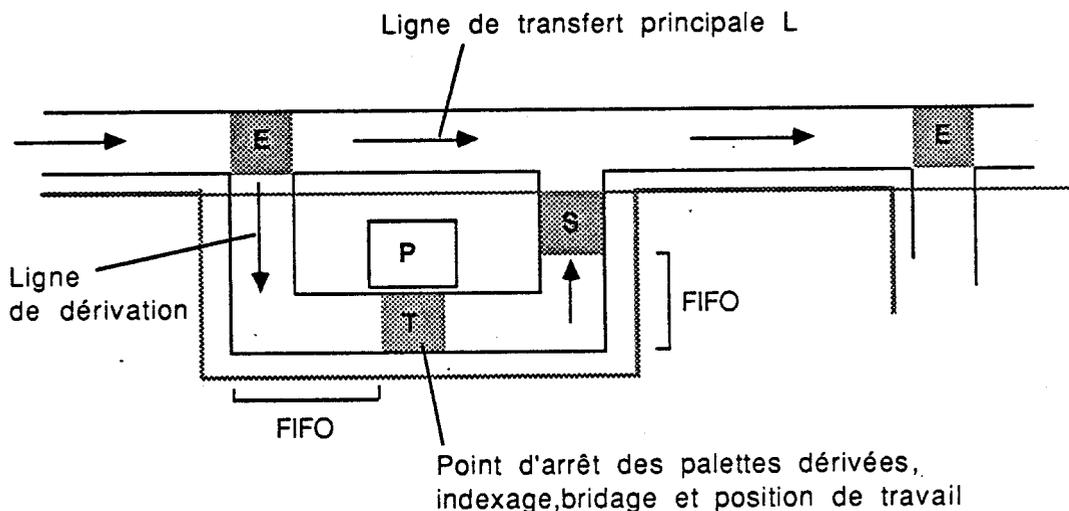


Figure II.43

En général ces modules standards comportent :

- un dispositif de sélection des palettes à transférer sur le module (dérivation d'entrée), comportant un détecteur de présence palette, un système de reconnaissance du type de pièce (plot de lecture de badge) et un vérin de dérivation d'entrée. Les palettes non sélectionnées continuent leur chemin sur la ligne de transfert principale L.

- un dispositif d'entraînement par courroies des palettes dans le module et une zone de stockage des palettes en attente de l'opération.
- un dispositif de travail comportant un détecteur de présence palette, un vérin d'arrêt de la palette, un vérin d'indexage et de bridage, et un plot de lecture-écriture permettant d'inscrire sur le badge les informations relatives aux opérations effectuées.
- un dispositif de transfert des palettes sur la ligne principale avec une zone d'attente pour le cas où cette ligne est encombrée.

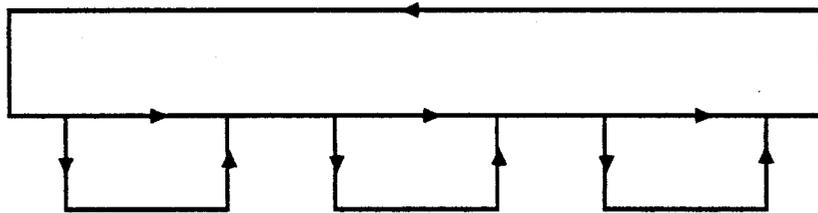


Figure II.44

Tous les modules sont dits "hors ligne" : opérant en dehors de la ligne principale de transfert des produits, ils choisissent parmi les différents produits qui passent ceux qu'ils doivent traiter.

On peut, grâce à cette architecture modulaire, modifier l'unité de production en fonction de l'évolution des produits et des quantités requises.

Supposons qu'on est en présence de cinq types de pièces palettisées  $po_1$ ,  $po_2$ ,  $po_3$ ,  $po_4$ , et  $po_5$ . Le poste T peut effectuer les opérations  $o_1$  et  $o_2$ . En cas d'encombrement du poste, les pièces  $po_1$ ,  $po_2$ ,  $po_3$  poursuivent leur chemin sur la ligne de transfert principale L.

On peut construire la représentation suivante sous forme de règles opératoires élémentaires :

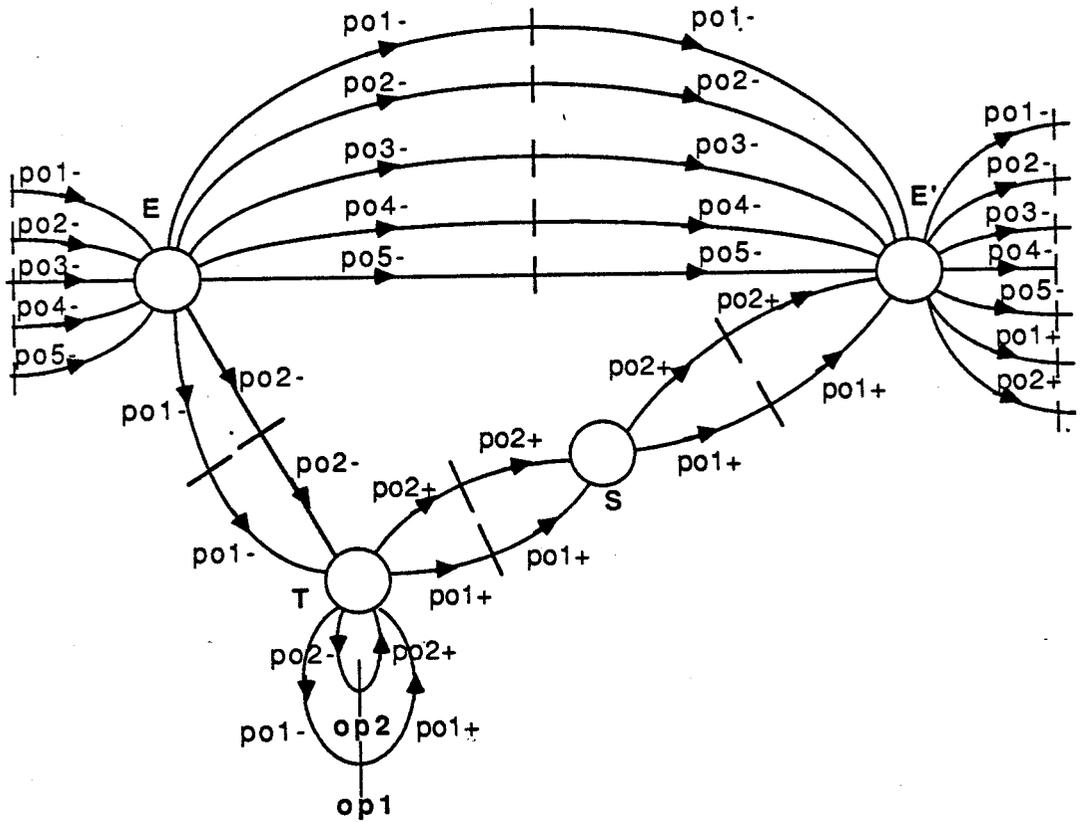


Figure II.45

- b) Si on veut donner à la ligne principale la possibilité de bloquer des pièces pour attribuer la priorité à la sortie de la dérivation, on doit faire apparaître un point d'arrêt A sur L.

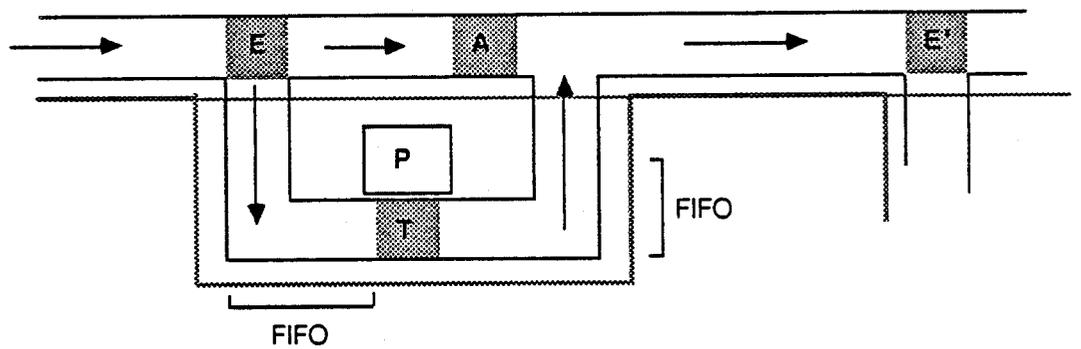


Figure II.46

- c) Il est possible également de laisser A et S pour prévoir dès le début une grande souplesse de fonctionnement et de reconfiguration de l'unité de production dans l'attribution des priorités d'accès à la zone critique.

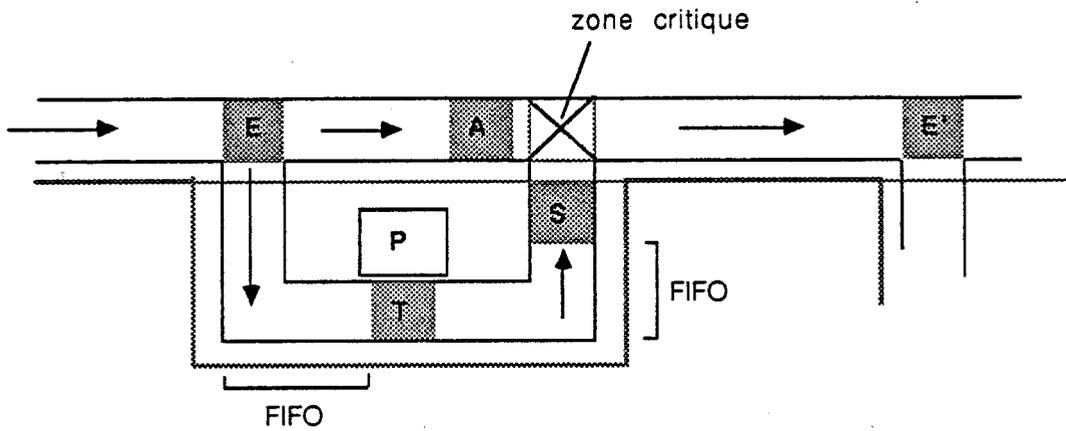


Figure II.47

**Remarque :**

Le développement ultérieur du graphe de commande fait apparaître une exclusion mutuelle systématique entre les transferts ayant pour arrivée le même tampon d'entrée d'un poste.

Sur les schémas ci-dessus on voit en outre apparaître une zone critique X liée à un croisement. Cette contrainte fera apparaître une exclusion entre les transferts vers E' venant de la ligne principale et de la dérivation. Cette contrainte ne doit pas être spécifiée au niveau de la description des actions élémentaires qui laissent la possibilité d'un parallélisme maximal. Cette contrainte qui est liée au système de transport apparaît dans la phase de développement ultérieure de la partie commande. La description des gammes n'est pas a priori liée à une architecture matérielle complètement définie.

Ainsi

La représentation de deux transferts :

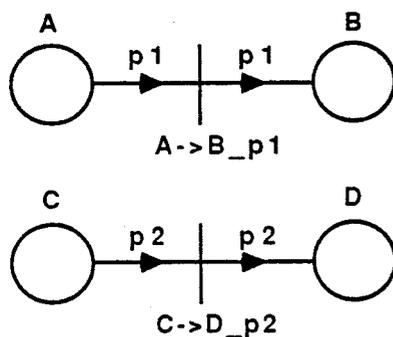


Figure II.48

pourra recouvrir des réalités matérielles différentes.

Par exemple :

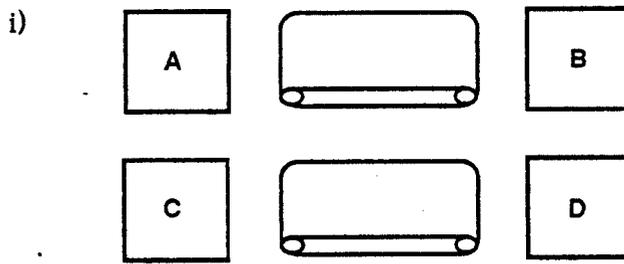


Figure II.49

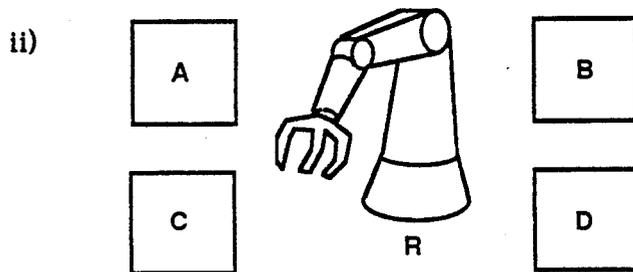


Figure II.50

iii)

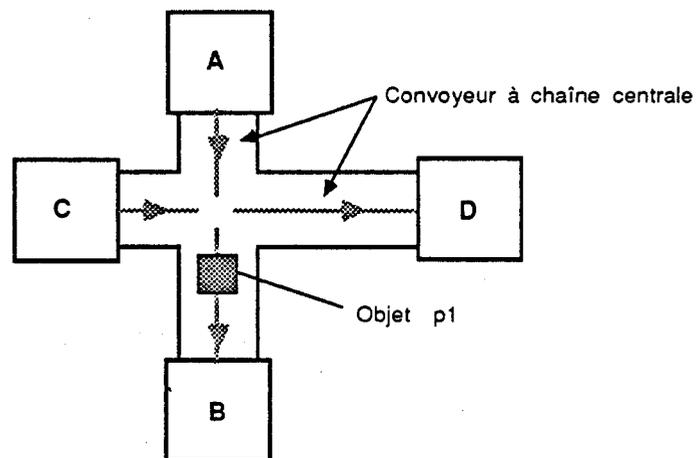


Figure II.51

ex.7 :

a) Considérons une machine M possédant un tampon d'entrée T.

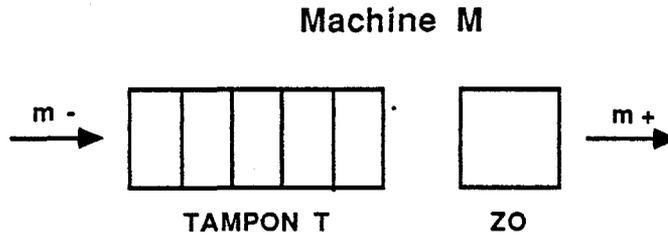


Figure II.52

Si le transfert d'une pièce "m-" du tampon T sur la zone opératoire est uniquement commandé par l'appel de la zone opératoire, la représentation de la machine sous forme de règles est la suivante :

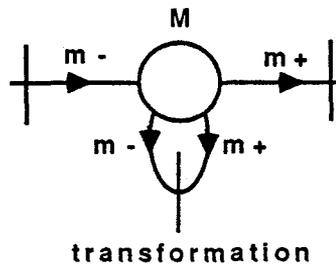


Figure II.53

Si le tampon a la possibilité de bloquer ou de dérouter vers une autre voie les pièces m- malgré un appel de la zone opératoire, on aura alors la représentation suivante :

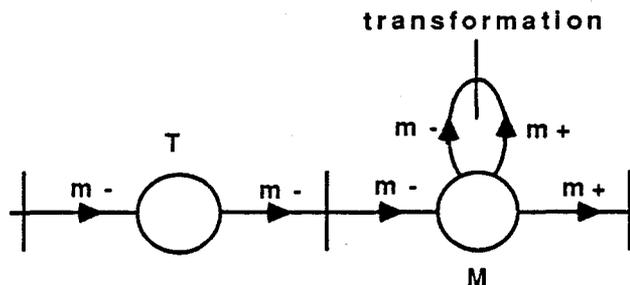


Figure II.54

- b) Considérons maintenant une machine  $M$  possédant un tampon d'entrée  $Te$  et un tampon de sortie  $Ts$ . Si les tampons  $Te$  et  $Ts$  peuvent bloquer ou orienter les pièces, c'est-à-dire s'ils sont le lieu d'une décision concernant l'évolution ultérieure de la pièce, alors on a la représentation suivante :

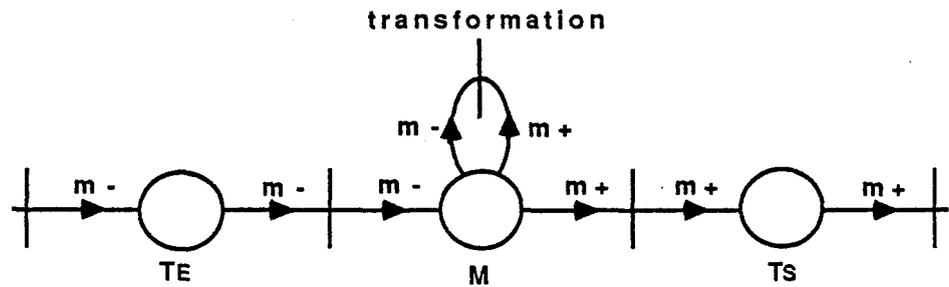


Figure II.55

La FIFO implicite associée à l'étape de structuration ultérieure de la place  $M$  est alors nulle.

## II.

### L'ANALYSE DE COHERENCE ET DE COMPLETUDE D'ENCHAÎNEMENT LOGIQUE DES REGLES OPERATOIRES

#### II.1 PRESENTATION GENERALE

Nous sommes maintenant en possession de la notion d'action élémentaire ou règle opératoire et du formalisme qui a été choisi pour sa représentation.

La première tâche dont nous proposons l'étude est la description et la décomposition en actions élémentaires de chacune des gammes de production des objets manufacturés. Pour chaque produit final réalisé par l'unité de production, l'objectif va être d'établir un lien sous forme d'un ensemble d'actions élémentaires entre un certain nombre de prédicats (présence objet lieu) de départ et d'arrivée (qu'on appelle faits initiaux et faits terminaux). Les enchaînements de ces actions vont permettre de définir clairement les cheminements des pièces nécessaires au cycle de fabrication et, éventuellement, à ses modes dégradés.

Afin de tout de suite éclaircir les idées, considérons la gamme opératoire  $t-1-m-/t+1+m+$  très simple présentée à la figure II.56.

Le rôle de cette gamme est d'obtenir à partir d'une pièce brute de type  $t-1-m-$  arrivant en ENTREE le produit fini ou semi-fini  $t+1+m+$  sur la SORTIE.

Cette pièce doit subir

- une opération de tournage  $t$  sur le poste TOUR
- une opération de lavage  $l$  sur le poste LAVAGE
- une opération de métrologie  $m$  sur le poste METROLOGIE.

Un certain nombre de sites intermédiaires  $S1, S2, S3, S4, S5, S6, S7, S8, S9$  et  $S10$  ont été également introduits.

L'ensemble de tous ces lieux peut correspondre à une architecture matérielle existante ou à des entités matérielles encore abstraites.

L'état d'avancement de la pièce est repéré par les signes "-" et "+"

0- signifie que l'opération 0 est à effectuer.

0+ signifie que l'opération 0 a été réalisée.

On fait en outre ici l'hypothèse que le transport des pièces est assuré par des supports p1 (par exemple des palettes).

La pièce t-l-m- sur son support p1 sera représentée par le symbole plt-l-m-.

L'ensemble des règles opératoires constituant cette gamme est donné plus explicitement ci-dessous :

SEQUENCE OPERATOIRE t-l-m-/t+l+m+

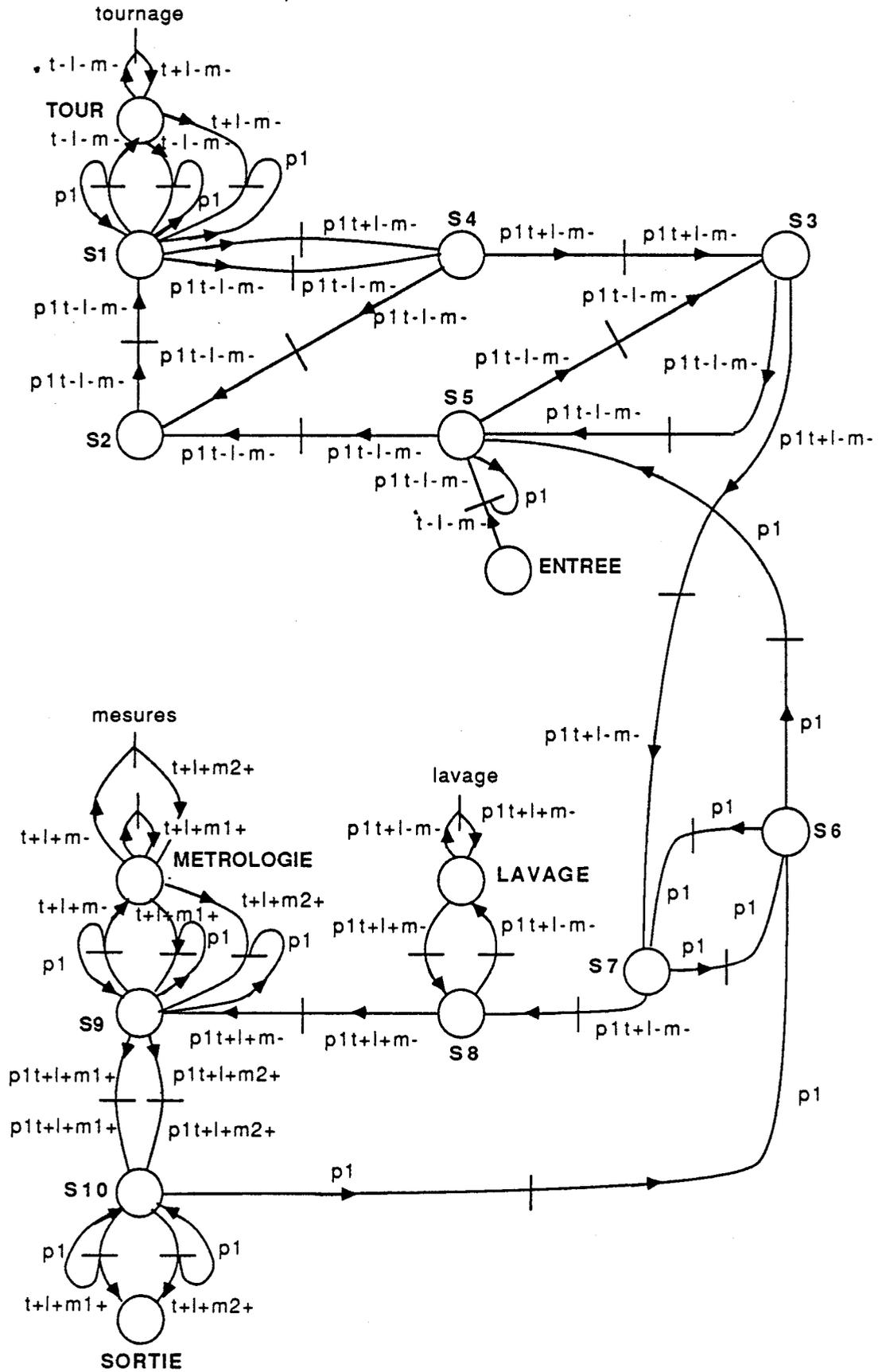


Figure II.56

règles de production en marche normale :

entrée->s5_t-l-m-	Transfert de la pièce brute t-l-m- depuis le tampon d'entrée jusqu'au poste de palettisation s5. Palettisation de t-l-m- sur une palette p1 en s5.
s5->s2_t-l-m-	Transfert simple de t-l-m- sur la palette p1 de s5 à s2.
s2->s1_t-l-m-	Transfert simple de t-l-m- sur la palette p1 de s2 à s1.
s1->tour_t-l-m-	Dépalettisation et transfert de t-l-m- depuis s1 jusqu'au tour.
tournage_t-l-m-/t+l-m-	Séquence de tournage.
tour->s1_t+l-m-	Evacuation de la pièce usinée t+l-m- du tour et repalettisation en s1.
s1->s4_t+l-m-	Transfert simple de la pièce t+l-m- sur une palette p1 de s1 vers s4.
s4->s3_t+l-m-	Transfert simple de la pièce t+l-m- sur une palette p1 de s4 vers s3.
s3->s7_t+l-m-	Transfert simple de la pièce t+l-m- sur une palette p1 de s3 vers s7.
s7->s8_t+l-m-	Transfert simple de la pièce t+l-m- sur une palette p1 de s7 vers s8.
s8->lavage_t+l-m-	Transfert simple de l'ensemble pièce t+l-m-/palette p1 vers le poste de lavage.
lavage_t+l-m-/t+l+m+	Opérations de lavage et nettoyage de la pièce t+l-m- sur sa palette p1.
lavage->s8_t+l+m-	Evacuation de l'ensemble pièce lavée t+l+m-/palette p1 vers s8.
s8->s9_t+l+m-	Transfert simple de l'ensemble p1t+l+m- de s8 vers s9.
s9->métrieologie_t+l+m-	Dépalettisation et transfert de la pièce usinée et lavée t+l+m- vers le poste de métrieologie.
métrieologie_t+l+m-/t+l+m1+ métrieologie_t+l+m-/t+l+m2+	Séquence d'opérations de mesures. Le résultat des mesures permet de classer la pièce dans 2 intervalles distincts qu'on représente par t+l+m1+ et t+l+m2+.
métrieologie->s9_t+l+m1+ métrieologie->s9_t+l+m2+	Evacuation du poste de métrieologie et repalettisation en s9.
s9->s10_t+l+m1+ s9->s10_t+l+m2+	Transfert simple de p1t+l+m1+ ou p1t+l+m2+ de s9 vers s10. En situation réelle, la classification de t+l+m+ en 2 intervalles pourrait également donner lieu à des trajets ultérieurs différents.
s10->sortie_t+l+m1+ s10->sortie_t+l+m2+	Dépalettisation de la pièce manufacturée t+l+m1+ ou t+l+m2+ en s10 et transfert simple de celle-ci sur la sortie.

s10->s6_p1	Evacuation et recyclage de la palette p1 vers s6.
s6->s7_p1 s7->s6_p1	Carrousel d'attente pour les palettes.
s6->s5_p1	Transfert d'une palette p1 de s6 vers le poste de palettisation s5.

règles de production pour dérivations de modes éventuellement dégradés du procédé :

tour->s1_t-l-m-	Evacuation d'une pièce non usinée t-l-m- du tour en cas de panne de ce dernier.
-----------------	---

s1->s4_t-l-m- s4->s2_t-l-m- s5->s3_t-l-m- s3->s5_t-l-m-	Chemins d'attente à différents niveaux en cas de panne ,par exemple du tour ou du poste de palettisation s1.
--	--

La rigueur de la description est très importante. Ce travail est la phase préparatoire d'une chaîne de conception complexe. Aussi il est capital d'assurer des propriétés de cohérence et de complétude d'enchaînement des actions élémentaires tout en permettant à l'opérateur une évolution progressive de sa description. La gamme opératoire représentée ici est relativement simple, dans un but d'illustration. Elle ne comporte qu'un seul usinage t. L'étude de cas réels est souvent plus complexe. L'expérience a prouvé que l'obtention à la main de telles descriptions pour chacune des gammes devenait rapidement chargée d'erreurs et d'oublis. Ces erreurs étant repercutées sur la construction de la partie commande et sur la simulation, il devenait de plus en plus délicat d'identifier les causes d'erreur au fur et à mesure de la progression dans la chaîne de conception.

C'est pour cette raison que nous avons proposé une assistance informatique de la phase de description de chacune des gammes opératoires sous forme d'actions élémentaires par un logiciel appelé "Analyse de cohérence et de complétude" écrit en langage Le-Lisp. Sa taille est d'environ 120 KOctets.

Le rôle de ce logiciel est de permettre à l'utilisateur d'obtenir une description correcte des gammes opératoires sous la forme formalisée et simple des règles de production.

De façon résumée on peut dire de ce logiciel :

- i) qu'il assiste le raisonnement de l'opérateur dans la construction étape après étape des règles de production décrivant les gammes opératoires de l'unité de production.
- ii) qu'il permet d'éviter les oublis et les lacunes de description en faisant le bilan des règles et des faits initiaux effectivement utilisés, des faits terminaux effectivement obtenus et des voies d'investigations incomplètes.
- iii) qu'il assure le bon enchaînement des règles de production et le lien entre un ensemble de faits initiaux et de faits terminaux eux-mêmes éventuellement déterminés progressivement.

Ce logiciel utilise un moteur d'inférence hybride activant des phases mutuellement coopérantes de chaînage avant et de chaînage arrière. Il est

destiné à clarifier tous les enchaînements logiques et toutes les carences de description telles que

- i) l'omission de règles, de faits initiaux ou terminaux.
- ii) la présence de règles, de faits initiaux ou terminaux inutiles.
- iii) l'investigation incomplète de voies de chaînage laissées volontairement en suspens.

Le but est d'assurer à la description des propriétés de cohérence et de complétude d'enchaînement. Ainsi, l'utilisateur est interrogé par le logiciel jusqu'à ce qu'une définition rigoureuse et satisfaisante des gammes opératoires soit donnée. La tâche est terminée quand tous les cheminements élémentaires sont précisés en respectant les choix de l'utilisateur et les conditions de cohérence et de complétude.

Dans la suite, nous allons présenter concrètement à partir d'une série de schémas la façon dont fonctionne l'étape d'analyse de cohérence et de complétude. Le plan du moteur d'inférence sera également présenté. Mais, afin de donner tout de suite une idée plus claire des concepts sur lesquels repose le logiciel, donnons quelques explications préliminaires simplifiées.

A partir d'un ensemble de faits de départ (ou faits initiaux) fixés arbitrairement (en général la présence de produits bruts sur un tampon d'entrée et du système de transport dans un certain état d'attente - ex. : palettes -), et à partir d'un ensemble de règles opératoires décrivant chacune des actions élémentaires, le moteur d'inférence du logiciel engage une exploration en chaînage avant et en profondeur d'abord (justifiée par le fait que l'on explore ici des cheminements de types d'objets). Les règles déclenchables sont ainsi chaînées progressivement par réécriture de la base de faits courante qui évolue avec le tir progressif de ces règles. Si aucune règle ne peut être déclenchée lors du pas d'inférence courant, alors le logiciel va s'intéresser aux règles candidates, dont une prémisse au moins se trouve dans la base de faits courante. Une phase de chaînage arrière va alors éventuellement permettre de relier les prémisses manquantes à des faits initiaux ou va permettre de reboucler sur des règles déjà examinées (c'est le cas, par exemple, des boucles de recyclage de matériels tels que des palettes, des platines d'usinage). Sinon, l'analyse va amener l'utilisateur, s'il le souhaite, à définir de nouveaux faits initiaux, ou de nouvelles règles, en chaînage arrière, de manière à prouver les prémisses manquantes, de façon continue et interactive. Les faits à partir desquels les prémisses manquantes

vont pouvoir être validées vont ainsi servir de départ à une phase de redescente en chaînage avant qui va permettre, entre autres, de déclencher la règle candidate dont on a pu rendre atteignables les prémisses manquantes. Si, lors du pas d'inférence courant, on constate l'absence de règles déclenchables et de règles candidates, alors l'utilisateur va être interrogé de façon à ajouter de nouvelles règles ou éventuellement préciser de nouveaux faits terminaux parmi les faits qui viennent juste d'être atteints. Le logiciel détecte la présence de faits qui ont été déclarés comme faits terminaux (c'est le cas, en général, de la présence d'un produit terminé sur un tampon de sortie). Il détecte également l'apparition d'une situation déjà rencontrée et complètement explorée (cas de chemins qui convergent ou qui bouclent). Lorsque le moteur d'inférences ne peut réaliser d'enchaînements ni questionner l'utilisateur à partir de la base de faits courante, des retours-arrière progressifs sont alors réalisés pour revenir aux différentes phases de l'exploration précédemment enchaînées. Les règles qui ont été précédemment tirées à partir de la situation courante sont récapitulées. De nouvelles inférences sont alors lancées avec les règles déclenchables ou candidates qui n'ont pas encore été regardées à ce niveau.

Lorsque, pour la base de faits courante, il ne subsiste plus aucune règle déclenchable ni aucune règle candidate, le logiciel interroge l'utilisateur sur la possibilité de créer, en partant de la base de faits courante, d'autres chemins avec de nouvelles règles. La flexibilité de circulation des pièces est accrue par le nombre de chemins d'évolution possibles pour chaque fait de la base de faits courante, c'est-à-dire par le nombre d'indéterminismes directionnels qui ont été engendrés.

L'utilisateur peut interrompre l'investigation d'un chemin s'il ne sait pas encore comment la poursuivre. La présence d'extrémités non terminales de l'arborescence fait que la gamme opératoire sera mémorisée comme gamme incomplète.

Lorsque la phase de retour-arrière est terminée, la plus grosse partie de l'analyse de cohérence et de complétude est réalisée. Pour finir l'analyse, le logiciel regarde la liste des règles, des faits initiaux et terminaux qui n'ont pas été utilisés par la première étape de l'analyse. L'analyseur va alors tenter de les relier au reste de la description en lançant un chaînage arrière sur les prémisses de chacune des règles encore inutilisées. Les règles et les faits qui n'auront pas été rattachés, à la suite de cette dernière étape, seront mémorisés comme information en attente mise de côté, ou simplement éliminés.

En résumé, disons que l'analyse de cohérence et de complétude garantit la cohérence et la complétude d'enchaînement logique des actions élémentaires qui constituent la description de chacune des gammes opératoires.

Afin de rendre les explications moins abstraites, nous allons présenter tout de suite un exemple d'utilisation avant de donner le plan détaillé du moteur d'inférences.

## II.2 EXEMPLE

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

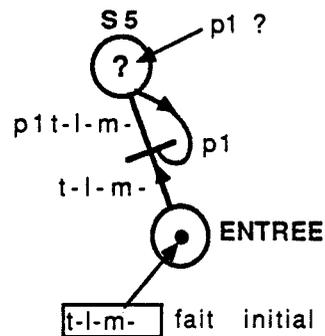


figure II.57 n°1

Point de départ :

On dispose d'une règle : la règle entrée->s5\_t-l-m- et d'un fait initial : (présence t-l-m- entrée).

Il n'y a pas de règles déclenchantes.

Il y a une règle candidate : entrée->s5 qui a pour prémisse manquante : (présence p1 s5).

L'analyse part en chaînage arrière et interroge l'utilisateur :

"La prémisse (présence p1 s5) doit-elle être considérée comme un fait initial de la gamme?"

## SEQUENCE OPERATOIRE t-l-m-/t+l+m+

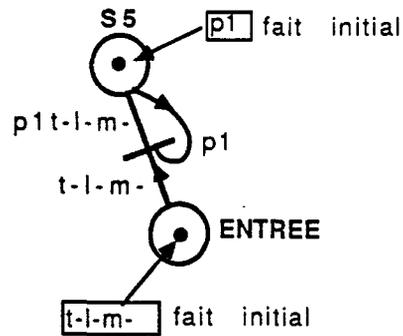


figure II.57 n°2

Il y a 3 possibilités de réponse à la question de la figure 1 :

i) Si la réponse est positive, le fait (présence p1 s5) est mémorisé comme fait initial de la gamme. L'analyse reprend son exploration en chaînage avant à partir des faits de redescente (présence t-l-m- entrée) (présence p1 s5). C'est ce que nous allons examiner dans les figures 3 à 23.

ii) Si la réponse est négative, l'analyse continue son investigation en chaînage arrière. Ceci est présenté dans les figures 24 à 38.

iii) L'utilisateur est indécis. L'analyse cesse son exploration sur cette règle et part en retour-arrière.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

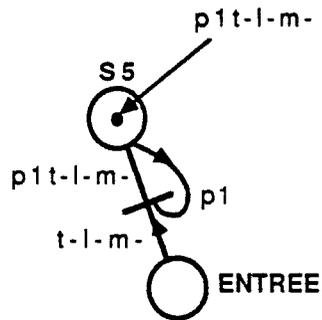


figure II.57 n°3

L'exploration des chainages logiques est lancée en chainage avant.  
La règle entrée->s5 t-l-m- étant déclenchable, elle est tirée.  
Cette première inférence conduit au fait (présence p1t-l-m- s5).

## SEQUENCE OPERATOIRE t-l-m-/t+l+m+

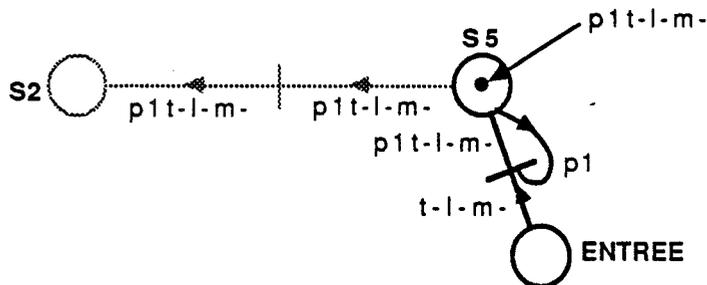


figure II.57 n°4

La base de faits courante est constituée du seul fait (présence p1t-l-m- s5). Le logiciel regarde si ce fait appartient à l'ensemble des buts, ce qui n'est pas le cas.

L'analyse demande alors à l'utilisateur si le fait (présence p1t-l-m- s5) doit être déclaré comme fait terminal (but) :

i) Si l'utilisateur répond oui, l'analyse part en retour-arrière. Nous reverrons des cas analogues aux figures 10 et 17.

ii) Si l'utilisateur répond non, le logiciel demande à l'utilisateur s'il veut compléter sa description en élaborant de nouvelles règles opératoires. C'est le cas ici. L'utilisateur construit la règle s5->s2\_t-l-m-.

iii) L'utilisateur est indécis. L'analyse part en retour-arrière.

Le fait (présence p1t-l-m- s5) est pour le moment mémorisé comme feuille non terminale de l'arborescence d'exploration. La voie de recherche est inachevée pour l'instant. Si, ultérieurement, une règle est tirée à partir de ce fait, ce dernier ne sera plus mémorisé comme feuille non terminale. Si, en fin d'analyse, des feuilles non terminales sont encore subsistantes, la gamme est mémorisée comme étant incomplète.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

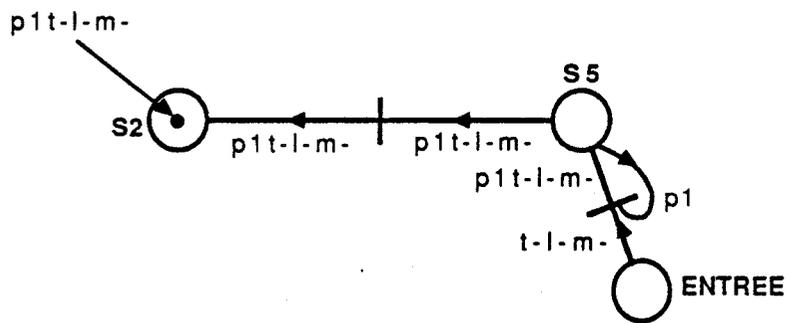


figure II.57 n°5

La règle  $s5 \rightarrow s2_{t-l-m-}$  est déclenchable.

L'inférence est effectuée. Elle conduit au fait (présence p1t-l-m- s2).

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

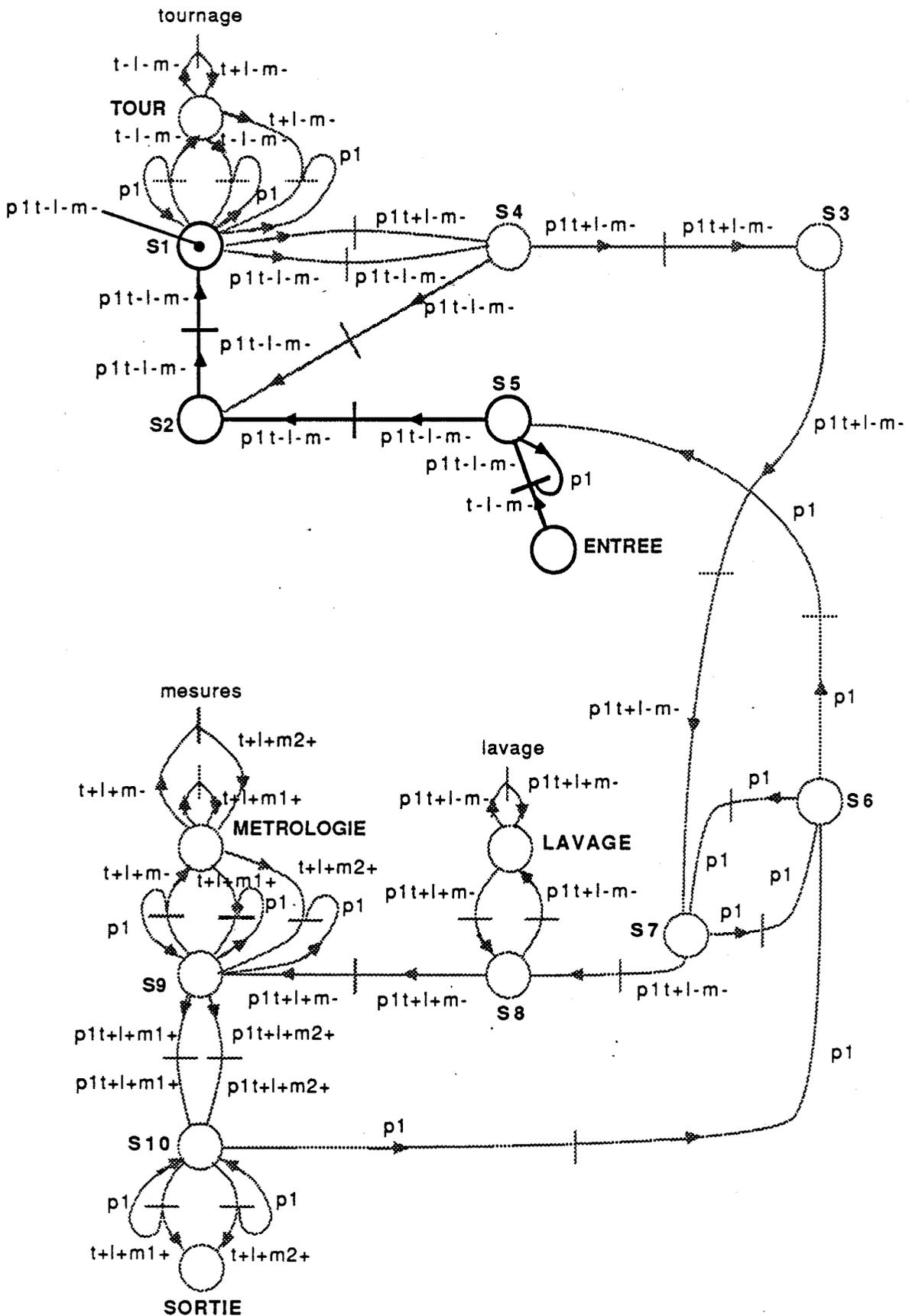


figure II.57 n°6

On suppose maintenant que les règles ne sont plus introduites une à une, mais qu'un ensemble de règles (ici en grisé fin) ont déjà été saisies à l'avance par l'utilisateur. Les inférences se poursuivent en chaînage avant sans intervention de l'opérateur. Seuls des messages précisant l'évolution des inférences lui sont communiqués.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

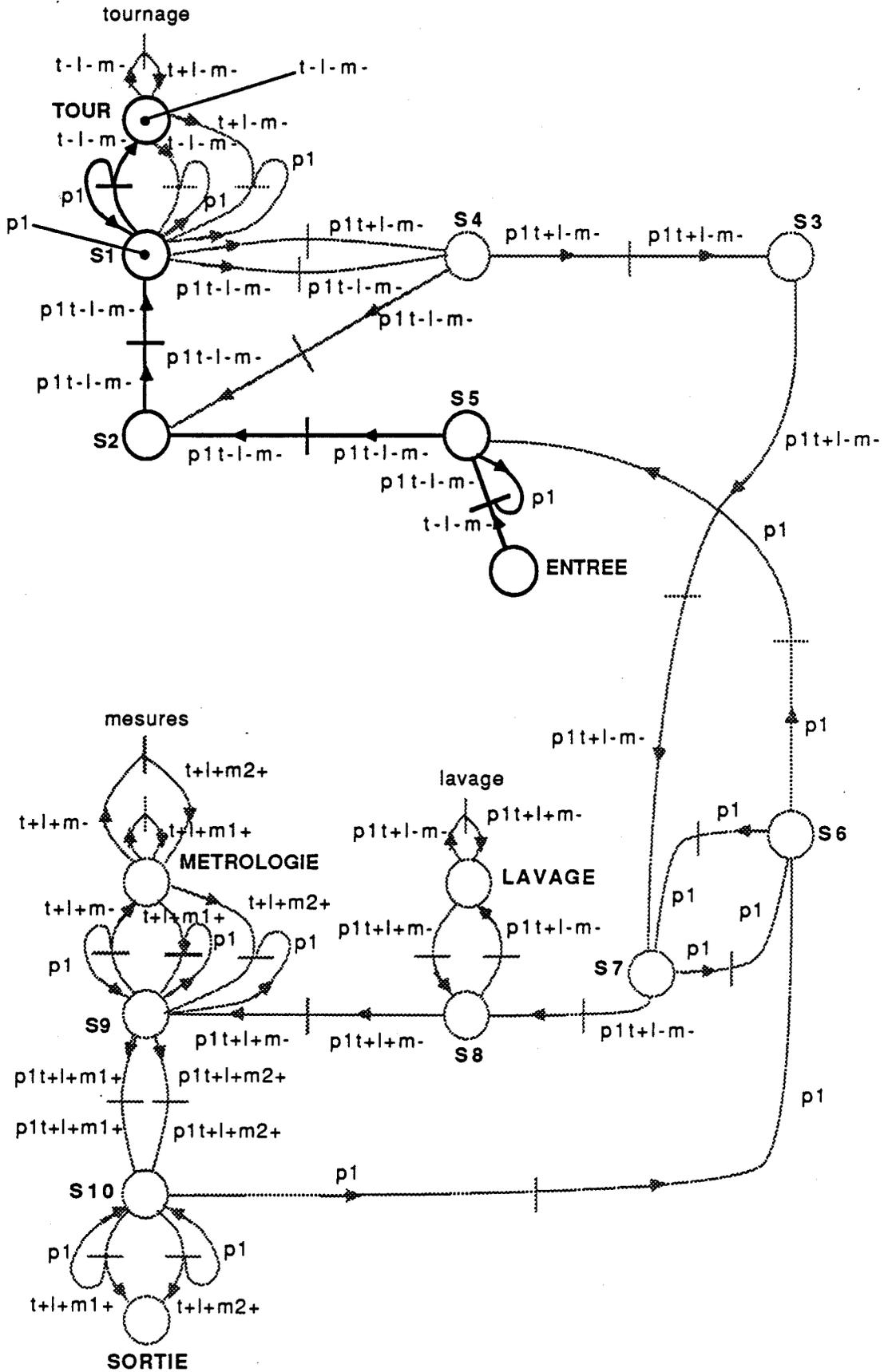


figure II.57 n°7

Les inférences en chaînage avant se poursuivent. Le fait (présence  $p1t-l-m- s1$ ) donne lieu à un indéterminisme directionnel. Deux règles sont en effet déclençables :  $s1 \rightarrow \text{tour } t-l-m-$  et  $s1 \rightarrow s4 \ t-l-m-$ . Ce point où un choix doit être fait constitue un point de retour-arrière. La règle  $s1 \rightarrow \text{tour } t-l-m-$  est tirée. La règle  $s1 \rightarrow s4 \ t-l-m-$  est laissée en attente de retour-arrière.



# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

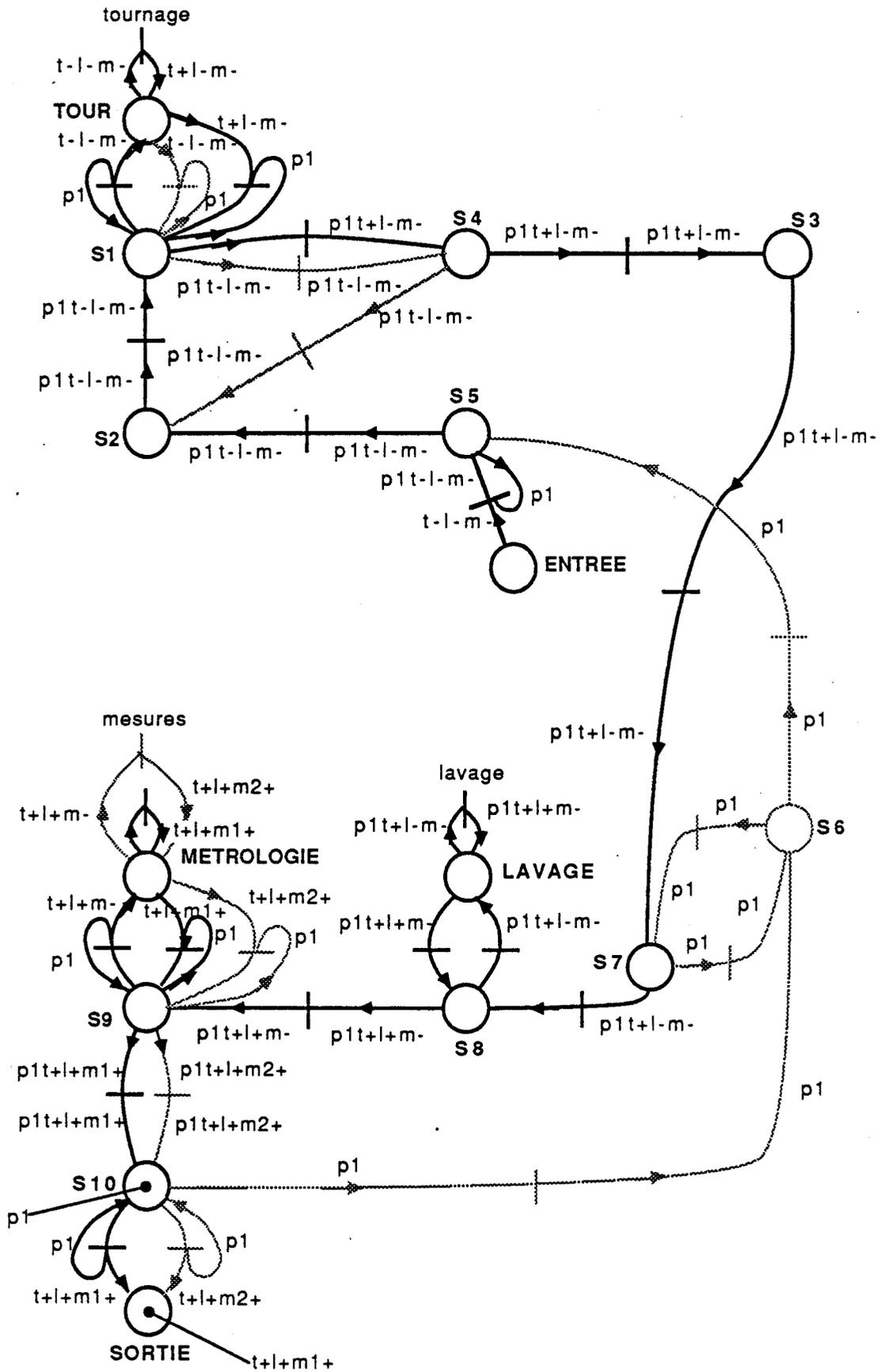


figure II.57 n°9

Le poste de métrologie pouvant classer la pièce  $t+l+m+$  dans 2 intervalles  $t+l+m1+$  et  $t+l+m2+$ , le fait (présence  $t+l+m-$  métrologie) est un point de choix et donc de retour-arrière. Il y a ici encore un indéterminisme directionnel qui est détecté. La règle  $mesures_{t+l+m-}/t+l+m1+$  est déclenchée. La règle  $mesures_{t+l+m-}/t+l+m2+$  est laissée en attente.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

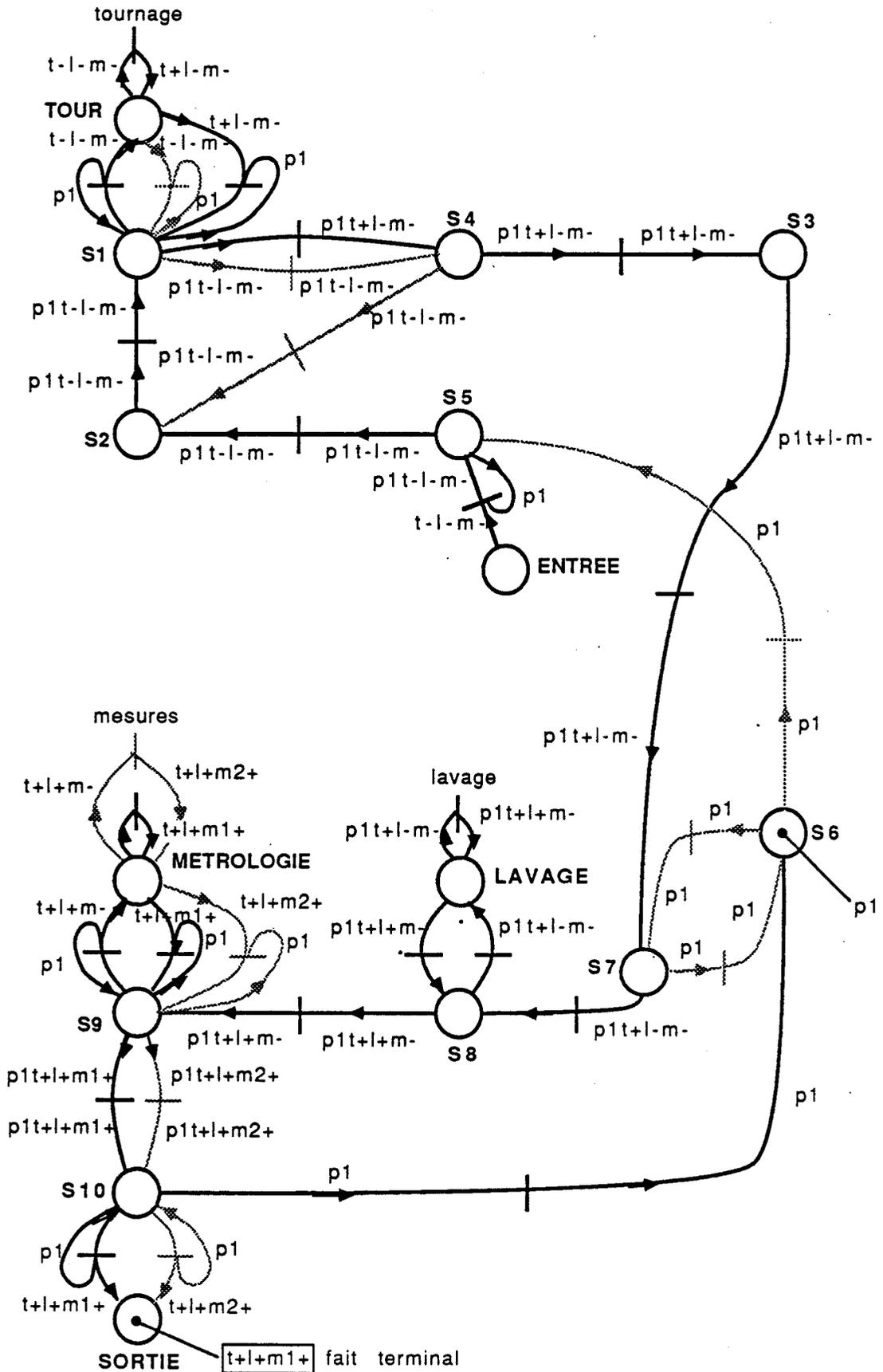


figure II.57 n°10

L'inférence s10->sortie\_t+l+m1+ a conduit au fait (présence t+l+m1+ sortie). Il est possible à ce moment de déclarer ce fait comme terminal. Si ce fait était déjà déclaré comme but, l'analyse le détecte. Ce fait est enlevé de la base de fait courante pour les évolutions aval. L'inférence s10->s6\_p1 est déclenchée à partir du fait non terminal (présence p1 s10).

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

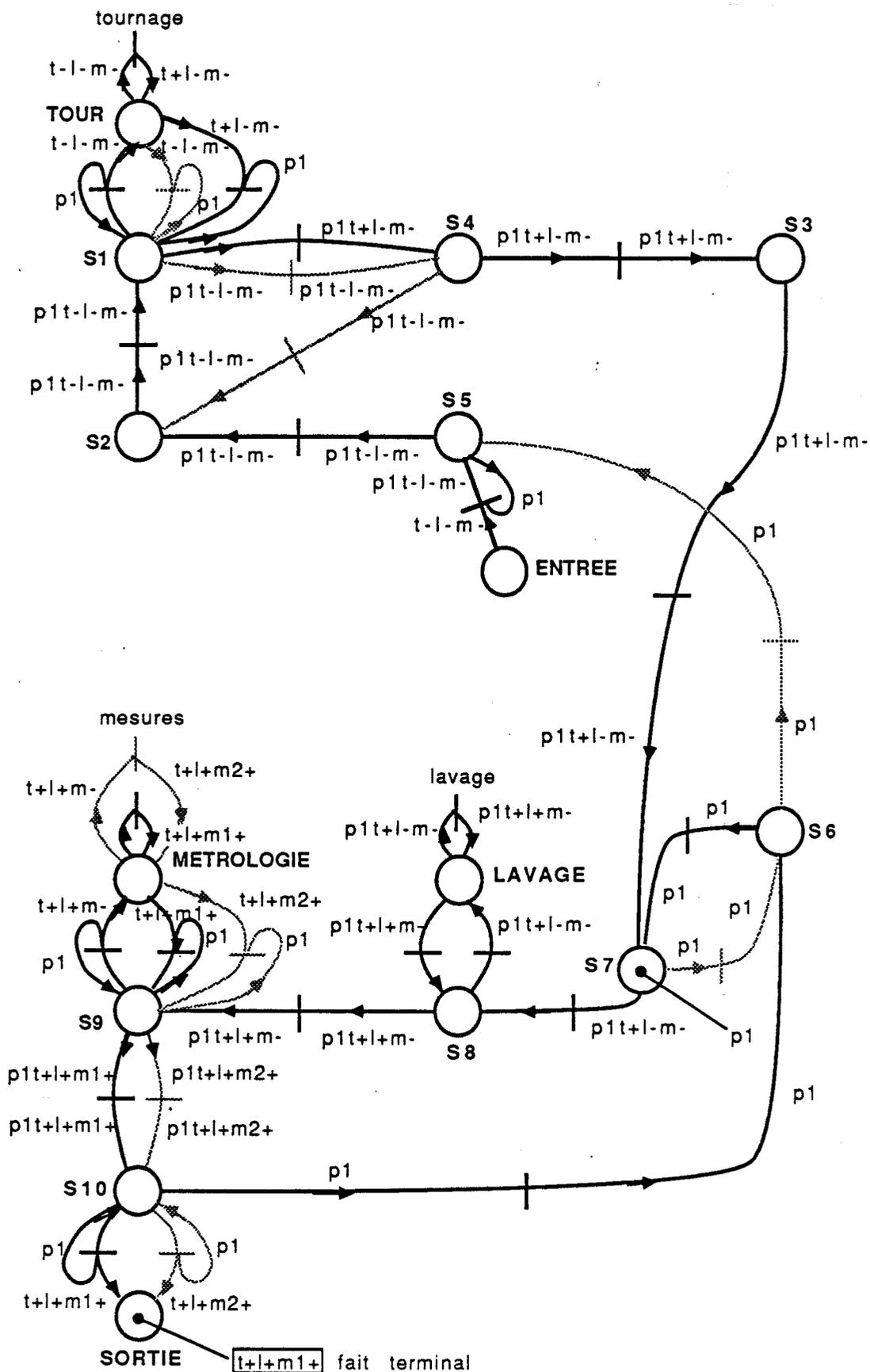


figure II.57 n°11

Le fait (présence  $p1$  s6) est un point d'indéterminisme et donc de retour-arrière .L'inférence  $s6 \rightarrow s7_{p1}$  est réalisée en premier. La règle  $s6 \rightarrow s5_{p1}$ , également déclenchable, est laissée en attente.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

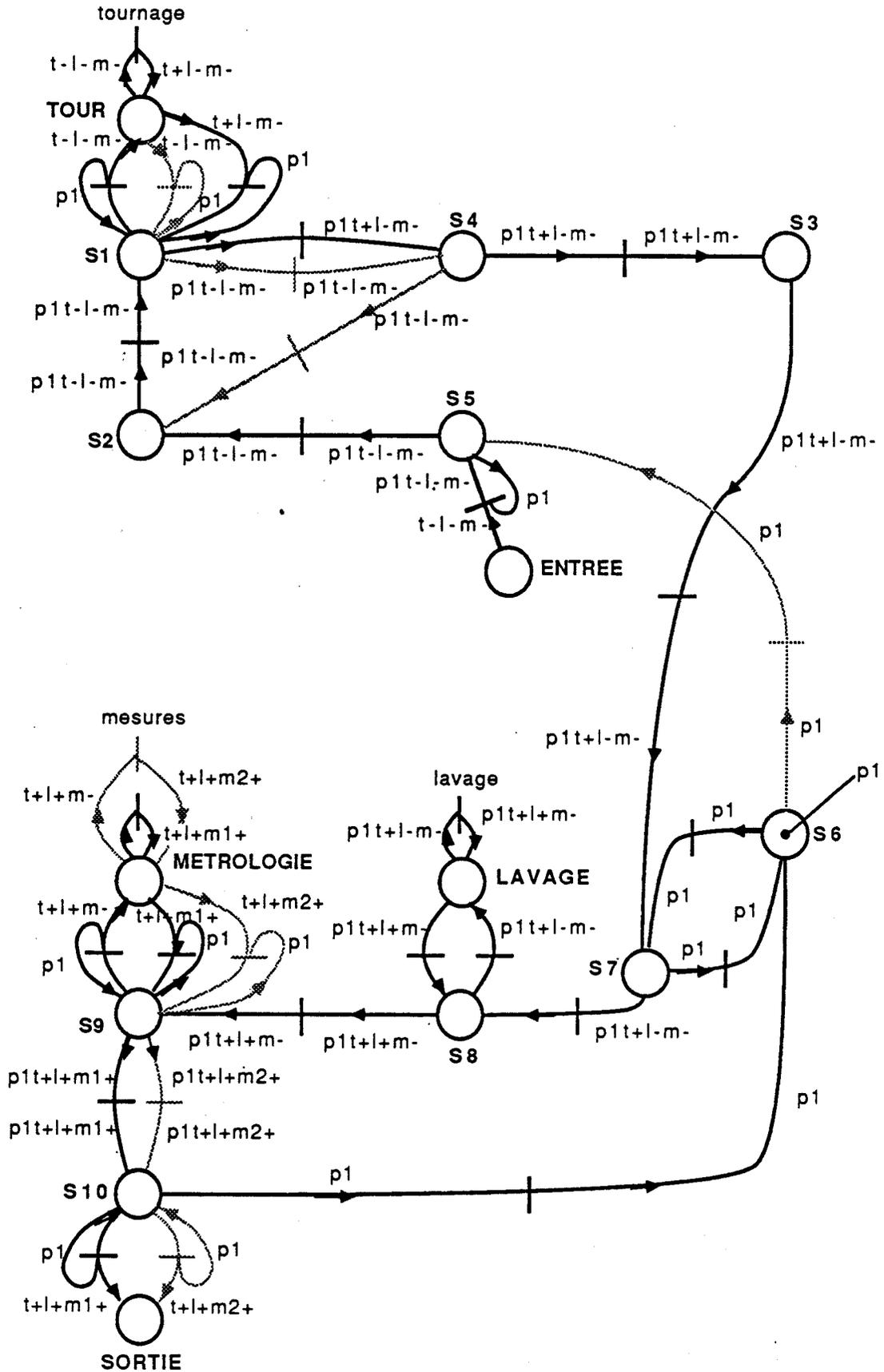


figure II.57 n°12

La base de faits courante est constituée du fait (présence p1 s6) déjà rencontré. La règle s6->s5\_p1 qui était en attente de retour-arrière peut toutefois être déclenchée maintenant à ce niveau (Ce choix de fonctionnement est arbitraire. Les règles candidates, par contre, ne peuvent être regardées qu'au premier niveau où elles sont détectées. Ce choix, justifié par l'expérience, peut facilement être modifié).

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

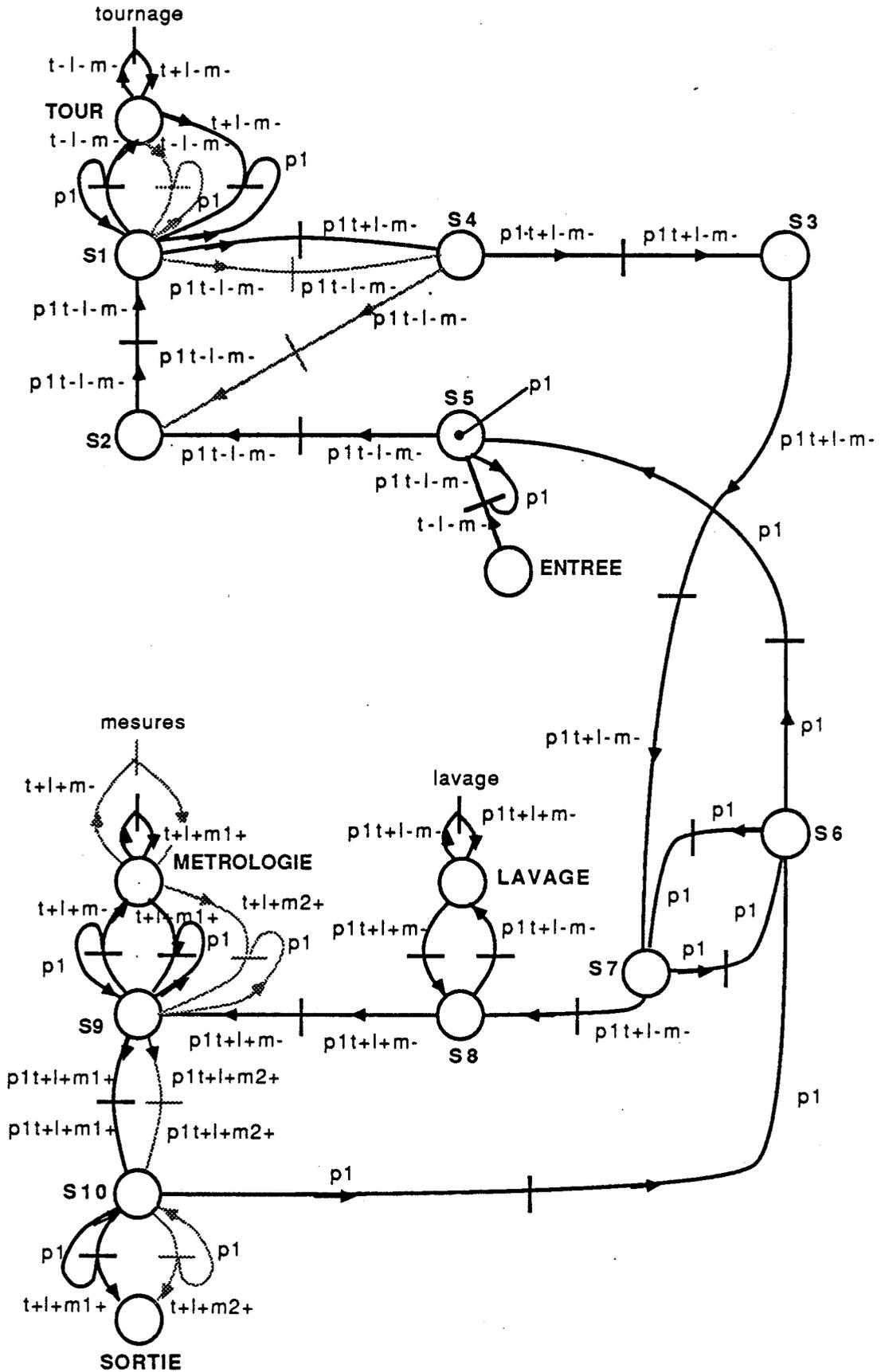


figure II.57 n°13

La base de faits courante est constituée du fait (présence p1 s5) déjà rencontré. Toutes les règles qui ont ce fait dans leurs prémisses ont déjà été utilisées (elles ont été parcourues en chaînage avant). Le processus d'analyse retombe sur une situation dont les règles d'évolution existantes ont toutes été explorées. L'analyse part en retour-arrière.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

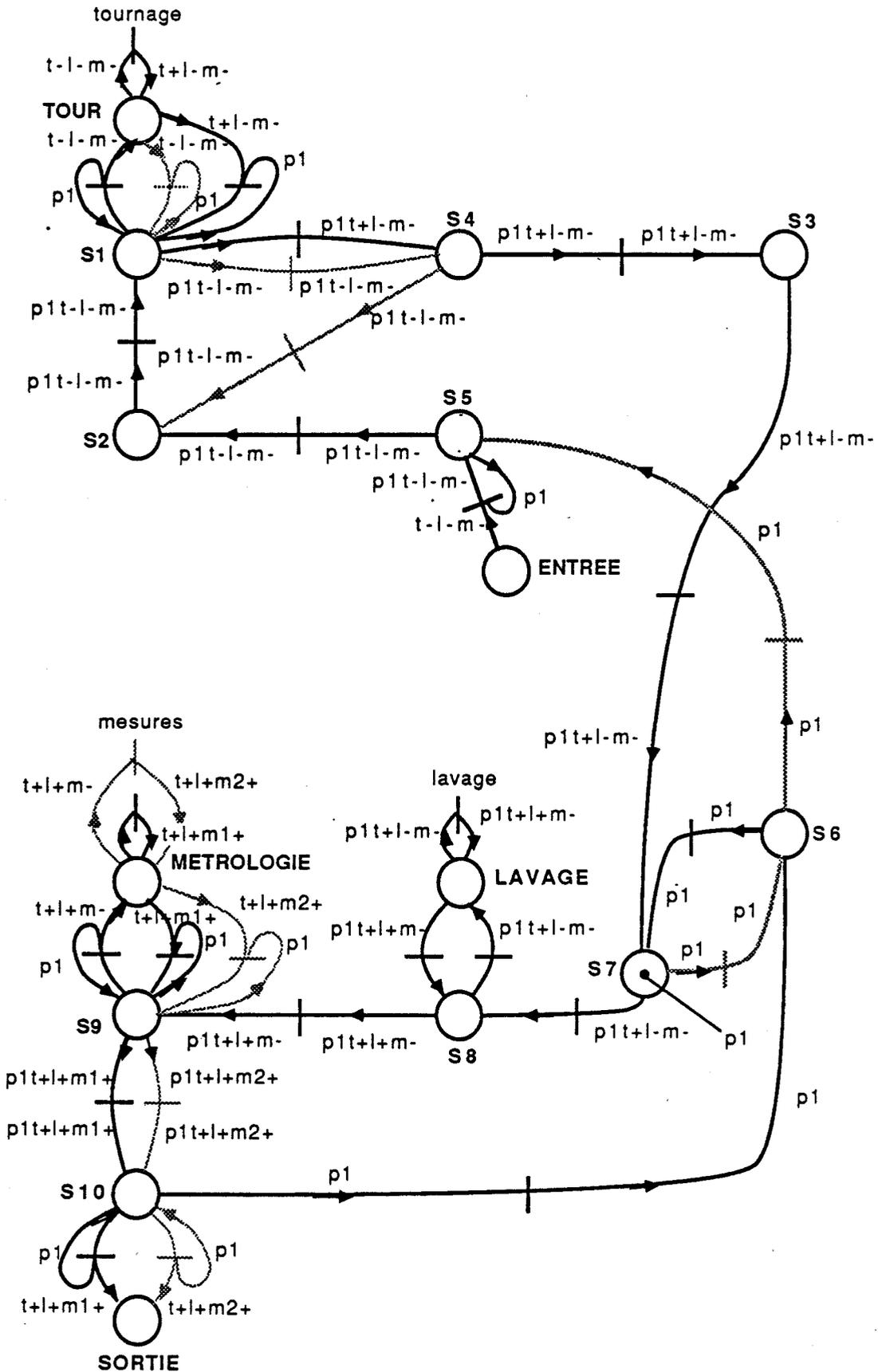


figure II.57 n°14

L'analyse effectuée des retours-arrière qui peuvent être de 3 types:  
 -retours-arrière pas à pas  
 -retours-arrière de point d'indéterminisme en attente en point d'indéterminisme en attente.  
 -retours-arrière identiques au précédent incluant des points intermédiaires à la demande.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

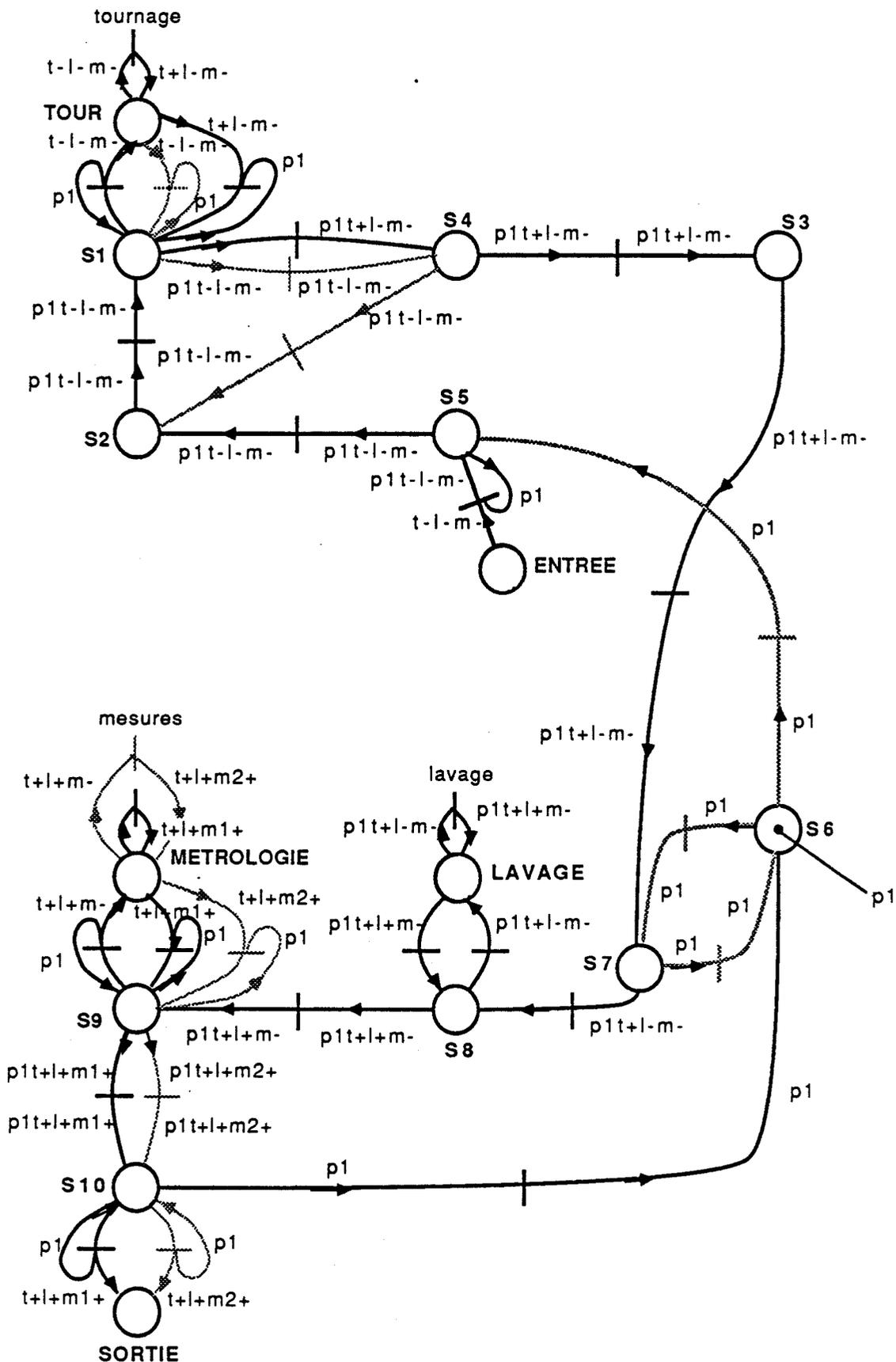


figure II.57 n°15

Le retour-arrière a conduit ici à la situation { (présence p1 s6)}. Le logiciel précise que les règles s6->s7\_p1 et s6->s5\_p1 ont déjà été déclenchées à partir de cette base de faits. Il n'y a plus de règles déclenchables ni de règles candidates. Le logiciel demande à l'utilisateur s'il veut construire de nouvelles règles à partir de cette base de faits afin de lancer ici de nouvelles voies d'exploration en profondeur.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

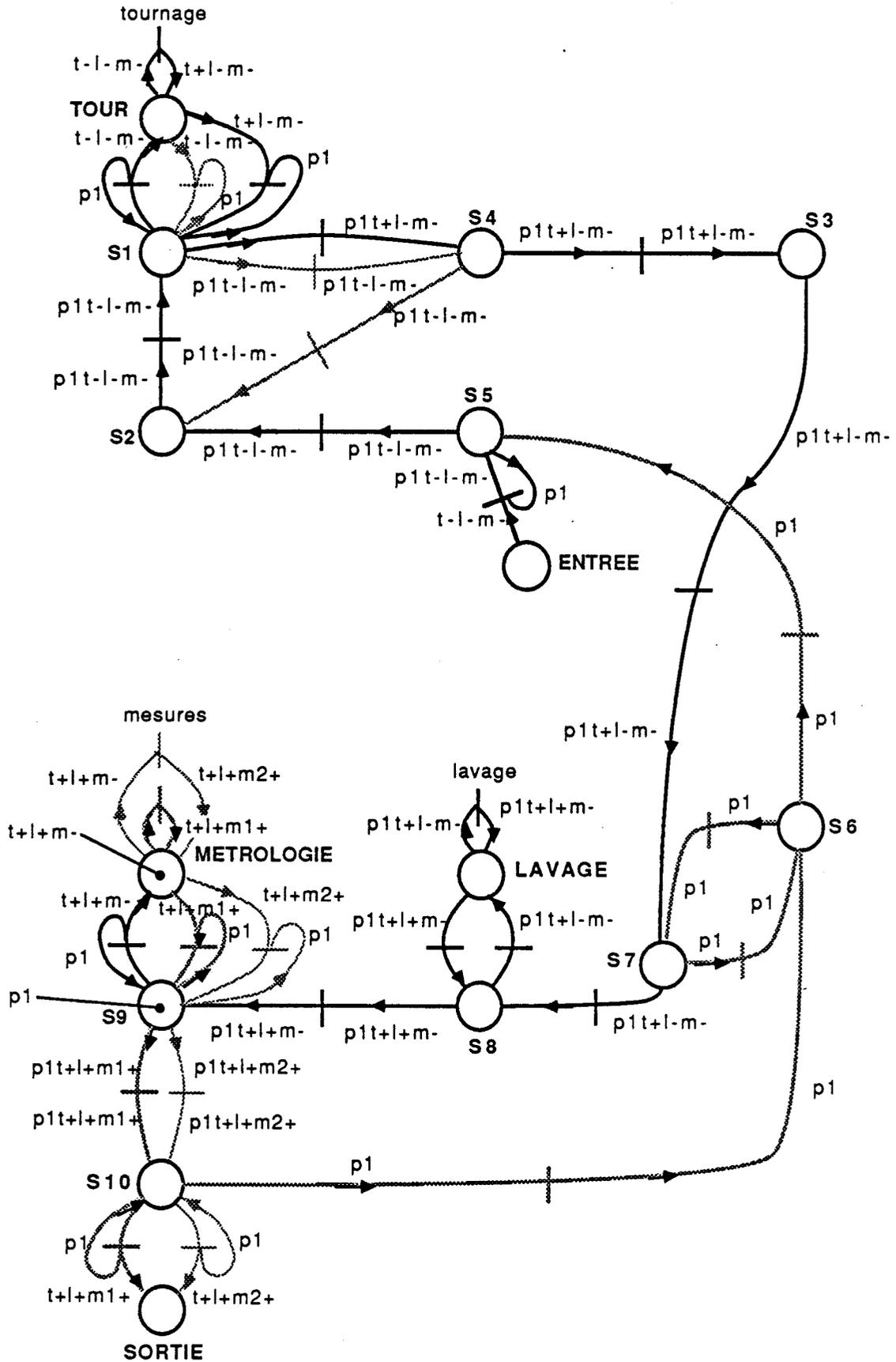


figure II.57 n°16

Le retour-arrière conduit ici à la situation {(présence t+l+m- métrologie) (présence p1 s9)}.

Le logiciel précise que la règle mesures\_t+l+m-/t+l+m1+ a déjà été tirée à partir de cette base de faits.

La règle mesures\_t+l+m-/t+l+m2+ est déclenchable, en attente. Elle va être déclenchée ici.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

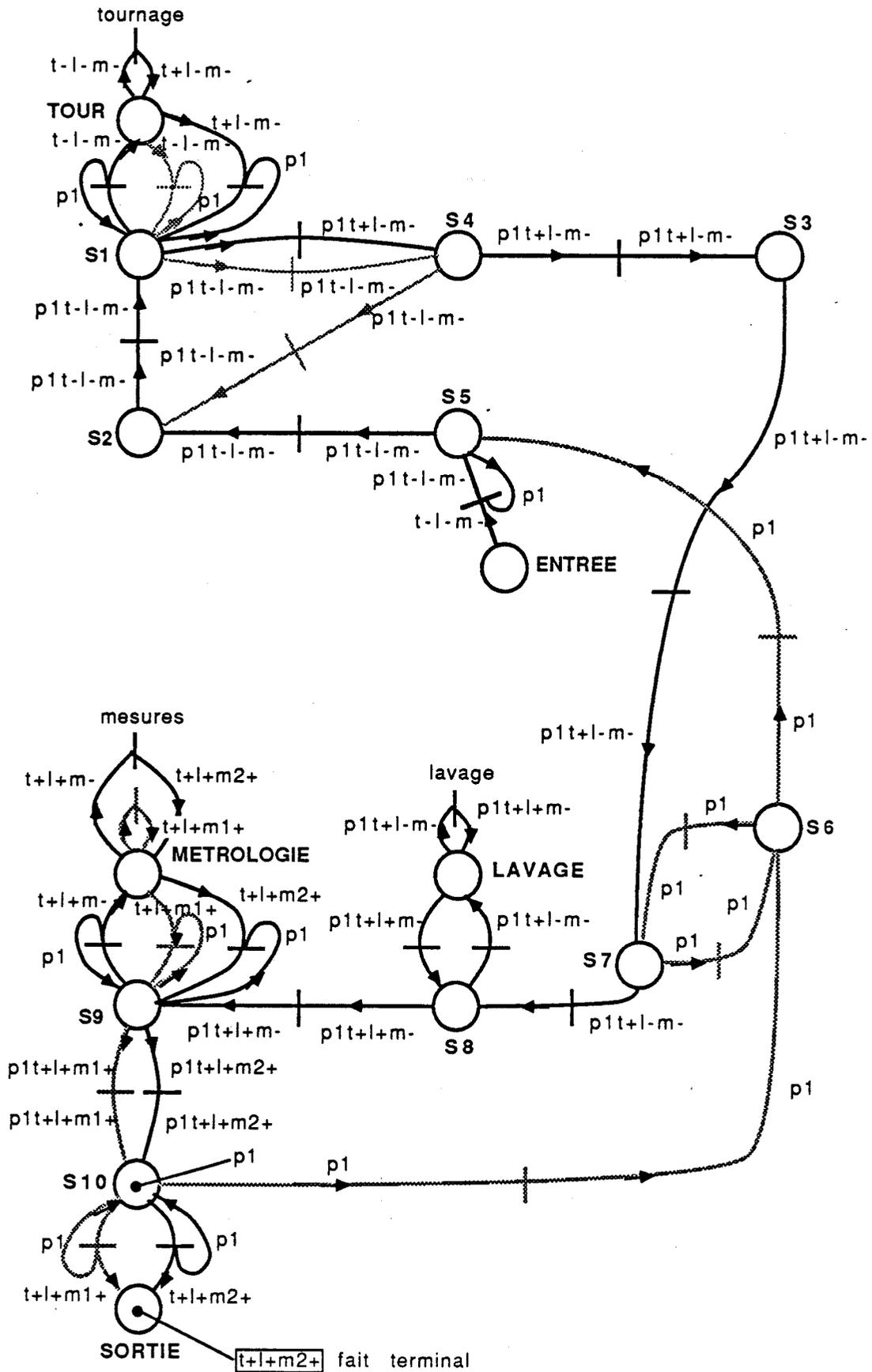


figure II.57 n°17

La règle mesures  $t+l+m-/t+l+m2+$  est inférée.

Le chemin constitué des règles mesures  $t+l+m-/t+l+m2+$ , métrologie  $\rightarrow$  s9  $t+l+m2+$ ,

s9  $\rightarrow$  s10  $t+l+m2+$ , s10  $\rightarrow$  sortie  $t+l+m2+$  est exploré à son tour.

Le but (présence  $t+l+m2+$  sortie) est atteint. Les règles partant du fait (présence p1 s10) ont toutes été utilisées. L'analyse repart en retour-arrière.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

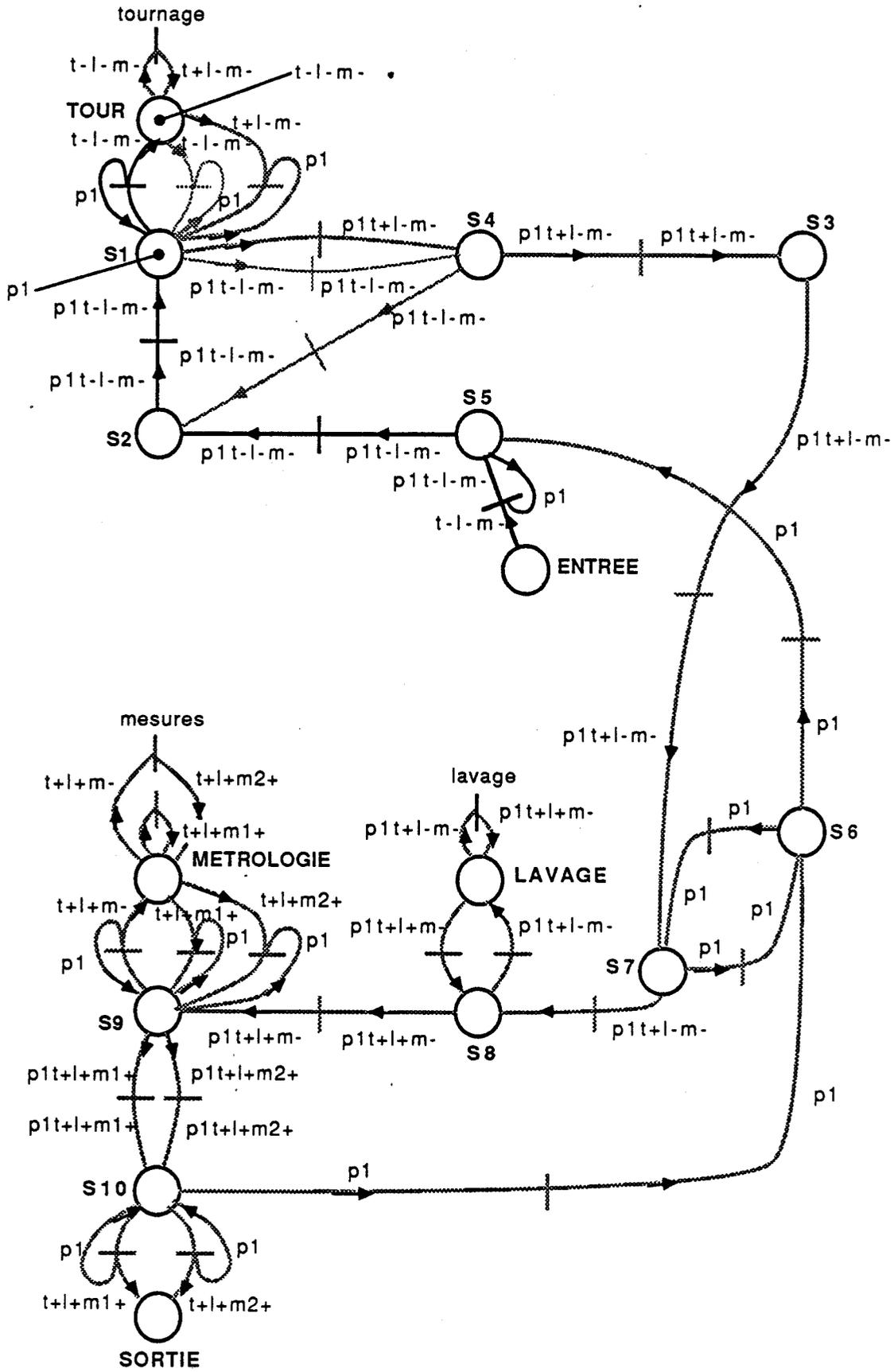


figure II.57 n°18  
 Nouveau retour-arrière sur le point d'indéterminisme directionnel {(présence t-l-m- tour) (présence p1 s1)}. Le logiciel précise que la règle tournage t-l-m-/t+l+m- a déjà été déclenchée à partir de cette situation. La règle déclenchable tour->s1\_t-l-m- représentant un chemin de mode dégradé du procédé va être déclenchée maintenant.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

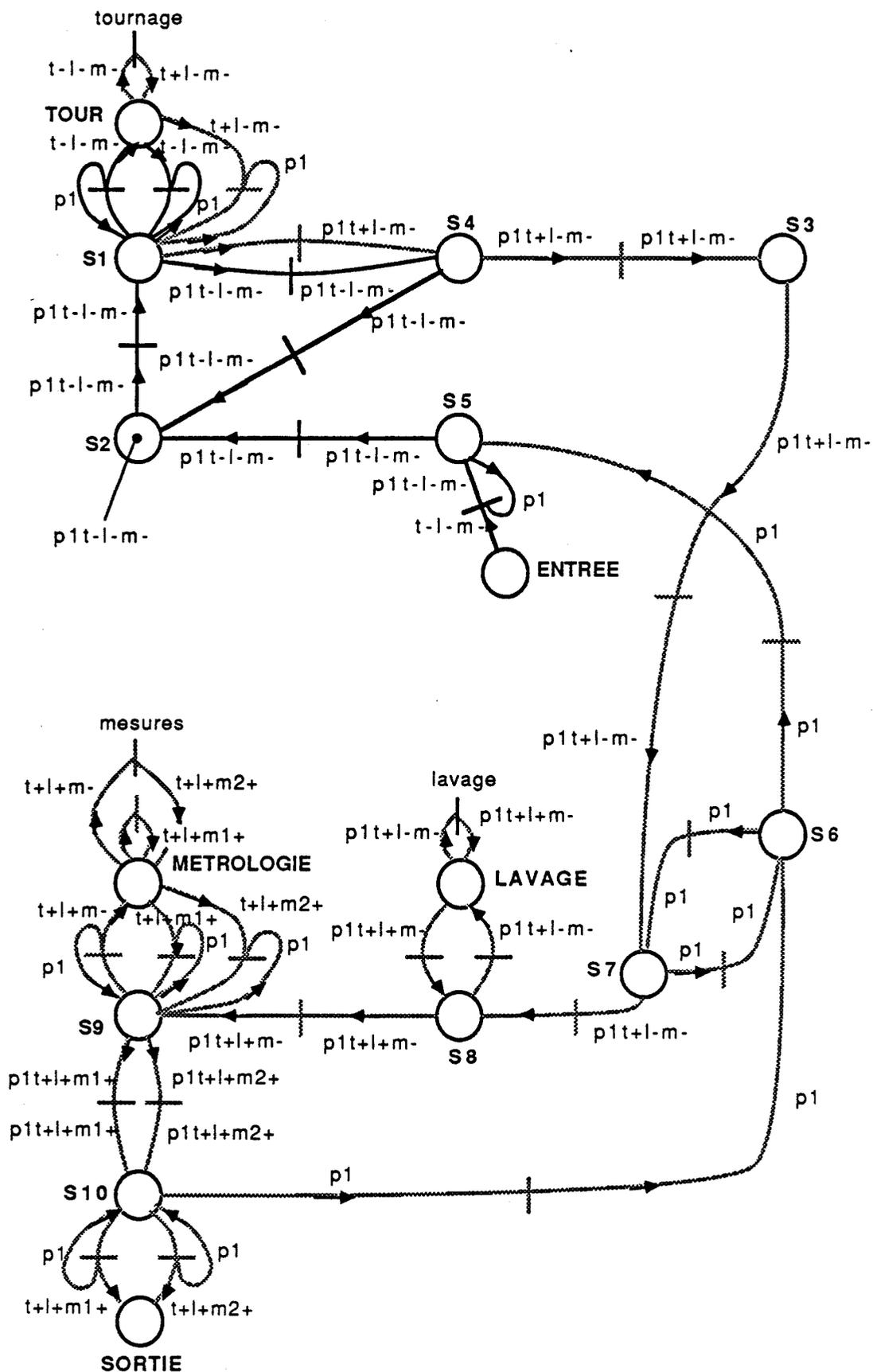


figure II.57 n°19

Les inférences relatives au chemin (tour->s1\_t-l-m-, s1->s4\_t-l-m-, s4->s2\_t-l-m-) sont effectuées. L'analyse de cette branche aboutit à la situation {(présence p1t-l-m- s2)} dont les règles d'évolution existantes ont toutes été explorées. L'analyse part en retour-arrière.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

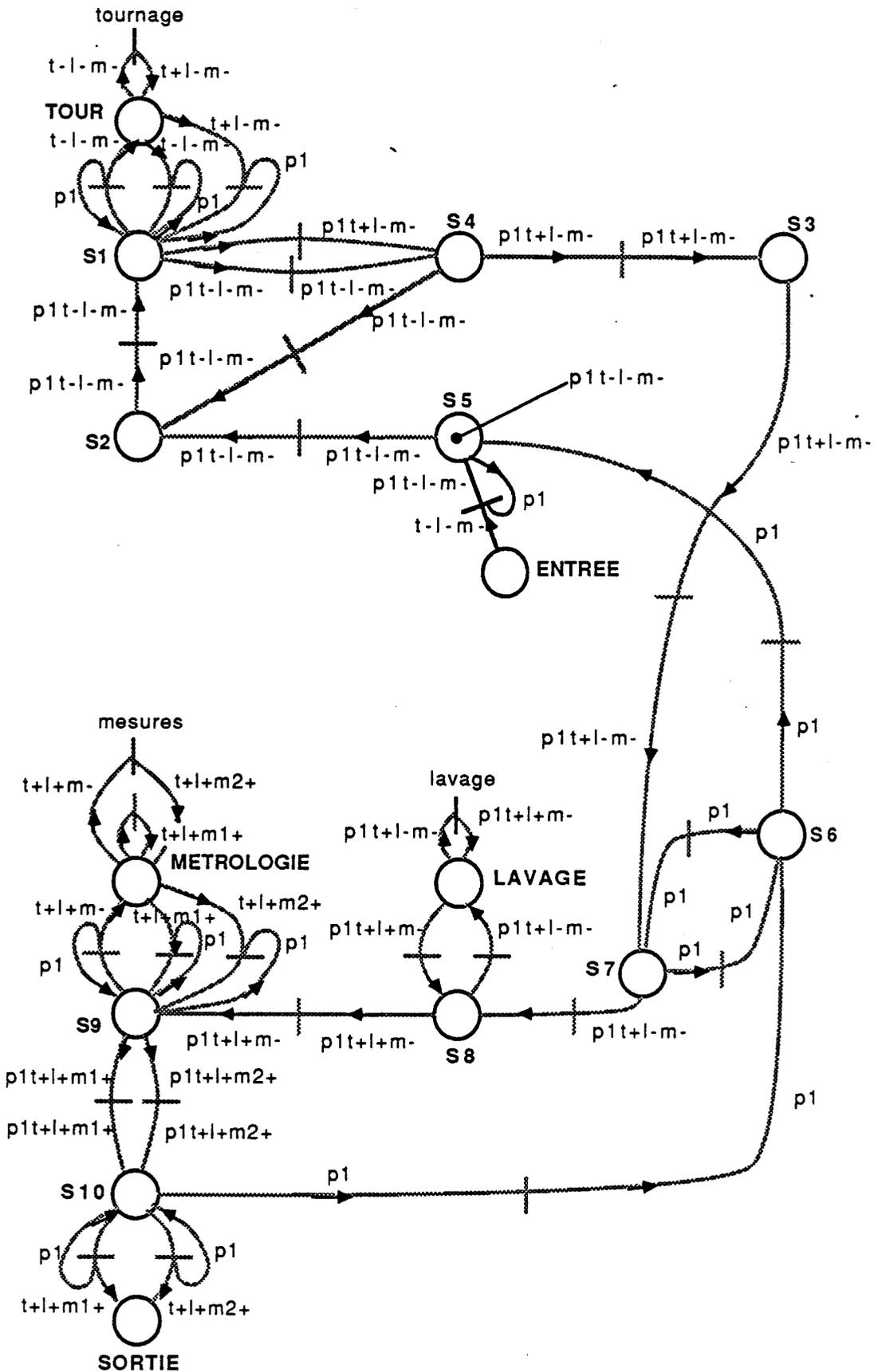


figure II.57 n°20

Retour-arrière jusqu'à la situation ((présence p1t-l-m- s5)).

Il n'y a plus de règles déclenchables à ce niveau.

Il n'y a pas de règles candidates à ce niveau.

Le logiciel demande à l'utilisateur s'il veut créer de nouvelles règles en partant de cette base de faits.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

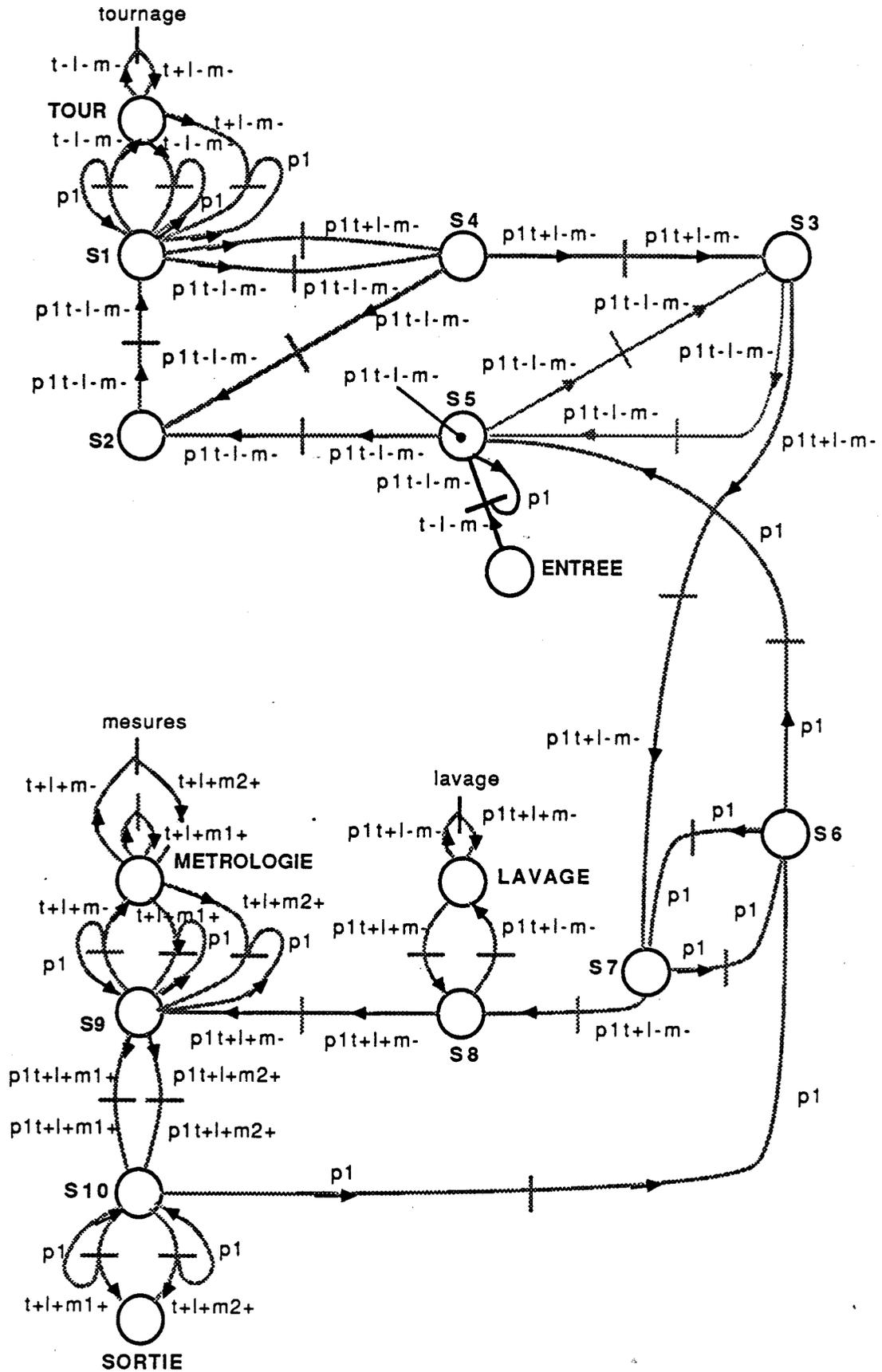


figure II.57 n°21

L'utilisateur construit les règles  $s5 \rightarrow s3_{t-l-m-}$  et  $s3 \rightarrow s5_{t-l-m-}$  pour représenter par exemple un mode dégradé du procédé ou un itinéraire de délestage.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

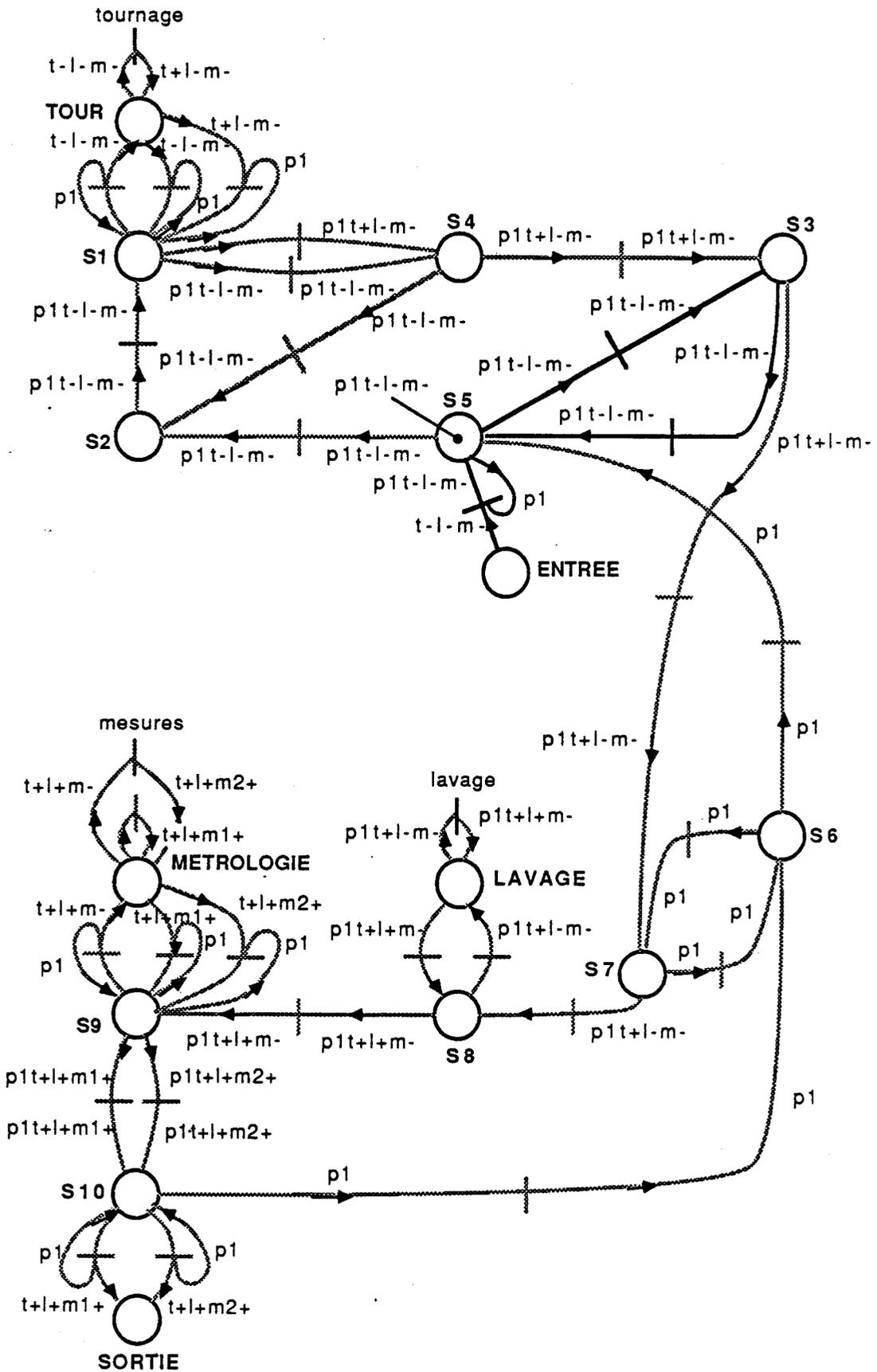


figure II.57 n°22

Les règles  $s5 \rightarrow s3_{t-l-m-}$  et  $s3 \rightarrow s5_{t-l-m-}$  sont déclenchées successivement.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

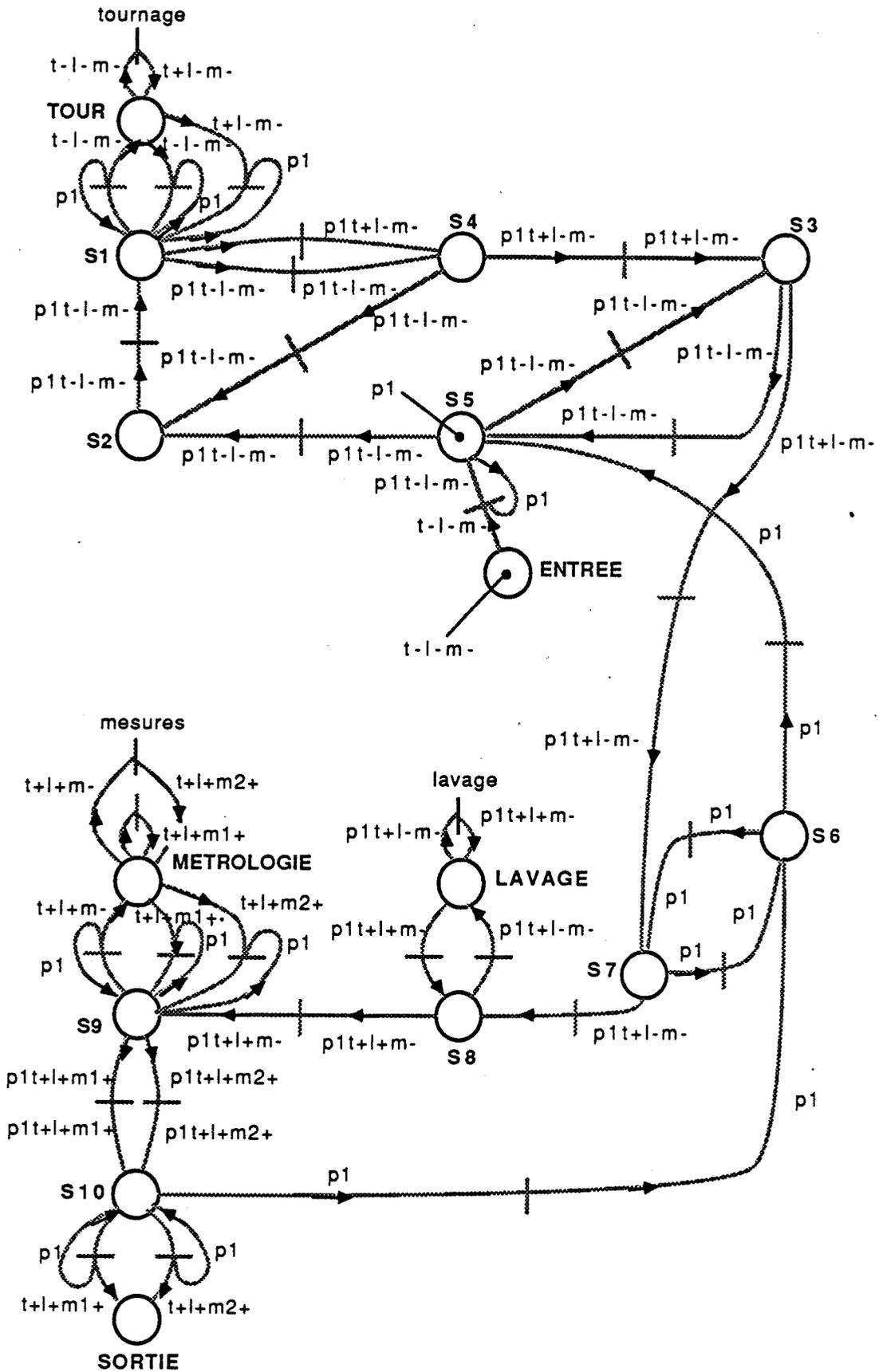


figure II.57 n°23

Retour-arrière jusqu'aux faits initiaux (présence t-l-m- entrée) et (présence p1 s5). La première étape de l'analyse est terminée. Le logiciel examine alors s'il existe des règles non utilisées, des faits initiaux non utilisés, des faits terminaux non atteints, et des feuilles non terminales. La deuxième étape de l'analyse consiste alors à examiner tous ces problèmes avec l'utilisateur. Aucun problème n'apparaît ici. L'analyse est terminée.

## SEQUENCE OPERATOIRE t-l-m-/t+l+m+

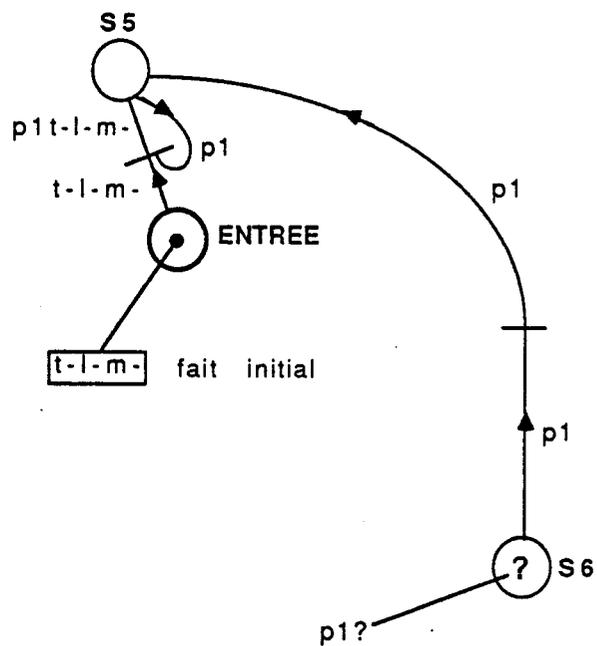


figure II.57 n°24

Nous allons examiner maintenant les possibilités de chaînage arrière de l'analyse.

Revenons au cas des figures 1 et 2.

A la question "La prémisse (présence p1 s5) doit-elle être considérée comme un fait initial de la gamme ?", l'utilisateur répond non.

Le logiciel demande alors si ce fait peut être déduit de règles qui n'ont pas encore été décrites.

Si l'utilisateur répond non ou est indécis, le chaînage arrière part en backtracking et la première étape de l'analyse se termine ici. Ce cas n'est pas intéressant.

Si l'utilisateur répond oui, et c'est ce cas qui offre un intérêt ici, l'utilisateur est conduit à construire au moins une règle, en l'occurrence la règle s6->s5\_p1. Une inférence est réalisée sur cette règle en chaînage arrière et c'est la question concernant l'origine du fait (présence p1 s6) qui est maintenant posée.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

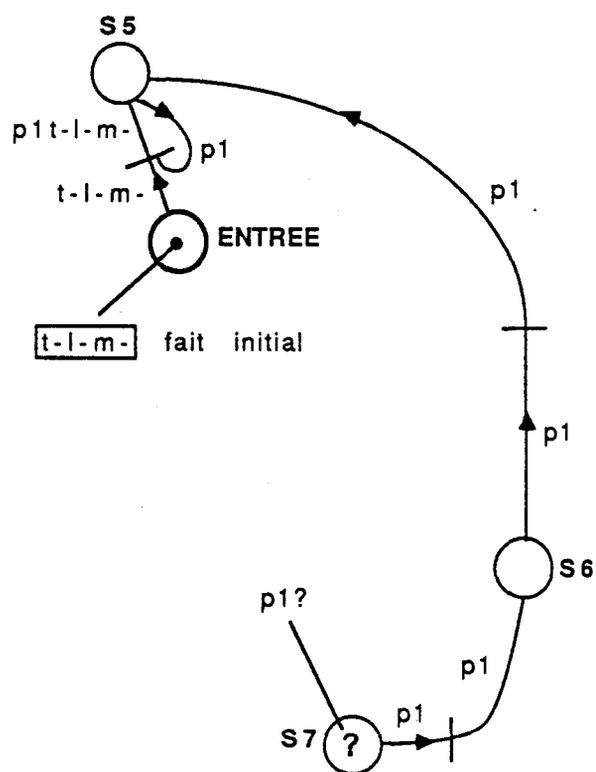


figure II.57 n°25

Si l'utilisateur précise que le fait (présence p1 s6) est un fait initial, alors les faits (présence t-l-m- entrée) et (présence p1 s6) sont pris comme faits de départ pour une redescente en chaînage avant. Les règles ne peuvent être liées définitivement à la description que par chaînage avant. Ce choix arbitraire est fait pour ne pas oublier de chemin d'exploration et pour assurer une certaine homogénéité dans l'examen des règles.

Ici, l'utilisateur a répondu que le fait (présence p1 s6) provenait d'une règle non encore décrite, la règle s7->s6\_p1, qu'il a alors construite.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

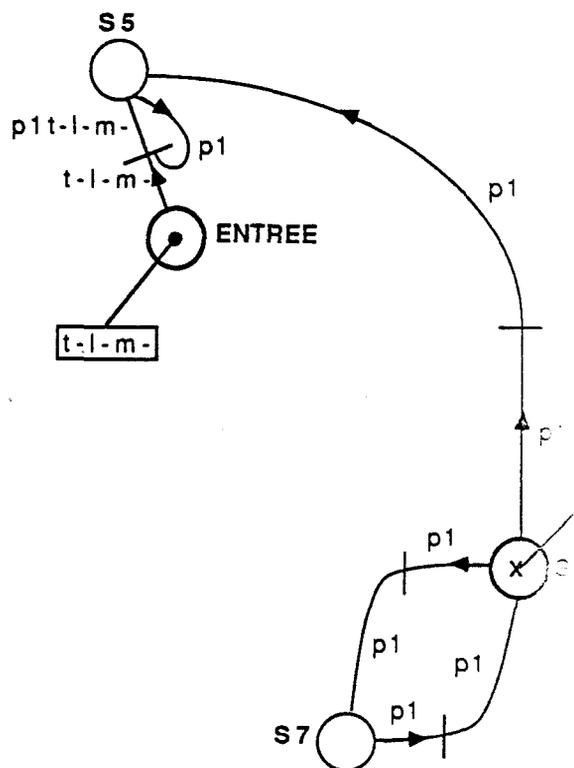


figure II.57 n°26

Supposons que l'utilisateur définisse maintenant la règle  $s6 \rightarrow s7$  pour justifier la provenance du fait (présence p1 s7). Le chaînage arrière poursuit. Le fait (présence p1 s6) provient de règles déjà examinées en chaînage arrière (Le comportement est identique si le fait provient de règles déjà utilisées en chaînage avant). Le fait est donc pris comme fait existant pour une redescente en chaînage avant.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

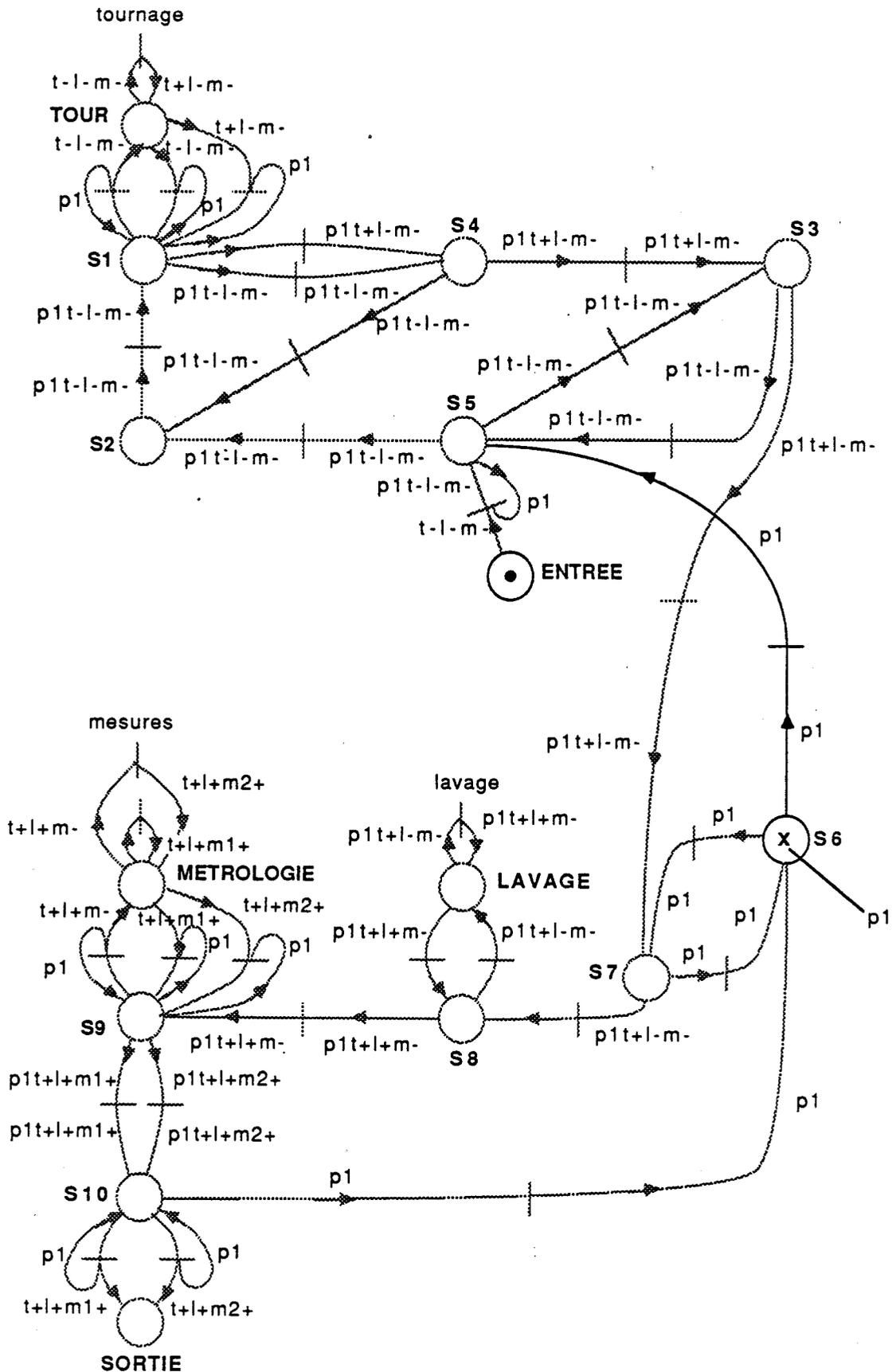


Figure II.57 n°27

Revenons maintenant au cas des figures 1 et 2. Supposons maintenant que les règles examinées en chaînage arrière ne sont plus créées au fur et à mesure de l'analyse en chaînage arrière, mais globalement, en vrac, au départ. Le chaînage arrière évolue comme précédemment mais sans solliciter l'utilisateur.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

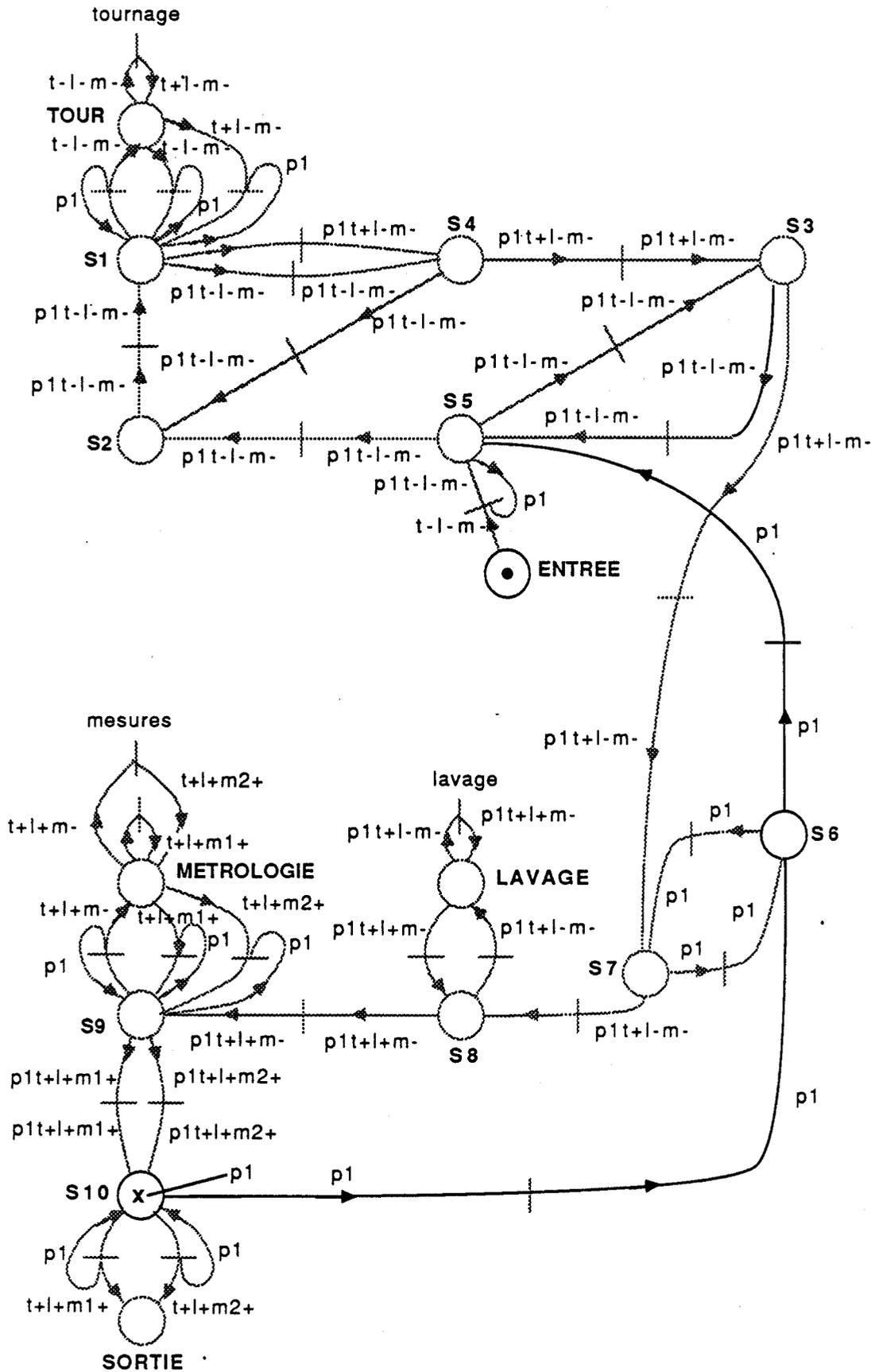


figure II.57 n°28

L'analyse évolue en chaînage arrière d'une façon pratiquement symétrique à ce qui se produit en chaînage avant. Nous rencontrons ici un point de choix (présence p1 s10) qui peut provenir de 2 règles : s10->sortie t+l+m1+ et s10->sortie t+l+m2+. Ces deux voies sont explorées à tour de rôle en profondeur d'abord.

## SEQUENCE OPERATOIRE t-l-m-/t+l+m+

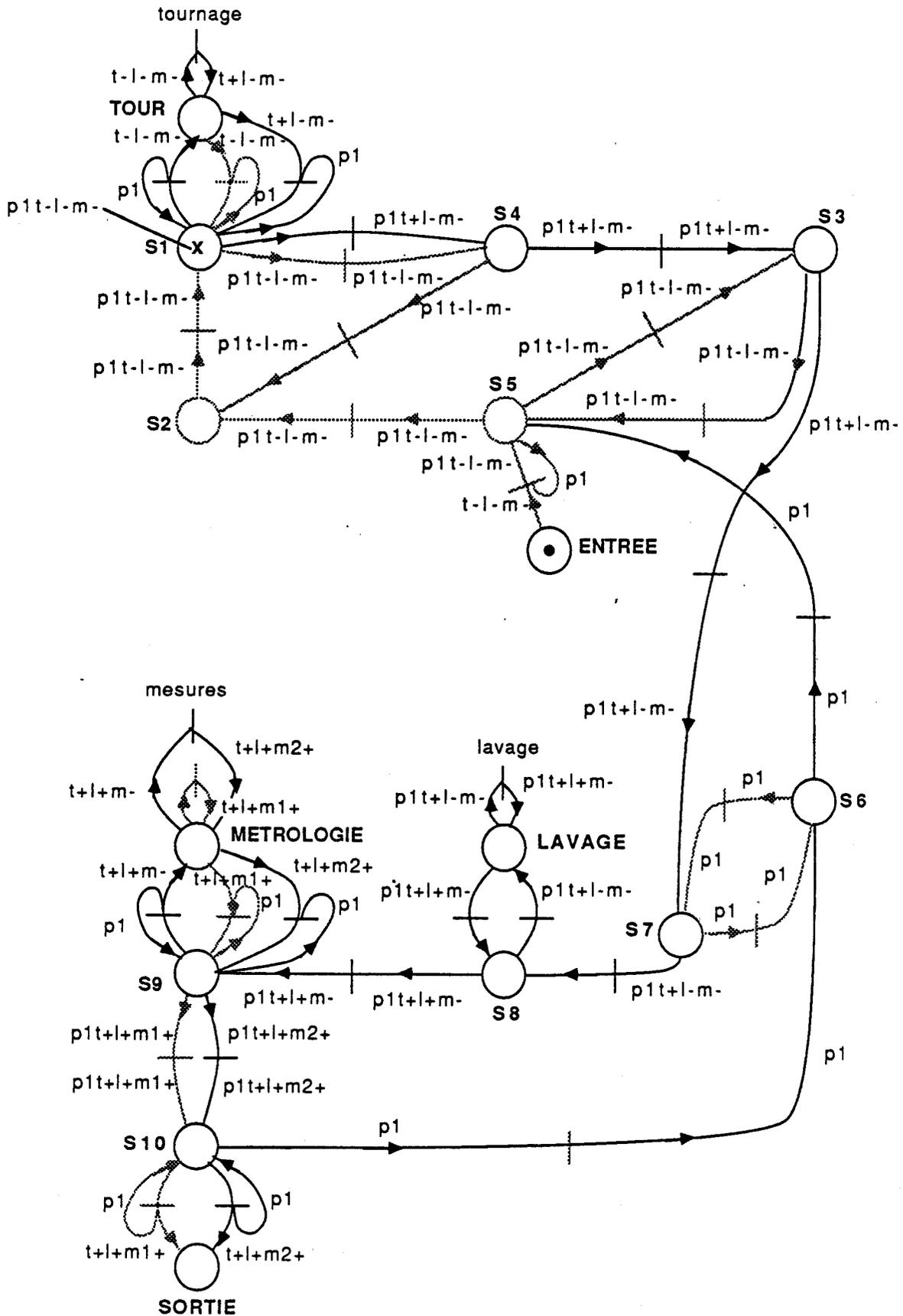


figure II.57 n°29

La première des deux voies est explorée en profondeur d'abord. On arrive au point  $\{(présence\ p1t-l-m-\ s1)\}$  qui est lui aussi un point de choix entre 2 voies :  $tour \rightarrow s1\_t-l-m-$  et  $s2 \rightarrow s1\_t-l-m-$ .

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

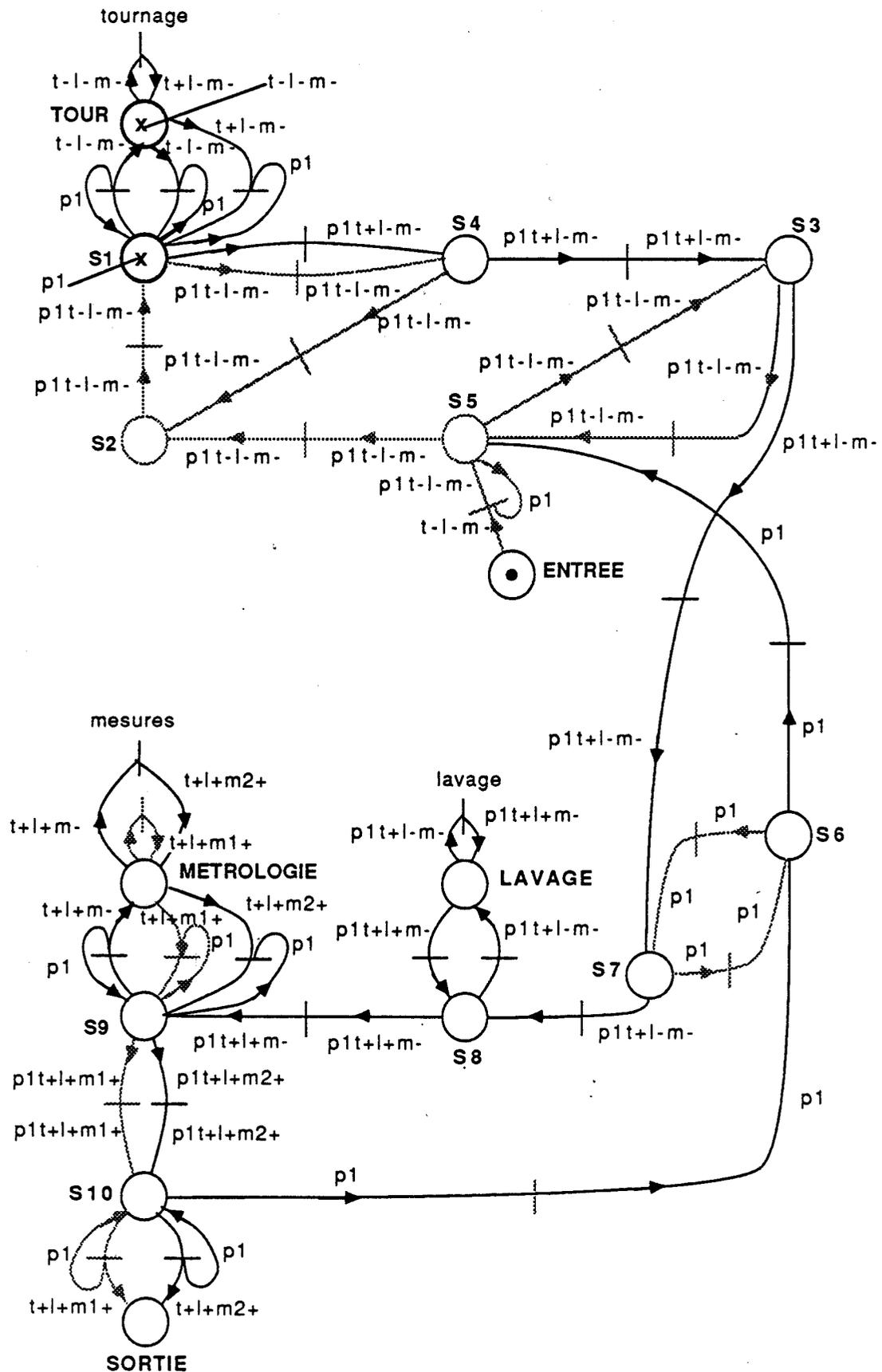


figure II.57 n°30

La première des deux voies est explorée en chaînage arrière. L'analyse rencontre la situation de bouclage {(présence t-l-m- tour) (présence p1 s1)} qui provient de règles déjà examinées en chaînage arrière. Les faits (présence t-l-m- tour) et (présence p1 s1) sont mémorisés comme faits de départ pour une redescente ultérieure en chaînage avant.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

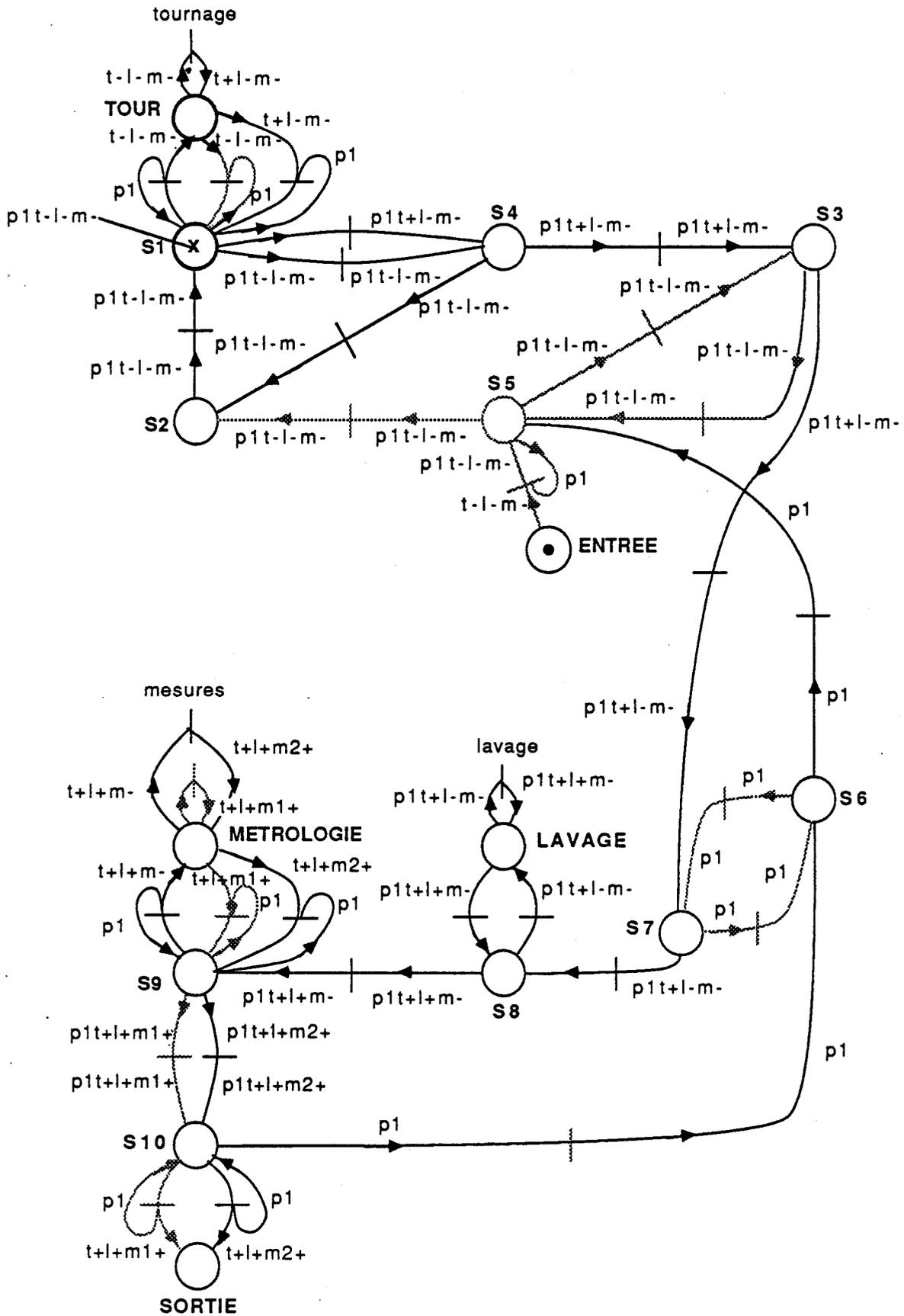


figure II.57 n°31

Retour-arrière du chaînage arrière sur la situation {(présence p1t-l-m- s1)}. La deuxième voie en attente, c'est-à-dire la voie correspondant aux règles s2->s1 t-l-m-, s4->s2 t-l-m- et s1->s4 t-l-m- est explorée à son tour. On retombe sur une situation de bouclage {(présence p1t-l-m- s1)} qui provient de règles déjà examinées en chaînage arrière.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

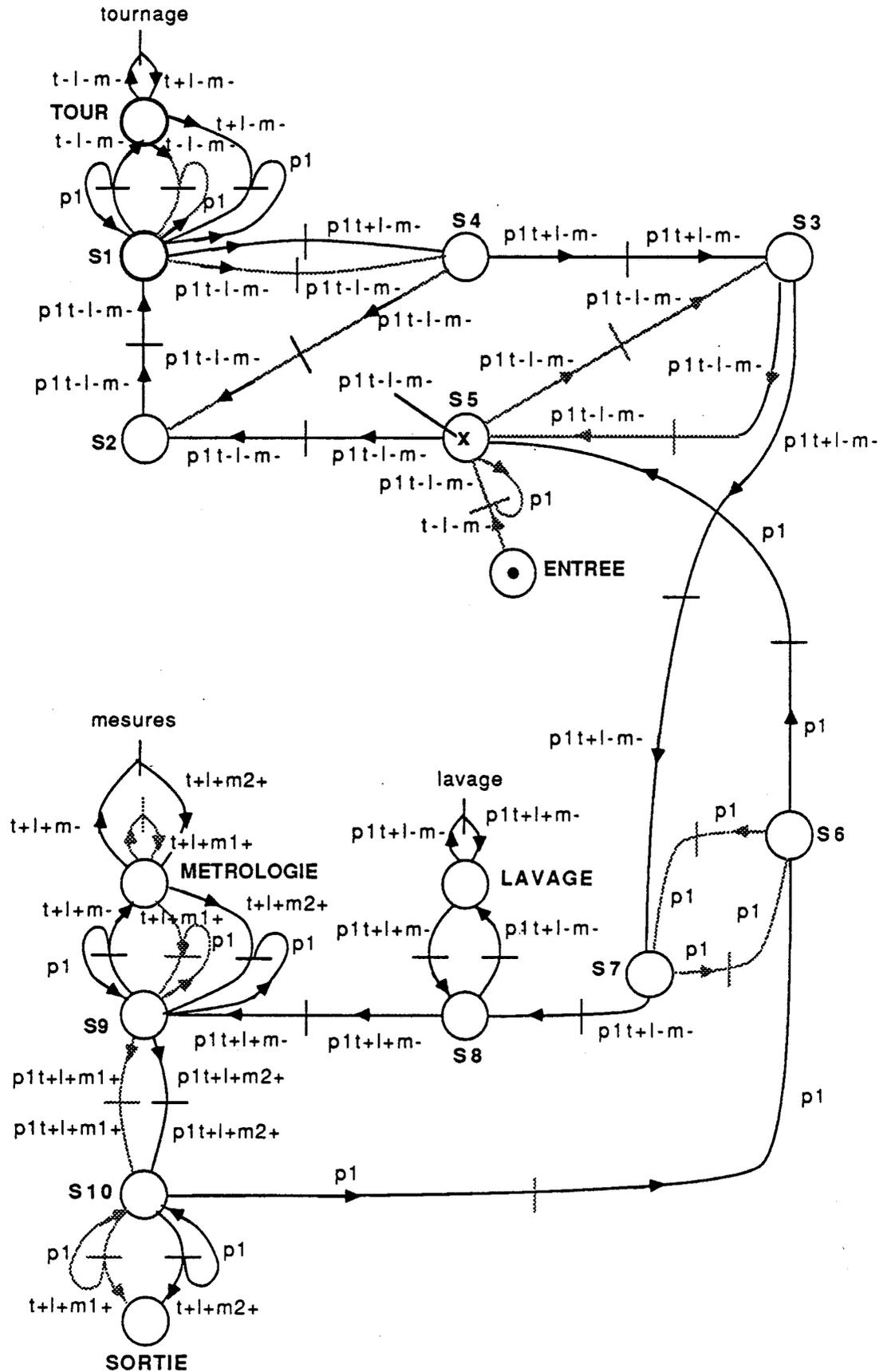


figure II.57 n°32

Le fait (présence p1t-l-m- s2) correspond à un point de choix entre les règles s4->s2 t-l-m- examinée précédemment et s5->s2 t-l-m- à laquelle l'analyse s'intéresse maintenant. Le chaînage conduit au fait (présence p1t-l-m- s5) qui est un nouveau point de choix.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

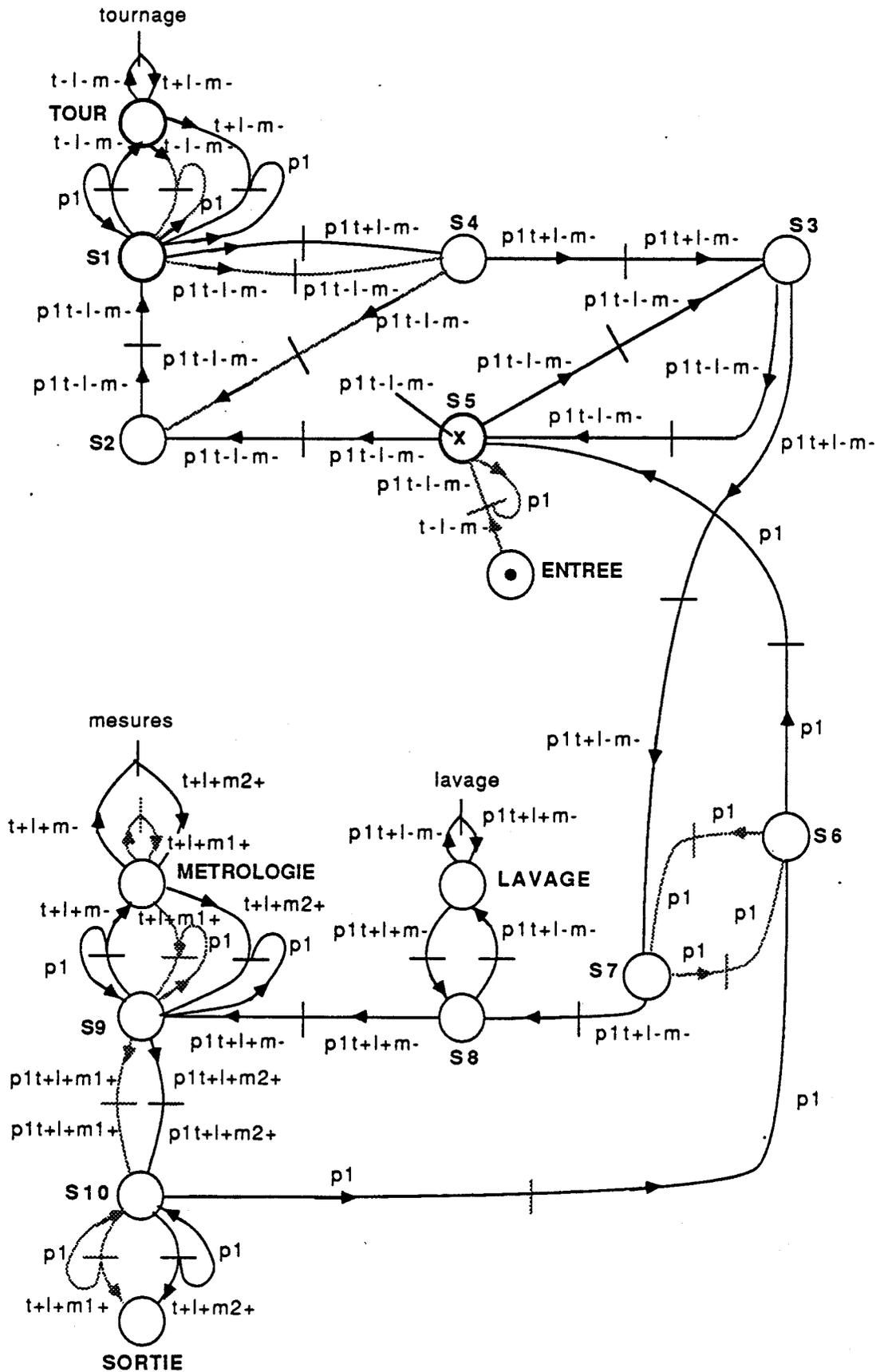


figure II.57 n°33

Le chemin constitué des règles  $s3 \rightarrow s5_{t-l-m-}$  et  $s5 \rightarrow s3_{t-l-m-}$  est exploré en chaînage arrière maintenant. L'analyse tombe sur la situation  $\{(présence\ p1t-l-m-\ s5)\}$  qui provient de règles déjà examinées en chaînage arrière ( $s3 \rightarrow s5_{t-l-m-}$ ) et de la règle  $entrée \rightarrow s5_{t-l-m-}$  explorée en chaînage avant et qui a lancé le chaînage arrière. Le fait  $\{(présence\ p1t-l-m-\ s5)\}$  est mémorisé comme fait de départ pour une redescente ultérieure en chaînage avant.

## SEQUENCE OPERATOIRE t-l-m-/t+l+m+

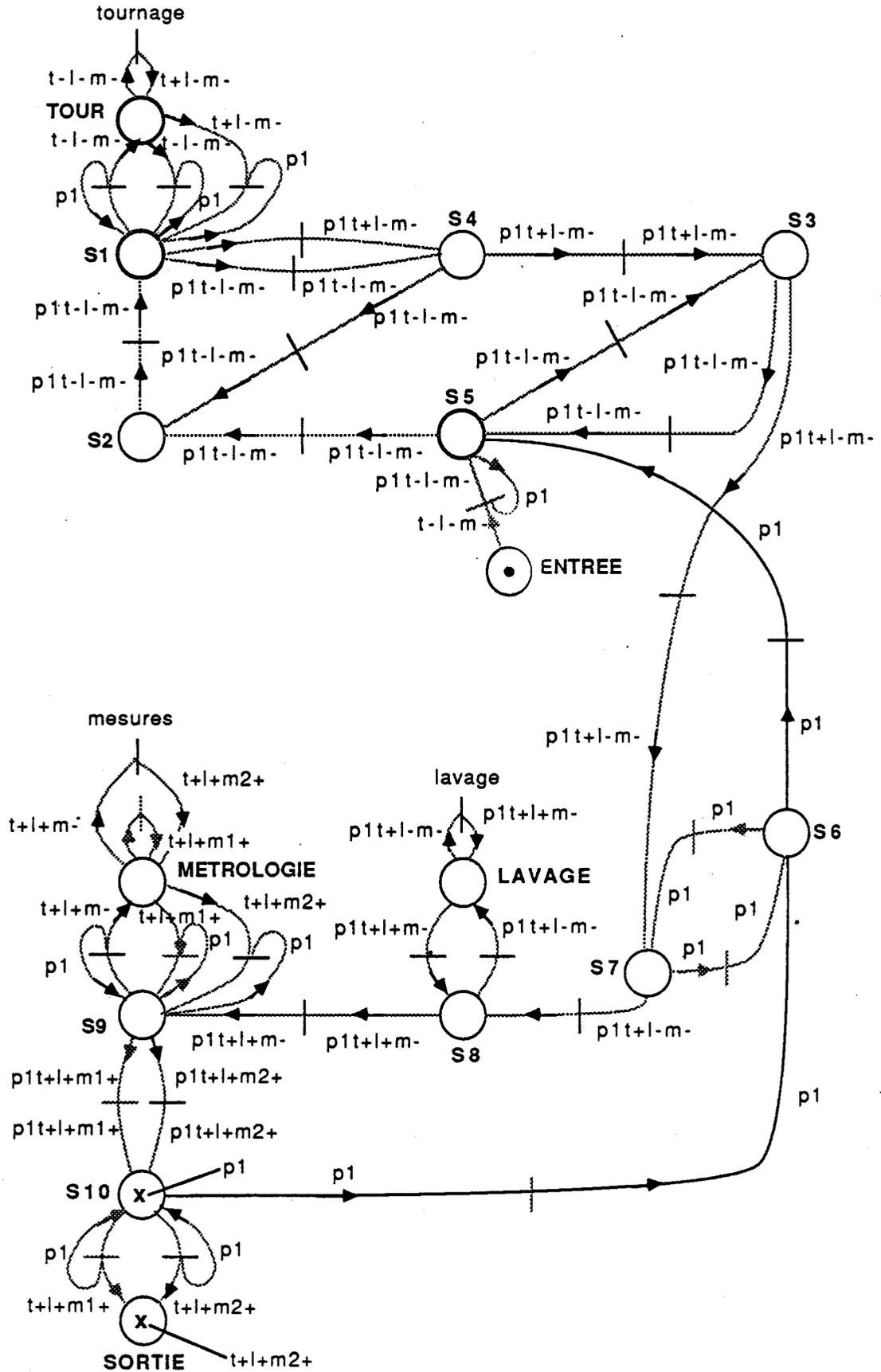


figure II.57 n°34

Un retour-arrière du chaînage arrière est alors effectué jusqu'au point de choix ((présence  $p1$  s10) (présence  $t+l+m2+$  sortie)). La règle  $s10 \rightarrow$  sortie  $t+l+m1+$  qui représente une possibilité en attente du chaînage arrière va être examinée maintenant.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

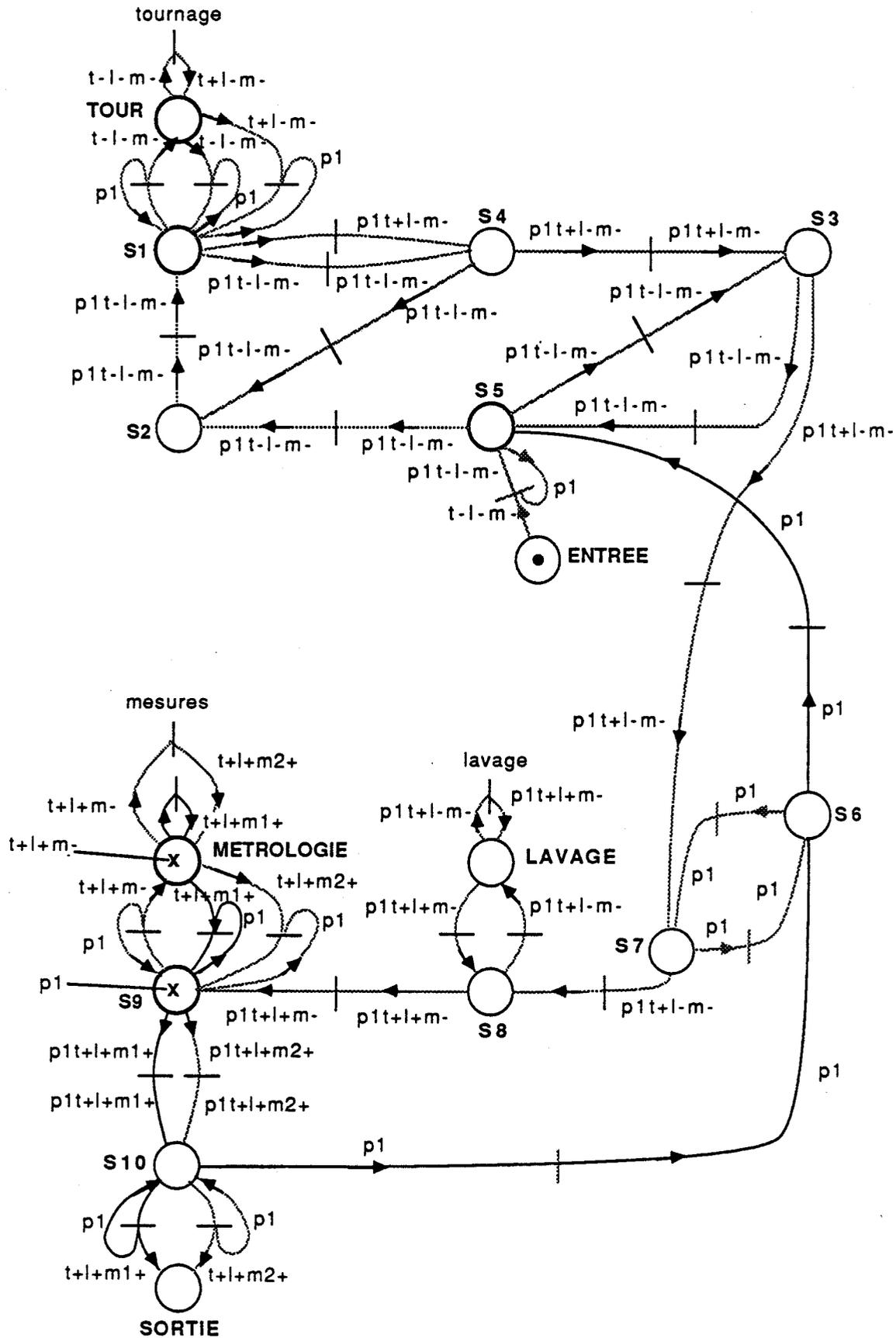


figure II.57 n°35

Le chemin constitué des règles  $s10 \rightarrow \text{sortie } t+l+m1+$ ,  $s9 \rightarrow s10 \ t+l+m1+$ ,  $\text{m\u00e9trologie} \rightarrow s9 \ t+l+m1+$  et  $\text{mesures } t+l+m- / t+l+m1+$  est explor\u00e9 en chaînage arri\u00e8re jusqu'aux faits (pr\u00e9sence  $p1$   $s9$ ) et (pr\u00e9sence  $t+l+m-$   $\text{m\u00e9trologie}$ ) qui proviennent de r\u00e8gles d\u00e9j\u00e0 examin\u00e9es en chaînage arri\u00e8re. Ces faits sont m\u00e9moris\u00e9s comme faits de d\u00e9part pour une redescente ult\u00e9rieure en chaînage avant.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

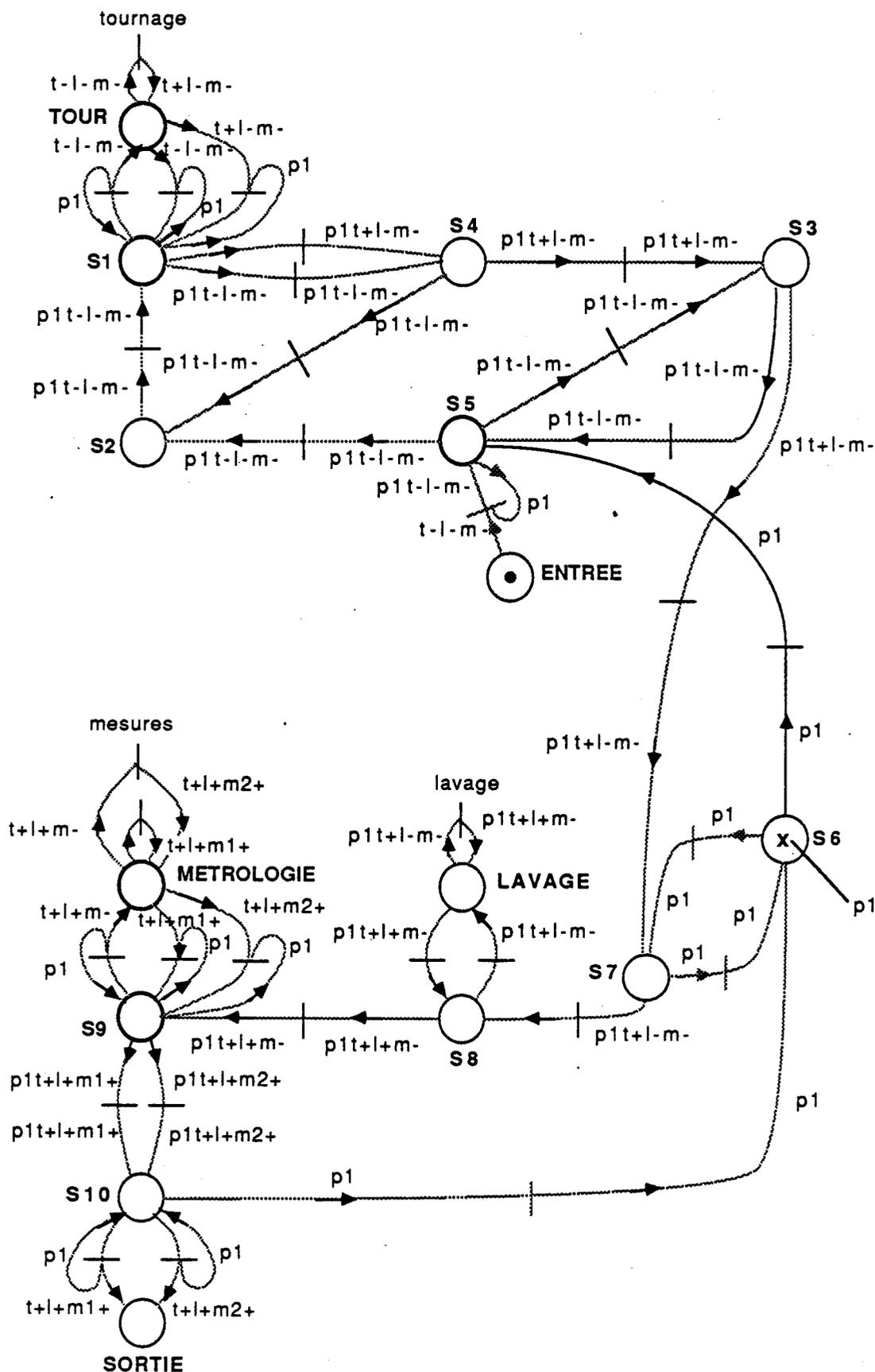


figure II.57 n°36

Un retour-arrière du chaînage arrière est alors effectué jusqu'au point de choix (présence p1 s6).

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

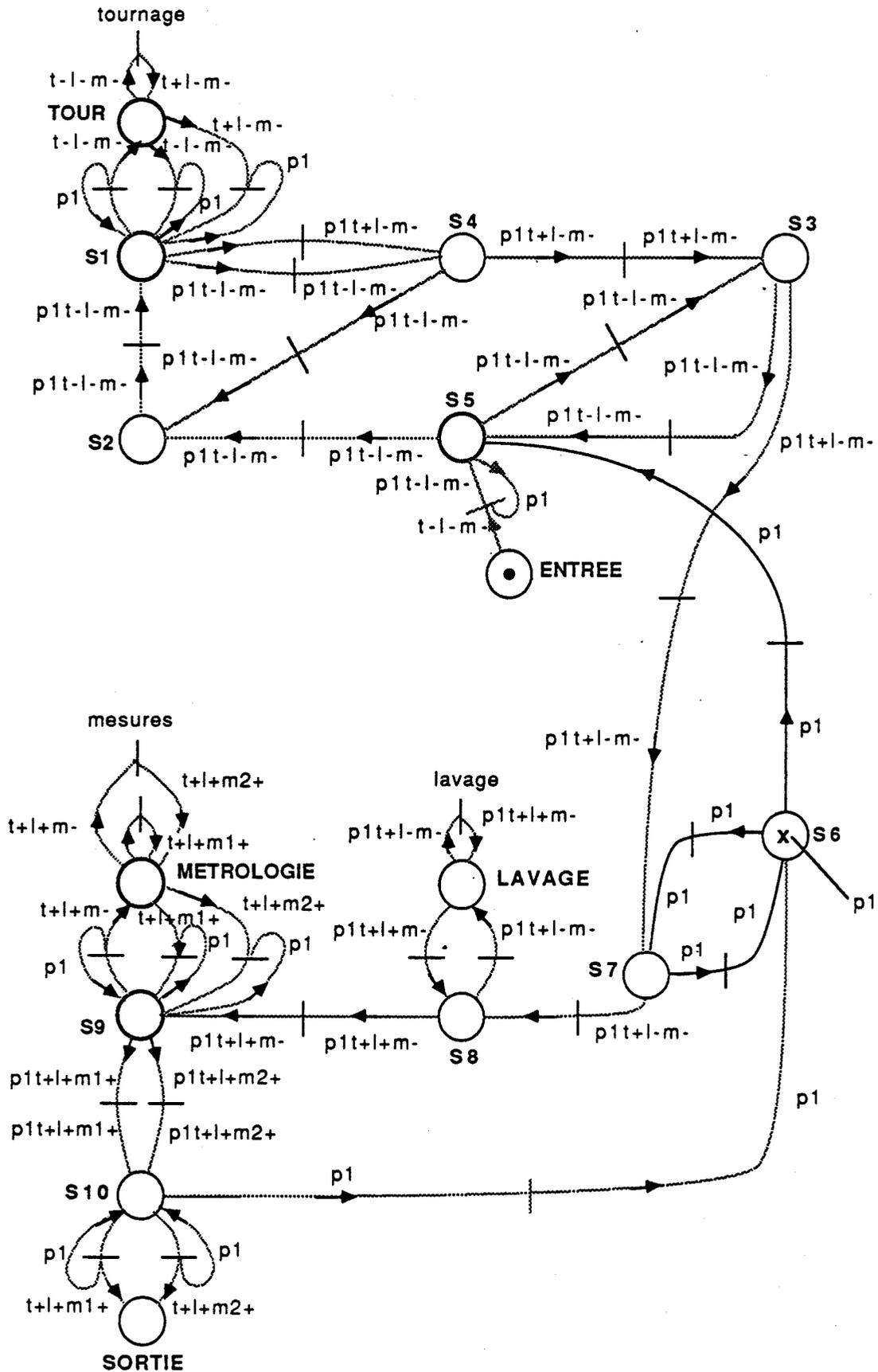


figure II.57 n°37

Le chemin constitué des règles  $s7 \rightarrow s6_{p1}$  et  $s6 \rightarrow s7_{p1}$  est alors exploré. Le chaînage arrière retrouve le fait (présence  $p1$   $s6$ ) qui provient de règles déjà examinées en chaînage arrière. Ce fait est mémorisé comme fait de départ pour une redescente ultérieure en chaînage avant.



# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

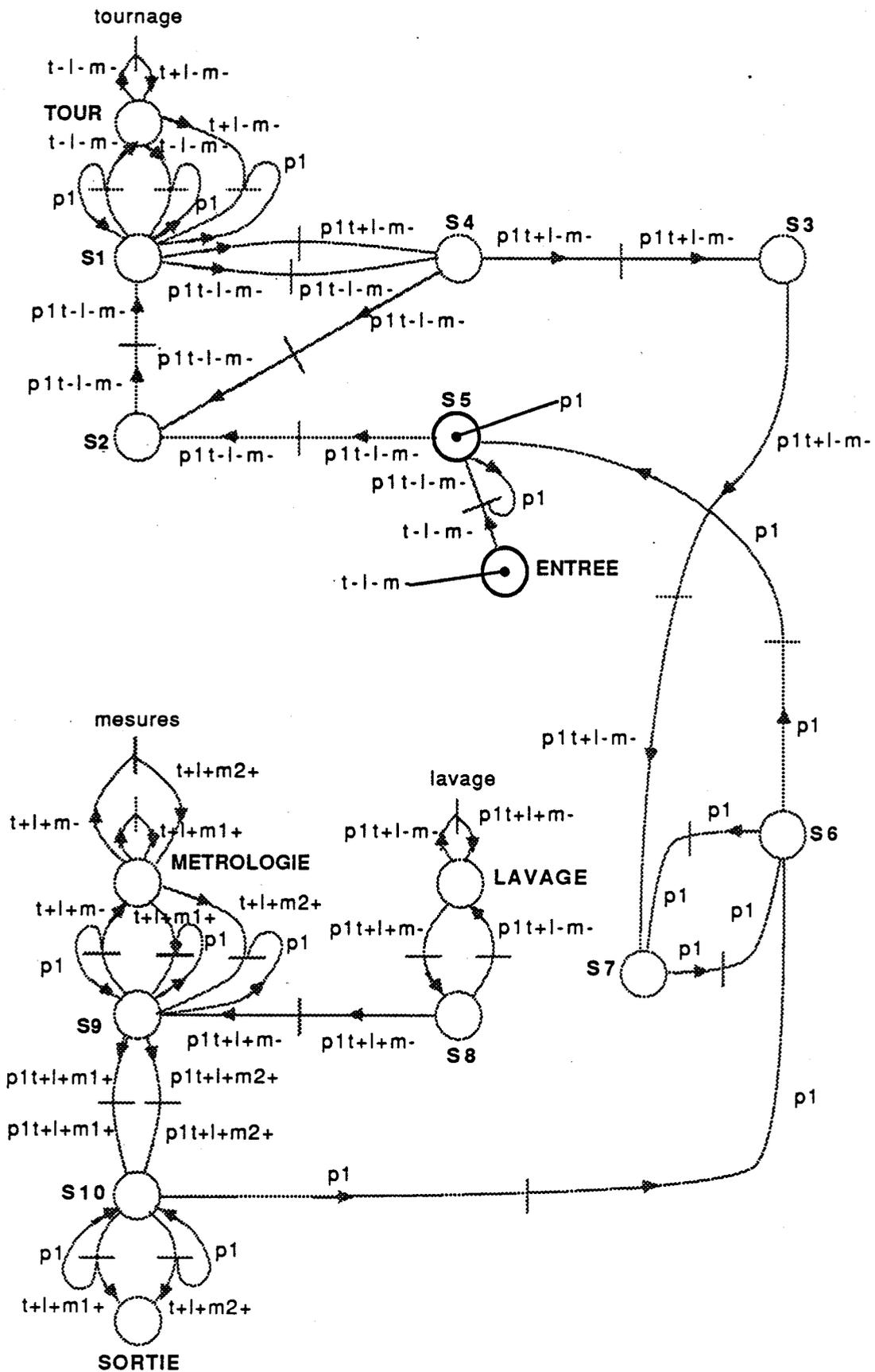


figure II.57 n°39

Nous allons présenter ici une des fonctions de la deuxième étape de l'analyse. Revenons aux premières figures. Supposons que l'utilisateur ait saisi en vrac l'ensemble des règles en grisé sur le dessin ci-dessus. Nous avons en outre les faits initiaux (présence t-l-m- entrée) et (présence p1 s5) à partir desquels une phase de chaînage avant peut être lancée. On note qu'il manque la règle s7->s8\_t-l-m-.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

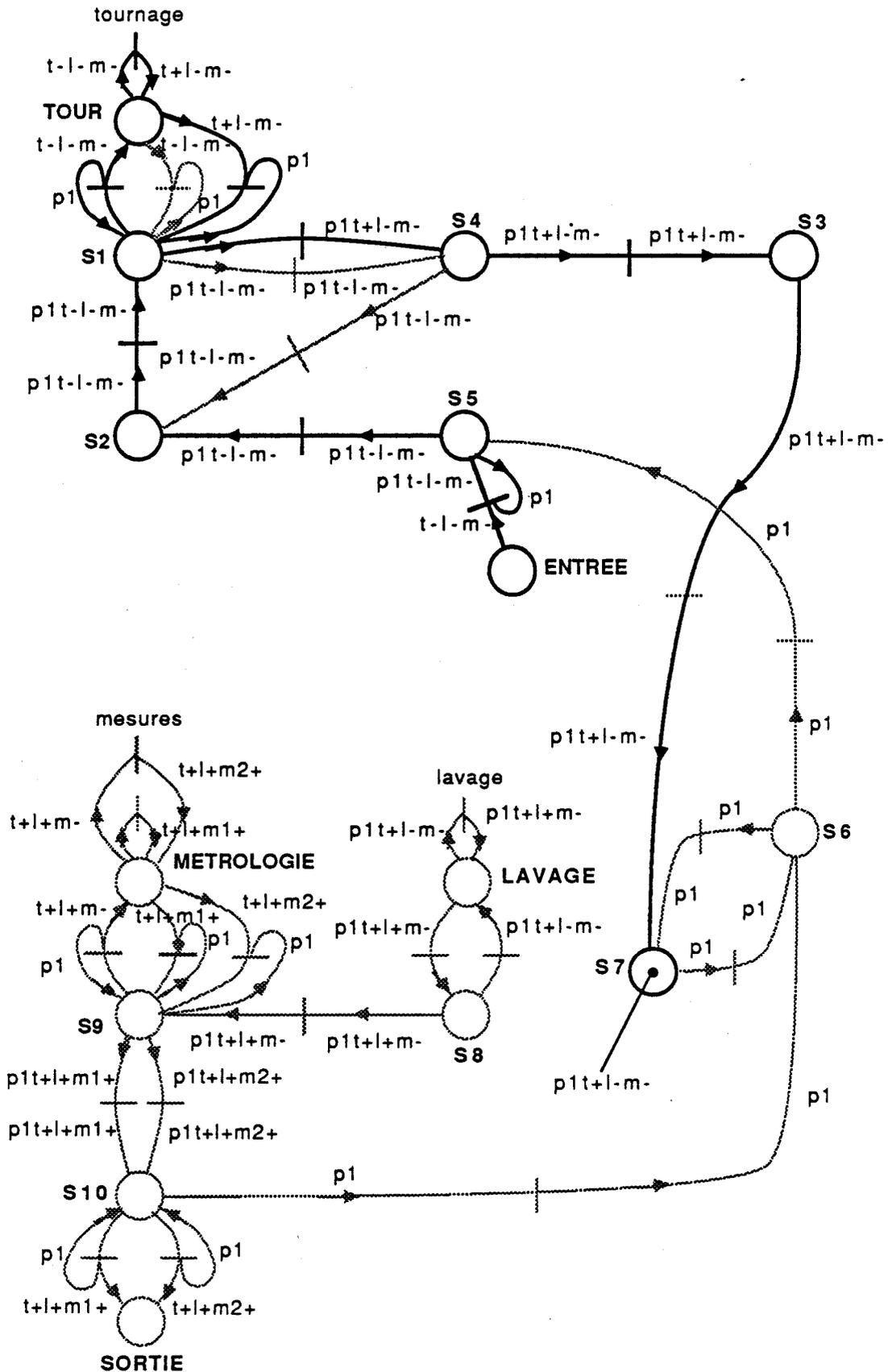


figure II.57 n°40

La phase de chaînage avant parvient au fait (présence  $p1t+l-m- s7$ ) et ne peut continuer. Le logiciel demande alors à l'utilisateur si ce fait doit être déclaré comme fait terminal, ou s'il désire ajouter ici des règles à la description opératoire. Supposons la réponse négative. Le fait (présence  $p1t+l-m- s7$ ) est alors mémorisé pour le moment comme feuille non terminale de l'arborescence des règles.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

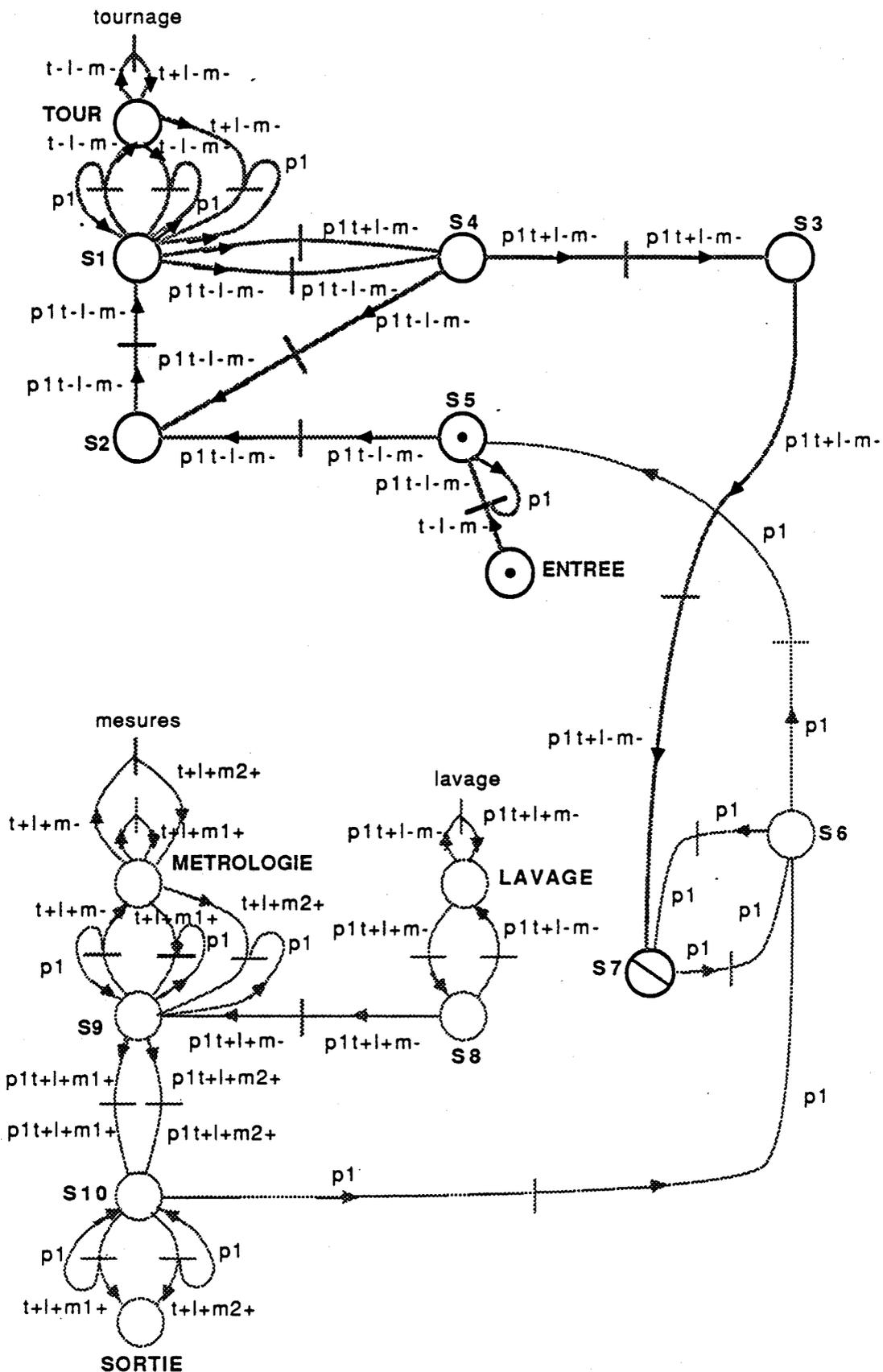


figure II.57 n°41

Les retours-arrière s'effectuent normalement comme précédemment. La fin d'une première étape d'analyse est atteinte. Il reste un certain nombre de règles en suspens qui n'ont pas été utilisées.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

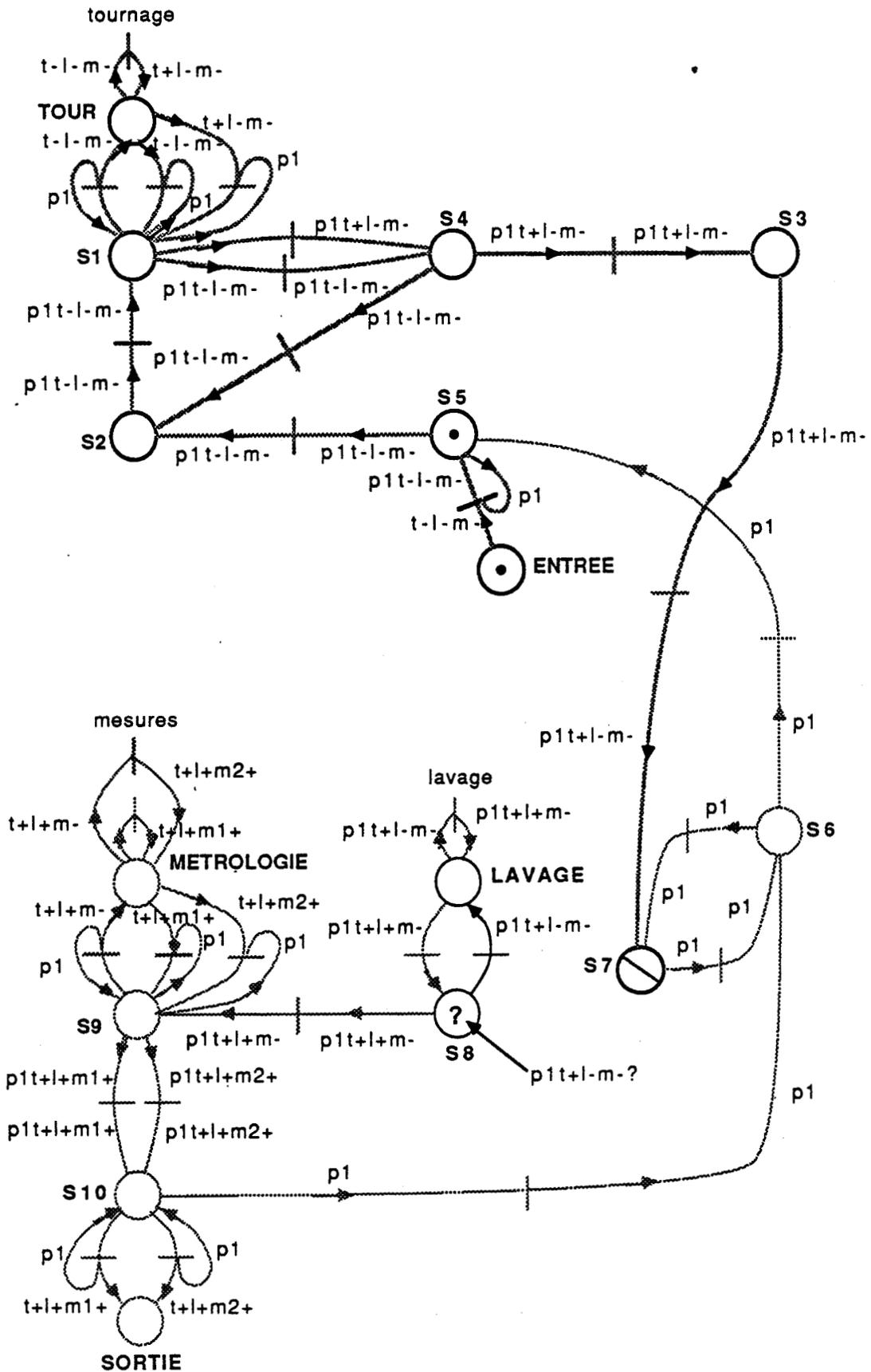


figure II.57 n°42

Le logiciel va alors, dans la 2<sup>ème</sup> étape de l'analyse, tenter de rattacher tout d'abord ces règles inutilisées au reste de la description. Les règles inutilisées vont être passées en revue. Le logiciel va demander à l'utilisateur s'il veut les supprimer. Si oui, il y a destruction des règles concernées. Sinon, le logiciel va tenter d'établir la provenance de leurs prémisses en chaînage arrière, comme on l'a déjà vu. Enfin, en cas d'indécision, ces règles peuvent être mises de côté dans une zone de débarras en marge de la gamme afin que l'utilisateur puisse les réintégrer plus tard, éventuellement.

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

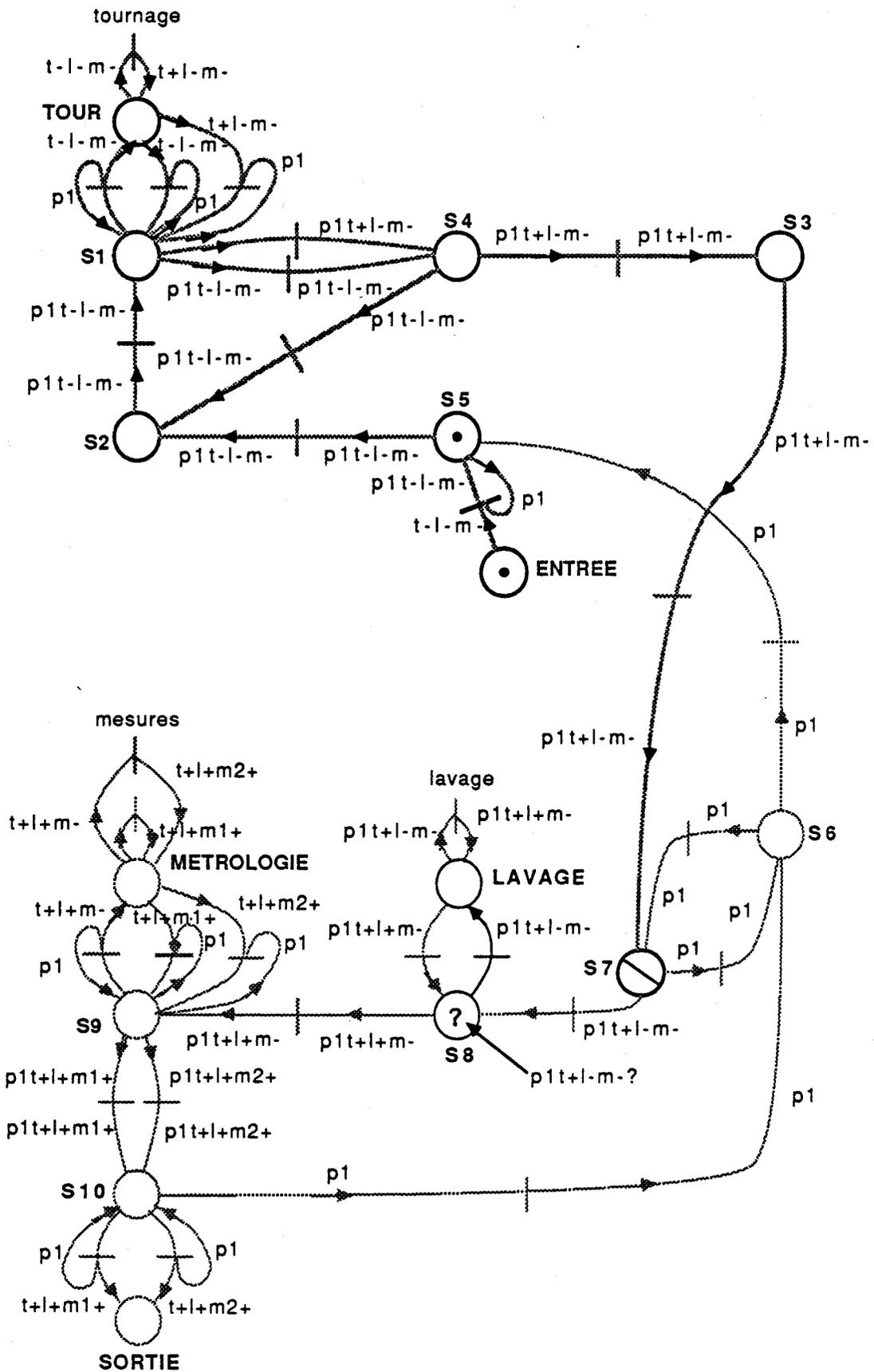


Figure II.57 n°43

Supposons que le logiciel s'intéresse à la règle inutilisée  $s8 \rightarrow \text{lavage } t+l-m-$  et cherche à établir la provenance de la prémisse (présence  $p1t+l-m- s8$ ). L'utilisateur ne souhaite pas déclarer ce fait comme fait initial. Le logiciel va donc demander si ce fait provient de règles non encore saisies. Supposons que l'utilisateur réponde positivement et introduise la règle  $s7 \rightarrow s8 \text{ } t+l-m-$ .

# SEQUENCE OPERATOIRE t-l-m-/t+l+m+

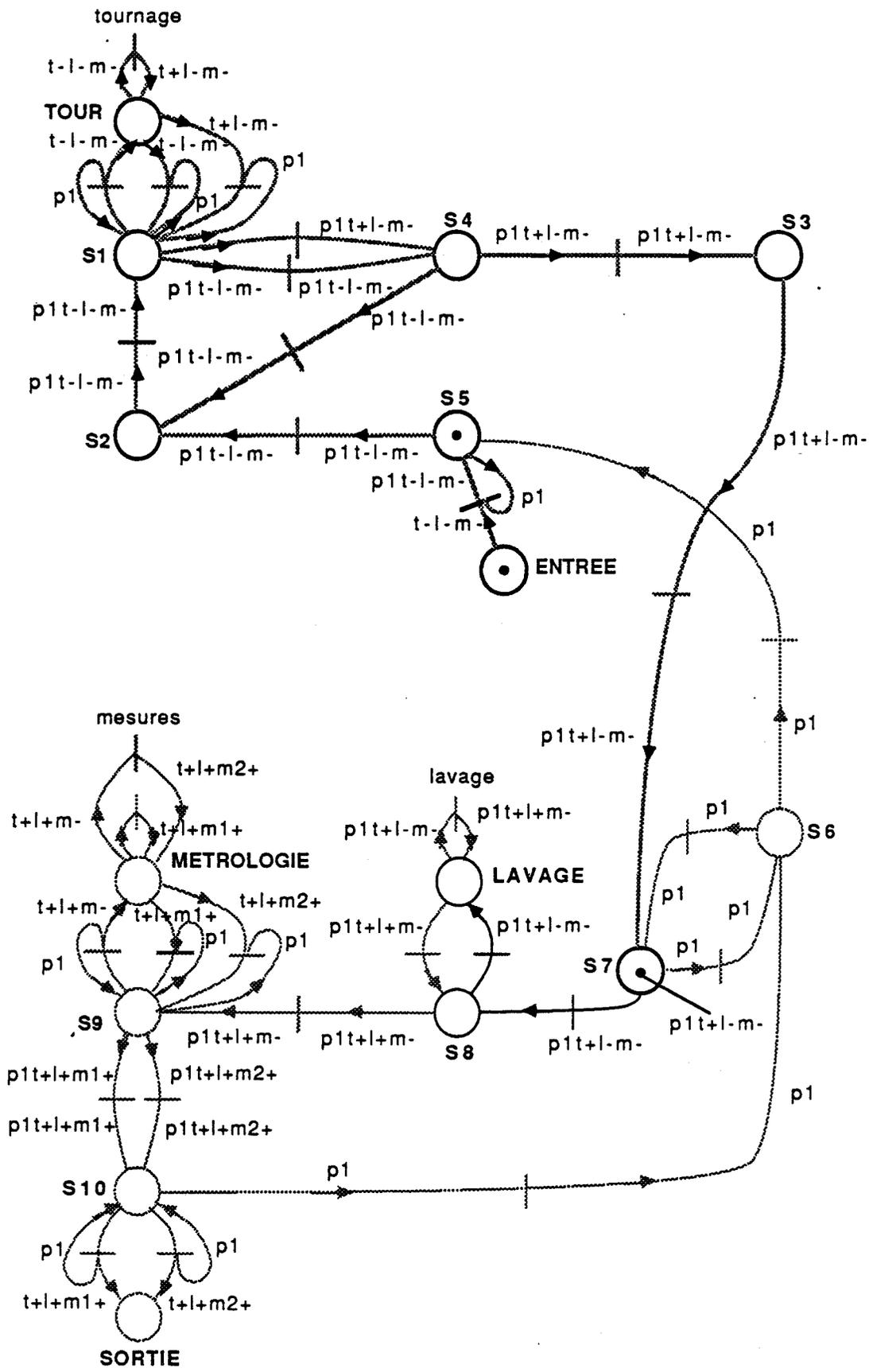


figure II.57 n°44

Le chaînage arrière se poursuit jusqu'au fait (présence  $p1t+l-m-$  s7) qui a déjà été atteint (en chaînage avant) et qui, par conséquent, va servir de fait de départ pour la redescente ultérieure en chaînage avant.

## SEQUENCE OPERATOIRE t-l-m-/t+l+m+

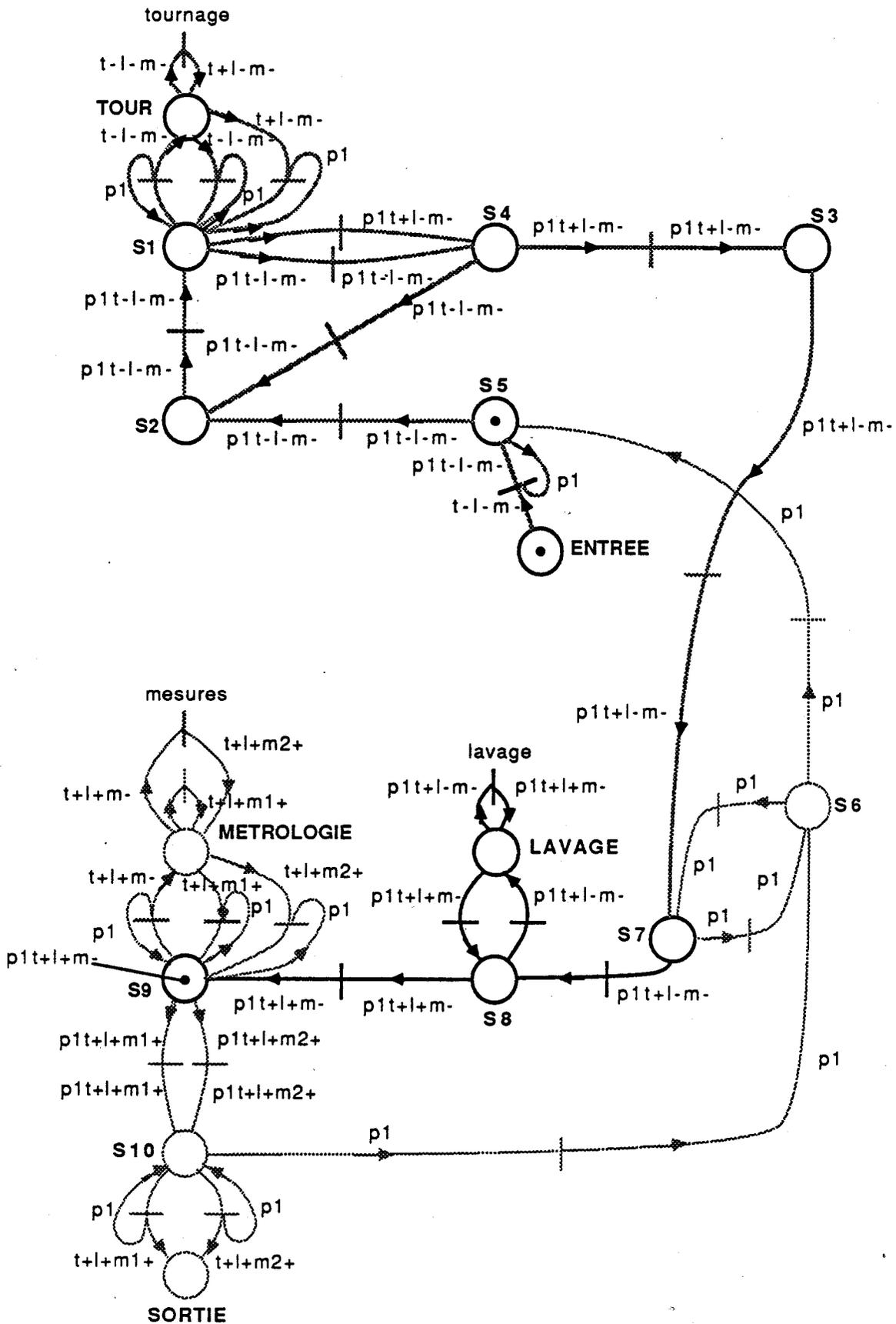


figure II.57 n°45

La phase de chaînage avant se déroule alors normalement à partir du fait (présence  $p1t+l-m-$  s7) qui n'est plus mémorisé comme extrémité non terminale. Lorsque toutes les règles inutilisées auront été étudiées, le logiciel amènera alors l'utilisateur à se prononcer sur les faits initiaux restés inutilisés et sur les faits terminaux qui n'ont pas été atteints. L'analyse complète sera à ce moment achevée.

## II.3 PLAN DU MOTEUR D'INFERENCE. SURVOL DE QUELQUES FONCTIONNALITES UTILITAIRES DE L'ANALYSEUR

### II.3.1. Le corps principal

La présentation du plan du moteur d'inférence de l'analyseur de cohérence et de complétude va être donné dans les pages qui suivent.

Bien que cette présentation soit limitée aux fonctions essentielles du moteur, la description reste technique et délicate. Elle peut être évitée en première lecture.

Le moteur d'inférence est constitué d'appels mutuels de procédures directement ou indirectement récursives. Il n'est donc pas évident d'en faire un synoptique simple. Afin de faciliter toutefois la lecture, disons de façon très simplifiée que la procédure centrale est la procédure EVOLUTION. Cette procédure est le point de départ de l'étude de l'état courant de la base de faits. Rappelons que la base de faits est constituée d'un ensemble de prédicats de la forme (présence objet lieu).

Le rôle de la procédure EVOLUTION peut se décomposer en trois phases :

- > La première phase consiste à détecter les points de bouclage du chaînage avant sur des situations déjà rencontrées et complètement explorées, et à entamer directement un retour arrière dans ce cas. La procédure EVOLUTION rend alors la main au niveau qui l'a appelée.
- > La deuxième phase qui intervient si la première phase n'a pas détecté un point d'arrêt et de retour arrière consiste d'abord à construire l'ensemble des règles déclençables et des règles candidates envisageables à partir de l'état courant.

Elle appelle alors la procédure EXAMINER qui va explorer en profondeur d'abord les possibilités de cheminement en chaînage avant de chacune des règles déclençables (appel de la procédure DECLENCHER) puis de chacune des règles candidates (appel de la procédure ANALYSE). Pour ces dernières, l'étude des prémisses manquantes va lancer une étape de chaînage arrière (appel de la procédure DEMONTRER et à travers elle des procédures PROUVER et VERIFIER) dont les résultats vont permettre éventuellement de relancer le chaînage avant.

Lorsque toutes les règles déclenchables et toutes les règles candidates ont été passées en revue, on retourne à la procédure EVOLUTION.

→ La troisième phase étudie les raisons pour lesquelles la procédure EXAMINER lui rend la main. Elles peuvent être de plusieurs types :

1 Si la base de faits courante est constituée de faits qui sont tous des buts alors la procédure EVOLUTION part en retour arrière en rendant la main au niveau qui l'a appelée.

2 Sinon la base de faits courante est constituée de faits qui ne sont pas tous des buts (et parmi lesquels on ne compte pas de buts dans le cas général).

On a alors deux possibilités :

a La main a été rendue à la procédure EVOLUTION en extrémité de phase descendante de chaînage avant et trois cas peuvent se présenter :

- i) L'utilisateur déclare les faits non terminaux de la base de faits comme étant des buts qui n'avaient pas été déclarés. On revient au cas 1.
- ii) L'utilisateur décide d'abandonner pour le moment le chaînage avant à partir de ces faits. Ces faits sont mémorisés comme extrémités non terminales de l'exploration laissées en suspens.
- iii) L'utilisateur décide de poursuivre le chaînage avant à partir de ces faits en introduisant à partir de la base de faits courante de nouvelles règles à examiner à ce niveau (procédure DEMANDER).

b La main a été rendue à la procédure EVOLUTION pour qu'elle poursuive une phase de retour arrière du chaînage avant commencée à des niveaux d'appel récursif de cette procédure plus profonds. On est à ce point en phase de remontée.

Alors l'utilisateur peut décider d'introduire de nouvelles règles à partir de ce point de retour intermédiaire (procédure DEMANDER).

Dans tous les cas, l'évolution se termine par le fait que la procédure EVOLUTION doit rendre la main à son niveau appelant. Le retour arrière est modulé alors suivant les modes d'utilisation de l'analyseur (mise au point, récapitulation) et suivant les exigences de l'utilisateur.

procédure EVOLUTION (base de règles, base de faits)

(on est dans une phase d'examen en chaînage avant de l'enchaînement des règles opératoires)

**Si** la phase d'exploration de l'arborescence du chaînage des règles est descendante (n'est pas en retour-arrière)

et

la base de faits est complètement constituée de faits prémisses de règles déjà utilisées

et

il n'existe pas dans la base de faits des faits prémisses de règles non encore utilisées

**->alors** -Message "Nous retombons ici sur une situation connue"  
-la phase d'exploration repart en retour-arrière vers le niveau appelant de la procédure DECLENCHER

**->sinon** -à partir de la base de règles et de la base de faits courante  
soit ecb l'ensemble des règles déclenchables.  
soit ec l'ensemble de ces règles déclenchables non encore utilisées.  
soit rc l'ensemble des règles candidates (règles dont au moins une prémisses est dans la base de faits) qui n'ont pas encore été utilisées.  
soit indéterminismes l'ensemble des doublets (fait prémisses de règles déclenchables, liste de règles déclenchables (appartenant à ecb) ayant ce fait pour prémisses).

**-EXAMINER** (ec,rc,base de faits)

(on va passer en revue tous les enchaînements possibles à partir de chacune des règles opératoires de ec, puis de rc s'il y a lieu)

**-l'examen est terminé pour ce niveau**

**si** ce dernier a conduit à un ensemble de faits qui sont tous des faits terminaux (buts), c'est-à-dire si l'examen a pu atteindre une situation terminale,

**->alors** la phase d'exploration part en retour-arrière vers le niveau appelant de la procédure DECLENCHER.

**->sinon**

si la phase d'exploration est descendante (n'est pas en retour-arrière)

**->alors** -Message "Considérons les faits de la base de faits qui ne sont pas des buts. Existe-t-il des faits terminaux parmi les faits observés (o/n)"

**-soit** la réponse est "oui",  
dans ce cas

-Message "Lesquels"

-les faits de la base de faits qui ne sont pas déjà des faits terminaux sont passés en revue. L'ensemble des buts est alors augmenté de chaque nouveau fait terminal.

**-si** les conséquents de la dernière règle déclenchée sont des faits terminaux mis à jour

**->alors** la phase d'exploration part en retour-arrière vers le niveau appelant de la procédure DECLENCHER.

**->sinon**

-les faits de la base de faits qui ne sont pas terminaux sont mémorisés comme extrémités non terminales (ce qui permet à l'utilisateur d'arrêter l'exploration du chaînage des opérations dans cette voie s'il ne sait pas comment la continuer pour le moment).

-(tant que l'utilisateur demande à créer de nouvelles règles pour continuer sa description de la gamme, on cherche à faire évoluer la gamme à partir des extrémités non terminales de la base de

faits courante)

= tant que DEMANDER

EVOLUTION (règles non encore utilisées,  
base de faits privée des faits terminaux)

-à la sortie de cette boucle, la phase d'exploration  
part en retour-arrière vers le niveau appelant de  
la procédure DECLENCHER.

-soit la réponse est "non",  
dans ce cas

-les faits non terminaux de la base de faits qui ne sont  
pas déclarés comme nouveaux faits terminaux sont  
mémorisés comme extrémités non terminales de  
l'arborescence.

-tant que DEMANDER (l'utilisateur demande à créer de  
nouvelles règles)

EVOLUTION (règles non encore utilisées,  
base de faits privée des faits terminaux)

->sinon (la phase d'exploration est en cours de retour-arrière)

-si l'utilisateur a demandé le mode mise au point

(vrai quand :1) il crée la description d'une gamme

2) il complète la description d'une gamme)

->alors

-tant que DEMANDER (l'utilisateur demande à  
créer de nouvelles règles)

EVOLUTION (règles non encore utilisées,  
base de faits privée des faits terminaux)

-récapitulation des faits initiaux nouvellement  
créés.

-la phase d'exploration part en retour-arrière  
vers le niveau appelant de la procédure  
DECLENCHER

->sinon (cas où l'utilisateur a demandé la  
récapitulation d'une gamme)

-la phase d'exploration continue son retour-  
arrière.

fin de la procédure.

procédure **EXAMINER** (règles déclenchables,règles candidates,base de faits)

(on va passer en revue tous les enchaînements possibles à partir de chacune des règles opératoires déclenchables,puis à partir de chacune des règles opératoires candidates).

règles déclenchables <- règles déclenchables passées en argument et non encore utilisées.  
règles candidates <- règles candidates passées en argument et non encore utilisées.

Si il n'y a plus de règles déclenchables à ce niveau

->alors -ANALYSE (règles candidates,base de faits)

Les règles candidates (c'est-à-dire dont au moins une prémisse est dans la base de faits) vont être passées en revue. On va chercher à valider les prémisses manquantes grâce à une phase d'exploration en chaînage arrière des enchaînements conduisant à ces prémisses.

->sinon -DECLENCHER (la 1<sup>ère</sup> des règles déclenchables,base de faits)

-S'il y a encore d'autres règles déclenchables ou candidates,afficher le message "Revenons maintenant à une situation précédemment rencontrée".

on affiche alors la base de faits (augmentée éventuellement des nouveaux faits initiaux qui ont été déclarés précédemment à ce niveau d'exploration ou à des niveaux plus profonds et qui ont permis le déclenchement de règles candidates)

-On passe en revue à partir de cette situation de base de faits l'ensemble des règles d'évolution qui ont déjà été déclenchées et on les affiche.

-EXAMINER (la suite de l'ensemble des règles déclenchables,base de faits)

on réitère le processus d'exploration de l'arborescence des enchaînements d'opérations en chaînage avant et en profondeur d'abord pour toutes les règles déclenchables.

**fin de la procédure.**

procédure **ANALYSE** (règles candidates, base de faits)

-Soit il n'y a plus de règles candidates  
dans ce cas

Si les conséquents de la dernière règle déclenchée sont tous des faits terminaux

-> **alors** on est en face d'une situation terminale.

La phase d'exploration effectue un retour-arrière vers le niveau de la  
procédure EVOLUTION appelant.

-> **sinon** la situation n'est pas reconnue comme terminale.

La phase d'exploration effectue un retour-arrière vers le niveau de la  
procédure EVOLUTION appelant.

-Soit la procédure DEMONTRER (les prémisses de la 1<sup>ère</sup> règle candidate qui ne sont pas dans  
la base de faits) arrive à un résultat positif ,c'est-à-dire à la validation des prémisses  
manquantes.

(DEMONTRER effectue une phase d'exploration en chaînage arrière dans l'arborescence des  
règles opératoires qui permettent de conduire aux prémisses de la 1<sup>ère</sup> règle candidate. En  
amenant éventuellement l'utilisateur à définir de nouveaux faits initiaux ou de nouvelles  
règles, DEMONTRER recherche des faits de départ de l'arborescence des chaînages (faits  
initiaux ou faits déjà rencontrés en chaînage avant, faits de rebouclage du chaînage arrière)  
à partir desquels une redescente en chaînage avant peut être effectuée)

-alors ces faits constituent les faits de départ pour la redescente en chaînage avant.

EVOLUTION (ensemble des règles opératoires, faits de départ pour la redescente).

-ANALYSE (la suite de l'ensemble des règles candidates, base de faits).

on réitère le processus de tentative de validation des prémisses manquantes pour les autres  
règles candidates.

-Soit la procédure DEMONTRER ne permet pas de valider les prémisses manquantes de la 1<sup>ère</sup>  
règle candidate

-alors ANALYSE (la suite de l'ensemble des règles candidates, base de faits).

on réitère le processus de tentative de validation des prémisses manquantes pour les autres  
règles candidates.

**fin de la procédure.**

procédure DEMONTRER (liste de prémisses manquantes d'une règle ,règle opératoire dont on veut valider la liste des prémisses manquantes)

(DEMONTRER lance une phase de chaînage arrière).

-On initialise la liste des règles examinées par la phase de chaînage arrière avec la règle dont on veut valider des prémisses manquantes.

-Dans ce contexte,on lance PROUVER (liste de prémisses manquantes de la règle,règle opératoire dont on veut valider la liste des prémisses manquantes).

**fin de la procédure.**

procédure PROUVER (liste de prémisses)

(Le rôle de cette procédure est de déterminer par chaînage arrière si la liste des prémisses est atteignable à partir de faits de situation déjà assurés).

-Soit il n'y a plus de prémisses à prouver.PROUVER retourne VRAI.

-Soit la procédure VERIFIER (la 1<sup>ère</sup> prémisses de la liste,à partir des règles qui ont cette prémisses dans la liste de leurs conséquents) retourne VRAI.Alors le résultat est celui de la procédure PROUVER appliquée au reste de la liste des prémisses.

-Soit la procédure VERIFIER retourne FAUX.Alors PROUVER retourne FAUX.

**fin de la procédure.**

procédure **VERIFIER** (prémisse, à partir d'une liste de règles)

(Le rôle de cette procédure est de déterminer par chaînage arrière si la prémisse est atteignable et de rechercher les faits d'où provient la prémisse).

-Soit la prémisse fait partie des faits initiaux, alors on ajoute cette prémisse à la liste des faits qui vont servir de départ à la redescente en chaînage avant. VERIFIER retourne VRAI.

-Soit la liste des règles est vide,

Ceci peut se produire dans 2 cas:

1) Si aucune règle de la gamme ne possède la prémisse dans ses conséquents, alors on est arrivé en chaînage arrière à des prémisses extrémités qui ne sont pas déclarées comme faits initiaux. Il est alors nécessaire d'INTERROGER l'utilisateur pour savoir si la prémisse est atteignable et pour éclaircir alors sa provenance en introduisant éventuellement de nouvelles règles opératoires ou en la déclarant comme nouveau fait initial).

2) Toutes les règles ont été passées en revue. Aucune n'a pu permettre de vérifier que la prémisse est atteignable par un chemin de règles. VERIFIER retourne FAUX.

-Soit la 1<sup>ère</sup> règle de la liste fait partie des règles déjà utilisées (en chaînage avant) ou déjà examinées (en chaînage arrière).

-Si il ne subsiste pas de règle qui possède la prémisse dans la liste de ses conséquents parmi les règles non encore utilisées et non encore examinées, alors on ajoute la prémisse à la liste des faits de départ pour la redescente en chaînage avant.

-VERIFIER retourne VRAI. On applique alors VERIFIER aux autres règles pour voir si la prémisse peut avoir d'autres origines que celles qui ont déjà été trouvées.

-Soit on n'est dans aucune des situations précédentes.

On place la 1<sup>ère</sup> règle de la liste dans la liste des règles déjà examinées en chaînage arrière.

Si la procédure PROUVER (les prémisses de la règles) retourne VRAI

->alors VERIFIER retourne VRAI. On applique alors VERIFIER aux autres règles pour voir si la prémisse peut avoir d'autres origines que celles qui ont déjà été trouvées.

->sinon VERIFIER retourne le résultat de VERIFIER appliquée récursivement à la suite de la liste de règles. On essaie de déterminer la provenance de la prémisse à partir des autres règles.

fin de la procédure.

procédure **DECLENCHER** (une règle, à partir d'une base de faits)

-La liste des indéterminismes détectés dans la procédure EVOLUTION appelante (via la procédure EXAMINER) est constituée d'une liste de doublets qui ont chacun la forme suivante : (prédictat, liste de règle déclenchables ayant ce prédicat dans leurs prémisses). Parmi ces indéterminismes, on sélectionne les indéterminismes relatifs à la règle à déclencher, c'est-à-dire dont le prédicat fait partie des prémisses de la règle.

Le logiciel affiche ces indéterminismes par l'intermédiaire d'un message.

-base de faits <- base de faits U {conséquents de la règle} - {prémisses de la règle}

-règles inutilisées <- règles inutilisées - {règle}

-dernière règle déclenchée <- règle

-descente <- VRAI (on est bien en phase de descente en chaînage avant).

-faits initiaux non utilisés <- faits initiaux non utilisés - {prémisses de la règle}

-extrémités non terminales <- extrémités non terminales - {prémisses de la règle}

-int <- {conséquents de la règle}  $\cap$  {buts}

si  $\text{int} \neq \emptyset$  alors: -Un message est affiché indiquant qu'un ou plusieurs buts ont été atteints.

-Si, en outre, on est en mode mise au point (c'est-à-dire si on crée, ou si on complète la description d'une gamme,  $\neq$  récapitulation) et si tous les faits de la base de faits ne sont pas des buts, alors le logiciel demande à l'utilisateur s'il n'y a pas parmi ces faits des faits qu'il faudrait déclarer comme faits terminaux. Si la réponse est affirmative, ces faits sont mémorisés comme faits terminaux atteints.

-La procédure EVOLUTION (appliquée à la base de règles, et à l'ensemble des faits de la base de faits qui ne sont pas des buts) est alors appelée récursivement pour poursuivre l'exploration.

**fin de la procédure.**

### II.3.2. Compléments de l'analyse principale

La procédure EVOLUTION constitue le noyau essentiel de l'analyseur qui est appelé par les procédures destinées à créer, compléter (ou modifier) et récapituler les descriptions sous forme de règles de chacune des gammes opératoires.

Comme on l'a déjà mentionné au II.1, il peut apparaître à la fin de la phase d'analyse des règles et des faits initiaux et terminaux qui s'avèrent être inutiles pour la description qui a été mise au point, et qui ne peuvent lui être rattachés par chaînage logique. Une étape de complément va alors intervenir et interroger de façon intelligente l'utilisateur sur le sort à réserver à ces données qui sont en trop. La technique va être de tenter de relier les règles inutilisées au reste de la description en lançant un chaînage arrière (procédure DEMONTRER) sur les prémisses de chacune des règles non encore utilisées. Ceci est illustré au II.2. par les figures II.57 n<sup>os</sup> 39 à 45.

Les règles et les faits qui n'auront pas été rattachés, à la suite de cette dernière étape, seront mémorisés comme information mise de côté (dans une zone "débarras"), ou simplement éliminés. A l'initialisation d'une étape d'analyse portant sur une gamme existante, le logiciel rappelle toujours le contenu du "débarras" et autorise l'utilisateur à y puiser des données (règles, faits initiaux/terminaux) dont il aurait besoin.

#### II.4 REMARQUE SUR LE PROBLEME DU CONTROLE DE COHERENCE DANS LES BASES DE CONNAISSANCES

Les Journées Nationales sur l'Intelligence Artificielle organisées par le PRC-GRECO Intelligence Artificielle (\*) à Aix-les-Bains (SAVOIE) en Novembre 1986 ont permis de faire le point sur l'état de l'art de la Recherche en Intelligence Artificielle. En particulier, dans le pôle concernant la Méthodologie des Applications, un bilan a été fait à propos du contrôle de cohérence dans les bases de connaissances par Marc AYE (LIA CHAMBERY) Eric PIPARD et Marie-Christine ROUSSET (LRI ORSAY) |AYE 86|. De très rares études ont été menées sur le sujet. elles montrent l'ambition du projet et la difficulté d'obtenir des résultats opérationnels. Ces études se sont essentiellement focalisées sur la technique des systèmes experts. Au niveau national, sur le plan théorique, deux laboratoires du PRC collaborent, avec des objectifs différents, à la définition et au contrôle de la cohérence : le LRI d'ORSAY et le LIA de CHAMBERY. En dehors du PRC, sont également cités les efforts de COGNITECH avec le logiciel TG-2, de Pierre LE BEUX et Dominique FONTAINE (Université Technologique de COMPIEGNE) |LE 86| et les possibilités offertes par des langages objets comme SRL. Dans le cas général, les vérifications de cohérence souhaitées sont ambitieuses. Elles ont pour objectif de porter à la fois sur la syntaxe, la logique et la sémantique des bases de connaissances. Le traitement du langage naturel donne un exemple de la complexité du problème abordé. La puissance déductive des règles rend en pratique le problème de la détection d'incohérences insoluble dans toute sa généralité. C'est pour cette raison que beaucoup de travaux se sont limités à la logique des propositions, la logique des prédicats d'ordre 1 et plus introduisant une puissance déductive encore beaucoup plus grande dans le cas général.

Ici, l'analyse de cohérence porte essentiellement sur la logique d'enchaînement des règles, et dans une certaine mesure sur la syntaxe des règles.

NB :

(\*) Le Ministère de la Recherche et le Centre National de la Recherche Scientifique ont mis en place en 1985 un programme de recherches concertées (PRC) en Intelligence Artificielle. Celui-ci a été créé sous la forme d'un Groupement de Recherches Concertées (GRECO) du CNRS. Le PRC Intelligence Artificielle est une action nationale de recherche qui s'est fixé pour objectifs de structurer et renforcer le potentiel de recherche français en Intelligence Artificielle et de constituer un repère scientifique et technologique dans ce domaine. Les actes des "Journées Nationales sur l'I.A." sont disponibles aux Editions CEPADUES (TOULOUSE).

Pour l'instant, seule la logique des propositions a été abordée de façon complète. La plupart des exemples industriels peuvent être décrits par des règles propositionnelles. Toutefois, l'introduction de prédicats d'ordre 1 n'est pas impossible a priori, puisque le champ des déductions reste limité en contexte industriel et que l'on effectue des vérifications de cohérence à partir de bases de faits données. Les exemples où les prédicats seraient d'utilisation intéressante sont des exemples où l'utilisateur est tenté de donner une description en compréhension et non plus en extension des opérations.

Il est possible de fournir à l'utilisateur une interface de description des règles qui permette certaines descriptions en compréhension plutôt qu'en extension, et de traduire ensuite sa description en extension de façon transparente pour effectuer l'analyse de cohérence. C'est ce qui est proposé pour le moment et qui se révèle satisfaisant pour les exemples industriels courants que nous traitons actuellement. Nous reparlerons des problèmes liés aux réseaux à prédicats au Chapitre III Partie II § 6 et en Annexe 3.

### III.

#### LA TRADUCTION ET L'AGREGATION DES REGLES OPERATOIRES SOUS FORME D'UN GRAPHE RDP COLORE : LE PREGRAPHE.

La méthode de description des gammes à l'aide des règles opératoires associées aux actions élémentaires est réitérée pour chacune des gammes prévues.

On étend la notion de gamme à la notion de recensement de l'ensemble des cheminements élémentaires devant être envisagés. Les règles opératoires répertoriées autour de chacune de ces gammes ou pôles de fabrication vont être réunies pour former l'ensemble des règles opératoires que l'atelier devra supporter. L'ensemble de ces règles va constituer la définition du rôle fonctionnel de l'unité de production.

Toutefois, cette description précise risque de s'avérer encombrante. Il vient donc naturellement à l'esprit l'idée de l'agréger sous une forme plus concise.

#### III.1 CONDITIONS D'AGREGATION

La condition nécessaire et suffisante pour agréger deux règles est qu'elles possèdent la même structure (mêmes sites prémisses et mêmes sites conséquents), que les informations complémentaires significatives associées (concernant entre autres leur classification de type) soient identiques et enfin que le dispositif technologique matériel qui réalisera ces actions soit le même.

#### Exemples :

- 1) Considérons les deux règles ci-dessous

#### Règle R1

#### Représentation littérale

prémisses : (présence a1 A) (présence b B)

conséquents (présence C1 C)

type : positionnelle

détail : assemblage

#### traduction RdP

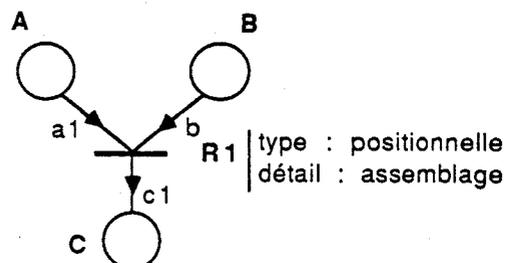


Figure II.58

**Règle R2**

Représentation littérale

prémises : (présence a2 A) (présence b B)

conséquents : (présence c2 C)

type : positionnelle

détail : assemblage

traduction RdP

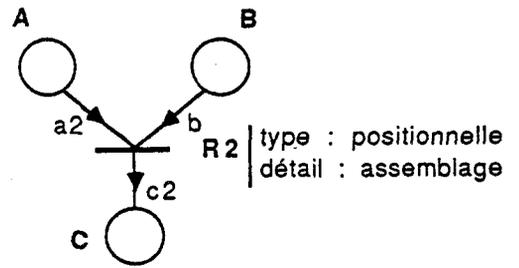


Figure II.59

Le formalisme retenu pour l'analyse de cohérence et de complétude utilise une description des règles sous forme de prédicats plus adaptée aux inférences logiques. Ce formalisme va devoir être traduit avec la phase d'agrégation dans un autre formalisme qui constitue la convention de description littérale des modèles RdP colorés utilisée dans toutes les phases ultérieures de la chaîne de C.A.O. et qui va assurer la compatibilité d'interface de tous les logiciels de la chaîne.

Les règles R1 et R2 vont alors pouvoir être traduites sous forme RdP coloré, et agrégées si elles utilisent le même dispositif technologique (ex. : bras robot) pour former la transition RdPC R ci-dessous :

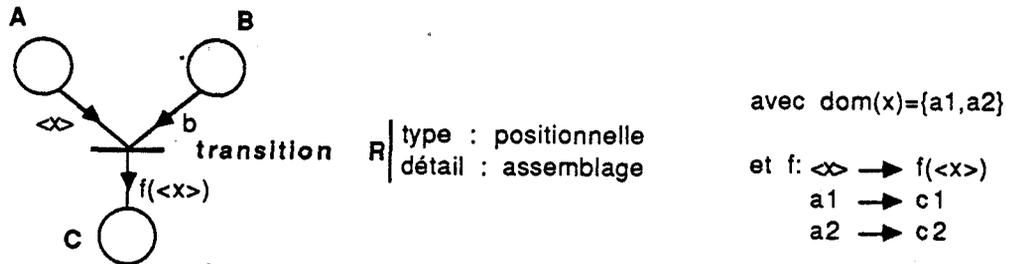


Figure II.60

- 2) Par contre, les deux règles ci-dessous ne pourront être agrégées parce qu'elles sont de types différents.

**Règle tournage**

Représentation littérale

prémises (présence t- T)

conséquents (présence t+ T)

type : transformationnelle

détail : usinage

traduction RdP

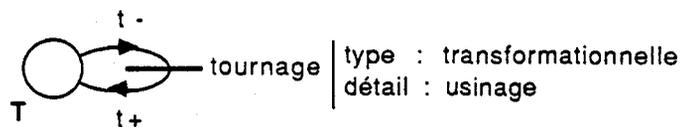


Figure II.61

et

**Règle retournement**

**Représentation littérale**

prémisses (présence r- T)

conséquents (présence r+ T)

type : positionnelle

détail : repositionnement

**traduction RdP**



Figure II.62

**III.2 METHODE D'AGREGATION**

La phase de traduction et d'agrégation est complètement assurée par un logiciel écrit en langage Le\_Lisp et dont la taille est d'environ 30 K Octets.

Montrons brièvement le type de résultats auxquels il parvient :

Prenons quatre règles R1, R2, R3 et R4 pouvant être agrégées.

La structure de ces quatre règles est donnée ci-dessous.

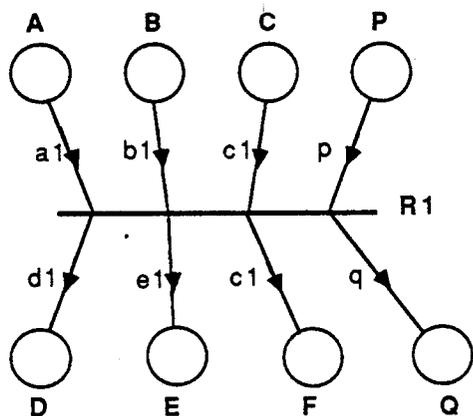


Figure II.63 n° 1

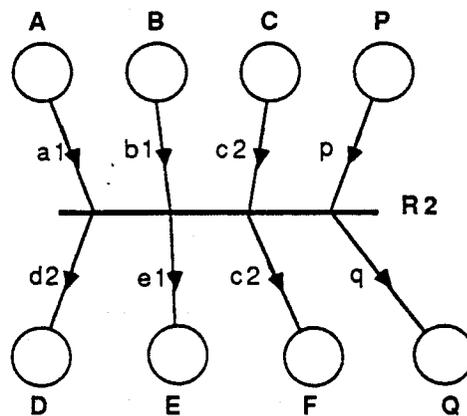


Figure II.63 n° 2

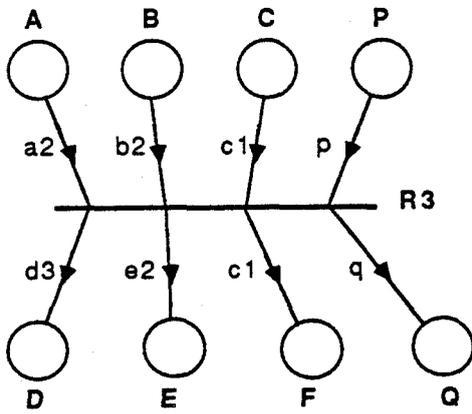


Figure II.63 n° 3

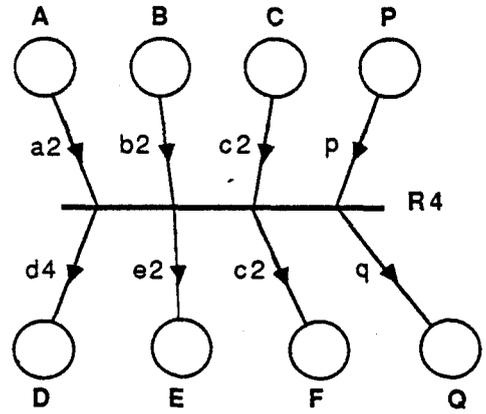
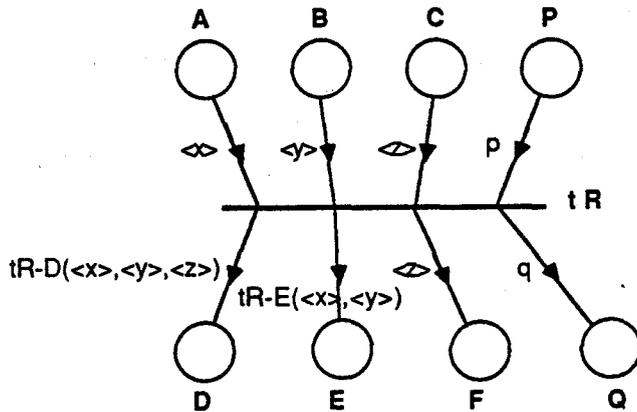


Figure II.63 n° 4

L'agrégation des quatre règles conduit à la transition  $tR$  suivante :



où  $x$ ,  $y$  et  $z$  sont des variables libres prenant leur valeurs respectivement dans les ensembles :  
 $\text{dom}(\langle x \rangle) = \{a1, a2\}$   
 $\text{dom}(\langle y \rangle) = \{b1, b2\}$   
 $\text{dom}(\langle z \rangle) = \{c1, c2\}$

Figure II.64

Les fonctions  $tR-D$  et  $tR-E$  sont générées automatiquement par le logiciel

$tR-D : (x,y,z) \rightarrow tr-D(x,y,z)$       et  $tR-E : (x,y) \rightarrow tR-E(x,y)$   
 $(a1,b1,c1) \rightarrow d1$        $(a1,b1) \rightarrow e1$   
 $(a1,b1,c2) \rightarrow d2$        $(a2,b2) \rightarrow e2$   
 $(a2,b2,c1) \rightarrow d3$   
 $(a2,b2,c2) \rightarrow d4$

S'il n'existe qu'un arc étiqueté par une fonction, cette fonction prend le nom  $tR$  de la transition (ex. :  $tR-D$  deviendrait  $tR$  en l'absence de l'arc  $tR \rightarrow E$ ). La transition  $tR$  se comporte comme une fonction vectorielle.

Les fonctions tR-D et tR-E sont écrites automatiquement en Le\_Lisp, et seront utilisées dans les phases ultérieures de la chaîne de C.A.O. comme les autres fonctions Lisp de la chaîne logicielle.

Le langage Le\_Lisp a la propriété très intéressante de ne faire a priori aucune différence entre données et code exécutable. Il est donc possible à un programme écrit en Le\_Lisp de générer certaines de ses propres procédures. Cette souplesse de programmation fait du LISP un outil de choix pour les techniques de manipulation symbolique et donc pour l'Intelligence Artificielle. C'est une des raisons qui nous a amené à préférer ce langage pour l'application décrite dans ce mémoire.

### III.3 UN EXEMPLE D'UTILISATION

Nous avons déjà regardé dans la partie II de ce chapitre la description d'une gamme t-l-m-/t+l+m+. Les figures qui suivent (figures II.65 n<sup>os</sup> 1 à 6) présentent, outre la gamme t-l-m-/t+l+m+, d'autres gammes prévues pour l'unité de production :

- f-l-m-/f+l+m+ pour des pièces devant subir
  - une séquence de fraisage f
  - un lavage l
  - une série de mesures m
  
- t-f-l-m-/t+f+l+m+ pour des pièces devant subir
  - une séquence de tournage de type t
  - une séquence de faisage de type f
  - un lavage l
  - une série de mesures m.

Notons ici que les opérations t et f sont permutable. Elles peuvent être exécutées dans un ordre indifférent en fonction de critères divers tels que l'encombrement des machines.

- t1-r-t2-l-m-/t1+r+t2+l+m+ pour des pièces devant subir
  - une séquence de tournage de type t1
  - une séquence de tournage de type t2
  - un retournement entre t1 et t2
  - un lavage l
  - une série de mesures m

- t1-f-t2-l-m-/t1+f+t2+l+m+ pour des pièces devant subir
  - une séquence de tournage de type t1
  - une séquence de fraisage de type f
  - une séquence de tournage de type t2
  - un lavage l
  - une série de mesures m.

Enfin, deux nouveaux lieux vont intervenir pour ces autres gammes. Il s'agit d'un centre d'usinage CU et d'un poste de bridage-débridage PBD destiné à brider les pièces à fraiser sur des platines p (supports d'usinage) avant leur entrée dans le centre d'usinage.

La traduction et l'agrégation des six gammes opératoires conduit au graphe Réseau de PETRI coloré qui est représenté sur les figures II.66 et qu'on appelle PREGRAPHE. Le PREGRAPHE constitue une description agrégée sous forme d'un graphe RdP coloré du rôle fonctionnel de l'unité de production. Il récapitule l'ensemble des cheminements élémentaires possibles et prévus pour les objets circulant dans l'atelier.

L'objectif essentiel atteint par l'élaboration du PREGRAPHE est de fournir une base de travail très propre (à la fois cohérente, complète et concise) pour la construction ultérieure structurée et modulaire du Réseau de PETRI structuré, adaptatif et coloré qui constitue le modèle de la Partie Commande (niveau bas du système de commande). Nous reparlerons dans le chapitre III de cette phase appelée DEVELOPPEMENT STRUCTURE ou, par abus de langage, STRUCTURATION.

#### III.4 LES VARIABLES LIBRES LOCALES DU PREGRAPHE

Nous avons vu que les règles opératoires sont traduites et éventuellement agrégées pour former des transitions du PREGRAPHE. L'agrégation va provoquer l'apparition de variables libres associées aux arcs amont et aval de chaque transition. Le domaine de variation de chacune de ces variables libres est constitué par l'ensemble des types d'objets pouvant circuler suivant l'arc étiqueté par cette variable libre.

Bien que ces variables puissent théoriquement se retrouver sur plusieurs transitions, nous avons choisi d'imposer des dénominations différentes pour chacune de ces transitions. Nous disons alors que les variables libres sont locales.

La dénomination plus globale aurait permis de rendre plus visuels certains cheminements sur le PREGRAPHE. Mais la dénomination locale donne une plus grande flexibilité de reconfiguration des modèles et en outre de leur implantation sur automates.

En effet, les variables libres obtenues sur le PREGRAPHE et les fonctions construites à partir de ces variables seront celles du modèle RdP développé de la Partie commande et en outre, pratiquement, celles qui seront utilisées pour son implantation. Ainsi, supposons qu'une nouvelle gamme doive être réalisée par l'unité de production. Si cette gamme utilise des chemins élémentaires déjà mentionnés sur le PREGRAPHE, ce qui est le cas si la conception a été faite à longue échéance, la structure du PREGRAPHE et du modèle de la partie commande ne sera pas modifiée. Seuls certains domaines de couleur relatifs aux variables libres des chemins empruntés par la nouvelle gamme devront être complétés.

SEQUENCE OPERATOIRE t-l-m-/t+l+m+

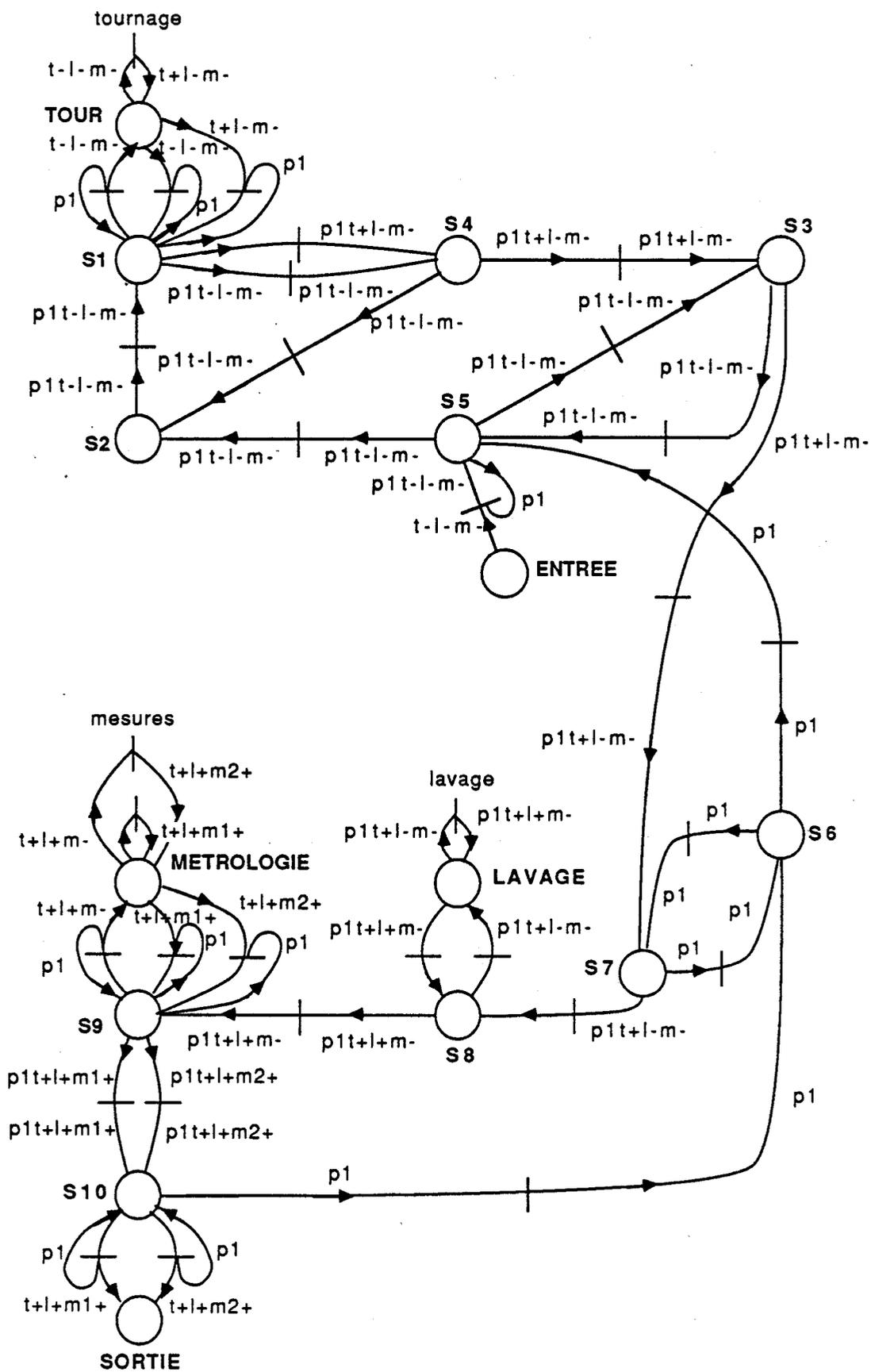


Figure II.65 n°1

SEQUENCE OPERATOIRE t1-r-t2-l-m-/t1+r+t2+l+m+

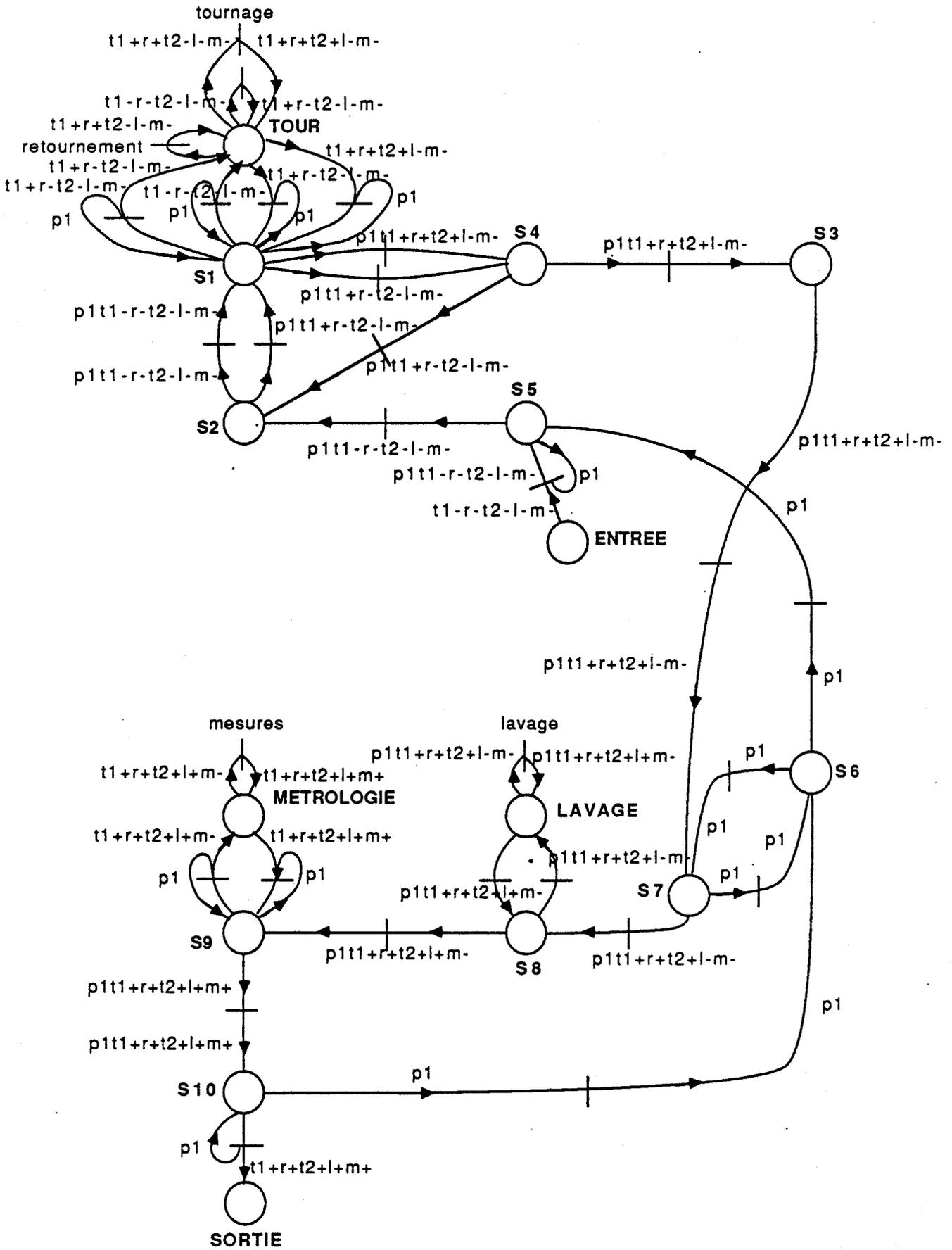


Figure II.65 n°2

# SEQUENCE OPERATOIRE t-f-l-m-/t+f+l+m+

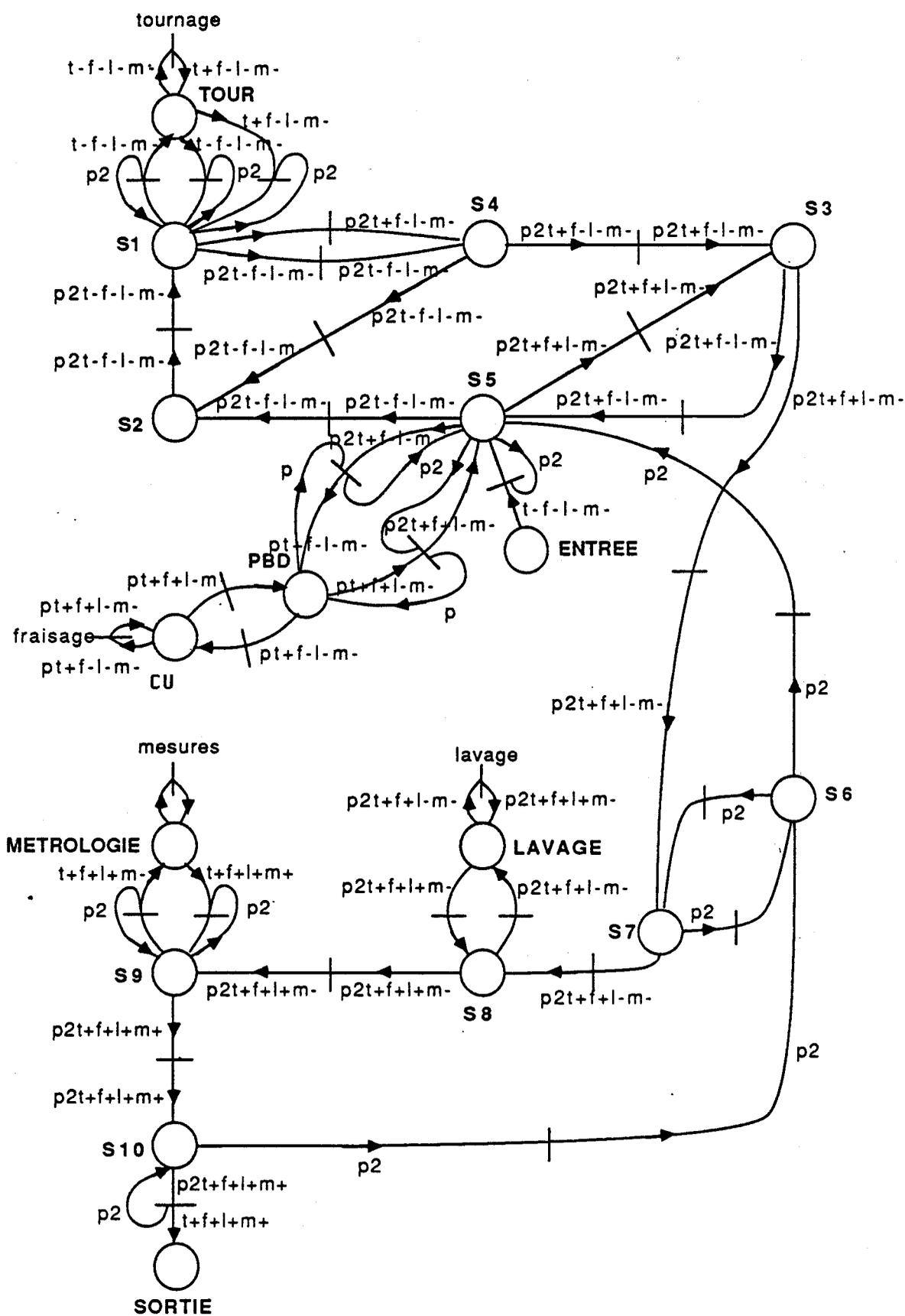


Figure II.65 n°3

**SEQUENCE OPERATOIRE t-f-l-m-/t+f+l+m+**  
(suite)

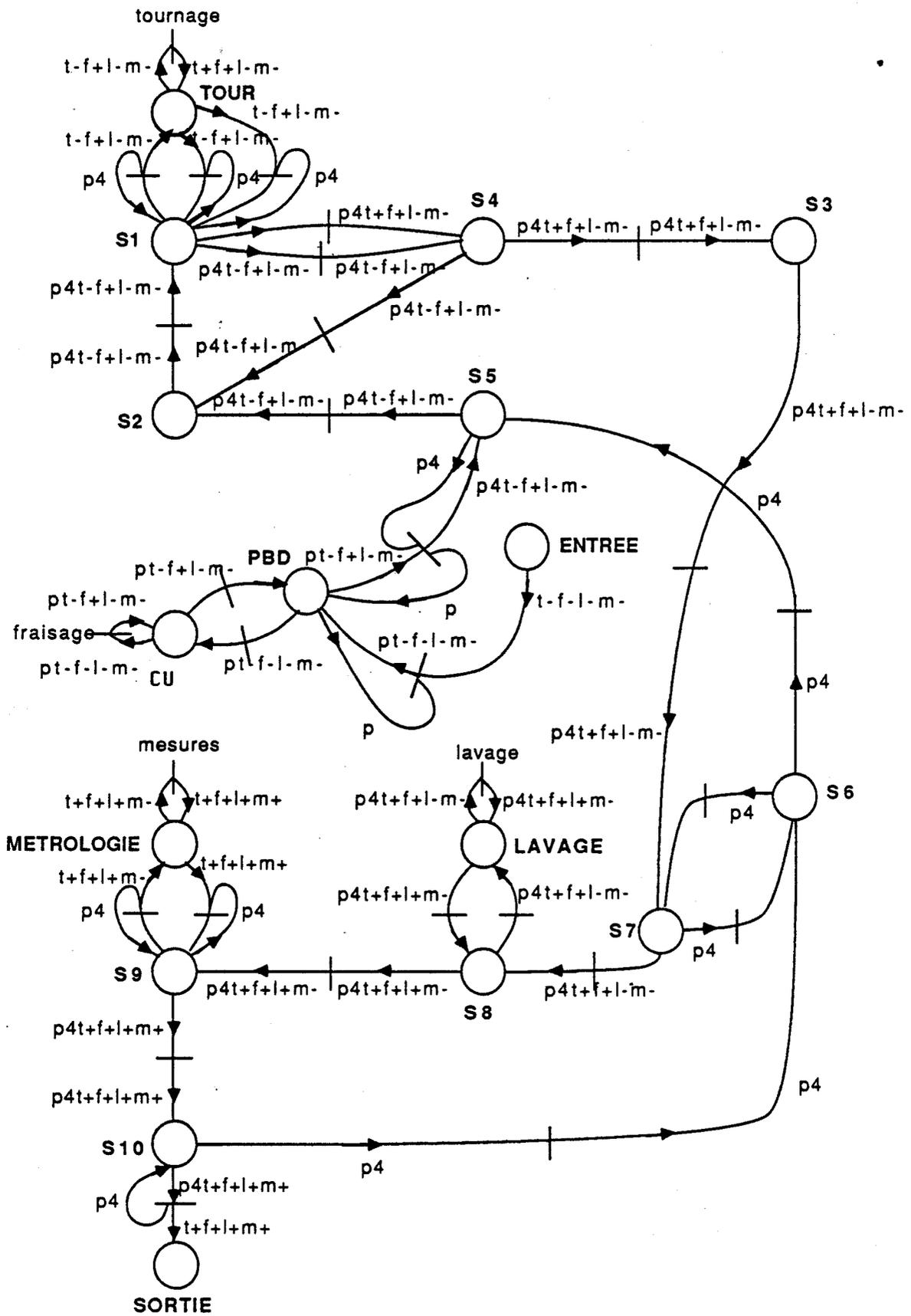


Figure II.65 n°4

SEQUENCE OPERATOIRE f-l-m-/f+l+m+

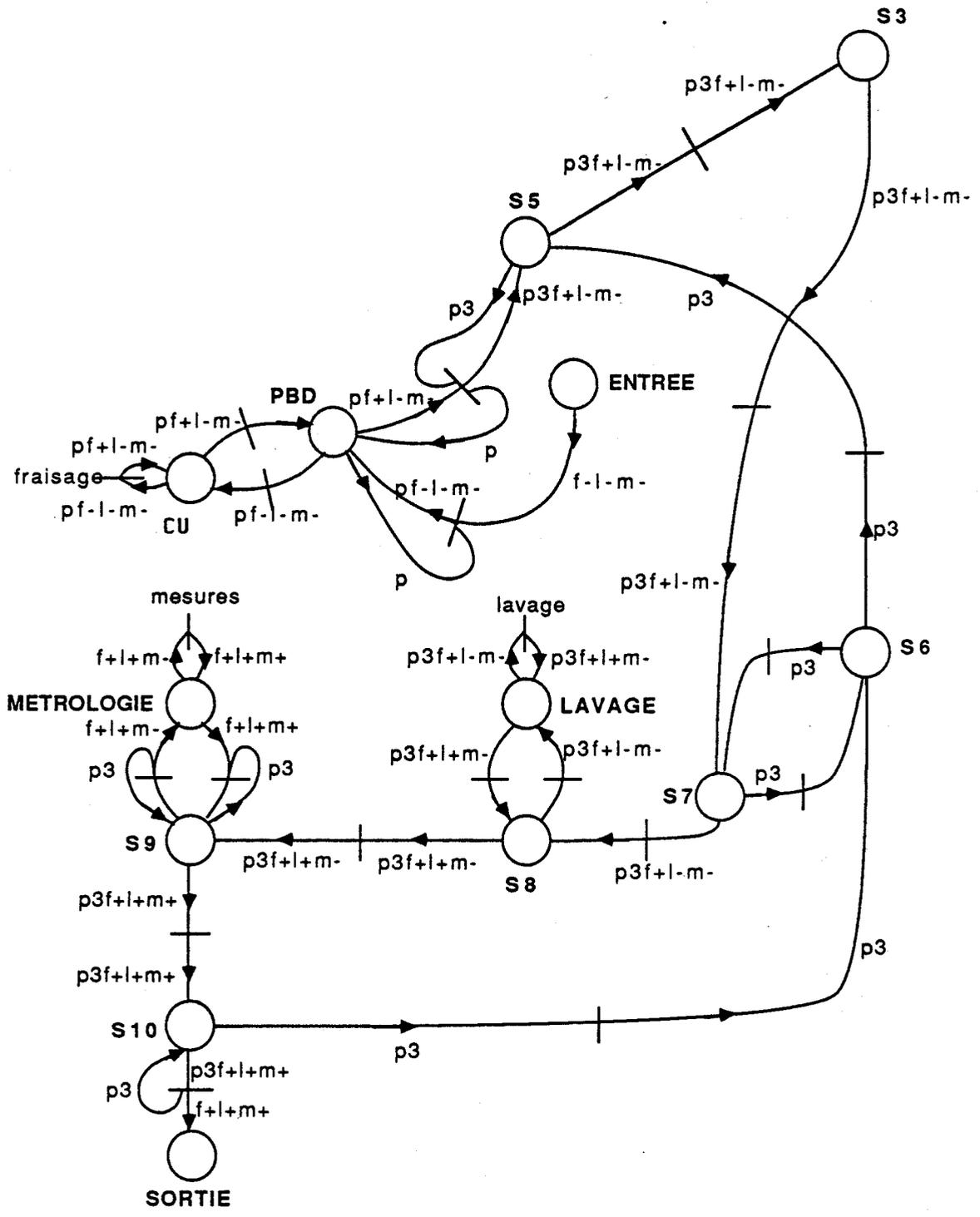


Figure II.65 n°5

SEQUENCE OPERATOIRE t1-f-t2-l-m-/t1+f+t2+l+m+

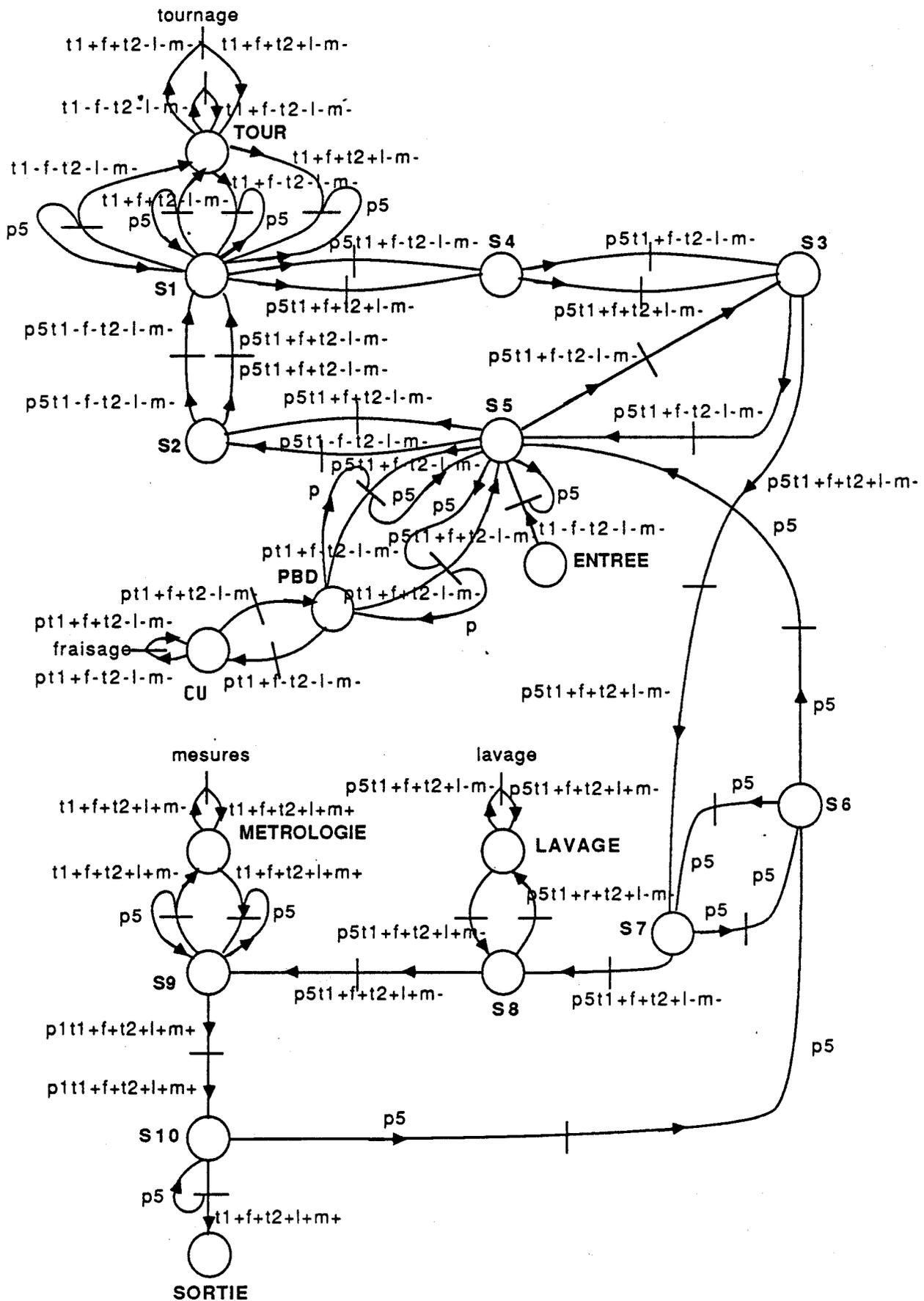


Figure II.65 n°6

# PREGRAPHE

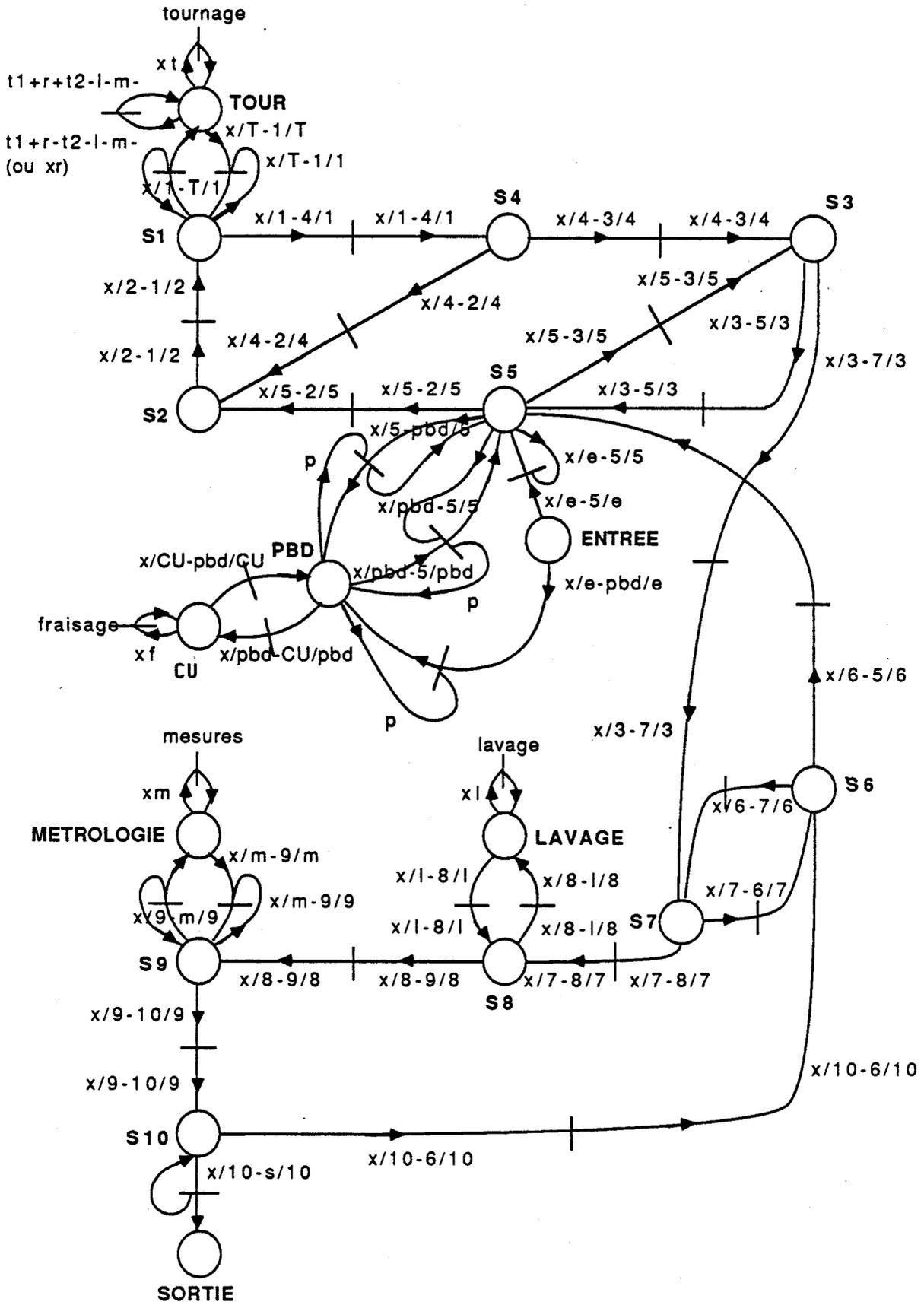


Figure II.66

**Figure II.66 : DOMAINES DE COULEURS DES VARIABLES LIBRES  
FONCTIONS RELATIVES AUX CHANGEMENTS DE COULEURS**

$$\text{dom}(x/e-5/e) = \{t-l-m-, t1-r-t2-l-m-, t-f-l-m-, t1-f-t2-l-m-\}$$

$$\text{dom}(x/e-5/5) = \{p1, p2, p5\}$$

entrée->s5 :	(x/e-5/e, x/e-5/5)	entrée->s5(x/e-5/e, x/e-5/5)
	(t-l-m-, p1)	p1t-l-m-
	(t1-r-t2-l-m-, p1)	p1t1-r-t2-l-m-
	(t-f-l-m-, p2)	p2t-f-l-m-
	(t1-f-t2-l-m-, p5)	p5t1-f-t2-l-m-

$$\text{dom}(x/e-pbd/e) = \{t-f-l-m-, f-l-m-\}$$

entrée->pbd :	x/e-pbd/e	entrée->pbd(x/e-pbd/e)
	t-f-l-m-	pt-f-l-m-
	f-l-m-	pf-l-m-

$$\text{dom}(x/5-pbd/5) = \{p2t+f-l-m-, p5t1+f-t2-l-m-\}$$

s5->pbd-pbd :	x/5-pbd/5	s5->pbd-pbd(x/5-pbd/5)
	p2t+f-l-m-	pt+f-l-m-
	p5t1+f-t2-l-m-	pt1+f-t2-l-m-

s5->pbd-s5 :	x/5-pbd/5	s5->pbd-s5(x/5-pbd/5)
	p2t+f-l-m-	p2
	p5t1+f-t2-l-m-	p5

$$\text{dom}(x/pbd-5/pbd) = \{pt+f+l-m-, pt-f+l-m-, pf+l-m-, pt1+f+t2-l-m-\}$$

$$\text{dom}(x/pbd-5/5) = \{p2, p4, p3, p5\}$$

pbd->s5-s5 :	(x/pbd-5/pbd, x/pbd-5/5)	pbd->s5-s5 (x/pbd-5/pbd, x/pbd-5/5)
	(pt+f+l-m-, p2)	p2t+f+l-m-
	(pt-f+l-m-, p4)	p4t-f+l-m-
	(pf+l-m-, p3)	p3f+l-m-
	(pt1+f+t2-l-m-, p5)	p5t1+f+t2-l-m-

$$\text{dom}(x/pbd-CU/pbd) = \{pt+f-l-m-, pt-f-l-m-, pf-l-m-, pt1+f-t2-l-m-\}$$

$$\text{dom}(x/CU-pbd/CU) = \{pt+f+l-m-, pt-f+l-m-, pf+l-m-, pt1+f+t2-l-m-\}$$

$$\text{dom}(x/5-2/5) = \{p1t-l-m-, p1t1-r-t2-l-m-, p2t-f-l-m-, p4t-f+l-m-, p5t1-f-t2-l-m-, p5t1+f+t2-l-m-\}$$

dom (x/2-1/2) = {p1t-l-m-,p1t1-r-t2-l-m-,p1t1+r-t2-l-m-,p2t-f-l-m-,  
p4t-f+l-m-,p5t1-f-t2-l-m-, p5t1+f+t2-l-m-}

dom (x/1-T/1) = {p1t-l-m-,p1t1-r-t2-l-m-,p1t1+r-t2-l-m-,p2t-f-l-m-,  
p4t-f+l-m-,p5t1-f-t2-l-m-, p5t1+f+t2-l-m-}

s1->T-T :	x/1-T/1	s1->T-T (x/1-T/1)
	p1t-l-m-	t-l-m-
	p1t1-r-t2-l-m-	t1-r-t2-l-m-
	p1t1+r-t2-l-m-	t1+r-t2-l-m-
	p2t-f-l-m-	t-f-l-m-
	p4t-f+l-m-	t-f+l-m-
	p5t1-f-t2-l-m-	t1-f-t2-l-m-
	p5t1+f+t2-l-m-	t1+f+t2-l-m-

s1->T-s1:	x/1-T/1	s1->T-s1 (x/1-T/1)
	p1t-l-m-	p1
	p1t1-r-t2-l-m-	p1
	p1t1+r-t2-l-m-	p1
	p2t-f-l-m-	p2
	p4t-f+l-m-	p4
	p5t1-f-t2-l-m-	p5
	p5t1+f+t2-l-m-	p5

dom (x/T-1/T) = {t-l-m-,t+l-m-,t1+r-t2-l-m-,t1+r+t2+l-m-,t-f-l-m-,  
t+f-l-m-,t-f+l-m-,t+f+l-m-,t1+f-t2-l-m-,t1+f+t2+l-m-}

dom (x/T-1/1) = {p1,p2,p4,p5}

s1->T-s1:	(x/T-1/T,x/T-1/1)	s1->T-s1(x/T-1/T,x/T-1/1)
	(t-l-m-,p1)	p1t-l-m-
	(t+l-m-,p1)	p1t+l-m-
	(t1+r-t2-l-m-,p1)	p1t1+r-t2-l-m-
	(t1+r+t2+l-m-,p1)	p1t1+r+t2+l-m-
	(t-f-l-m-,p2)	p2t-f-l-m-
	(t+f-l-m-,p2)	p2t+f-l-m-
	(t-f+l-m-,p4)	p4t-f+l-m-
	(t+f+l-m-,p4)	p4t+f+l-m-
	(t1+f-t2-l-m-,p5)	p5 t1+f-t2-l-m-
	(t1+f+t2+l-m-,p5)	p5t1+f+t2+l-m-

dom (x/1-4/1) = {p1t-l-m-,p1t+l-m-,p1t1+r-t2-l-m-,p1t1+r+t2+l-m-,  
p2t-f-l-m-,p2t+f-l-m-,p4t-f+l-m-,p4t+f+l-m-,  
p5t1+f-t2-l-m-,p5t1+f+t2+l-m-}

dom (x/4-2/4) = {p1t-l-m-,p1t1+r-t2-l-m-,p2t-f-l-m-,p4t-f+l-m-}

dom (x/4-3/4) = {p1t+l-m-,p1t1+r+t2+l-m-,p2t+f-l-m-,p4t+f+l-m-,  
p5t1+f-t2-l-m-,p5t1+f+t2+l-m-}

dom (x/3-5/3) = {p1t-l-m-,p2t+f-l-m-,p5t1+f-t2-l-m-}

dom (x/5-3/5) = {p1t-l-m-,p2t+f+l-m-,p3f+l-m-,p5t1+f-t2-l-m-}

dom (x/3-7/3) = {p1t+l-m-,p1t1+r+t2+l-m-,p2t+f+l-m-,p4t+f+l-m-,p3f+l-m-,  
p5t1+f+t2+l-m-}

dom (x/7-8/7) = {p1t+l-m-,p1t1+r+t2+l-m-,p2t+f+l-m-,p4t+f+l-m-,p3f+l-m-,  
p5t1+f+t2+l-m-}

dom (x/8-l/8) = {p1t+l-m-,p1t1+r+t2+l-m-,p2t+f+l-m-,p4t+f+l-m-,p3f+l-m-,  
p5t1+f+t2+l-m-}

dom (x/l-8/l) = {p1t+l+m-,p1t1+r+t2+l+m-,p2t+f+l+m-,p4t+f+l+m-,p3f+l+m-,  
p5t1+f+t2+l+m-}

dom (x/8-9/8) = {p1t+l+m-,p1t1+r+t2+l+m-,p2t+f+l+m-,p4t+f+l+m-,p3f+l+m-,  
p5t1+f+t2+l+m-}

dom (x/9-m/9) = {p1t+l+m-,p1t1+r+t2+l+m-,p2t+f+l+m-,p4t+f+l+m-,  
p3f+l+m-,p5t1+f+t2+l+m-}

s9->metrologie-metrologie : x/9-m/9 s9->metrologie-metrologie (x/9-m/9)

p1t+l+m-	t+l+m-
p1t1+r+t2+l+m-	t1+r+t2+l+m-
p2t+f+l+m-	t+f+l+m-
p4t+f+l+m-	t+f+l+m-
p3f+l+m-	f+l+m-
p5t1+f+t2+l+m-	t1+f+t2+l+m-

s9->metrologie-s9 : x/9-m/9 s9->metrologie-s9(x/9-m/9)

p1t+l+m-	p1
p1t1+r+t2+l+m-	p1
p2t+f+l+m-	p2
p4t+f+l+m-	p4
p3f+l+m-	p3
p5t1+f+t2+l+m-	p5

dom (x/m-9/m) = {t+l+m1+, t+l+m2+, t1+r+t2+l+m+, t+f+l+m+, f+l+m+,  
t1+f+t2+l+m+}

dom (x/m-9/9) = {p1,p2,p4,p3,p5}

metrologie->s9 :	(x/m-9/m, x/m-9/9)	metrologie->s9 (x/m-9/m, x/m-9/9)
	(t+l+m1+,p1)	p1t+l+m1+
	(t+l+m2+,p1)	p1t+l+m2+
	(t1+r+t2+l+m+,p1)	p1t1+r+t2+l+m+
	(t+f+l+m+,p2)	p2t+f+l+m+
	(t+f+l+m+,p4)	p4t+f+l+m+
	(f+l+m+,p3)	p3f+l+m+
	(t1+f+t2+l+m+,p5)	p5t1+f+t2+l+m+

dom (x/9-10/9) = {p1t+l+m1+, p1t+l+m2+,p1t1+r+t2+l+m+,p2t+f+l+m+,  
p4t+f+l+m+,p3f+l+m+,p1t1+f+t2+l+m+}

dom (x/10-s/10) = {p1t+l+m1+, p1t+l+m2+,p1t1+r+t2+l+m+,p2t+f+l+m+,  
p4t+f+l+m+,p3f+l+m+,p1t1+f+t2+l+m+}

s10->sortie-sortie :	x/10-s/10	s10->sortie-sortie(x/10-s/10)
	p1t+l+m1+	t+l+m1+
	p1t+l+m2+	t+l+m2+
	p1t1+r+t2+l+m+	t1+r+t2+l+m+
	p2t+f+l+m+	t+f+l+m+
	p4t+f+l+m+	t+f+l+m+
	p3f+l+m+	f+l+m+
	p1t1+f+t2+l+m+	t1+f+t2+l+m+

s10->sortie-s10:	x/10-s/10	s10->sortie-s10(x/10-s/10)
	p1t+l+m1+	p1
	p1t+l+m2+	p1
	p1t1+r+t2+l+m+	p1
	p2t+f+l+m+	p2
	p4t+f+l+m+	p4
	p3f+l+m+	p3
	p1t1+f+t2+l+m+	p1

dom (x/10-6/10) = {p1,p2,p4,p3,p5}

dom (x/6-5/6) = {p1,p2,p4,p3,p5}

dom (x/6-7/6) = {p1,p2,p4,p3,p5}

dom (x/7-6/7) = {p1,p2,p4,p3,p5}

dom (xt) = {t-l-m-,t1-r-t2-l-m-,t1+r+t2-l-m-,t-f-l-m-,t-f+l-m-,  
t1-f-t2-l-m-,t1+f+t2-l-m-}

tournage :	xt	tournage(xt)
	t-l-m-	t+l-m-
	t1-r-t2-l-m-	t1+r-t2-l-m-
	t1+r+t2-l-m-	t1+r+t2+l-m-
	t-f-l-m-	t+f-l-m-
	t-f+l-m-	t+f+l-m-
	t1-f-t2-l-m-	t1+f-t2-l-m-
	t1+f+t2-l-m-	t1+f+t2+l-m-

dom (xf) = {pt+f-l-m-,pt-f-l-m-,pf-l-m-,pt1+f-t2-l-m-}

fraisage :	xf	fraisage(xf)
	pt+f-l-m-	pt+f+l-m-
	pt-f-l-m-	pt-f+l-m-
	pf-l-m-	pf+l-m-
	pt1+f-t2-l-m-	pt1+f+t2-l-m-

dom (xr) = {t1+r-t2-l-m-}

retournement :	xr	retournement(xr)
	t1+r-t2-l-m-	t1+r+t2-l-m-

dom (xl) = {p1t+l-m-,p1t1+r+t2+l-m-,p2t+f+l-m-,p4t+f+l-m-,p3f+l-m-,  
p5t1+f+t2+l-m-}

lavage :	xl	lavage(xl)
	p1t+l-m-	p1t+l+m-
	p1t1+r+t2+l-m-	p1t1+r+t2+l+m-
	p2t+f+l-m-	p2t+f+l+m-
	p4t+f+l-m-	p4t+f+l+m-
	p3f+l-m-	p3f+l+m-
	p5t1+f+t2+l-m-	p5t1+f+t2+l+m-

$\text{dom}(xm) = \{t+l+m-, t1+r+t2+l+m-, t+f+l+m-, f+l+m-, t1+f+t2+l+m-\}$

mesures : xm

t+l+m-

t1+r+t2+l+m-

t+f+l+m-

f+l+m-

t1+f+t2+l+m-

mesures(xm, facteur aleatoire)

dans (t+l+m1+, t+l+m2+)

avec une distribution à préciser ultérieurement dans la phase de structuration du prégraphe.

t1+r+t2+l+m+

t+f+l+m+

f+l+m+

t1+f+t2+l+m+

# SEQUENCES OPERATOIRES

- t-l-m- (p1)
- t1-r-t2-l-m- (p1)
- t-f-l-m- (p2)/f-t-l-m- (p4)
- f-l-m- (p3)
- t1-f-t2-l-m- (p5)

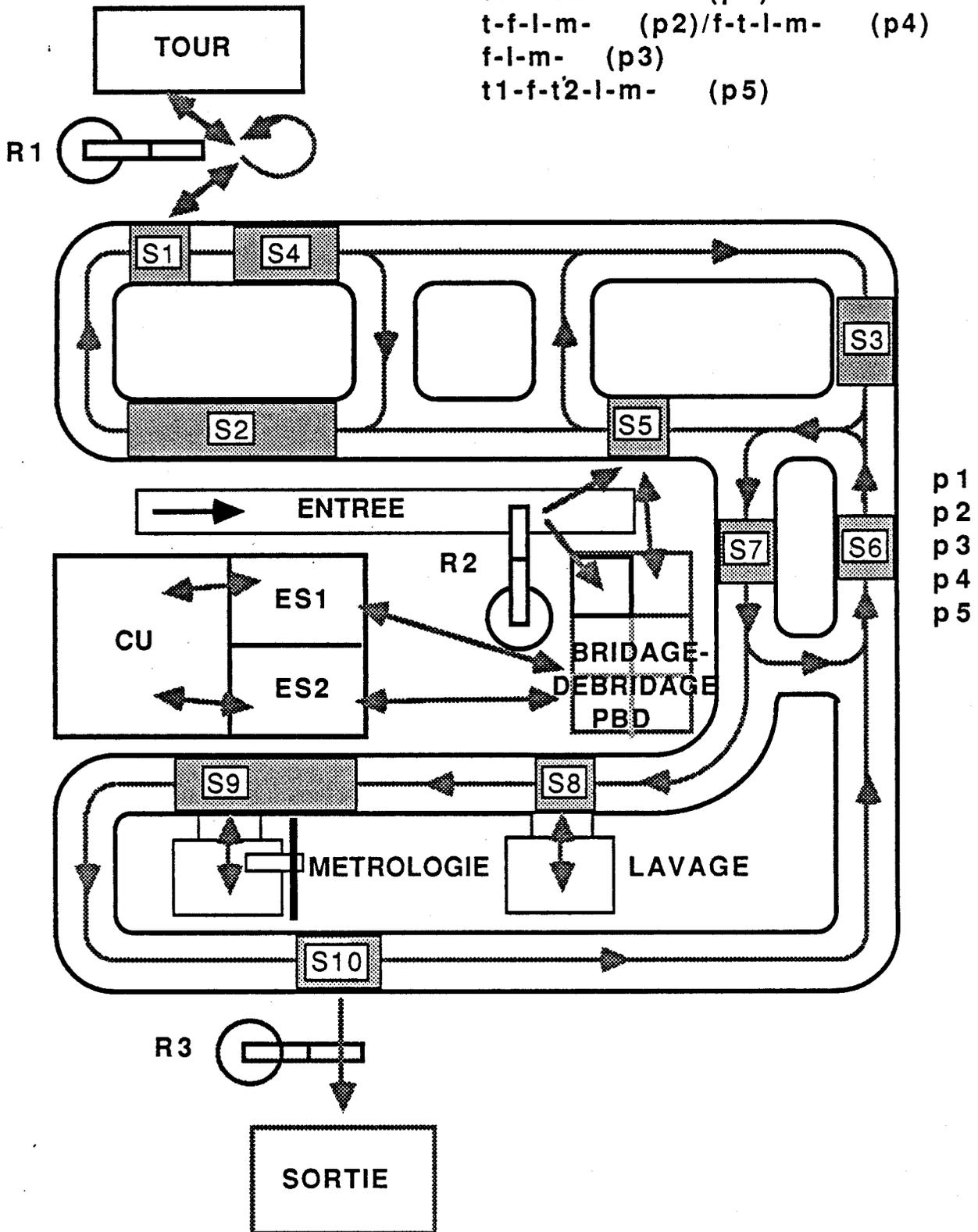


Figure II.67

**Remarque très importante : La place de l'architecture matérielle réelle vis à vis du PRÉGRAPHE. (Figure II.67)**

La figure précédente II.67 représente une architecture matérielle correspondant au Prégraphe des figures II.66.

Cette architecture est organisée autour d'un convoyeur à chaîne centrale débrayable entraînant des palettes (qui peuvent donc être arrêtées individuellement par des systèmes de butée sans stopper l'ensemble du convoyeur).

Les sites S1, S2, S3, S4, S5, S6, S7, S8, S9, S10 correspondent à des butées. Ces lieux servent à arrêter les palettes afin d'effectuer une opération ou afin de créer une file d'attente (S2, S3).

Les flèches indiquent les différents transferts. Ceux-ci sont assurés, en dehors du convoyeur central, par trois robots satellites R1, R2 et R3.

Cette architecture répond aux fonctionnalités données par le Prégraphe.

**Il faut bien noter ici que le Prégraphe peut être construit à partir d'une architecture matérielle existante mais qu'il ne nécessite absolument pas la définition préalable de cette architecture qui peut être élaborée également après la construction du Prégraphe. Les deux démarches sont possibles et sont en fait destinées à coopérer.**

De même, l'étape ultérieure de développement ou Structuration de la partie commande (cf chapitre III Partie I) pourra s'arrêter à des niveaux de structuration et de détail divers en fonction du niveau de spécification de la définition matérielle réelle, jusqu'à l'interfaçage ultime avec les capteurs et les préactionneurs.

La modularité et le caractère structuré de la démarche sont destinés à permettre des retours-arrière faciles sur les spécifications et à faire coopérer des approches d'analyse descendante et montante.

#### IV.

#### LES ETAPES DE MODULATION

Les paragraphes précédents ont montré comment une description sous forme de règles de production puis sous forme de PREGRAPHE pouvait être obtenue. Jointe à chacune de ces étapes de description, une étape de modulation permet de les moduler sans modifier les propriétés de cohérence et de complétude.

Ces étapes de modulation comportent actuellement quatre possibilités :

- la multiplication de sites.
- la multiplication d'ensembles de sites.
- l'ajout de lieux de stockage intermédiaires.
- l'agrégation de sites.

Ces étapes sont écrites en Le-Lisp. Elles occupent environ 2x15 K Octets.

##### IV.1 La multiplication de sites

###### a) A partir de la description sous forme de PREGRAPHE

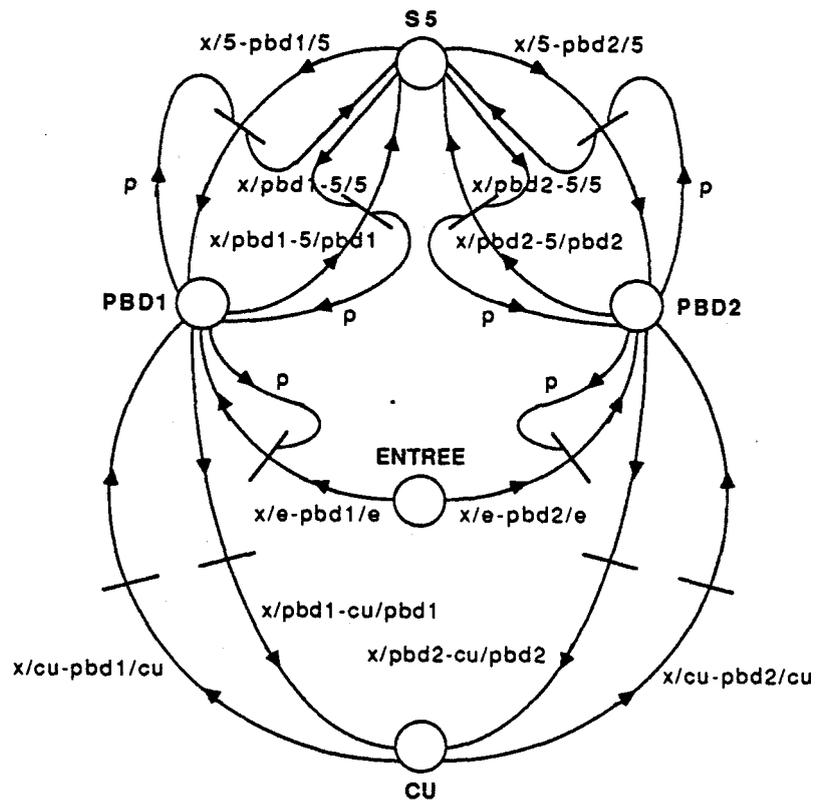
La méthode est la suivante :

- Le logiciel demande à l'utilisateur le nom de la place P à dupliquer.
- Il interroge ensuite l'utilisateur sur le nombre de nouvelles places identiques qui vont remplacer P. La liste des places duplicatas est alors construite.
- Le logiciel établit alors la liste des transitions concernées (celles qui ont la place P dans leurs sites prédécesseurs ou successeurs).
- Pour chaque place duplicata remplaçant P
  - . La liste des transitions concernées est passée en revue.  
Chaque transition est recopiée avec un nouveau nom et reçoit la liste des propriétés de la transition mère dans laquelle le nom de la place "P" a été substitué par le nom du duplicata de P.
  - . La liste de ces nouvelles transitions est ajoutée à la liste des transitions du PREGRAPHE.
- On supprime alors la liste des transitions concernées d'origine de la liste des transitions du PREGRAPHE.

ex. : Reprenons le PREGRAPHE de la figure II.66.

L'étude de l'enchaînement des opérations n'a fait apparaître qu'un poste de bridage-débridage PBD pour des raisons de simplicité. L'étape de modulation va permettre de multiplier ce poste pour correspondre aux exigences données par l'étude des flux et des performances réelles (à partir d'un prédimensionnement et/ou d'une simulation). Cette modulation ne porte pas sur l'enchaînement des opérations et donc n'altère en aucune manière les propriétés de cohérence et de complétude associées à la description.

A titre indicatif, la figure ci-dessous représente le doublement du poste PBD.



Il est clair que l'automatisation de la démarche évite un grand nombre d'erreurs.

b) A partir de la description sous forme de règles opératoires

Cette fois, il est possible de limiter l'ensemble des règles sur lesquelles va porter la duplication. Par exemple, la duplication de sites peut ne porter que sur une seule gamme opératoire. On est alors toutefois conduit à modifier dans les autres gammes le nom de la place remplacée.

La méthode est alors la même qu'au a), les règles opératoires jouant alors le rôle des transitions.

IV.2 La multiplication de blocs ou ensembles de sites

a) A partir de la description sous forme de PREGRAPHE

La méthode est la suivante :

- Le logiciel demande à l'utilisateur les noms  $P_k$  des places constituant l'ensemble B (ou bloc) à dupliquer.
- Il interroge ensuite l'utilisateur sur le nombre de blocs identiques qui vont remplacer B. La liste des blocs duplicatas est alors construite, avec, pour chaque bloc duplicata, la liste des nouveaux noms des places constitutives.

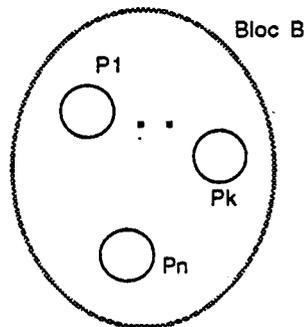


Figure II.69

- Le logiciel établit alors la liste des transitions concernées (celles qui ont au moins une des places  $P_k$  dans leurs sites prédécesseurs ou successeurs).
- Pour chaque bloc duplicata
  - . La liste des transitions concernées est passée en revue.  
Chaque transition est recopiée avec un nouveau nom et reçoit la liste des propriétés de la transition mère dans laquelle les noms des places  $P_k$  de B ont été substitués par les noms respectifs des duplicatas des  $P_k$ .

. La liste de ces nouvelles transitions est ajoutée à la liste des transitions du PREGRAPHE.

- On supprime alors la liste des transitions concernées d'origine de la liste des transitions du PREGRAPHE.

ex. Reprenons le PREGRAPHE de la figure II.66.

Considérons le bloc B constitué du TOUR et de son poste de chargement-déchargement S1.

La figure ci-dessous représente le triplement du bloc B.

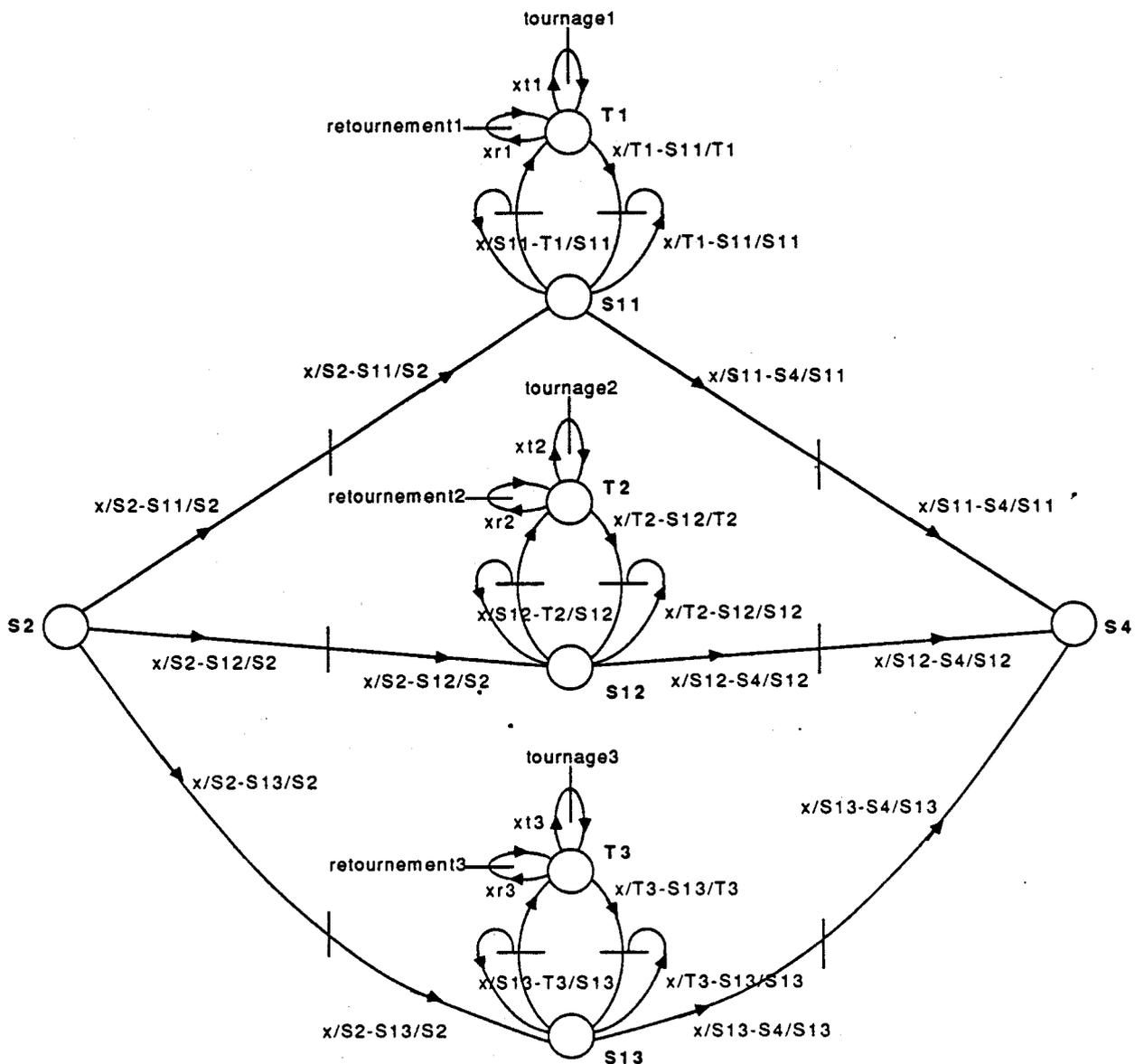


Figure II.70

b) A partir de la description sous forme de règles opératoires

Mêmes remarques que pour la multiplication de sites.

IV.3 L'ajout de lieux de stockage intermédiaires

Que ce soit sur la description sous forme de règles ou sur le PREGRAPHE, il est possible d'ajouter des lieux de stockage intermédiaires.

ex. Considérons la transition tR

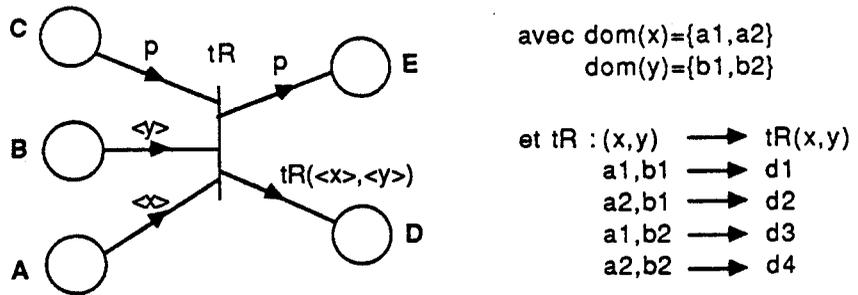


Figure II.71

On peut par exemple ajouter un lieu de stockage ST entre C et tR :

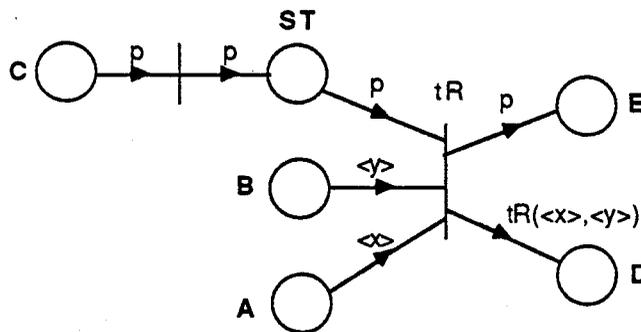


Figure II.72

ou entre tR et E.

On peut placer ST entre A et tR :

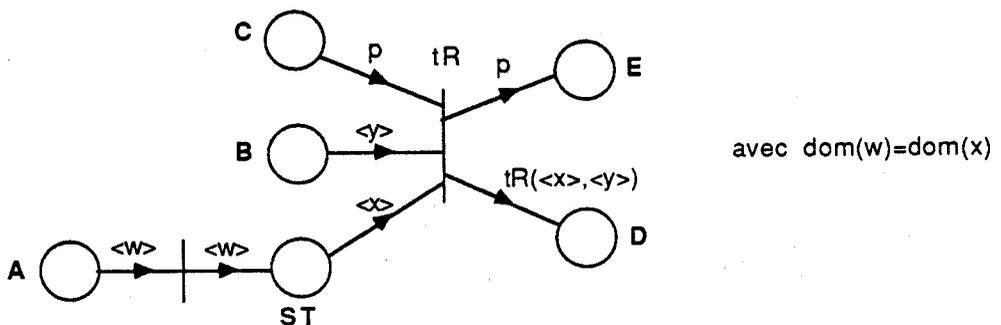
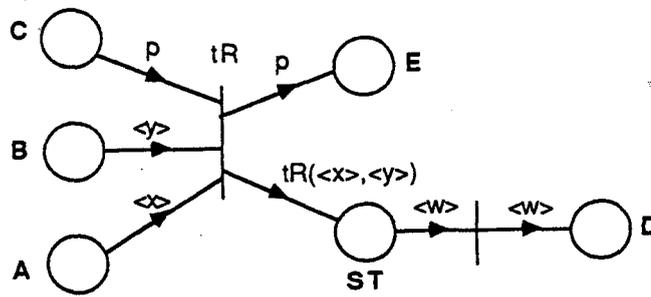


Figure II.73

on peut placer ST entre tR et D :



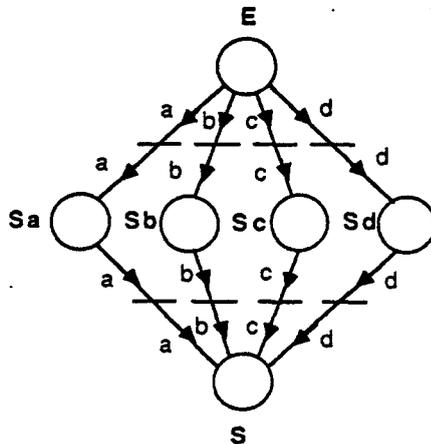
avec  $\text{dom}(w) = \{d1, d2, d3, d4\}$

Il est clair que l'ajout d'un site de stockage tampon intermédiaire n'altère aucunement les propriétés de cohérence et de complétude du chaînage logique des actions.

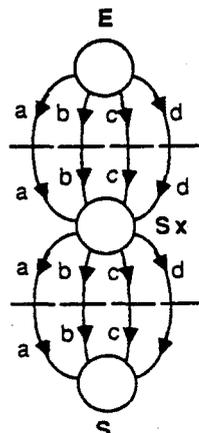
#### IV.4 L'agrégation de sites

Dans l'étape de modulation, la possibilité est donnée d'agréger plusieurs sites  $S_k$  correspondant par exemple à des machines spécifiques pour former un seul site S polyvalent.

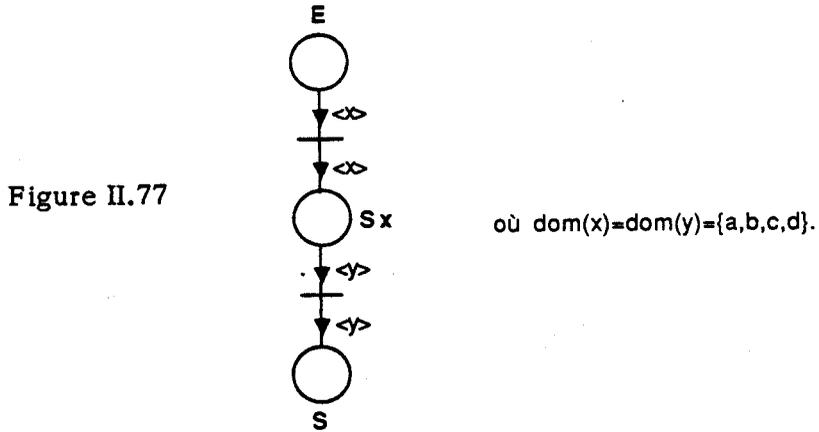
ex.



devient par exemple :



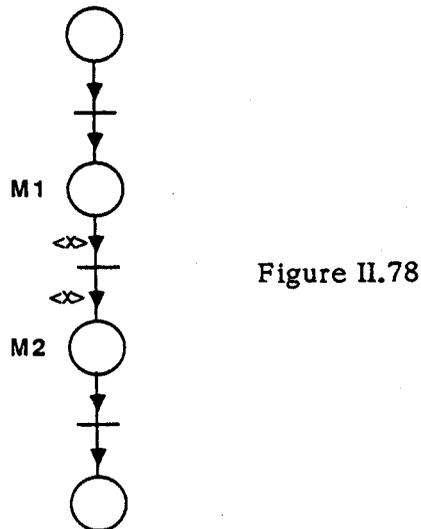
qu'il est encore possible ensuite d'agréger sous forme de PREGRAPHE.



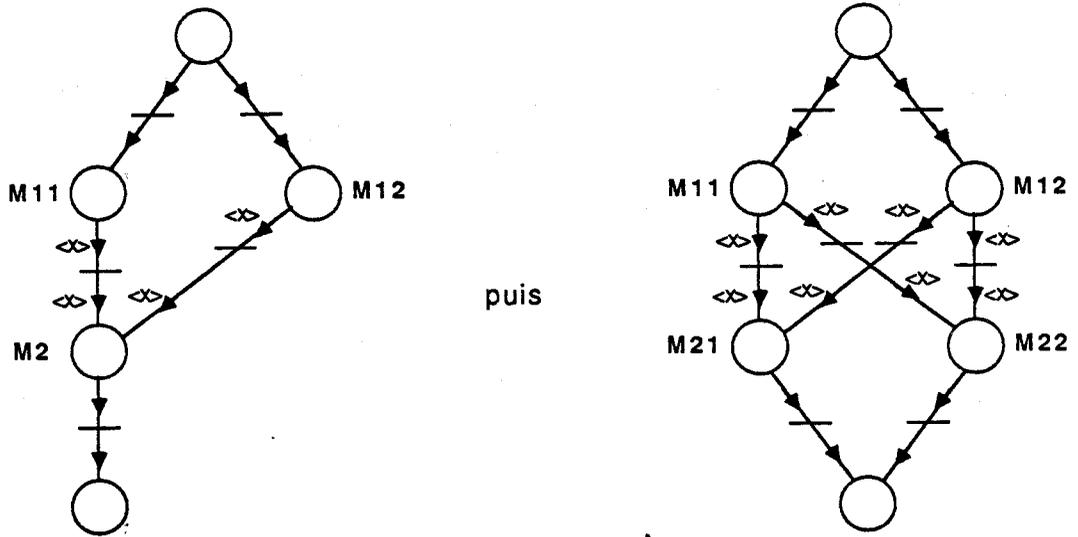
#### IV.5 Un mot sur la flexibilité

Pour terminer cette partie, précisons que l'étape de modulation constitue une étape importante destinée à augmenter la souplesse de circulation des pièces et ainsi, la flexibilité de l'unité de production.

ex. 1 :      Considérons un transfert simple entre deux postes M1 et M2

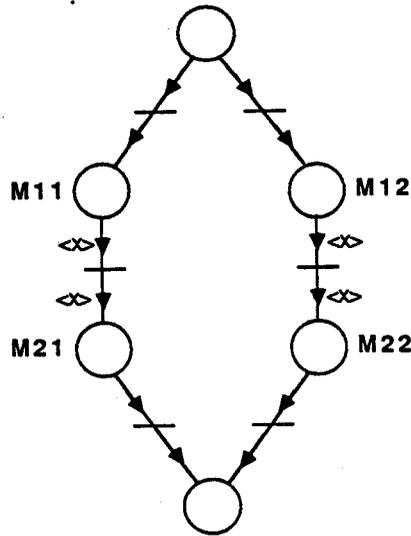


La duplication de M1 puis de M2 va conduire aux schémas suivants :

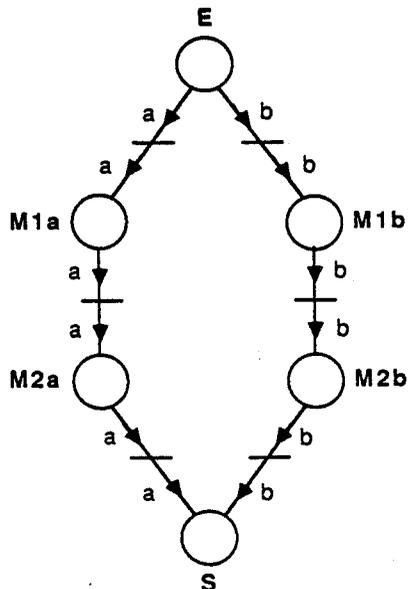


La duplication du bloc (M1, M2) va conduire au schéma ci-dessous

Figure II.80



ex. 2 : Considérons maintenant la description ci-dessous

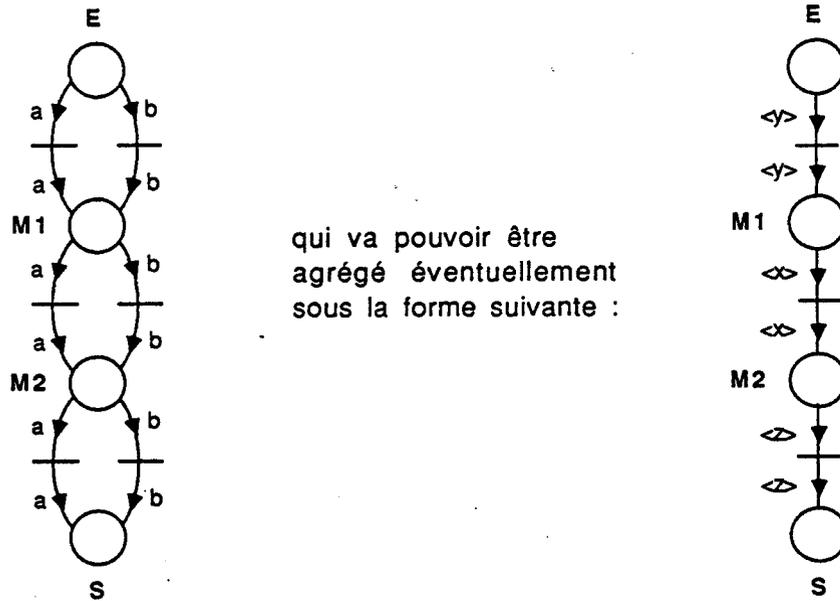


On suppose que :

- les postes M1a et M2a sont spécifiques à la pièce a.
- les postes M1b et M2b sont spécifiques à la pièce b.

Figure II.81

L'agrégation de sites va par exemple permettre de remplacer dans la description sous forme de règles M1a et M1b par un seul poste polyvalent M1, et M2a et M2b par un seul poste polyvalent M2. La description va alors être celle de la figure ci-dessous :



qui va pouvoir être  
agrégé éventuellement  
sous la forme suivante :

Figure II.82

Cette dernière description va pouvoir alors être traitée comme dans l'exemple 1 par duplication des postes.

Une autre possibilité consiste encore par exemple à dupliquer M2a et M2b. On a alors :

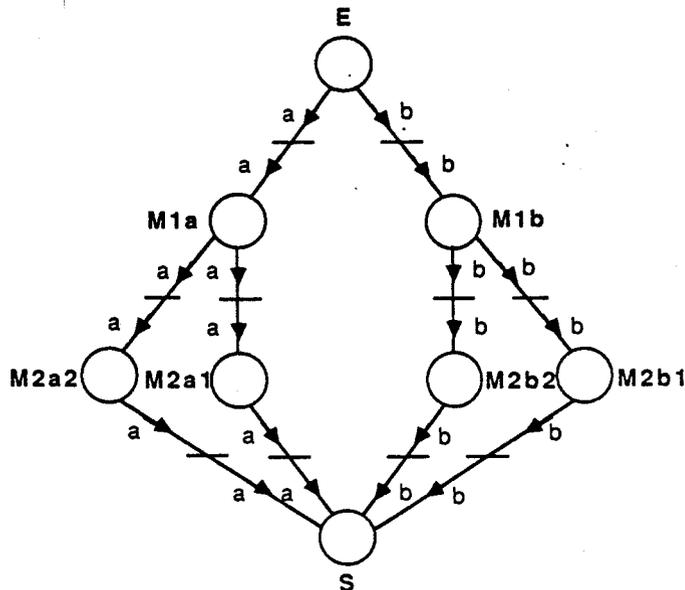


Figure II.83

Supposons que M2a-1 et M2b-1 ainsi que M2a-2 et M2b-2 soient respectivement agrégés pour former deux postes M2-1 et M2-2 polyvalents. On a

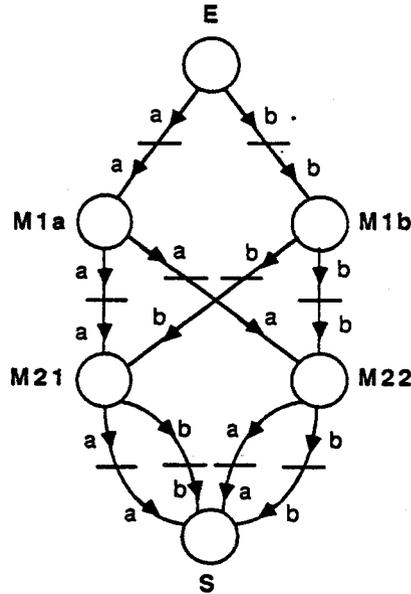


Figure II.84

Ainsi, la modulation permet, à partir de schémas de chaînes transferts linéaires avec des machines éventuellement spécifiques, de construire des figures de treillis flexibles très variés. Remarquons que l'augmentation de la flexibilité de circulation des pièces se traduit par l'accroissement du nombre d'orientations possibles pour une pièce à partir d'un lieu, c'est-à-dire par l'augmentation du nombre des indéterminismes directionnels. La résolution de ces indéterminismes fait appel au Niveau Décisionnel ou Hiérarchique. Un compromis doit être trouvé pour ne pas surcharger ce dernier et pour conserver un niveau de flexibilité de circulation intéressant.

Enfin, disons qu'une architecture flexible performante ne doit pas générer de stocks d'attente importants. Toutefois la possibilité d'ajout de lieux de stockage tampon intermédiaires a été donnée dans l'étape de modulation pour pouvoir étudier facilement et avec beaucoup de souplesse des cas de figures très variés.

V.

L'ETUDE SIMPLIFIEE DE LA COMBINATOIRE DES GAMMES

Les parties précédentes de ce chapitre ont permis de montrer ce qu'était la description des gammes et comment, à partir de cette description, on obtenait le PREGRAPHE.

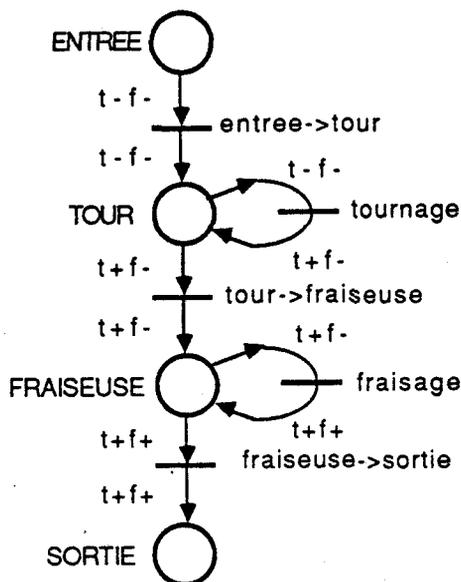
Afin d'obtenir une architecture flexible pour le PREGRAPHE (souplesse de circulation des pièces, facilité de reconfiguration des rôles des machines), une étude préalable précise et à long terme des gammes d'usinage et d'assemblage doit être réalisée.

En particulier, une recherche des combinaisons opératoires à l'intérieur de ces gammes est nécessaire pour mettre en lumière toutes les possibilités de permutations d'opérations. Cette phase d'étude de flexibilité des séquences opératoires est essentielle si l'on veut atteindre un degré élevé de flexibilité pour le système de commande.

Par exemple, si une pièce doit subir une opération de tournage  $t$  et si toutes les machines (spécifiques ou polyvalentes) pouvant effectuer cet usinage sont indisponibles (saturées ou en panne), il est intéressant de prévoir toutes les permutations opératoires possibles dans la gamme de façon à ce que la pièce puisse s'orienter immédiatement vers une autre étape de transformation sans être placée en attente dans un stock tampon intermédiaire.

On cherche ainsi à remplacer le schéma ci-dessous de chaîne transfert linéaire.

Figure II.85



par le treillis suivant, plus flexible :

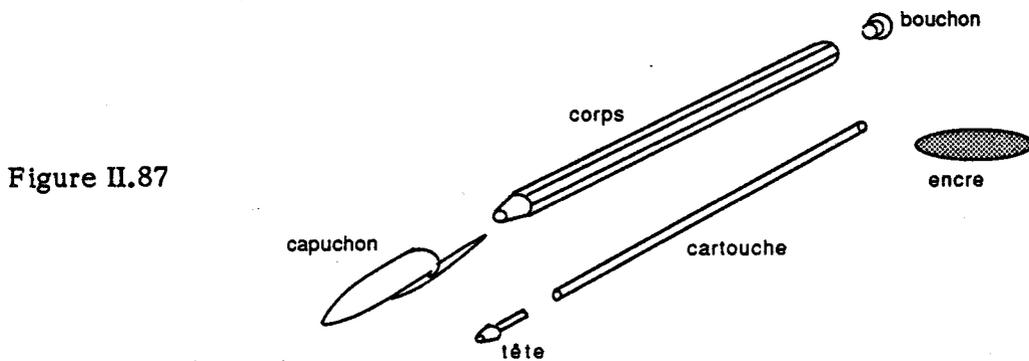


### V.1 La méthode présentée par Alain BOURJAULT

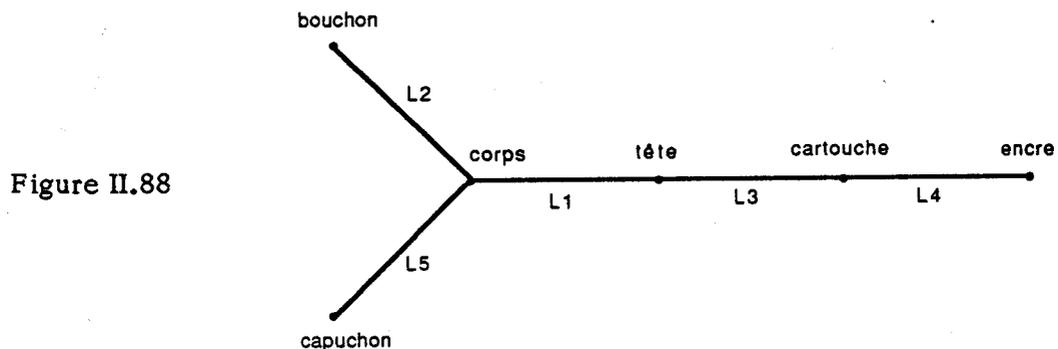
Le cas des séquences d'usinages présente une complexité restreinte, car les contraintes d'antériorité sont en général simples à établir. Ces contraintes étant fortes, la combinatoire de permutation des opérations reste de ce fait très limitée. La thèse d'Alain BOURJAULT s'est donc concentrée sur les gammes d'assemblage. Le panachage de séquences d'assemblage et d'usinage se fait simplement par réunion des contraintes d'antériorité.

L'idée est ici de décomposer les assemblages en liaisons fonctionnelles  $L_i$  binaires (à deux extrémités) entre composants de l'assemblage final.

Ainsi pour un crayon à bille.



on obtient le graphe des liaisons fonctionnelles ci-dessous



Pour chacune des liaisons  $L_i$ , on cherche à obtenir la condition de réalisabilité  $C_i$  de  $L_i$  à partir d'une condition logique portant sur l'existence

préalable ou l'absence des autres liaisons Lk. Ainsi si  $C_i = L_1 \bar{L}_2$ ,  $L_i$  ne peut être réalisée que si  $L_1$  est réalisée et que si  $L_2$  n'est pas encore réalisée.

Pour établir les conditions  $C_i$ , une méthode de conception a été élaborée, basée sur la définition de quatre relations portant sur les liaisons.

2 relations binaires R et S

$R(L_i, L_j)$  VRAIE  $\Leftrightarrow$  il est impossible de réaliser  $L_i$  lorsque  $L_j$  est établie

$S(L_i, L_j)$  VRAIE  $\Leftrightarrow$  il est impossible de réaliser  $L_j$  lorsque  $L_i$  n'est pas établie

2 relations p-aires  $R_p$  et  $S_p$

$R_p(L_i, L_k, \dots, L_p)$  VRAIE  $\Leftrightarrow$  il est impossible de réaliser  $L_i$  lorsque  $L_i$  et ... et  $L_p$  sont établies

$S_p(L_i, L_k, \dots, L_p)$  VRAIE  $\Leftrightarrow$  il est impossible de réaliser  $L_i$  lorsque ni  $L_k$  ni ... ni  $L_p$  ne sont établies.

Compte tenu des propriétés de ces relations et des propriétés du graphe des liaisons, un questionnaire minimal est recherché et une méthode de construction systématique des  $C_i$  est donnée.

On trouve alors pour les  $C_i$  des expressions logiques du genre de  $C_1$  :

$$C_1 = \bar{1}.5. (2.3.4 + \bar{2}.3.4 + \bar{2}.3.\bar{4} + \bar{2}.\bar{3}.\bar{4})$$

où 4 signifie la présence de la liaison  $L_4$

$\bar{4}$  signifie l'absence de la liaison  $L_4$ .

Ces expressions peuvent être simplifiées en utilisant les méthodes classiques de réduction d'expressions logiques (ex : QUINE - Mc CLUSKEY)

par exemple, pour le stylo à billes, on obtient

$$C_1 = \bar{1}. (\bar{2}.3.\bar{5} + \bar{2}.\bar{4}.\bar{5} + 3.4.\bar{5})$$

$$C_2 = \bar{2}. (\bar{1}.3.\bar{5} + 3.4.\bar{5} + \bar{1}.\bar{4}.\bar{5} + 1.3.4)$$

$$C_3 = \bar{3}. (1.\bar{2}.\bar{4} + \bar{2}.\bar{4}.\bar{5} + \bar{1}.\bar{4}.\bar{5})$$

$$C_4 = \bar{4}. (1.\bar{2}.3 + \bar{1}.3.\bar{5} + \bar{2}.3.\bar{5})$$

$$C_5 = \bar{5}. (1.3.4 + 1.\bar{2}.3 + 1.\bar{2}.\bar{4})$$

Un tableau primitif comme le tableau primitif direct ci-dessous est alors construit à partir des Ci.

r \ AFi	AF1	AF2	AF3	AF4	AF5
1				impossible	impossible
2	3	3	1+2	3	1
3	3.4	3.4	1.5	1.3+2.3	1.3
4	2.3.4	1.3.4	impossible	1.3.5	1.3.4
5	impossible	1.3.4.5	impossible	impossible	1.2.3.4

Figure II.39

- AFi désigne l'action fonctionnelle de réalisation de la liaison Li.
- r désigne le rang possible de réalisation de AFi dans l'ordre de réalisation des liaisons.
- dans la case de coordonnées (r, AFi) on lit la condition nécessaire pour que la liaison Li soit réalisée au rang r, c'est-à-dire en r-ième position.

Une exploration systématique du tableau du type des explorations arborescentes en profondeur d'abord est alors menée. Les chemins qui permettent de conduire à l'assemblage final sont ainsi détectés. Un exemple de chemin est donné en pointillés sur le tableau ci-dessus.

L'étude générale a été également étendue aux problèmes des actions d'assemblage simultanées.

L'établissement des contraintes d'antériorité a fait l'objet de recherches de simplification à partir du graphe de liaisons.

Des cas de figure relatifs aux actions indifférentes, à l'agrégation d'actions fonctionnelles en macro-actions et à la problématique des sous-assemblages ont été abordés.

Les développements récents ont fait l'objet de publications dont la liste est donnée ci-contre (|BOU 86a| |BOU 86b| |BOU 87a| |BOU 87b| |CHA 88|).

## V.2 Adaptation

### a) Obtention des contraintes d'antériorité

L'idée a été de ne pas chercher à exprimer les conditions de réalisabilité Ci mais de travailler directement à partir de l'ensemble minimal des relations non réductibles R, S, Rp et Sp obtenu par la méthode développée dans la thèse d'Alain BOURJAULT et qui a été adaptée à l'aide d'outils et de techniques utilisées en intelligence artificielle. Il n'est pas possible ici de rentrer dans le détail de la méthode. Disons simplement qu'elle nécessite un nombre important d'explorations spécifiques des graphes de liaisons.

L'ensemble minimal des relations R, S, Rp et Sp constitue alors l'ensemble des contraintes physiques de réalisation des assemblages qu'on appelle contraintes primaires.

Un ensemble de contraintes dites secondaires non physiques est construit par heuristique à partir des contraintes primaires afin de réduire de façon importante l'exploration ultérieure de l'arborescence des choix de permutations.

Prenons un exemple simple à propos du stylo à bille.

- la liaison tête-corps est impossible à réaliser si la liaison corps-capuchon est déjà établie. Il s'agit d'une contrainte physique primaire.
- on comprend alors aisément que bien que la liaison corps-capuchon soit possible physiquement si la liaison tête-corps n'a pas été établie au préalable, il est inutile de réaliser tout de suite la liaison corps-capuchon qui conduirait ultérieurement à une impasse. Ceci constitue alors une contrainte heuristique ou secondaire.

### b) Etude combinatoire en fonction des contraintes

L'ensemble des contraintes étant établi, il reste alors à réaliser l'exploration combinatoire des possibilités de permutation des opérations en fonction de ces contraintes.

La méthode est simple.



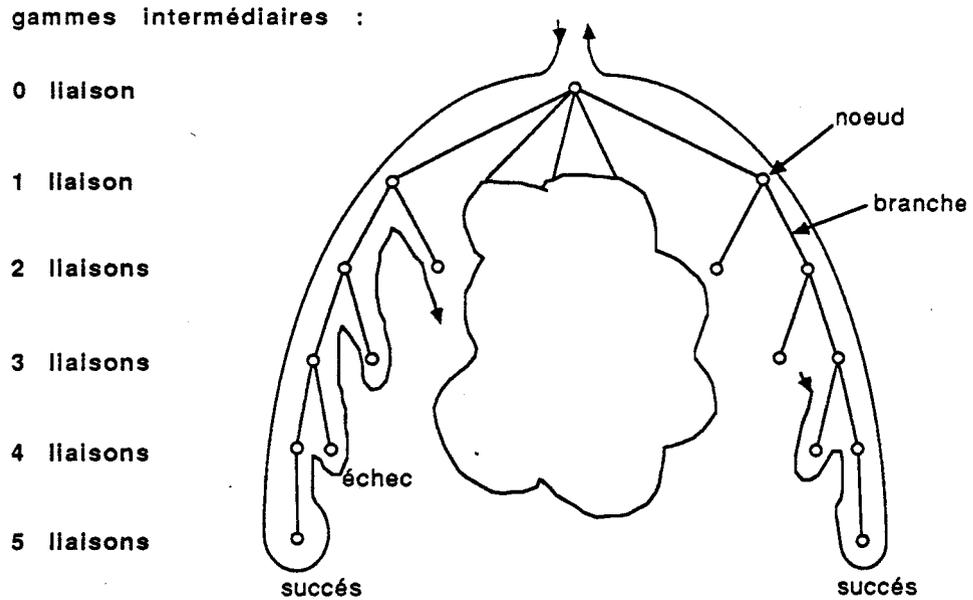


Figure II.90

Chaque noeud désigne une gamme intermédiaire et chaque branche représente l'ajout réussi d'une liaison.

Cette démarche est tout à fait élémentaire. La difficulté consiste à éliminer au plus tôt les explorations inutiles en construisant correctement l'ensemble des contraintes et en exploitant au plus tôt toutes les informations qu'elles contiennent. C'est la raison pour laquelle on a introduit les contraintes secondaires à partir des contraintes primaires.

La difficulté essentielle provient de l'élaboration des contraintes qui est très délicate dans le cas général comme le montre la différence de taille des deux logiciels écrits en Le-Lisp et qui assistent l'analyse de la combinatoire des gammes.

- |                               |            |
|-------------------------------|------------|
| - Elaboration des contraintes | 23 Koctets |
| - Etude combinatoire          | 3 Koctets  |

**Remarque :** Ajoutons une remarque à ce paragraphe qu'il n'est pas possible de développer davantage dans le cadre de ce mémoire. Elle concerne le traitement des actions nécessairement simultanées. Prenons l'exemple d'un piston et d'une bielle. Ils sont tous les deux reliés par l'intermédiaire d'un axe.

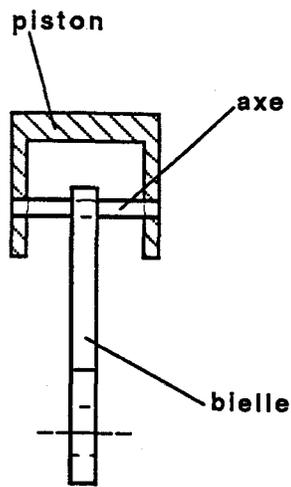


Figure II.91

La liaison axe-piston ne peut être réalisée si la liaison axe-bielle est déjà présente  
et

La liaison axe-bielle ne peut être réalisée si la liaison axe-piston est déjà présente.

Les liaisons doivent être considérées comme étant établies simultanément sauf si on décompose leur réalisation en sous-étapes.

Ces situations sont détectées par examen des contraintes binaires. Si l'utilisateur ne souhaite pas détailler davantage, les liaisons simultanées sont considérées comme une liaison unique représentée par une sous-liste dans la gamme.

Sur le PREGRAPHE, l'antériorité et la simultanéité de deux liaisons a-b et b-c se traduiront par exemple sous la forme suivante :

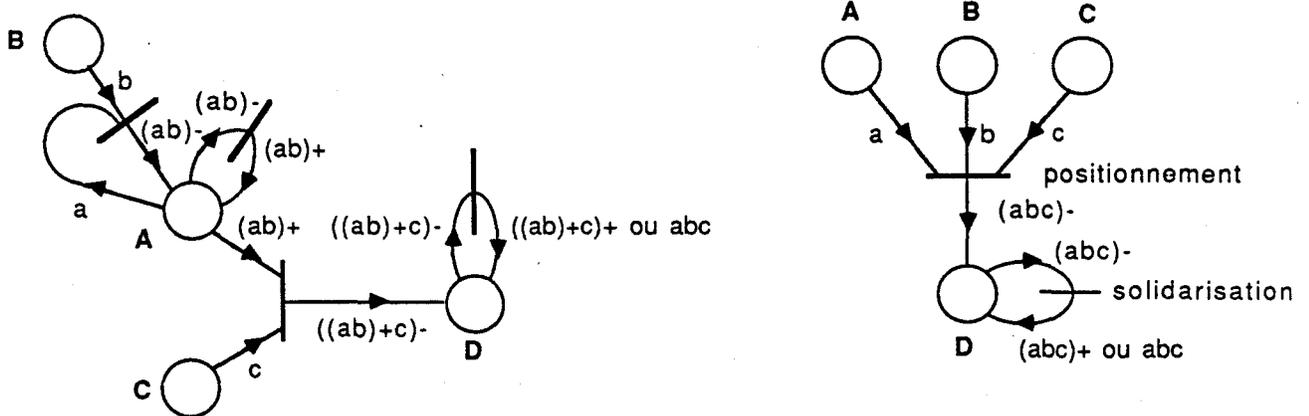
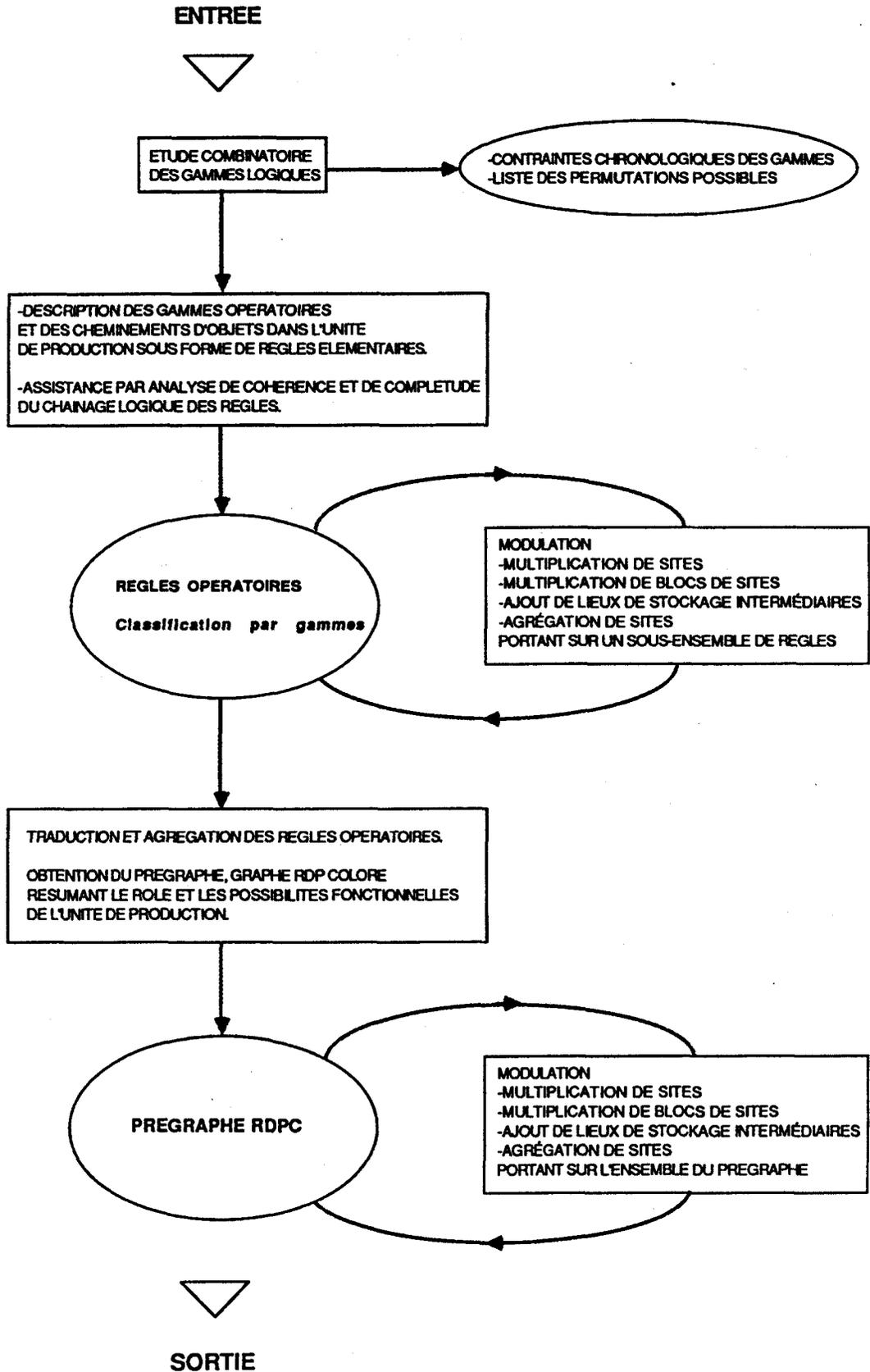


Figure II.92

## CONCLUSION

La chaîne de conception qui vient d'être présentée ici peut être schématiquement décrite sur la figure ci-dessous :



Rappelons que cette chaîne est supportée par un ensemble de logiciels informatiques écrits en langage Le-Lisp (INRIA : Jérôme CHAILLOUX) et dont la totalité représente une taille de 210 Koctets. Cette chaîne logicielle a été mise au point sur un mini ordinateur VAX 11/750 puis portée sur micro-ordinateur Apple Mac Intosh, comme d'ailleurs cela a été fait pour l'ensemble de la chaîne de C.A.O. existante du projet CASPAIM (NB : la conception des logiciels a été réalisée de façon suffisamment portable pour pouvoir transférer les fichiers sources d'une machine à l'autre par l'intermédiaire de logiciels de communication tels que KERMIT). Les programmes sources sont alors interprétés par des interpréteurs Le\_Lisp installés sur des machines différentes. Il existe également des possibilités de compilation des programmes Le\_Lisp à partir du code LLM3. Les essais qui ont été faits n'ont pas permis de noter une réelle différence de vitesse d'exécution, les compilateurs générant un code systématique non optimum et assez encombrant.

-----

Par son allure, le PREGRAPHE fait penser aux graphes de flux qui sont utilisés pour la construction des schémas directeurs dans les processus actuels d'automatisation et d'intégration des usines. Ces schémas directeurs ont pour but d'organiser la gestion des projets d'automatisation et d'intégration informatique de façon dirigée, cohérente et modulaire. Le rôle de tels graphes est d'obtenir un premier découpage de l'objectif fonctionnel à réaliser en sous-problèmes. Ils constituent ainsi le squelette fondamental et la ligne directrice de la gestion des projets d'usine automatisée et/ou intégrée (voir [BEN 86] "L'usine intégrée par ordinateur" chapitre 3, et [AUG 87]). Les approches méthodologiques de ces projets sont venues des méthodes apparues pour la gestion des projets informatiques et qui ont donné naissance à des systèmes de spécification tels que SADT [AUG 87]. Peu de choses ont été faites encore pour la Productique.

Le rôle du PREGRAPHE est, quant à lui pour le moment, de fournir le schéma d'organisation de base de la construction structurée d'une Partie Commande flexible (et donc en particulier polyvalente et modulable).

Toutefois, les informations qu'il manipule constituent l'essence fonctionnelle de l'unité de production. L'importance de base de ces informations feront du PREGRAPHE un outil essentiel pour l'ensemble de la démarche de

conception de l'unité de production (configurations matérielles et spatiales générales, prédimensionnements analytiques et pré-simulations) et pas seulement pour la partie commande, ce qui est son utilisation actuelle. C'est sur le rôle actuel du PREGRAPHE dans le projet CASPAIM et sur ses utilisations et développements attendus que va porter maintenant le chapitre III qui va suivre.

**CHAPITRE III**

**UTILISATION DANS LA CHAINE DE CONCEPTION  
DU PROJET CASPAIM DE LA BASE DE TRAVAIL  
FOURNIE PAR LE PREGRAPHE**



## INTRODUCTION

La notion de PREGRAPHE est maintenant définie. Le but du chapitre III va être de montrer le rôle de ce graphe intermédiaire dans la chaîne de conception du projet CASPAIM ainsi que les développements qui sont en cours autour de cet outil central.

La première partie montrera comment, à partir du Prégraphe, le modèle RdPSAC de la partie Commande est obtenu par une étape appelée STRUCTURATION. Cette partie montrera comment le prégraphe est, sur le plan structurel, le squelette du modèle de la Partie Commande.

Les résultats de la première partie étant acquis, la deuxième partie s'intéressera, à travers un exemple délicat à traiter, à des problèmes limites de modélisation. La question de l'extension du modèle de la Partie Commande et donc du Prégraphe sera discutée.

Le rôle essentiel du Prégraphe est actuellement de servir de base à la construction de la Partie Commande. C'est sur ce point que se concentrent les deux premières parties du chapitre III. La troisième partie va traiter, quant à elle, d'une tout autre utilisation où des modèles dérivés du Prégraphe peuvent avoir un rôle important à jouer. Il s'agit des problèmes de prédimensionnement de l'architecture des unités de production et de préétude simplifiée de leur comportement. Bien que cette étude soit actuellement en cours, des indications significatives sur l'orientation des travaux seront apportées.

INTRODUCTION

La notion de PERSONNE est maintenant définie. Le but du chapitre III va être de montrer le rôle de ce graphisme intermédiaire dans la chaîne de conception de projet CASTALM ainsi que les développements qui sont en cours autour de cet outil central.

La première partie montrera comment, à partir du paragraphe 1.5 modèle RBPAC de la partie Commande est obtenu par une étape appelée STRUCTURATION. Cette partie montrera comment le graphisme est sur le plan structurel, la relation du modèle de la partie Commande.

Les résultats de la première partie étant acquis, la deuxième partie s'attachera à travers un exemple concret à traiter à des problèmes limités de modélisation. La notion de l'extension du modèle de la partie Commande et dans le paragraphe sera discutée.

Le rôle essentiel du graphisme est notamment de servir de base à la construction de la partie Commande. C'est sur ce point que se concentrent les deux premières parties du chapitre III. La troisième partie va traiter, dans la suite, tout autre utilisation de ces modèles dérivés du graphisme pour avoir un rôle important à jouer. Il s'agit des problèmes de post-traitement de l'architecture des unités de production et de problèmes algorithmiques de leur comportement. Bien que cette étude soit naturellement en cours, des indications significatives sur l'orientation des travaux seront apportées.

## I. STRUCTURATION, A PARTIR DU PREGRAPHE, DU MODELE RdPSAC DE LA PARTIE COMMANDE

Nous avons montré dans le chapitre II comment était obtenu le PREGRAPHE et comment ce graphe décrivait les opérations fondamentales du rôle fonctionnel de l'unité de production. Ce graphe est destiné à servir de schéma de montage du modèle Réseau de Petri, Structuré, Adaptatif, Coloré de la Partie Commande. Rappelons que la Partie Commande constitue le niveau bas du système de commande, le niveau haut étant le Niveau Hiérarchique ou Niveau Décisionnel. Cette opération de construction, à partir du prégraphe, du modèle RdPSAC de la Partie Commande s'appelle structuration ou encore, développement structuré du Prégraphe.

C'est cette étape que nous nous proposons de décrire brièvement maintenant.

### L1 L'IDEE GENERALE

La base de travail est donc le Prégraphe. Les places représentent les différents lieux où une séquence de commandes doit être appliquée par la Partie Commande au Procédé lors du transit d'un objet. Les transitions entre les places représentent les opérations élémentaires (éventuellement macroscopiques) exigées du rôle fonctionnel de l'unité de production.

L'idée de départ est alors d'associer à chacune des places des modules Réseau de Petri appelés processus associés et reliés entre eux par des modules de liaison.

Chacun de ces modules possède des entrées-sorties standardisées. Il est ainsi possible de construire un premier modèle de la Partie Commande à partir du Prégraphe tel un jeu de construction en positionnant les différents modules bout à bout.

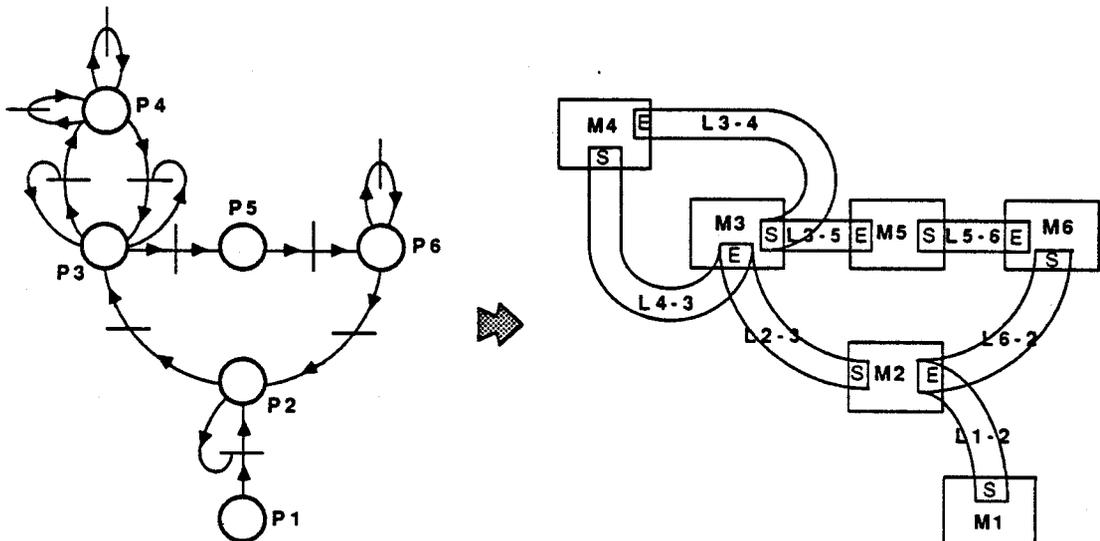


Figure III.1

Mi = module RdPSAC représentant le processus associé à la place Pi.  
Lij = module RdPSAC de liaison entre Mi et Mj.

La construction de chacun de ces modules est transparente à l'utilisateur en ce qui concerne la modélisation RdP. Elle est toutefois réalisée à partir d'un dialogue simple avec l'utilisateur qui est interrogé sur des spécifications de haut niveau. Chacun des modules est ainsi construit de façon stéréotypé par l'intermédiaire d'un questionnaire court et simple. Les schémas de base sont appelés modules génériques ou primitives.

L'élaboration de chacun des modules et l'assemblage, à partir de ces modules, de l'ensemble de la Partie Commande ont fait l'objet du travail d'une thèse récente [BOU 88]. Nous allons dans la suite de cette première partie du chapitre III rappeler et parfois compléter les grandes lignes de ce travail.

## L2 LES MODULES ASSOCIES AUX PLACES DU PREGRAPHE

Le premier point de la structuration consiste à construire les modules associés aux places du Prégraphe.

L'étude de cas concrets industriels a conduit à associer à chacune des places du Prégraphe la représentation suivante :

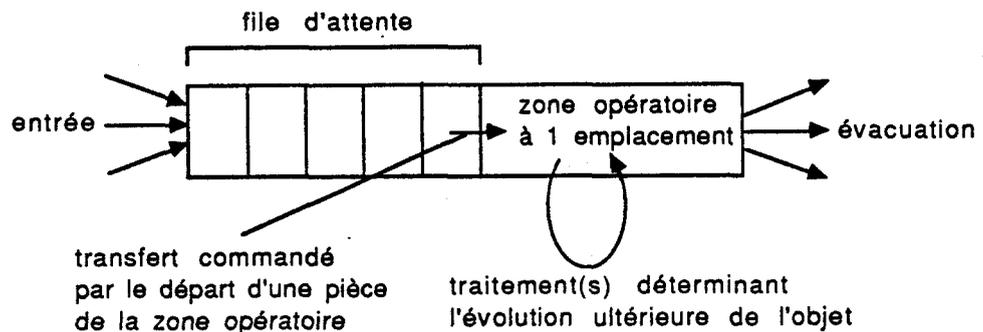


Figure III.2

A partir de cette décomposition fonctionnelle des lieux représentés par les places du prégraphe, un module RdPSC correspondant est élaboré (voir figure suivante III.3).

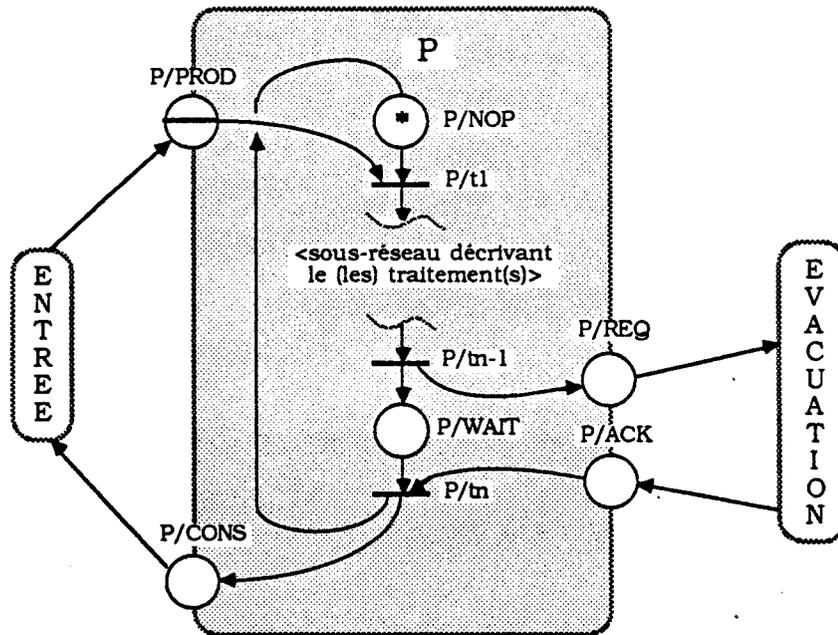


Figure III.3

Ce module RdP est construit de telle façon :

- i) la liaison producteur/consommateur représente la file d'attente dont la taille effective sera précisée lors de l'étape de dimensionnement. Dans cette liaison, le nombre de marques de la place P/CONS indique aux processus d'entrée le nombre d'emplacements disponibles dans la file d'attente, tandis que la macro-place P/PROD contient les marques représentant les objets présents dans cette file,
- ii) l'architecture est construite autour d'un processus P (voir la notion dans le chapitre I) qui assure le transfert d'un objet depuis la file d'attente jusqu'à la zone opératoire, ainsi que la gestion des opérations effectuées sur l'objet et déterminant son évolution ultérieure (ex. : transformations).
- iii) la liaison de synchronisation avec accusé de réception permet d'indiquer aux processus chargés de l'évacuation qu'un objet est prêt à être évacué. Lorsque l'objet est évacué, les processus d'évacuation émettent un accusé de réception (place P/ACK). Ceci permet de libérer la zone opératoire, donc d'autoriser le transfert d'une nouvelle pièce depuis la file d'attente en libérant un emplacement dans la file (ajout d'une marque dans la place P/CONS après le tir de la transition P/tn),

- iv) les processus d'entrée et d'évacuation d'objets sont des processus de transfert entre deux lieux du système.

Dans le cas le plus général, plusieurs types d'opérations peuvent être effectués en un lieu. On introduit ainsi la structure plus générale de la figure ci-dessous :

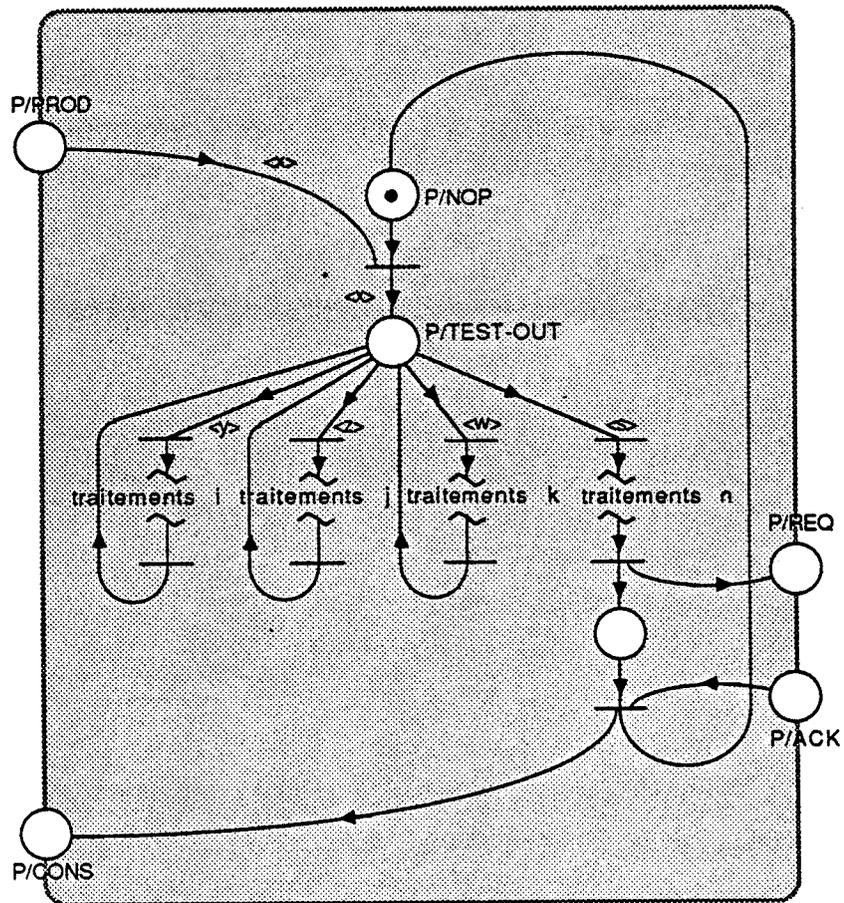


Figure III.4

Sur cette structure qu'on ne détaillera pas ici davantage apparaît une place TEST-OUT sur laquelle peuvent se brancher de façon modulaire un grand nombre de boucles de traitement. Ces boucles correspondent aux rôles fonctionnels des différentes transitions qui entourent la place étudiée du Prégraphe. C'est ainsi que sont construites les opérations relatives aux sites du Prégraphe en fonction de la nature des transitions qui entourent ces sites.

On obtient grâce à cette architecture une structure de processus facilement reconfigurable. Il est en particulier possible d'intégrer avec facilité des traitements relatifs à des modes dégradés. Les indéterminismes éventuels qui se présentent pour le tir des marques de la place TEST-OUT sont résolus par le niveau hiérarchique.

C'est sur ce schéma que sont construits différents exemples donnés dans la thèse de Jean-Pierre BOUREY (BOU 88) dont :

- les zones de stockage intermédiaire
- les machines :
  - a) machines sans tampon d'entrée/sortie
  - b) machines à un tampon d'entrée/sortie
  - c) machines à plusieurs tampons d'entrée/sortie équivalents
  - d) machines à un tampon d'entrée et un tampon de sortie.
- les assemblages et désassemblages.  
exemples des palettisations et dépalettisations.
- les repositionnements :
  - a) repositionnements internes
  - b) repositionnements externes.

Afin d'illustrer ce qui vient d'être dit et de bien faire comprendre le rôle structurel du Prégraphe, nous allons reprendre rapidement quelques-uns de ces exemples.

**Exemples**

**1. Les zones de stockage intermédiaire**

Ce sont des zones sur lesquelles les objets ne subissent pas d'opération de transformation. C'est le cas, par exemple, d'une zone de stockage de palettes vides. Cette zone est définie sur un convoyeur par le positionnement d'une butée.

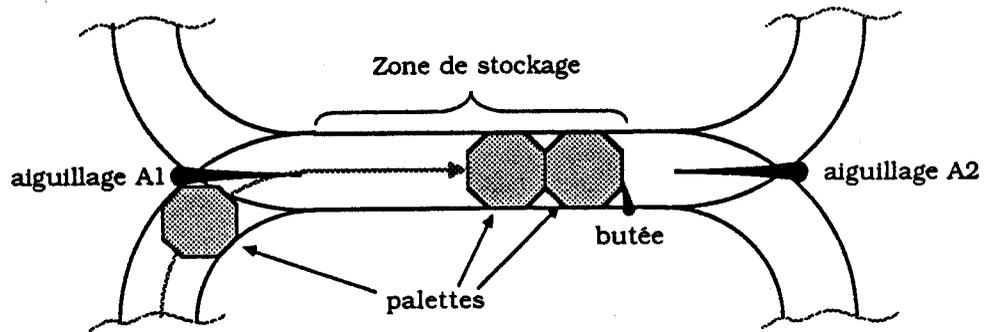


Figure III.5

La représentation de ces zones sur le Prégraphe est du type de celle qui est présentée à la figure ci-dessous.

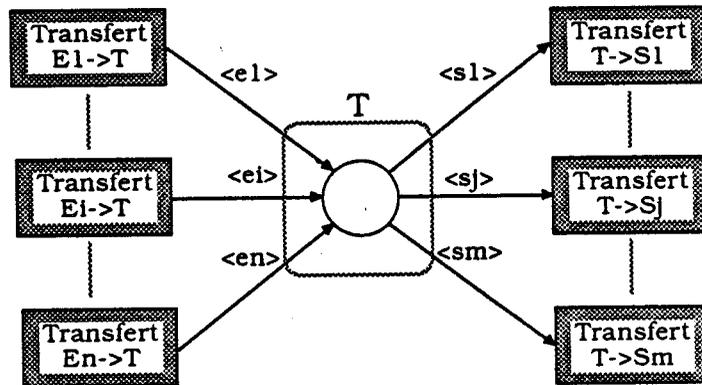


Figure III.6

- La place T est le point d'arrivée de différents transferts  $E_i \rightarrow T$ ,  $i \in \{1, \dots, n\}$  depuis un ensemble de places  $E_i$ ,  $i \in \{1, \dots, n\}$ . T est également le point de départ de différents transferts  $T \rightarrow S_j$ ,  $j \in \{1, \dots, m\}$  vers un ensemble de places  $S_j$ ,  $j \in \{1, \dots, m\}$ .
- $\langle ei \rangle$  (resp.  $\langle sj \rangle$ ) est une variable dont le domaine de couleurs représente l'ensemble des types d'objets transférés de  $E_i$  vers T (resp. T vers  $S_j$ ).

Etant donné que T est une zone de stockage intermédiaire, tout objet qui entre dans cette zone en sort à un niveau d'avancement identique dans sa gamme opératoire. De ce fait, il n'y a pas de transformation d'objet, donc pas de transformation de couleur. Ceci se traduit par l'égalité suivante :

$$\bigcup_{i=1}^n \text{dom} (\langle ei \rangle) = \bigcup_{j=1}^m \text{dom} (\langle sj \rangle)$$

Le RdP structuré et coloré qui est donné sur le schéma ci-dessous a été obtenu en utilisant le principe de la figure III.4.

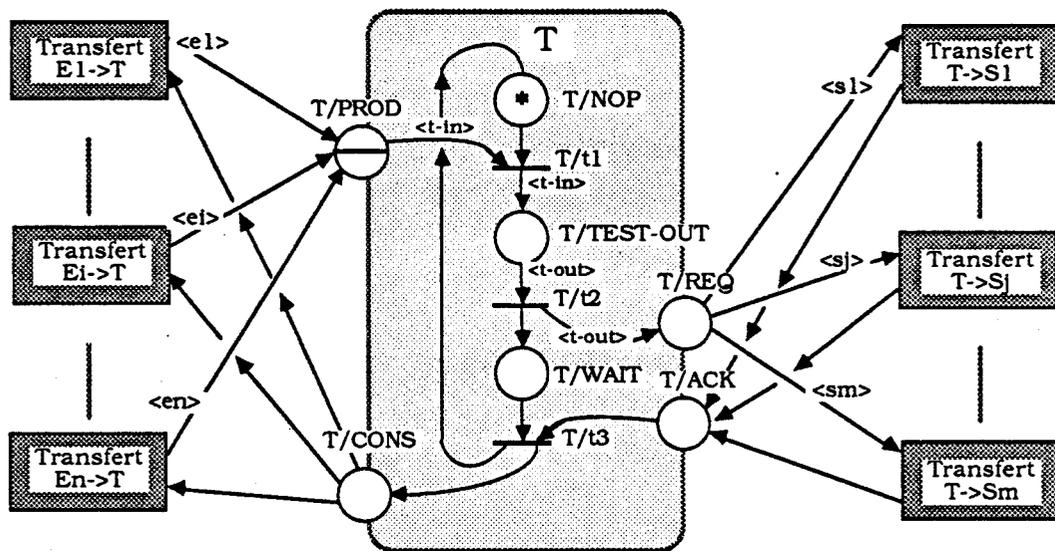


Figure III.7

Les rectangles  $E_i \rightarrow T$  (resp.  $T \rightarrow S_j$ ) correspondent au développement structuré du processus de transfert des objets de  $E_i$  vers  $T$  (resp.  $T$  vers  $S_j$ ). Nous reviendrons plus loin sur ces modules de liaison entre places.

Le schéma ainsi obtenu constitue une primitive de base du développement structuré du Prégraphe.

Dans ce RdPS apparaît la place T/TEST-OUT qui n'est pas utilisée pour les zones de stockage simple mais qui s'avère intéressante dans l'optique d'une modification a posteriori du logiciel de commande lors de l'exploitation de l'installation. En effet, si la zone de stockage est transformée en station d'assemblage, par exemple, une "boucle d'assemblage" (cf. exemple 3) pourra

être aisément ajoutée au niveau de la place TEST-OUT. Ceci permet d'accroître la flexibilité de reconfiguration du système de commande généré.

Une remarque doit être faite à propos des places de liaison du module avec l'extérieur (place PROD, CONS et REQ, ACK).

i) la liaison (T/REQ, T/ACK) permet d'aiguiller en sortie une pièce vers une zone et une seule. Ce type de liaison génère donc un conflit lorsque  $\exists i, j / \text{dom}(\langle si \rangle) \cap \text{dom}(\langle sj \rangle) \neq \emptyset$ . C'est au Niveau Décisionnel que reviendra la tâche de résoudre ce conflit.

ii) la liaison (T/PROD, T/CONS) ne permet pas, quant à elle, d'éviter que deux transferts aient lieu simultanément. Afin d'éviter ces collisions, une exclusion mutuelle est générée automatiquement entre tous les processus de transfert intervenant dans cette liaison.

## 2. Les machines

Nous allons maintenant donner une idée de la façon dont est abordée la construction des modules correspondant aux zones opératoires des machines effectuant des actions de transformation de pièces. Ces modules correspondent au développement des places du Prégraphe qui sont de la forme :

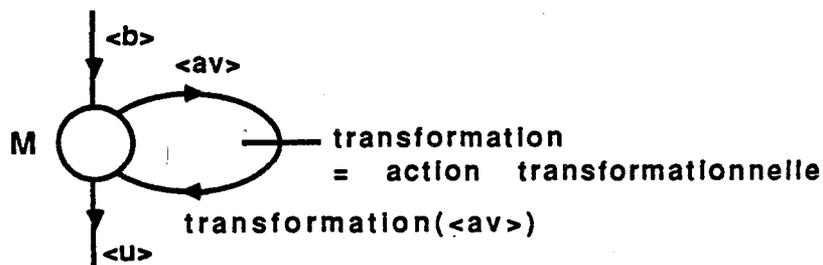


Figure III.8

### Notation :

- Dans les dessins qui vont suivre, on notera USINAGE l'opération de transformation représentée par la transition afin d'illustrer les idées. Il pourrait bien sûr s'agir de n'importe quel type de transformation.
- <b>, <av>, <ap>, <u> sont des variables libres dont les domaines de couleurs correspondent respectivement aux types de pièces :





(de fraisage (xf)

(cond

((equal xf 'pt+f-l-m-)'pt+f+l-m-)

((equal xf 'pt-f-l-m-)'pt-f+l-m-)

((equal xf 'pf-l-m-)'pf+l-m-)

((equal xf 'pt1+f-t2-l-m-)'pt1+f+t2-l-m-)))

Cette fonction est complètement construite par le logiciel d'obtention du Prégraphe.

Prenons maintenant la fonction mesures

Cette fonction est écrite en LISP sous la forme :

(de mesures (xm)

(cond

((equal xm 't+l+m-)'(t+l+m1+ t+l+m2+))

((equal xm 't1+r+t2+l+m-)'t1+r+t2+l+m+)

((equal xm 't+f+l+m-)'t+f+l+m+)

((equal xm 'f+l+m-)'f+l+m+)

((equal xm 't1+f+t2+l+m-)'t1+f+t2+l+m+)))

une mesure peut classer un objet dans plusieurs intervalles.

C'est le cas de  $t+l+m-$  qui peut conduire à  $t+l+m1+$  ou à  $t+l+m2+$  suivant le résultat de la mesure. L'image de  $t+l+m-$  par la fonction mesures sera alors représentée sous forme de la liste  $(t+l+m1+ t+l+m2+)$ .

C'est l'étape de développement du prégraphe qui remplacera cette liste par le résultat d'une fonction de choix aléatoire

(de mesures (xm)

(cond

((equal xm 't+l+m-)(aléatoire '(t+l+m1+ t+l+m2+)))

((equal xm 't1+r+t2+l+m-)'t1+r+t2+l+m+)

((equal xm 't+f+l+m-)'t+f+l+m+)

((equal xm 'f+l+m-)'f+l+m+)

((equal xm 't1+f+t2+l+m-)'t1+f+t2+l+m+)))

ou "aléatoire" est une variable aléatoire.

Par exemple si la répartition est uniforme, aléatoire aura par exemple la définition suivante en LISP.

(de aléatoire (liste)

(nth (random 0 1) liste)).

Bien entendu des fonctions aléatoires plus complexes peuvent être fournies en fonction des besoins (répartition gaussienne, etc...).

Les modules correspondant aux actions informationnelles seront ainsi du même type que les modules associés aux actions transformationnelles. Seules les définitions des fonctions de changement de couleurs telles que "usinage" ou "mesures" ont donc des traitements différents.

### 3. Exemples d'assemblages

#### 3.1 Positionnement d'assemblage

Nous allons maintenant considérer le cas d'un positionnement d'assemblage.

Il s'agit d'une action positionnelle.

Prenons le cas de la palettisation d'une pièce a sur une palette p. On a le séquençement suivant :

- arrivée d'une palette p sur la station de palettisation SP.
- transfert depuis un lieu A de la pièce à palettiser a sur la palette p.
- dépôt et éventuellement positionnement de la pièce a sur la palette p. On résume cette opération par le terme "palettisation".
- départ de l'ensemble pa de la station de palettisation.

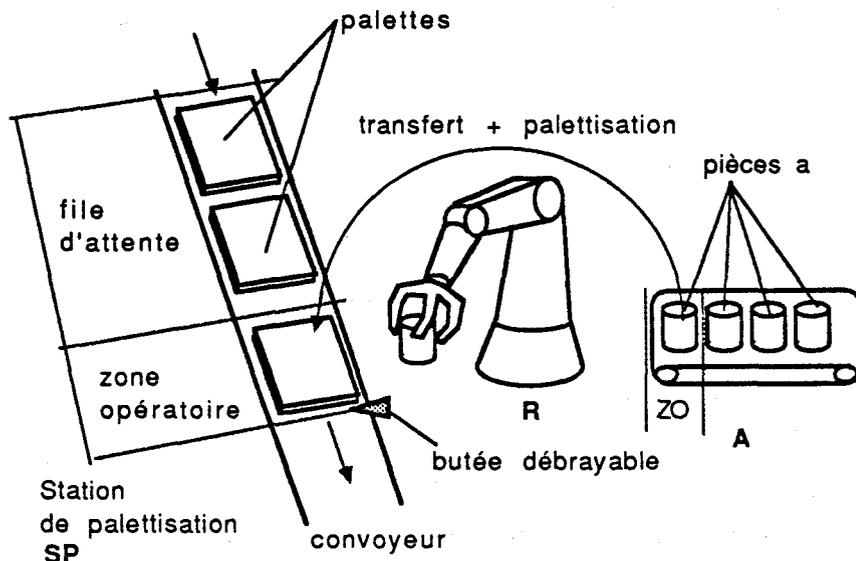


Figure III.11

Ceci est représenté sur le Prégraphe par le schéma suivant :

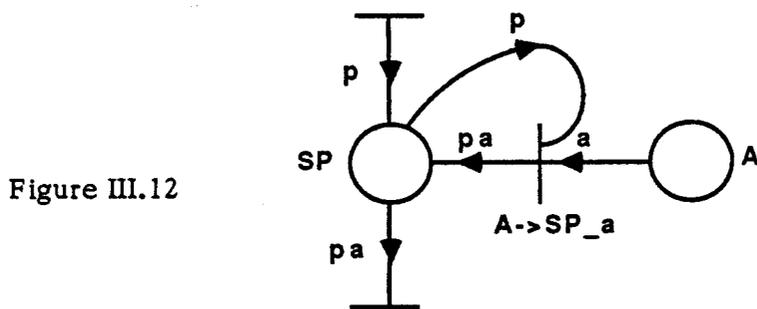


Figure III.12

En décomposant les différentes opérations (arrivée de la palette p sur SP, requêtes pour le transfert de la pièce, palettisation, évacuation de la pièce palettisée), on obtient le graphe structuré ci-dessous.

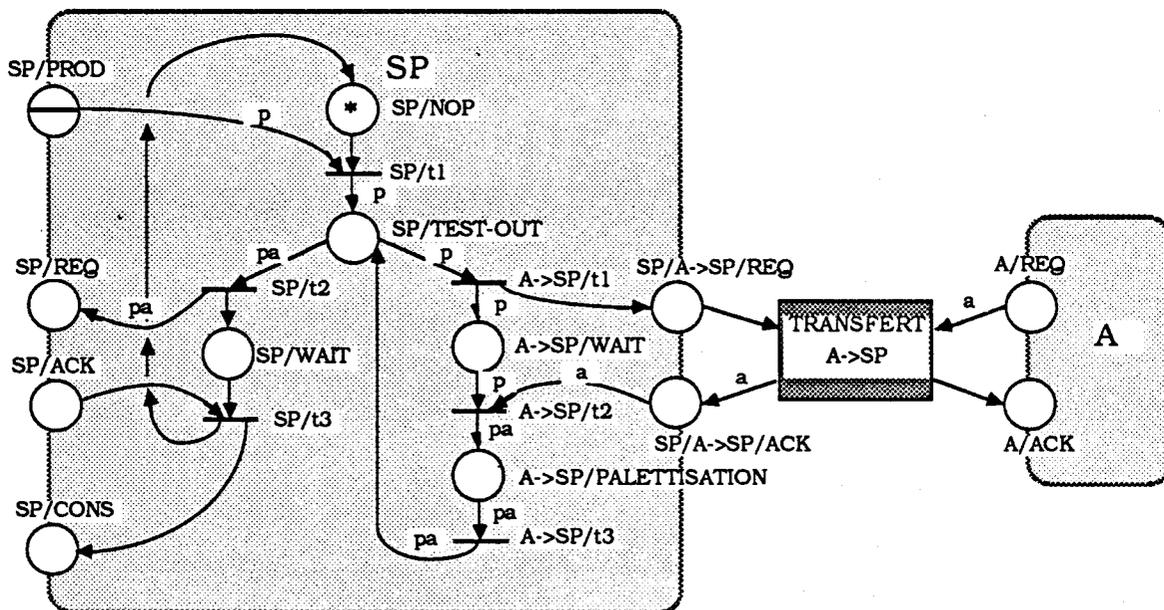


Figure III.13

### 3.2 Positionnement et solidarisation d'assemblage

Reprenons le cas de figure précédent, mais supposons cette fois que les palettes p sont remplacées par des ensembles pièce b/palette p qu'on appelle pb. Supposons que l'on assemble a sur b. La première opération consiste à positionner a sur b (action positionnelle) pour obtenir l'ensemble noté (ba)-. La deuxième opération consiste à solidariser l'assemblage (ba)- par un procédé quelconque (ex. : soudage, action transformationnelle) afin d'obtenir l'assemblage solidarisé qu'on appellera par exemple ici (ba)+.

On représente ceci de la façon suivante sur le Prégraphe :

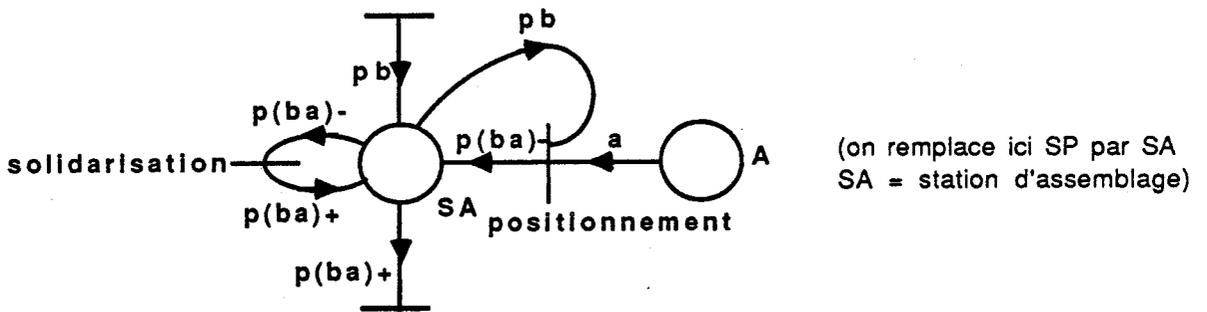


Figure III.14

Le processus structuré correspondant est donné ci-dessous

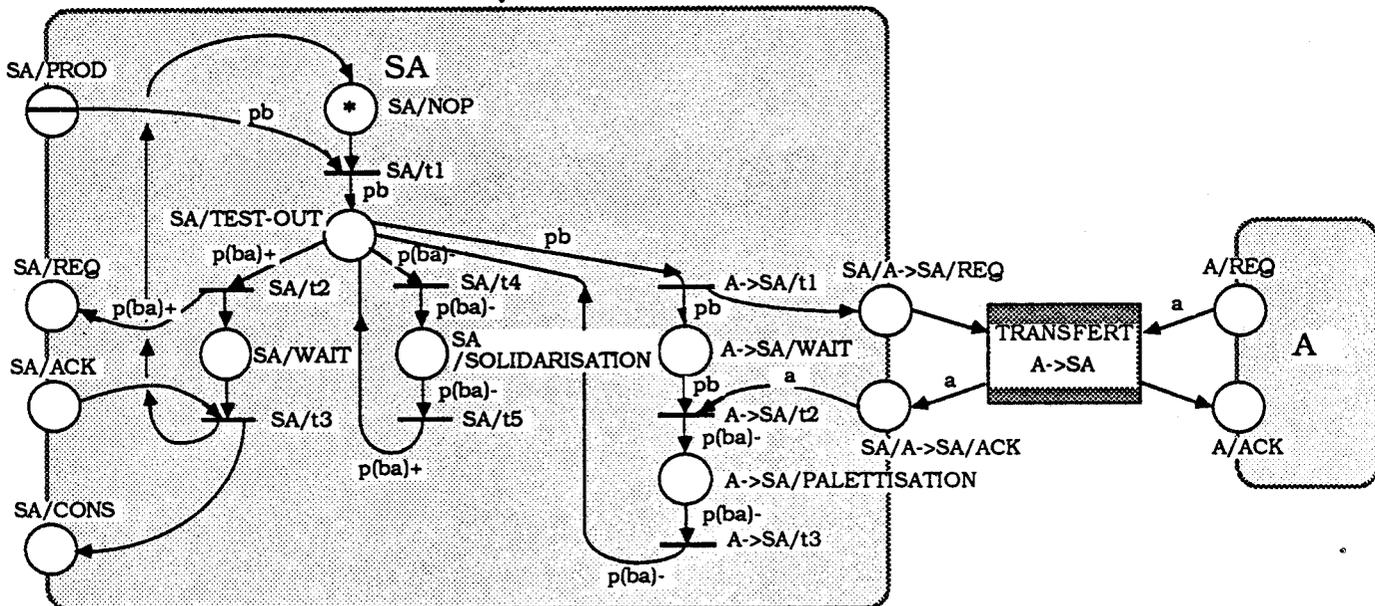


Figure III.15

Dans le cas où la station d'assemblage assure les opérations de plusieurs assemblages similaires, les arcs sont bien sûr étiquetés par des variables libres.

### 3.3 Remarques

- a) - La souplesse présentée par la place TEST-OUT permet de construire une structure répétitive multiple

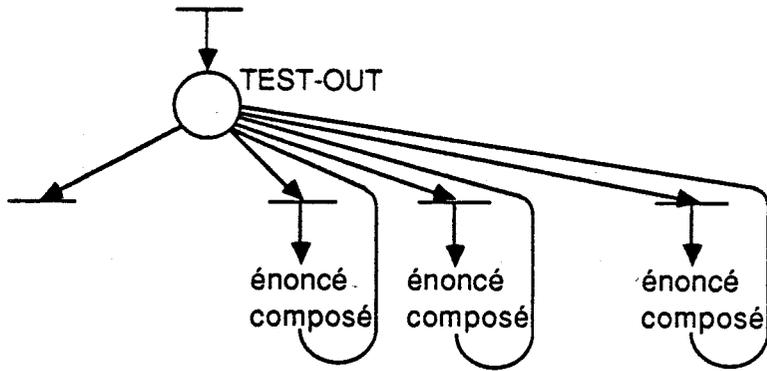


Figure III.16

Grâce à ce schéma, il est possible de représenter des stations de montage sur lesquelles sont assemblées des pièces provenant de lieux différents ou sur lesquelles sont effectuées des opérations d'assemblage avec contraintes d'antériorité.

Ces opérations se traduisent sur le Prégraphe par des constructions comme celles qui sont données ci-dessous :

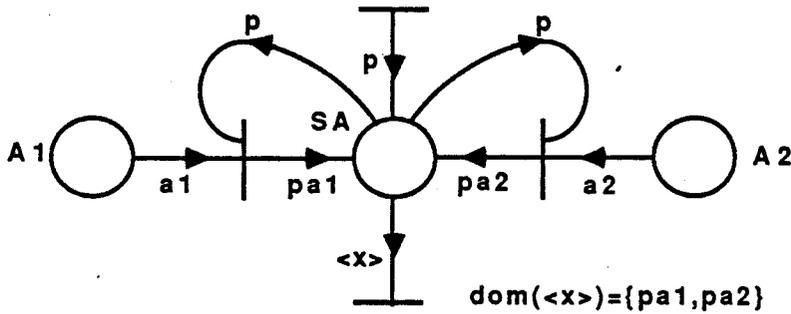
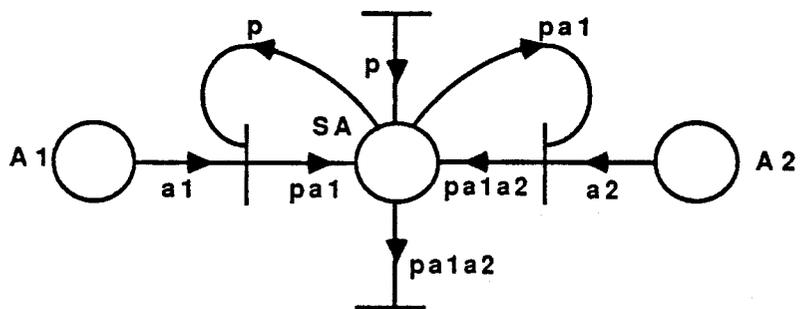


Figure III.17

Figure III.18



- Egalement, des liaisons de synchronisation vers les processus de transfert peuvent être multipliées pour décrire des cas tels que l'assemblage sur

un support p de plusieurs pièces arrivant de lieux différents et dans un ordre quelconque.

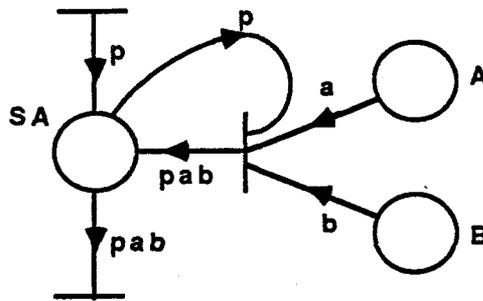
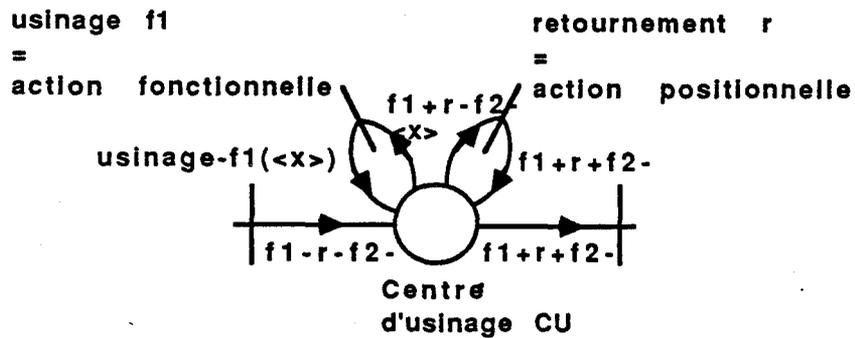


Figure III.19

b) Les desassemblages sont traités de façon symétrique.

#### 4. Les repositionnements

Les repositionnements correspondent à des actions positionnelles qui ont une structure en boucle sur une place dans le Prégraphe, comme les actions de transformation.



$$\text{dom}(\langle x \rangle) = \{f1-r-f2-, f1+r+f2-\}$$

$$\begin{aligned} \text{usage-f1} : \langle x \rangle &\longrightarrow \text{usage-f1}(\langle x \rangle) \\ f1-r-f2- &\longrightarrow f1+r-f2- \\ f1+r+f2- &\longrightarrow f1-r+f2- \end{aligned}$$

Figure III.20

Toutefois, il correspond à ces actions un développement particulier dans le processus RdP du module relatif à la place.

Prenons par exemple le cas d'un repositionnement externe à la machine.

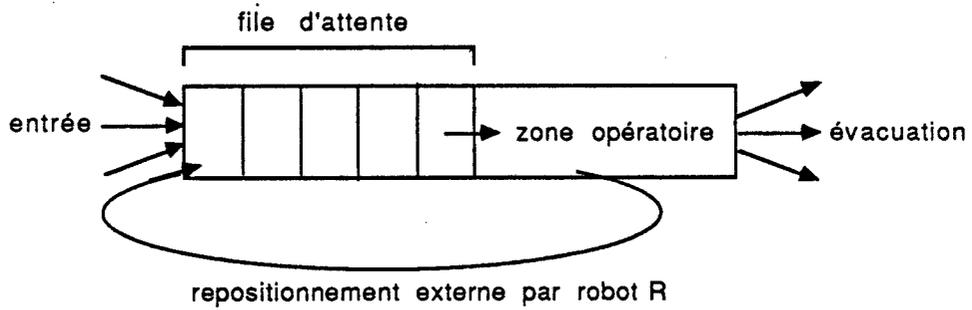


Figure III.21

le développement correspondant est donné ci-dessous sur le module RdP associé au centre d'usinage CU.

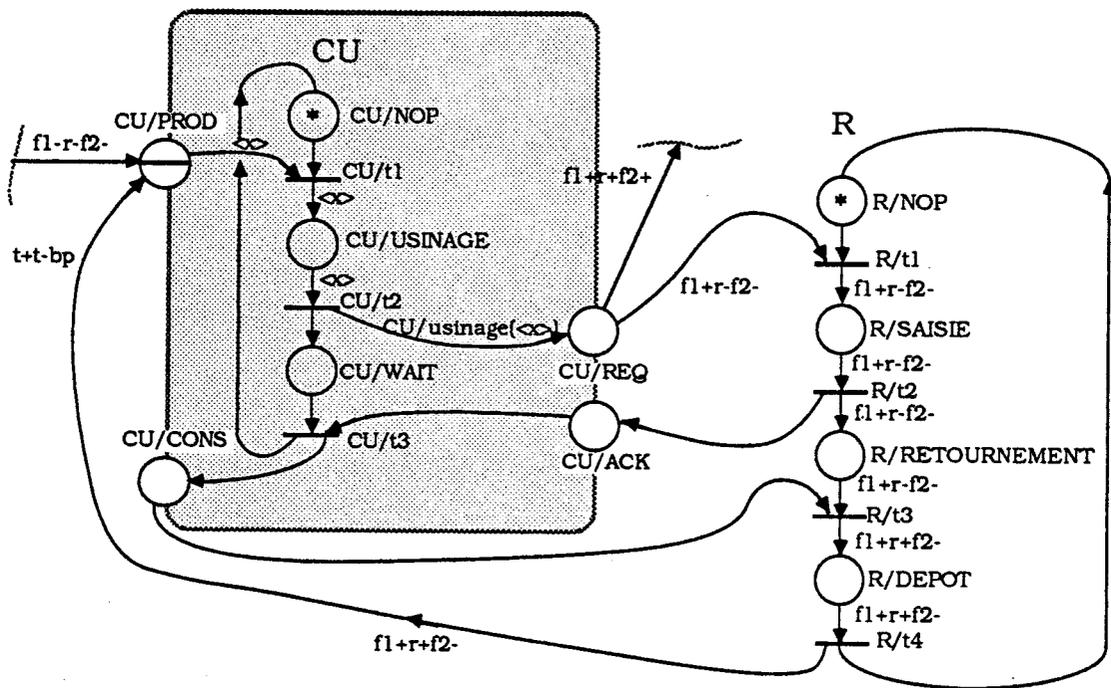


Figure III.22

Nous n'irons pas plus loin dans la description des modules associés aux places du Prégraphe. Ceci est l'objet du travail présenté dans [BOU 88]. Les explications précédentes, incomplètes, ne sont destinées qu'à faire comprendre

l'intérêt des informations portées par le Prégraphe. Notons que l'étape de développement ou structuration interroge l'utilisateur afin d'obtenir des informations supplémentaires appartenant à un niveau de spécification plus fin. La description s'enrichit donc autour du noyau qui est fourni par le Prégraphe.

### L3 LES MODULES DE LIAISON

Nous allons maintenant donner une idée rapide de la façon dont sont construits les modules de liaison.

Ces modules correspondent au développement des actions positionnelles entre places différentes. C'est le cas par exemple des transferts simples.

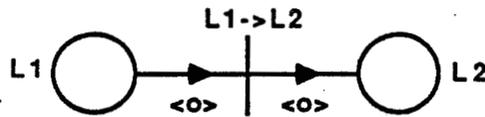


Figure III.23

$$\text{dom}(\langle o \rangle) = \{o1, o2, \dots, oN\}$$

Les transferts des pièces  $o1, o2, \dots, oN$  sont réalisés par le même système de transport. Les règles de transfert sont agrégées pour former la transition unique  $L1 \rightarrow L2$ . La façon la plus générale dont cette action est développée sur la partie commande est donnée ci-dessous :

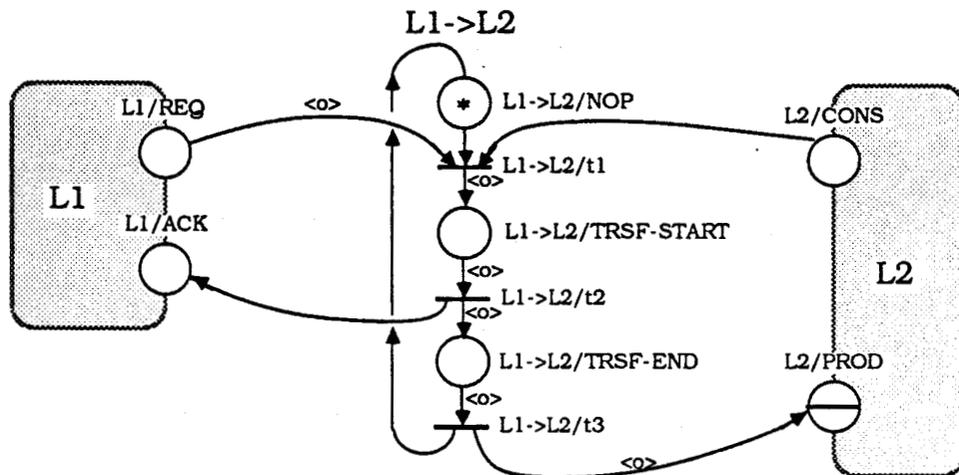


Figure III.24

Ceci correspond à une modélisation de la commande du transfert en trois phases :

### Phase 1

Pour que le transfert ait lieu, il faut :

- i) qu'il existe un objet à transférer depuis L1 (place L1/REQ)
- ii) qu'il existe un emplacement vide sur L2 (place L2/CONS)

### Phase 2

La décision de transfert ayant été prise, le transfert peut débuter. Cette phase correspond à la place notée "trsf-start".

Cette phase se prolonge jusqu'à ce que le système de transport ait quitté la zone de travail critique, permettant ainsi à l'objet suivant de L1 d'être transféré sur la zone opératoire. Ceci se traduit, sur le graphe structuré par l'émission de l'accusé de réception vers L1.

### Phase 3

Cette phase est constituée par la fin du transfert, le dépôt de l'objet sur le lieu de destination et l'arrêt du système de transport dans une position d'attente ou de repos. L'objet est alors dans la file d'attente du lieu de destination. Cette phase correspond à la phase notée "trsf-end".

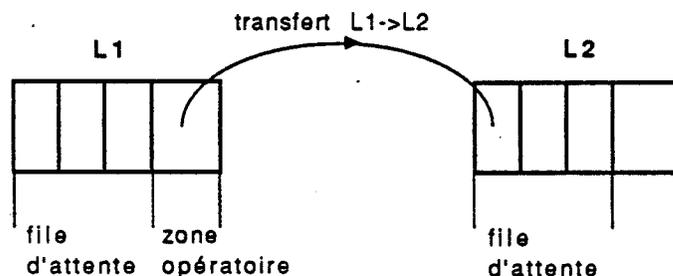


Figure III.25

Ce type de développement est utilisé pour la traduction en RdPS de la plupart des opérations de transfert. Il permet de représenter un parallélisme de fonctionnement maximal et d'assurer la sûreté d'exécution du transfert.

Il est nécessaire de faire maintenant une remarque importante.

La structure prégraphe qui représente un transfert simple ne comporte en fait

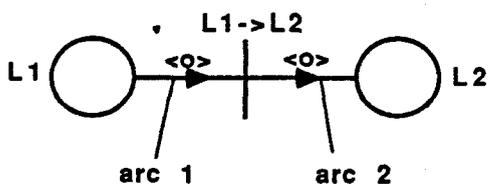


Figure III.26

que les arcs 1 et 2 qui correspondent respectivement aux arcs 1 et 2 du module de liaison développé.

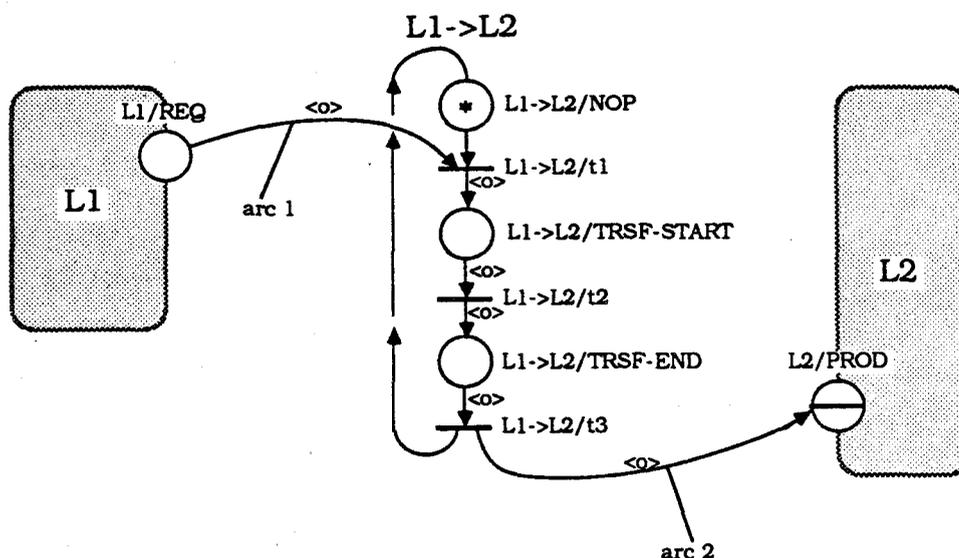


Figure III.27

Ces arcs sont indispensables au transfert de la marque colorée. Ils permettent d'avoir au sein du modèle de la Partie Commande une image de l'évolution du procédé, c'est à dire ici du transit de la pièce <0> sans avoir à développer un modèle spécifique pour le procédé (capteurs, actionneurs, pièces). Bien sûr, ces arcs n'apparaissent plus explicitement en phase d'implantation car ils sont remplacés par l'évolution du procédé réel. La coloration de ces deux arcs impose l'identification du type de la pièce dans la commande du transfert et dans la commande du processus L2. Ces arcs repèrent la fonction essentielle du transfert.

Par contre, les arcs 3 et 4 ne sont pas nécessaires au transfert.

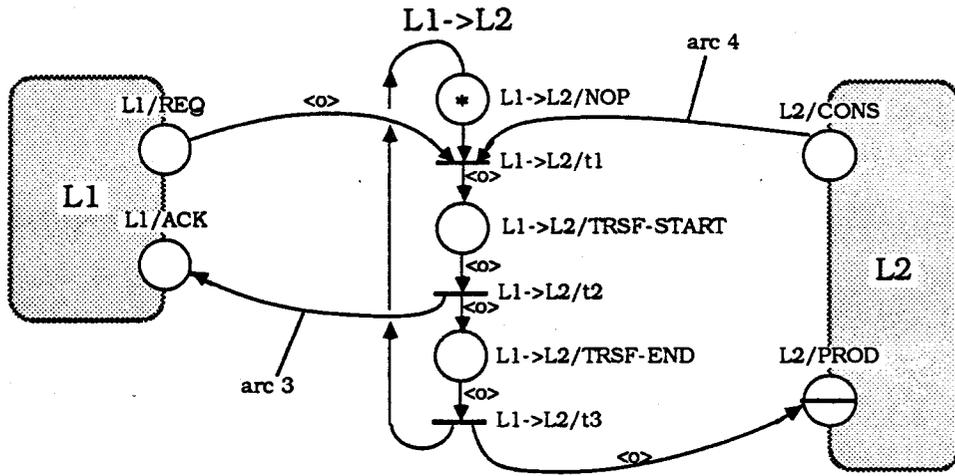


Figure III.28

L'arc 3 permet de représenter l'information "zone de sécurité de la place de départ libérée". Sa représentation impose la construction de cette information lors de l'implantation grâce à des capteurs de détection.

L'arc 4 permet de représenter l'information "place d'arrivée libre".

De la même façon que précédemment, sa représentation impose la construction de cette information lors de l'implantation.

Ces arcs qui sont d'une grande utilité ne doivent pourtant pas apparaître pour certains transferts. Prenons l'exemple d'un manipulateur rotatif R à 2 places.

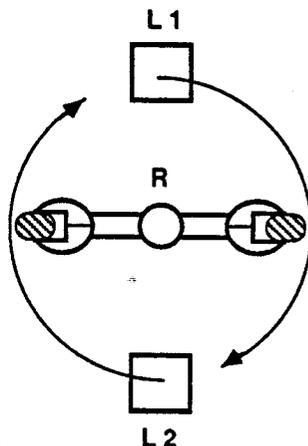


Figure III.29

Ce manipulateur permet d'effectuer les deux transferts  $L1 \rightarrow L2$  et  $L2 \rightarrow L1$  simultanément. Chacun des transferts ne nécessite pas l'information "place libre sur le poste d'arrivée" au moment initial. Les arcs 4 des 2 transferts sont donc à éliminer ainsi que les places  $L1/CONS$  et  $L2/CONS$  des processus  $L1$  et  $L2$ , sauf si elles servent à d'autres transferts. De même, l'information "place de départ libérée" est inutile pour  $L1 \rightarrow L2$  et  $L2 \rightarrow L1$ , ce qui nous amène à supprimer les arcs 3 ainsi que les places  $L1/ACK$  et  $L2/ACK$ , sauf si elles servent également à d'autres transferts.

**NB:**

Pour éviter qu'une marque provenant d'un autre transfert que  $L1 \rightarrow L2$  et  $L2 \rightarrow L1$  n'arrive sur un poste  $L1$  ou  $L2$  pendant la rotation du manipulateur (qui laisse donc temporairement les zones opératoires de  $L1$  et  $L2$  libres), on manipulera des marques "v" qui permettront de laisser les zones opératoires libres pendant la rotation de R.

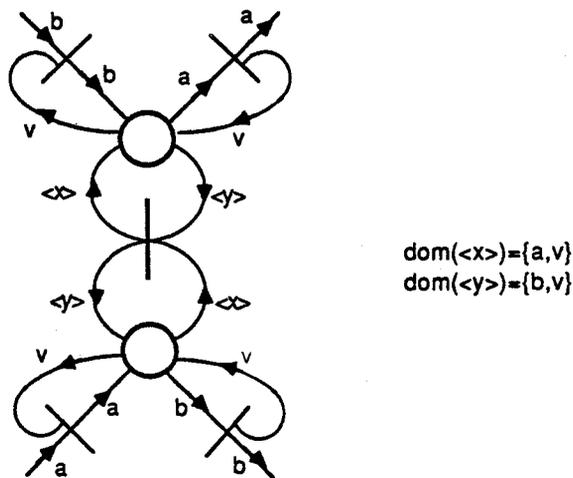


Figure III.30

Les indications concernant la présence ou l'absence des arcs 3 et 4 sont apportées par l'utilisateur. Sinon, en cas d'absence d'indications, tous les arcs sont représentés par défaut.

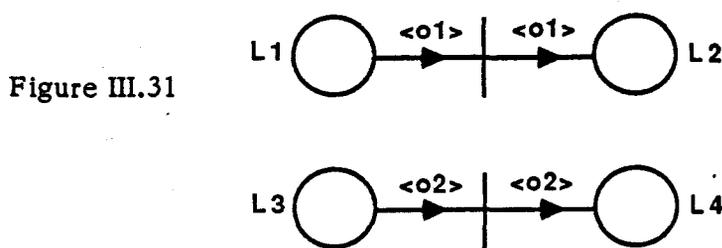
La prise en compte des contraintes de synchronisation ou d'exclusion entre transferts, liées aux caractéristiques spécifiques du système de transport, est effectuée dans une étape ultérieure que nous allons voir rapidement maintenant.

#### L4 REDUCTION DU PARALLELISME DE COMMANDE

De par sa conception, le Réseau de Petri développé issu de l'étape de structuration précédente met en évidence le parallélisme maximal du système et ne permet pas de prendre en compte directement le fait que, par exemple, deux tâches de transfert soient effectuées de façon exclusive ou, au contraire, synchronisée. Ce type de contrainte est pris en charge par une étape supplémentaire de structuration qui consiste à réduire les parallélismes.

##### I.4.1. Problèmes d'exclusion mutuelle

Prenons deux transferts :



Ces deux transferts peuvent avoir des moyens de transports indépendants. C'est le cas qui est implicitement traité par la phase de développement de la Partie Commande avant qu'intervienne toute réduction du parallélisme liée aux contraintes matérielles. Le parallélisme du système de commande obtenu est alors maximal.

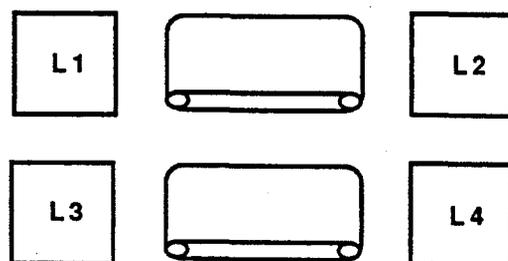
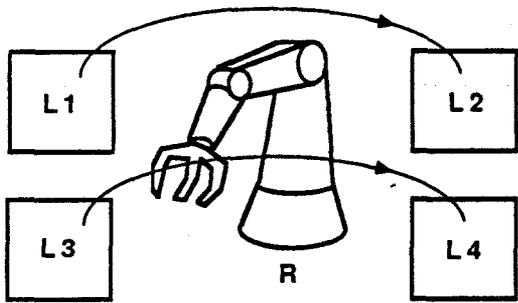


Figure III.32

Les deux opérations de transferts ont ici leur moyen propre de transport.

Ces deux transferts peuvent également utiliser une ressource critique, pour des raisons qui sont souvent liées à des problèmes de rentabilité.

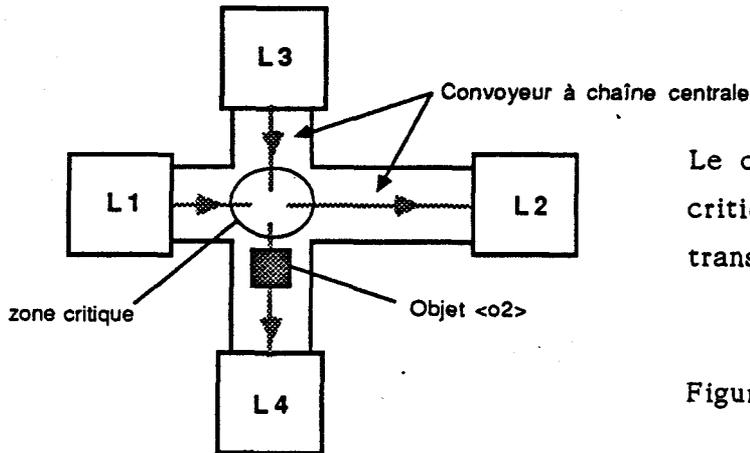


C'est le cas par exemple d'un bras robot qui permet d'effectuer les deux transferts séparément, ainsi d'ailleurs que d'autres tâches éventuelles.

Figure III.33

Ce cas de figure se rencontre par exemple dans les cellules de fonderie où un bras manipulateur central dessert un ensemble de postes qui l'entourent (moulage, refroidissement, ébavurage) ou dans des cellules d'usinage pour assurer des tâches de manutention, de chargement/déchargement de machines outils. Les temps de cycle des différentes machines permettent d'utiliser un unique bras robot en temps partagé. Le bras robot offre une grande souplesse d'application ainsi qu'une certaine facilité de reconfiguration. Un grand nombre d'applications industrielles de ce type sont présentées dans (|ENG 81|).

Un autre exemple de ressource critique est donné sur la figure ci-dessous :



Le croisement constitue une zone critique partagée par les deux transferts.

Figure III.34

Si l'utilisateur précise l'existence d'une ressource critique, quelle qu'elle soit, entre les deux transferts, ceci se traduira sur le graphe de commande par l'ajout d'une exclusion mutuelle entre les processus qui les commandent.

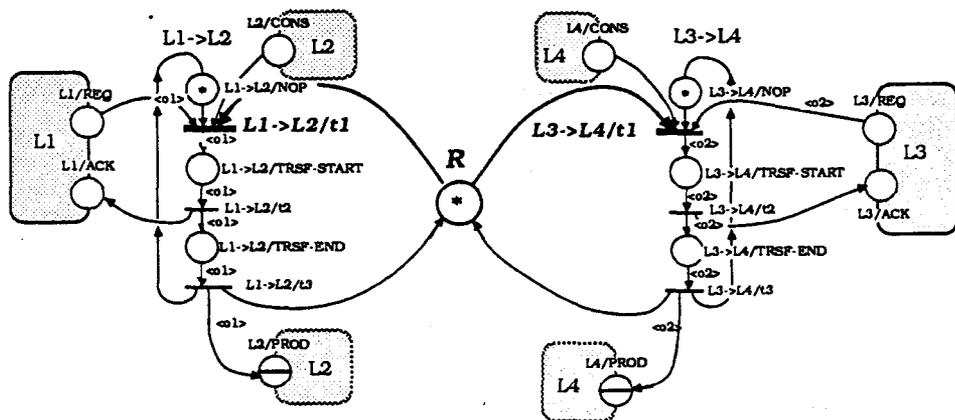


Figure III.35

I.4.2. Problèmes de synchronisation

Reprenons les deux transferts précédents.

- a) Supposons que l'utilisateur ait déjà construit le système de commande en supposant que les deux transferts soient indépendants.

Si l'utilisateur désire à ce moment étudier l'influence sur les performances du système de la synchronisation des deux transferts, un processus permettant la synchronisation peut être rapporté aux deux modules de liaisons existants. C'est ce qui est représenté sur la figure ci-dessous :

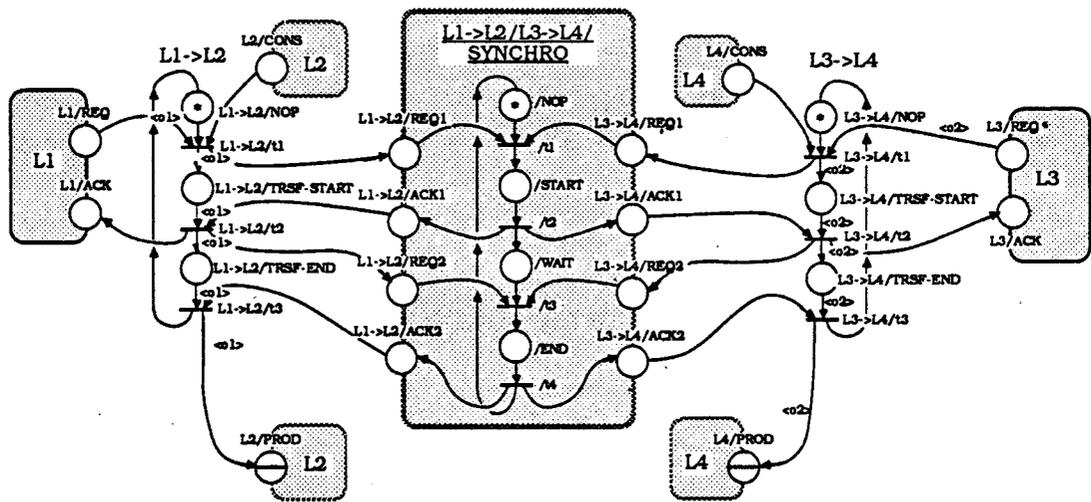
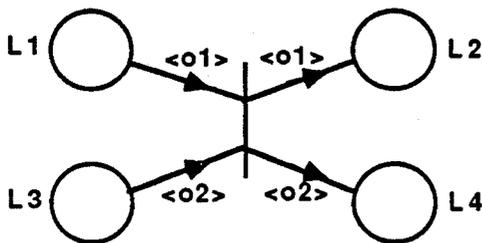


Figure III.36

- b) La synchronisation peut également être représentée directement sur le Prégraphe. Par exemple, pour les deux transferts précédents, on a la représentation suivante :



la transition L1/L3->L2/L4 étant une transition positionnelle

Figure III.37

Il existe une primitive de développement direct de cette structure. Elle est donnée ci-dessous et a l'avantage d'être plus simple et plus compacte que la précédente.

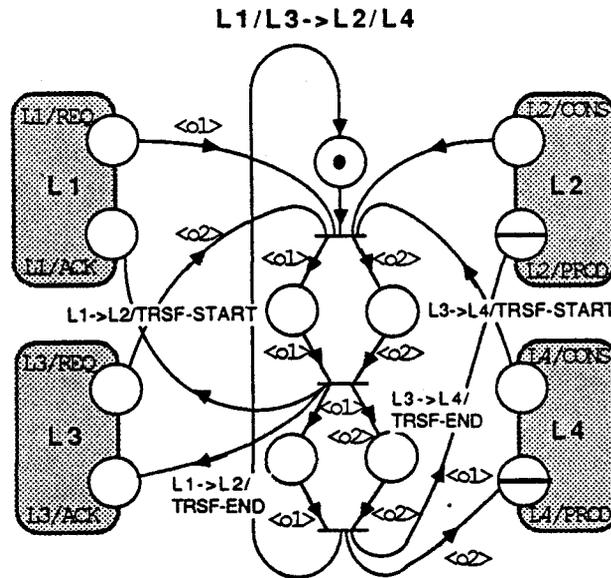


Figure III.38

- c) Les problèmes de synchronisation se posent dans tous les cas de transferts liés tels que les carrousels ou les convoyeurs à bande. Par exemple, dans son mode traditionnel, l'assemblage automatisé utilise un mode de transfert lié. En effet, l'assemblage automatisé exige en général un outillage de pose spécifique à chaque pièce. C'est pourquoi il est nécessaire de limiter le nombre de pièces à poser à chaque poste en multipliant le nombre de postes le long d'une ligne de transfert de produits. Dans le cas des transferts liés, les postes démarrent leur travail simultanément et l'accomplissent à l'intérieur du même temps de cycle global. Que ce soit dans leur forme circulaire (carrousel ci-dessous) ou dans leur forme linéaire, les transferts

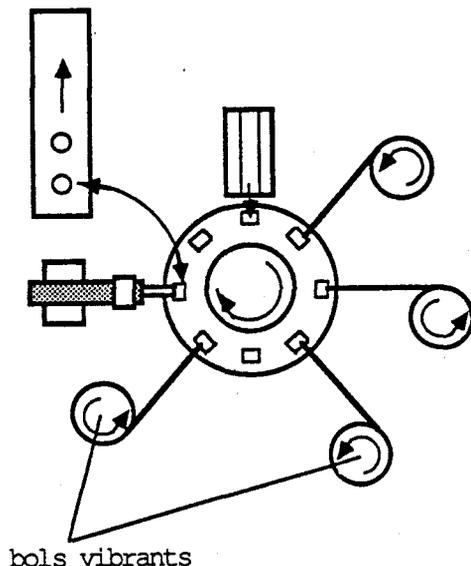


Figure III.39

liés sont généralement simples et efficaces dans le cas des productions cadencées de grande série, mais répondent mal aux besoins croissants de flexibilité et d'évolution.

C'est pour cette raison que pour des productions flexibles, on utilise le schéma des transferts libres modulaires dont on a déjà parlé au début du chapitre II. Les postes opèrent indépendamment les uns des autres sur des pièces véhiculées en général sur des palettes qui circulent librement de poste en poste.

Sur les transferts libres "en ligne", les opérations se font sur la ligne principale de circulation. Au contraire, sur les transferts libres "hors ligne" (dessin ci-dessous), les opérations se font hors de la ligne principale de circulation. Chaque poste peut ainsi choisir la palette "porte-pièce" sur laquelle il doit opérer, plusieurs variantes de produits pouvant ainsi être traitées simultanément. La flexibilité obtenue est complétée d'une grande facilité d'évolution de l'unité par adjonction ou remplacement de modules.

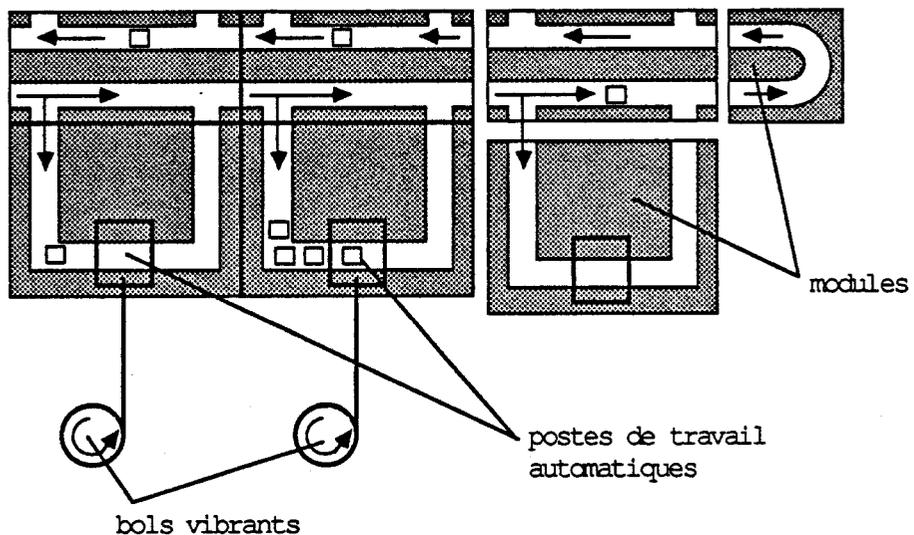


Figure III.40

- d) La prise en compte des contraintes a été présentée ici de façon très résumée sur des opérations de transfert. Les traitements précédents peuvent également porter sur des opérations de types différents (ex. : transfert + usinage). Toutes ces contraintes d'exclusion ou de synchronisation sont bien sûr réalisées de façon interactive et assistée.

## L5 NIVEAUX DE STRUCTURATION TERMINAUX

La partie commande issue du processus de structuration précédent aboutit en fait à un graphe de coordination de macro-places de commande. La plupart des places telles que les places USINAGE, TRSF-START, TRSF-END, etc... représentent elles-mêmes des séquences de commande simples. Le développement progressif des macro-étapes permet de descendre au niveau des (pré-)actionneurs et des capteurs.

Le graphe développé obtenu jusqu'à maintenant est un modèle Réseau de Petri. Il faut noter qu'il n'est pas nécessaire de développer les macro-places avec le même outil de représentation. Ainsi, par exemple, l'implantation Grafcet peut comporter des étapes qui envoient des ordres à des commandes numériques locales.

Afin d'illustrer cette notion de structuration progressive, prenons un exemple simple tiré de la cellule de production installée dans le laboratoire de Mécanique de l'I.D.N.

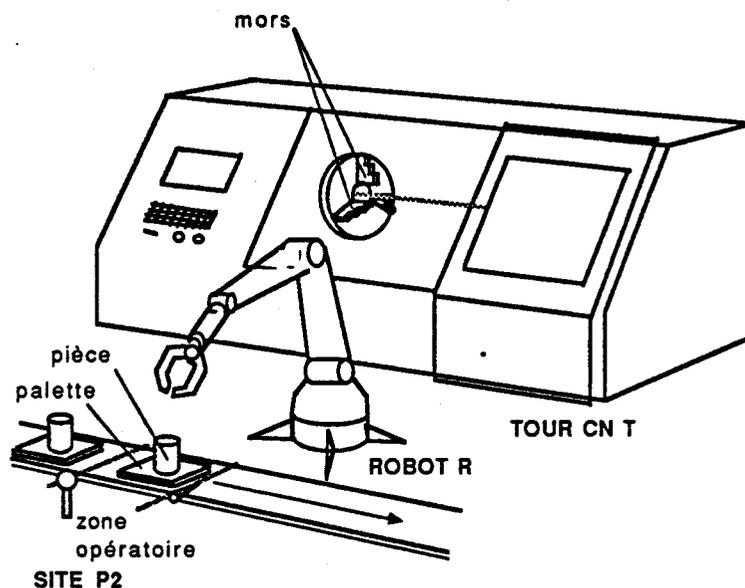


Figure III.41

Intéressons nous aux étapes de chargement et déchargement du tour, d'usinage, et de retournement d'une pièce en vue d'un second usinage.

Supposons que la première étape de conception ait conduit au Prégraphe ci-après :

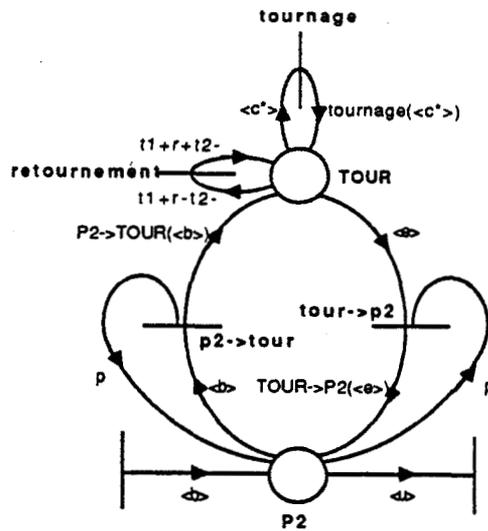


Figure III.42

avec  $\text{dom}(\langle b \rangle) = \{pt-, pt1-r-t2-\}$   
 $\text{dom}(\langle c \rangle) = P2 \rightarrow T (\text{dom}(\langle b \rangle)) = \{t-, t1-r-t2-\}$   
 $\text{dom}(\langle c^* \rangle) = \text{dom}(\langle c \rangle) \cup \{t1+r+t2-\} = \{t-, t1-r-t2-, t1+r+t2-\}$   
 $\text{dom}(\langle e \rangle) = \text{tournage}(\text{dom}(\langle c^* \rangle)) - \{t1+r-t2-\} = \{t+, t1+r+t2+\}$   
 $\text{dom}(\langle u \rangle) = T \rightarrow P2 (\text{dom}(\langle e \rangle)) = \{pt+, pt1+r+t2+\}$

et où les fonctions de changement de couleur sont données simplement par les correspondances suivantes : (A titre indicatif, ces fonctions sont données sous forme Le\_Lisp)

(de  $P2 \rightarrow T$  (b))

```
(cond
  ((equal b 'pt-) 't-)
  ((equal b 'pt1-r-t2-) 't1-r-t2-)))
```

de  $T \rightarrow P2$  (e)

```
(cond
  ((equal e 't+) 'pt+)
  ((equal e 't1+r+t2+) 't1+r+t2+)))
```

(de  $\text{tournage}(c^*)$ )

```
(cond
  ((equal c* 't-) 't+)
  ((equal c* 't1-r-t2-) 't1+r-t2-)
  ((equal c* 't1+r+t2-) 't1+r+t2+)))
```

Le développement effectué à partir du Prégraphe va conduire au graphe RdPSC suivant constitué du module associé à la place P2,

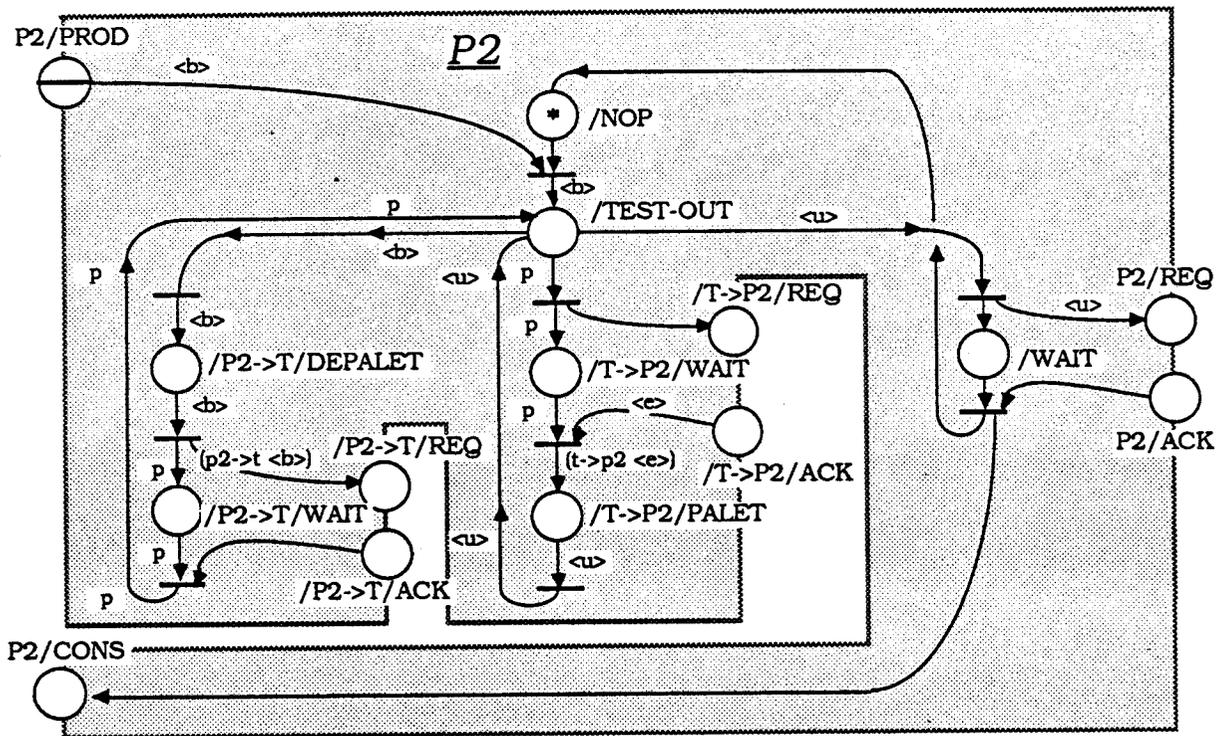


Figure III.43

et constitué des modules respectivement associés au TOUR et aux liaisons P2→TOUR et TOUR→P2 :

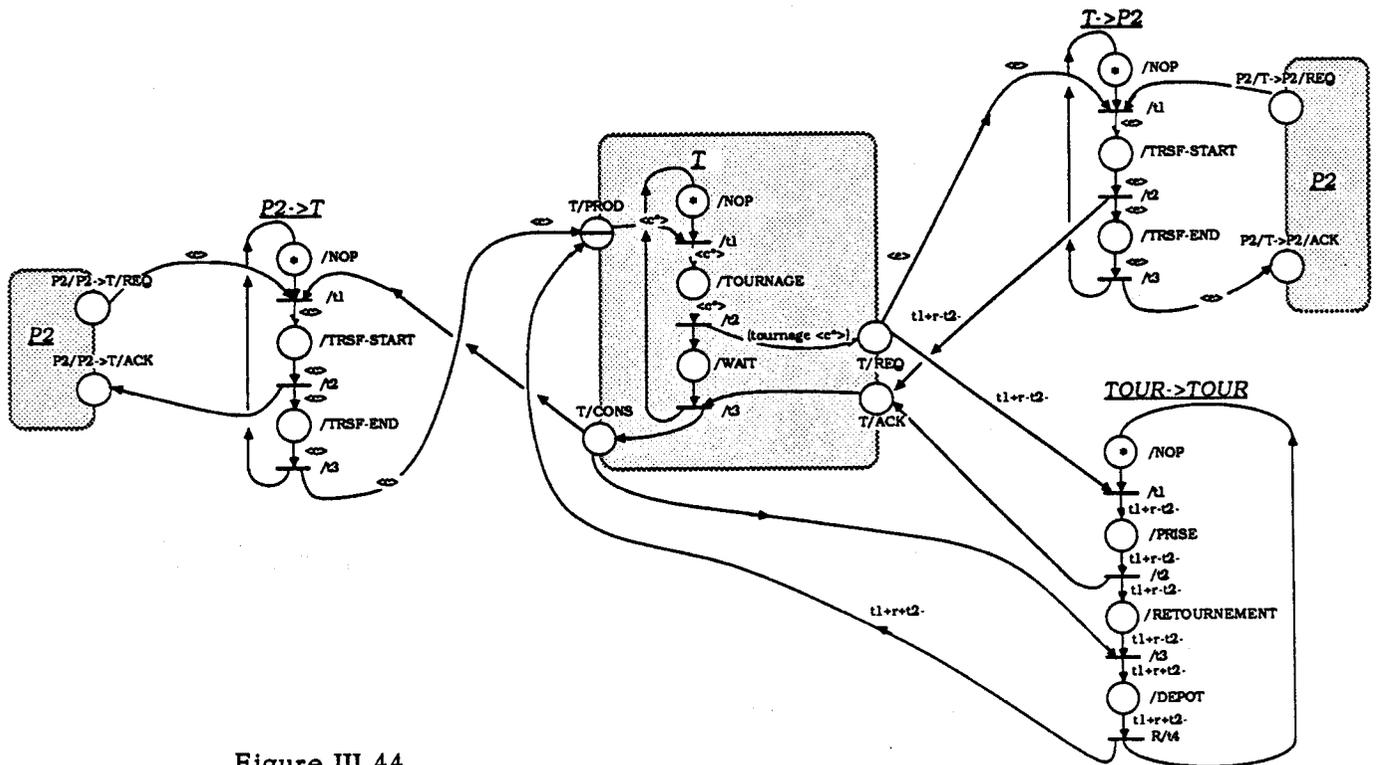


Figure III.44

Intéressons nous par exemple aux places TRSF-START et TRSF-END des processus de transfert  $P2 \rightarrow T$  et  $T \rightarrow P2$ . Les figures ci-dessous montrent comment peut être représenté sous forme RdP interprété le développement de ces macro-places.

Transfert  $P2 \rightarrow TOUR$

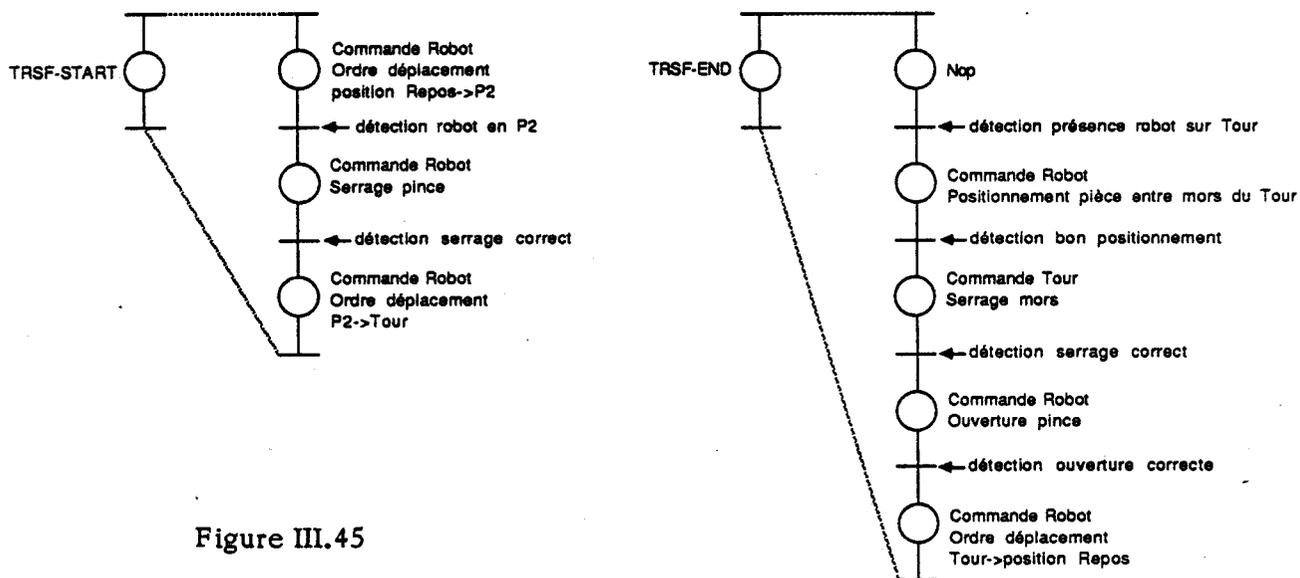


Figure III.45

Transfert TOUR  $\rightarrow P2$

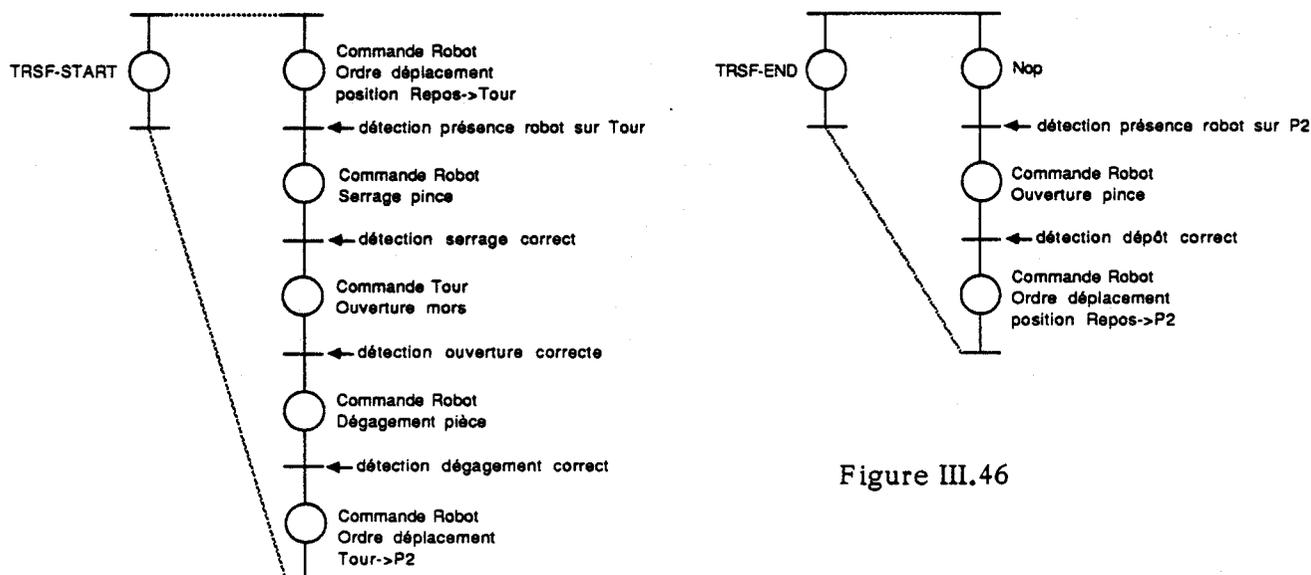


Figure III.46

(N.B. : Les places désignent des ordres donnés aux actionneurs, les transitions désignent des évènements provenant des capteurs. Dans cet exemple est représenté volontairement un grand nombre de tests de capteurs. La réalité industrielle montre qu'il n'est pas souhaitable d'abuser du nombre des capteurs. Ceux-ci sont à l'origine d'un grand nombre de pannes. On remplace alors ces évènements externes par une temporisation interne de la partie commande).

Considérons maintenant l'étape USINAGE du processus associé à la place TOUR. Supposons que cette place soit destinée à être une macro-commande adressée par exemple à une commande numérique, lorsque le graphe sera implanté sur des automates. Il n'y a pas lieu ici de développer davantage cette place puisque l'implantation répond à une autre forme de représentation.

Le développement des macro-places s'adresse à des démarches de C.F.A.O. connues. En particulier, dans le Laboratoire de Mécanique de l'I.D.N., une équipe d'informaticiens travaille actuellement sur la génération assistée et le téléchargement de programmes d'usinage sur la commande numérique des machines (DNC Direct Numerical Control). Cette démarche est maintenant classique. L'apport du projet CASPAIM se situe, lui, dans la conception assistée du graphe de commande de coordination de l'unité de production.

Citons pour terminer ce paragraphe un outil logiciel de conception d'applications pour automates programmables. Il s'agit de l'Atelier Logiciel GRAL de la Société SYGRAL (LES ULIS). Ce système industriel recouvre un champ d'application qui est complémentaire de la démarche CASPAIM.

GRAL utilise une méthode de programmation structurée et organisée de logiciels pour automates programmables. La méthode repose sur trois niveaux :

- L'analyse fonctionnelle où chaque application est découpée en tâches élémentaires.
- La conception de l'application qui consiste à décrire chacune des tâches à partir de trois composants de base (diagrammes de séquences, boîtes fonctionnelles et tâches littérales).

- La compilation et la mise au point sur automate cible en tenant compte des contraintes de l'automate. Les compilateurs disponibles en Septembre 1987 concernaient les automates programmables des gammes PB et SMC de APRIL.

En exploitation, GRAL peut également servir de système de détection de panne de premier niveau par visualisation en dynamique des diagrammes de séquences, des boîtes fonctionnelles, des tâches littérales et des variables des automates programmables.

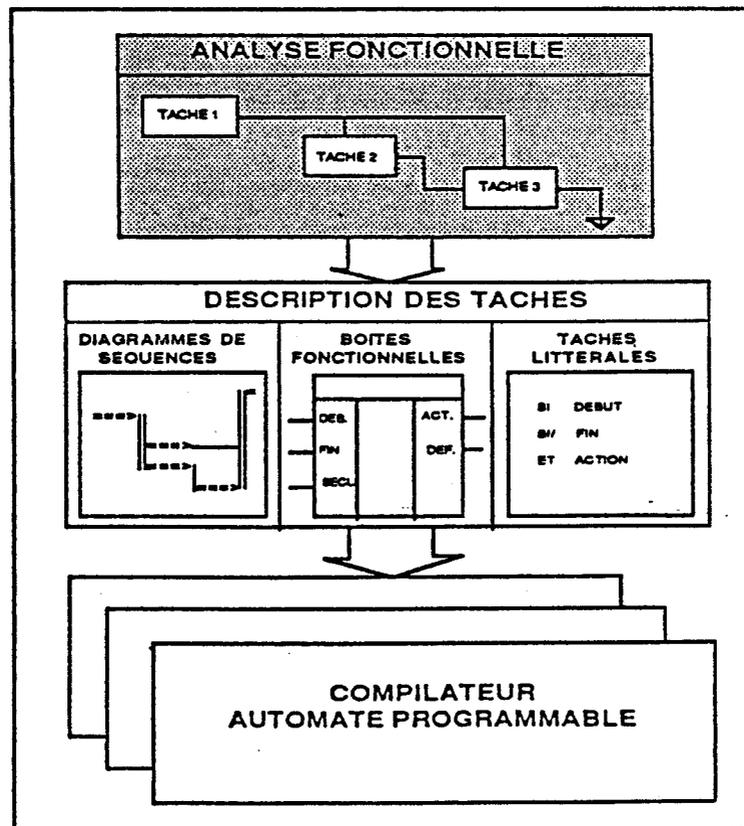


Figure III.47

A ce sujet, précisons qu'une méthode de traduction en GRAFCET des graphes RdPSAC de la Partie Commande existe pour le projet CASPAIM. Moyennant le respect de règles énoncées en annexes, la transposition est systématique et ne pose pas de problème de programmation. Elle a été réalisée pour le moment à la main sur des automates TSX 67 de la TELEMECANIQUE. La transposition pourrait être automatisée si l'on disposait d'un code source GRAFCET compilable sur les automates programmables cibles, ce qui n'est pas le cas pour le moment. La TELEMECANIQUE a actuellement à l'étude un compilateur C pour certains de ses contrôleurs, ce qui apportera une première réponse à ce problème.

## L6 CONCLUSION SUR LA STRUCTURATION

La démarche de conception assistée de la Partie Commande qui vient d'être présentée s'identifie à un processus de compilation avec enrichissement.

Prenons l'exemple d'un langage de haut niveau tel que PASCAL. Un certain nombre de structures syntaxiques qui peuvent être représentées sous la forme des diagrammes de CONWAY définissent la grammaire autorisée du langage. De même, la construction du PREGRAPHE utilise des structures et des informations qui doivent être comprises par l'étape de DEVELOPPEMENT. Les compilateurs PASCAL vérifient la syntaxe du texte programme source et le traduisent en code exécutable. De même, l'étape de DEVELOPPEMENT DU PREGRAPHE construit, à partir du PREGRAPHE, le code RdP correspondant à la Partie Commande. Toutefois, les étapes de compilation habituelles sont automatiques. Ici la compilation interroge l'utilisateur afin d'enrichir progressivement la traduction. Un projet envisagé au L.A.I.I. est de permettre à l'utilisateur de construire à partir de spécifications de haut niveau les fonctions mêmes de construction des modules RdP. L'utilisateur pourrait lui-même ainsi définir certaines parties du compilateur RdP.

Pour le moment, notons que la démarche de construction de la Partie Commande est tout à fait structurée. La spécification des solutions est progressive et intervient en même temps que la spécification des problèmes qui, ainsi, ne sont pas introduits prématurément. L'utilisateur peut ainsi définir son dossier de conception progressivement, en apportant ses spécifications au moment exact où elles doivent intervenir. Il est possible, à l'issue de la démarche de conception, de retourner à l'utilisateur le dossier de conception structuré et hiérarchisé correspondant à ses réponses successives aux questions du logiciel. Les informations interviennent ainsi dans l'ordre de leur importance, sans être mélangées.

Enfin, remarquons l'importance du PREGRAPHE dans cette démarche dont il constitue le schéma directeur. Le PREGRAPHE constitue la base de travail de l'étape de développement. Il fait apparaître la plupart des variables et des fonctions qui seront utilisées sur la Partie Commande. Il constitue l'ossature de cette Partie Commande.

Avant la création et l'informatisation de la méthode de conception, le travail était effectué à la main. Les erreurs étaient multiples et difficilement détectables à l'examen des résultats de simulation.

Le travail réalisé maintenant est encore un peu fastidieux dans la mesure où l'utilisateur est amené à définir avec précision ses exigences, mais il est plus sûr, rigoureux et plus rapide.

Signalons que l'évolution actuelle tend vers un enrichissement du Prégraphe destiné à une version simulable (voir Partie III de ce chapitre, § III.2.2.b.3). Un grand nombre de questions qui étaient posées jusqu'à maintenant au cours de la phase de développement structuré seront posées lors de la construction de l'extension simulable du prégraphe (capacité et gestion des tampons, réduction des parallélismes : ressources communes, temporisations, ...).

Pour terminer, mentionnons la réalisation de deux interfaces graphiques pour la chaîne de conception de la Partie Commande.

- La première approche (|PRE 87| |TAI 87|) part du fichier de description littéral du Réseau de Petri de la Partie Commande. Elle permet de sortir graphiquement chacun des processus avec le nom des places et des variables (réalisé sur BFM 86 et traceur HP 7475A).
- La deuxième approche initialisée dans le cadre de ce mémoire a eu pour point de départ l'idée d'associer au Prégraphe une image graphique générée à partir d'un terminal virtuel géré par la couche langage objet de Le-Lisp.



Cette démarche a plusieurs buts :

- 1) Le premier but est bien sûr de visualiser la description jusqu'alors littérale d'un Prégraphe.
- 2) Le deuxième but est de visualiser à plus long terme l'évolution d'une simulation effectuée sur une version simulable du Prégraphe, ou sur le modèle même de la Partie Commande issue du Prégraphe. (Dans ce dernier cas, la correspondance entre les places du Prégraphe et les processus de la Partie Commande est connue. Le passage d'une marque dans un processus sera donc visualisée par la présence d'une marque dans la place correspondante du Prégraphe. On aura ainsi une "macro"-visualisation de la simulation fine, associée à des possibilités de zooms sur les processus).

La réalisation pratique d'une partie de ce travail a été prise en charge par deux étudiants (|HEB 88|, |HEI 88|), ce qui permet à l'heure actuelle de disposer déjà d'une chaîne de conception relativement complète pour la Partie Commande. Ce travail a été entrepris dans un premier temps et dans un but expérimental sur Mac Intosh SE et sera transposé par la suite sur un poste de travail plus puissant (VS 2000/DIGITAL).

## II. LA QUESTION DE L'EXTENSION DU MODELE DE LA PARTIE COMMANDE

### II.1 LE PROBLEME POSE

La modélisation d'une unité de production flexible s'articule dans le projet CASPAIM autour de trois pôles : la Partie Commande, le Niveau Décisionnel et le Procédé.

Le Procédé est, rappelons-le, constitué de la Partie Opérative et de l'ensemble des équipements et matériels intervenant dans le processus de Production (machines, pièces, outils, automates, programmes, ...).

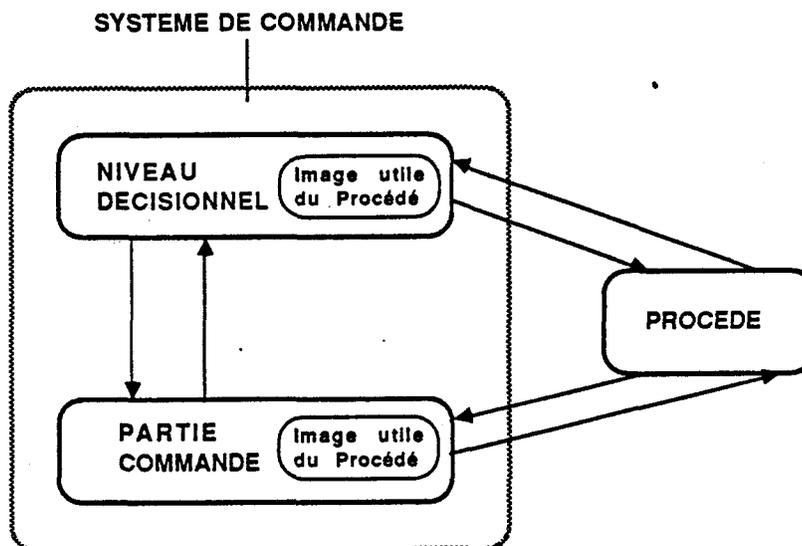


Figure III.49

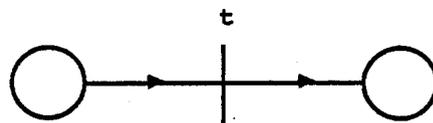
Une image utile du Procédé est intégrée au système de Commande pour lui permettre d'évoluer sans surcharger ses consultations du Procédé.

Le modèle du Procédé a, quant à lui, la tâche de représenter l'évolution asynchrone du Procédé réel. Il doit bien être distingué des images intégrées au système de Commande.

Le modèle de la Partie Commande est actuellement le noyau central de la modélisation dans le projet. La tentation est grande d'accroître la complexité de ce modèle afin de pouvoir effectuer de premières évaluations en se dispensant de développer le modèle complémentaire propre du Niveau

Décisionnel et de l'image du Procédé dont il a besoin ainsi que le modèle propre du Procédé. Un réflexe qui est alors courant dans la conception des systèmes de Commande est d'intégrer une représentation de plus en plus fine du Procédé au sein du modèle de la Partie Commande. On avait déjà recours à la coloration des marques, à la temporisation de places et de transitions, à la représentation des arcs de liaisons (absents du GRAFCET). La reconnaissance de plus en plus fine des pièces nécessaire à la commande exigerait alors d'introduire une image plus complexe au travers du marquage du réseau de Petri de la Partie Commande. On peut introduire ainsi des outils tels que les réseaux à prédicats ou même des réseaux à base de filtres permettant de filtrer le passage de marques à structure très complexe.

Exemple :



Filtre=((support -plateau)  
(sur-plateau pièce)  
(pièce boîtier)  
(nombre ?n)  
(1<?n<=21))

Figure III.50

La transition t ne peut être franchie que par des marques qui peuvent s'unifier avec le filtre. Le filtre est constitué de prédicats, d'ordre  $\leq 1$  dans l'exemple ci-dessus, les variables étant précédées du symbole "?". La représentation du filtre sous forme de listes peut bien sûr être assurée par d'autres moyens.

Cette extension de la modélisation n'est pas souhaitable.

En effet :

- i) L'accroissement de la complexité du modèle de la Partie Commande fait qu'il est rapidement difficile et même impossible de l'implanter convenablement sur des dispositifs technologiques industriels répondant aux exigences industrielles de temps de cycle.

- ii) La complexité de la modélisation doit être décomposée et ne pas reposer sur le seul modèle de la Partie Commande. L'analyse des problèmes au sein du projet CASPAIM a permis de dégager trois pôles de plus forte connexité : Partie Commande, Niveau Décisionnel, Procédé. Dans la mesure du possible et de l'efficacité résultante, il est plus raisonnable de partager la complexité de modélisation sur ces trois pôles.

Ainsi, supposons que le Niveau Hiérarchique ait besoin d'informations concernant le procédé pour résoudre un indéterminisme de la Partie Commande. On sait que la Partie Commande a également besoin, pour son fonctionnement, d'une image du procédé. L'erreur serait de compliquer le modèle de la Partie Commande afin d'y intégrer une image du procédé plus complexe et consultable par le Niveau Décisionnel. Il est préférable de ne pas tout mêler et de construire une autre image du procédé, spécifique au Niveau Décisionnel. Les deux niveaux du Système de Commande portent alors chacun une image du procédé limitée à leurs propres besoins, ce qui permet de limiter la complexité du modèle de la Partie Commande. Le problème est alors bien sûr d'assurer la cohérence des images du procédé entre elles et vis à vis du modèle même du procédé.

## **IL.2 PRESENTATION D'UN EXEMPLE**

Nous allons présenter maintenant un exemple dans lequel l'image du procédé nécessaire à la commande, surtout utile au Niveau Décisionnel, est trop complexe pour pouvoir être représentée sur le modèle de la Partie Commande.

Il s'agit d'un stockeur rotatif SR comportant six étagères à deux emplacements. Chaque emplacement peut recevoir un plateau. Le stockeur rotatif est placé comme lieu de stockage entre un convoyeur CV et une station de chargement et déchargement flexible SCDF.

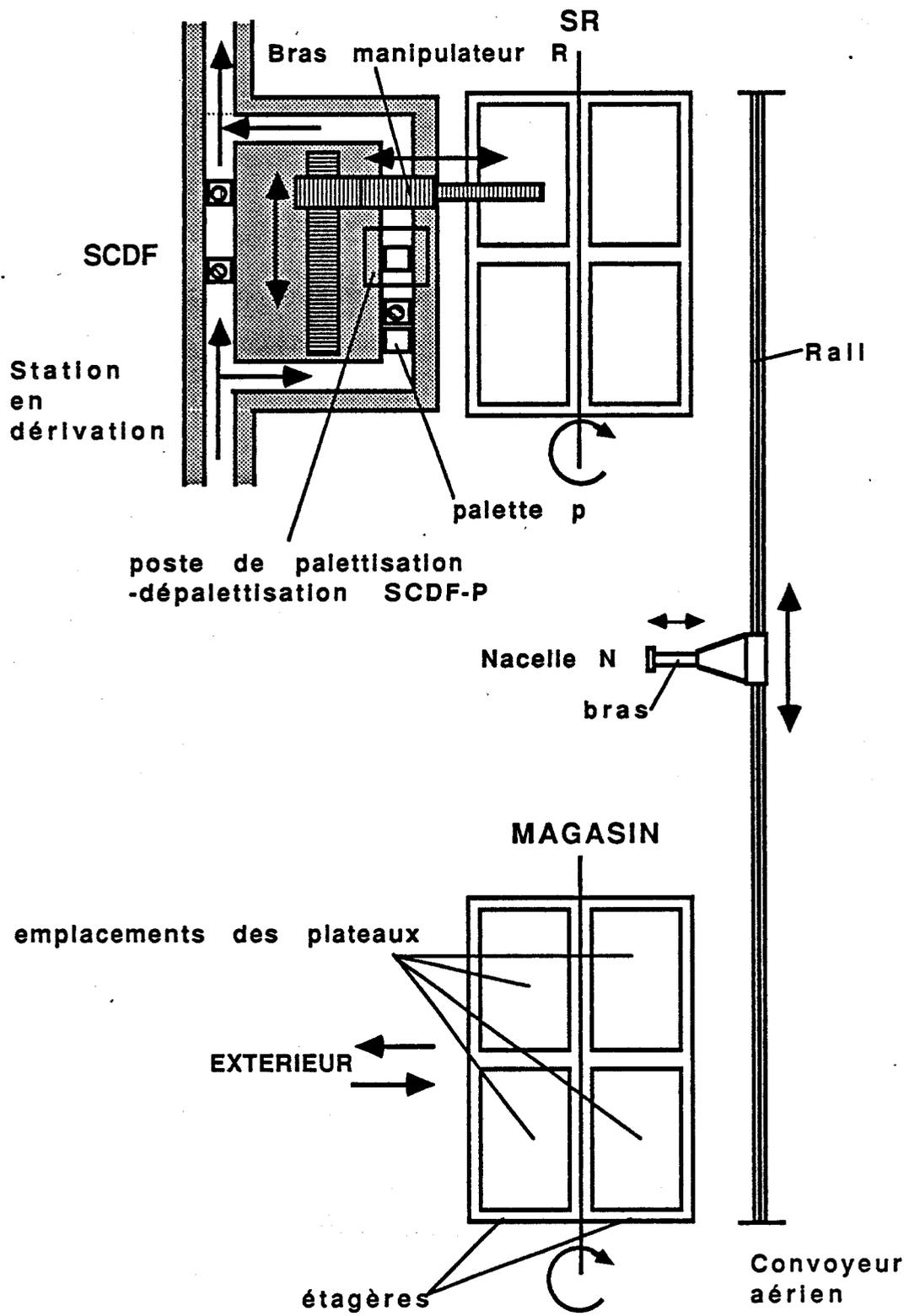


Figure III.51  
 Vue de dessus

A l'autre extrémité du rail de CV se trouve un magasin qui est constitué d'un stockeur similaire à SR et qui, en dehors du convoyeur, est chargé et déchargé par un processus qu'on ne représentera pas.

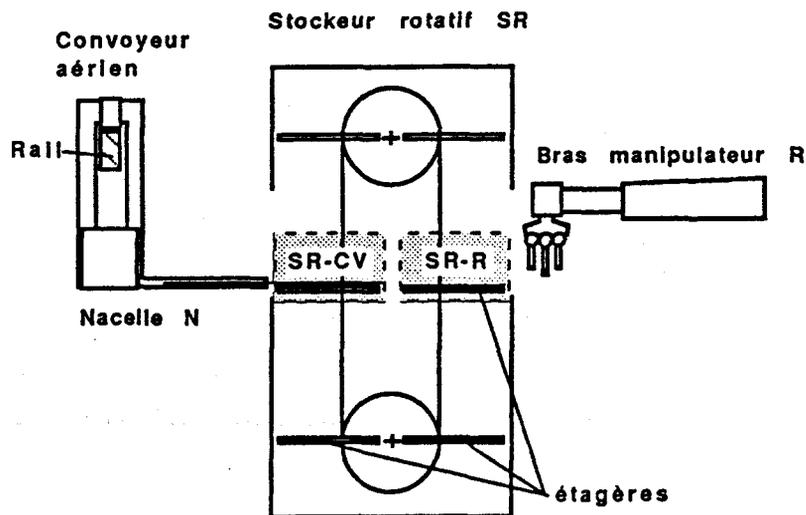
Les plateaux, tous de forme identique, peuvent contenir :

- 1) soit six cartes électroniques vierges ci ou assemblées cf
- 2) soit 12 boîtiers de contacteurs électriques nus bi ou assemblés bf.

Le bras manipulateur R vide les plateaux des pièces ci ou bi pour les palettiser chacune sur une palette au niveau du poste SCDF-P. Il les remplit en retour des pièces cf ou bf dépalettisées également au niveau du poste SCDF-P. La palettisation et la dépalettisation se font pièce à pièce.

Le convoyeur assure le transfert des plateaux entre le MAGASIN et le stockeur rotatif SR. En marche normale, le convoyeur apporte au stockeur rotatif des plateaux pleins contenant un type de pièce bi ou ci, et il reprend des plateaux pleins contenant un type de pièce bf ou cf.

Le stockeur rotatif SR comporte une zone SR-CV accessible au bras robot R et une zone SR-CV accessible à la nacelle du convoyeur.



VUE DE FACE

Figure III.52

### IL.3 PROBLEMES RENCONTRES POUR LA MODELISATION DE LA PARTIE COMMANDE

Il est possible, en face de cet exemple, de construire plusieurs prégraphes en fonction de la complexité que l'on veut et que l'on peut faire porter sur la Partie Commande Réseau de Petri.

Dans le cas présent, il est clair qu'un certain nombre de choix et de levées d'indéterminismes vont devoir être effectués par le système de commande. Ce rôle est dévolu au Niveau Hiérarchique. La Partie Commande RdP a pour but essentiel d'assurer la coordination des actions et le respect des contraintes de séquentialité.

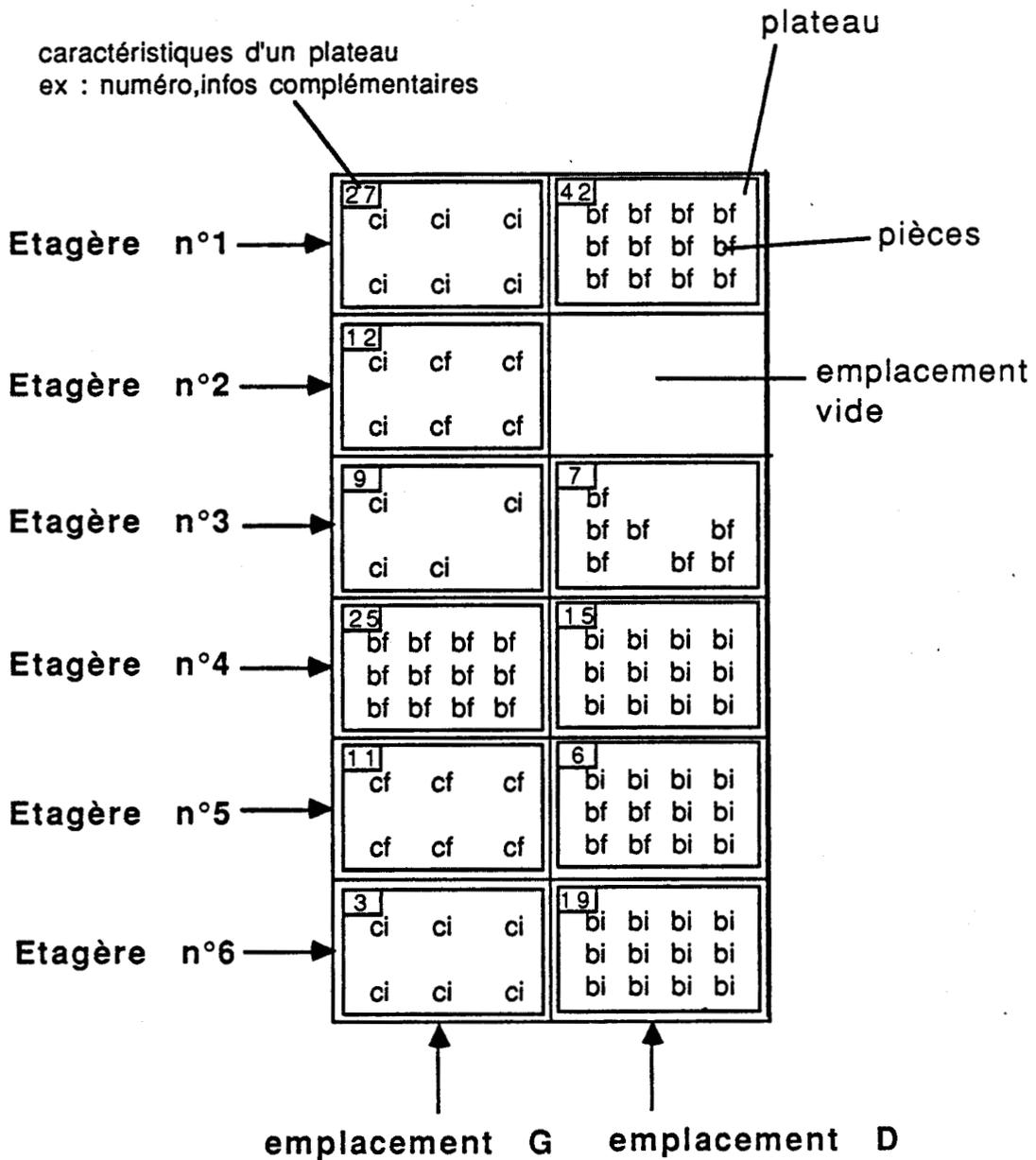
Le niveau Hiérarchique a besoin, pour élaborer une décision, d'un certain nombre d'informations relatives à l'état de la Partie Commande RdP, à l'état du Procédé et aussi à son état interne. Dans de nombreux cas relativement simples, le Niveau Hiérarchique peut se satisfaire de l'image du procédé qui est utilisée par la Partie Commande. Moyennant quelques extensions de modélisation, la Partie Commande peut continuer à supporter une image du procédé consultable par le Niveau Hiérarchique dans des cas un peu plus complexes. Le Niveau Hiérarchique a, dans la plupart de ces cas, un rôle très local et pourra être aisément décentralisé lors de l'implantation sur automates(\*). Nous estimons pour le moment que l'extension de la modélisation de la Partie Commande nécessaire à un Niveau Hiérarchique local ne devra pas être étendu au delà des réseaux à prédicats compte tenu des dispositifs technologiques industriels existant pour l'implantation.

Dans des cas très complexes comme celui de l'exemple présent, la représentation sur le modèle RdP d'une image du procédé suffisante pour le Niveau Hiérarchique n'est pas envisageable compte tenu de la finesse exigée. Il est ainsi nécessaire de développer une image spécifique du procédé, sollicitée par le Niveau Hiérarchique pour ses propres besoins (voir figures III.53 pour l'exemple présent). Le Niveau Hiérarchique nécessite alors parfois des moyens d'implantation puissants, le plus souvent extérieurs aux automates, tels que des micro-ordinateurs. Le modèle de la Partie Commande est en revanche simplifié et limité au seul rôle de coordination.

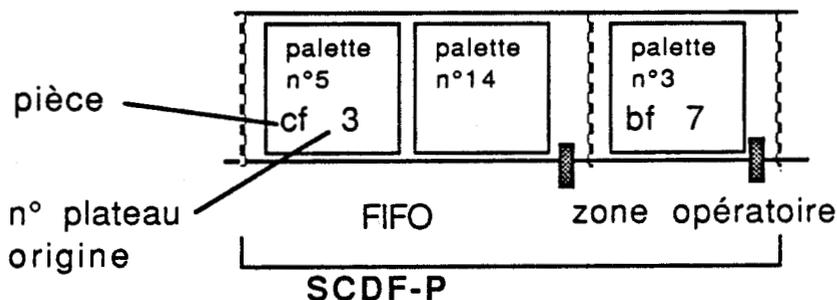
(\*) NB : Ce niveau hiérarchique local peut être représenté souvent de façon plus simple que par règles de production et par places d'interfaces partie Commande/Niveau Hiérarchique. Le caractère adaptatif ou l'interprétation des transitions par l'activité de certaines places peuvent être utilisés pour régler de façon autonome et simple certaines priorités sur le graphe RdP.

Figures III.53 :  
Exemples d'images du procédé nécessaires au Niveau Hiérarchique

\* Image de l'occupation du stockeur rotatif



\* Image de l'occupation du poste de palettisation-dépalettisation SCDF-P



C'est dans cet esprit que le Prégraphe suivant a été construit, pour cet exemple :

#### II.4 PREGRAPHE

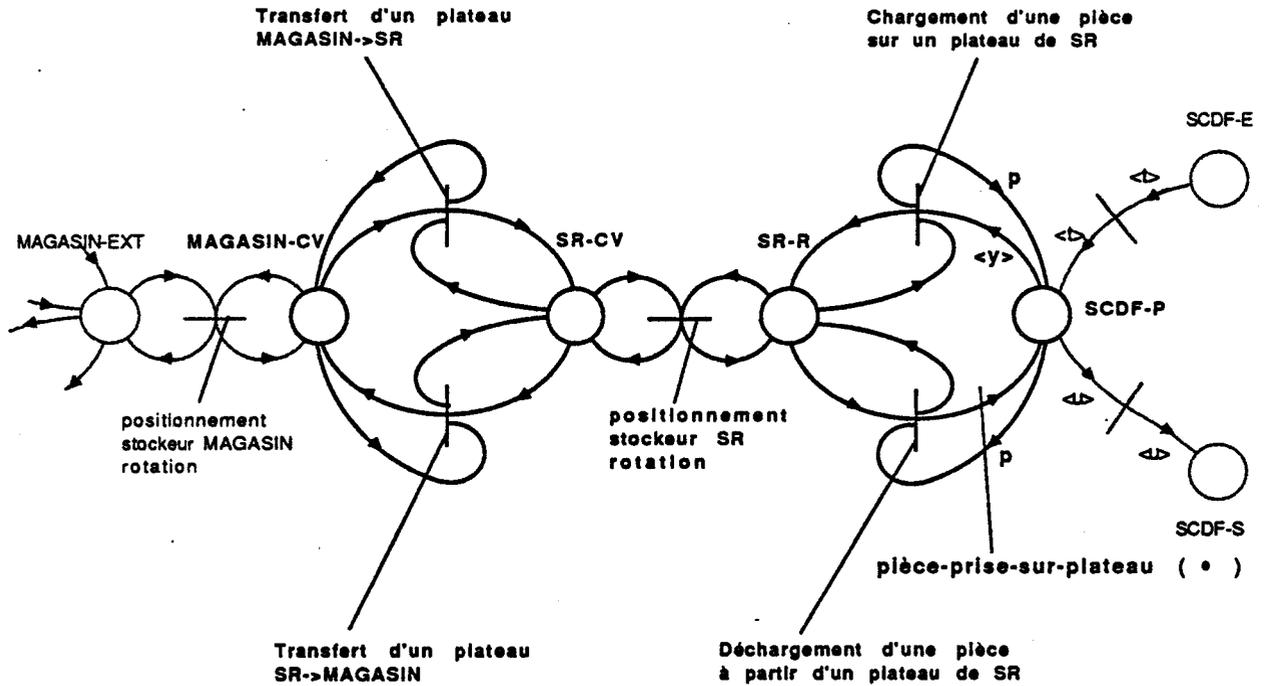


Figure III.54

Le stockeur rotatif SR est représenté par ses deux zones opératoires SR-CV et SR-R accessibles respectivement à la nacelle du convoyeur CV et au bras du robot R. A chacune de ces deux zones est associée une place du Prégraphe.

Le stockeur MAGASIN, partiellement représenté, obéit au même formalisme de représentation.

La place SCDF-P représente le poste de travail de la station de chargement et déchargement flexible. Cette station est constituée d'une dérivation. On a déjà vu précédemment (Chapître 2, Partie I, ex. 6) comment on représentait sur un prégraphe ce genre de dérivation. On ne fait référence ici qu'au poste de palettisation-dépalettisation SCDF-P.

Afin de faciliter la compréhension, donnons quelques indications sur le sens des transitions essentielles :

Transition MAGASIN-CV  $\rightarrow$  SR-CV

Il s'agit d'une transition positionnelle.

Elle représente le transfert d'un plateau entre le stockeur MAGASIN et le stockeur SR. Cette action nécessite la présence préalable de la bonne étagère d'origine du plateau sur la zone opératoire MAGASIN-CV du stockeur MAGASIN (arc 3). Elle nécessite également la présence préalable de la bonne étagère de destination du plateau sur la zone opératoire SR-CV du stockeur rotatif SR (arc 4).

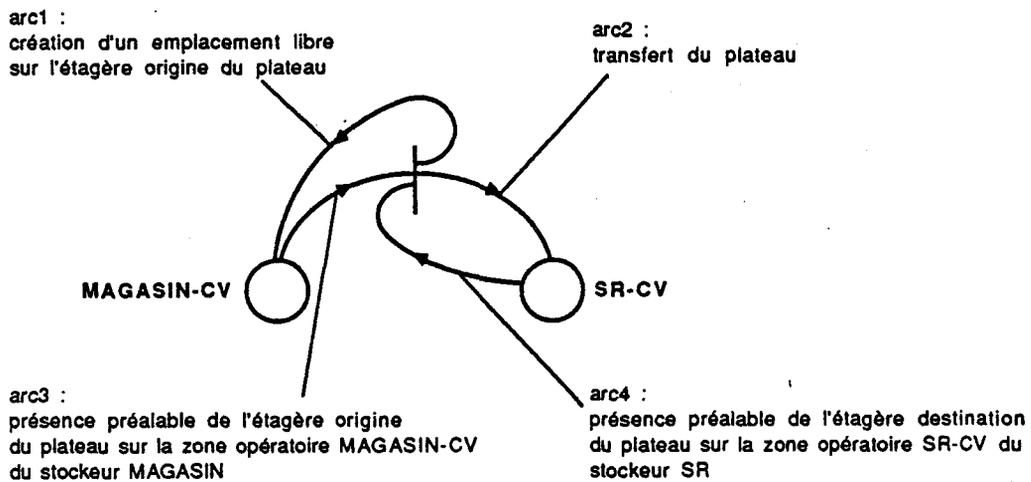


Figure III.55

Les autres arcs représentent le transport d'un plateau depuis le stockeur MAGASIN jusqu'au stockeur SR (arc 2) et la libération d'un emplacement sur l'étagère origine (arc 1).

La présence de marques colorées n'est pas utile dans la Partie Commande ici. Avant de laisser la Partie Commande assurer la commande du transfert, le Niveau Hiérarchique a dû effectuer un certain nombre de choix : par exemple choix d'un emplacement origine où prendre un plateau sur une étagère du stockeur MAGASIN, choix d'un emplacement vide destination sur

une étagère du stockeur SR. La détermination de ces choix utilise une image du procédé précise et spécifique au Niveau Hiérarchique qui n'a pas lieu d'être portée ici par la Partie Commande.

Transition SR-CV → MAGASIN-CV

Commentaires analogues à ceux qui ont été faits pour la transition MAGASIN-CV → SR-CV

Transition SCDF-P → SR-R

Il s'agit d'une transition positionnelle.

Elle représente le transfert d'une pièce bf ou cf entre le poste SCDF-P où elle est dépalettisée et le stockeur rotatif SR où elle est déposée dans un plateau. Cette action nécessite la présence préalable d'une pièce palettisée pbf ou pcf sur la zone opératoire du poste SCDF-P (arc 4). Elle nécessite également la présence préalable de l'étagère qui porte le bon plateau destination sur la zone opératoire SR-R du stockeur rotatif SR (arc 3). Les arcs 1 et 2 représentent le transfert de la pièce bf ou cf par l'intermédiaire du bras robot R et la libération de la palette P sur le poste SCDF-P.

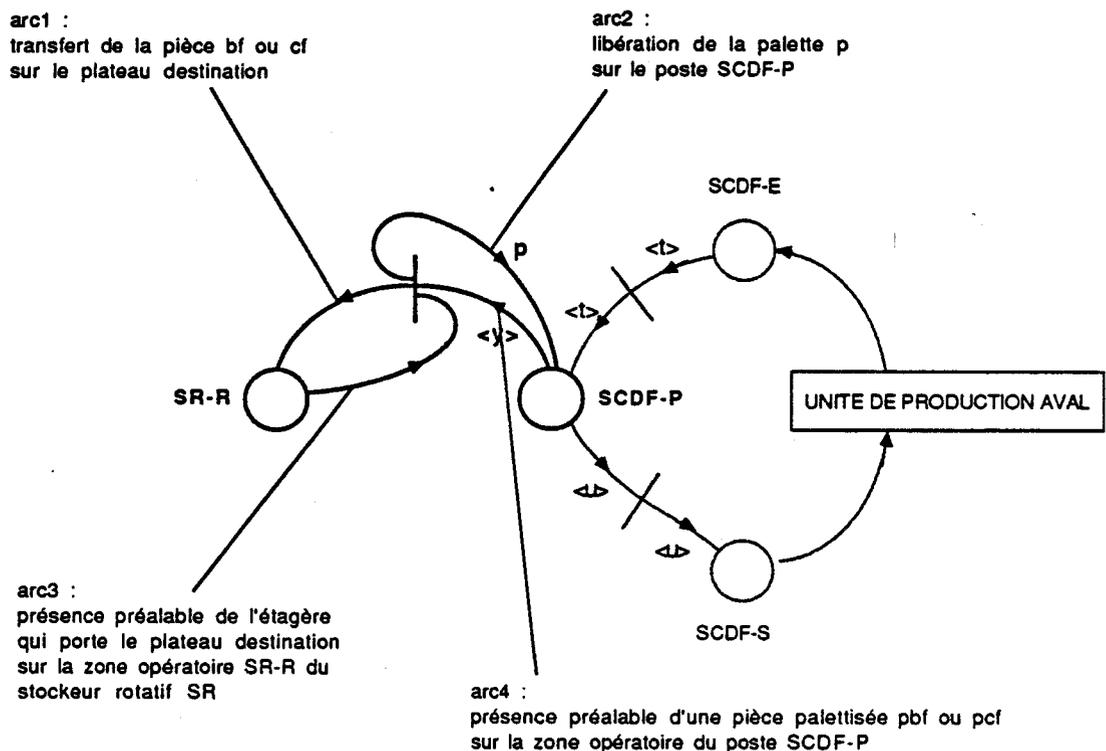


Figure III.56

La marque colorée  $p$  et la variable libre  $\langle y \rangle$  (où  $\text{dom}(\langle y \rangle) = \{pbf, pcf\}$ ) interviennent ici.

Elles sont utiles au fonctionnement du RdP représentant la Partie Commande de l'unité de production aval desservie par le poste de palettisation-dépalettisation.

### Transition SR-R $\rightarrow$ SCDF-P

Cette transition est elle aussi une transition positionnelle.

Elle représente le transfert d'une pièce  $bi$  ou  $ci$  entre le stockeur rotatif SR où elle était dans un plateau et le poste de palettisation-dépalettisation SCDF-P où elle est palettisée.

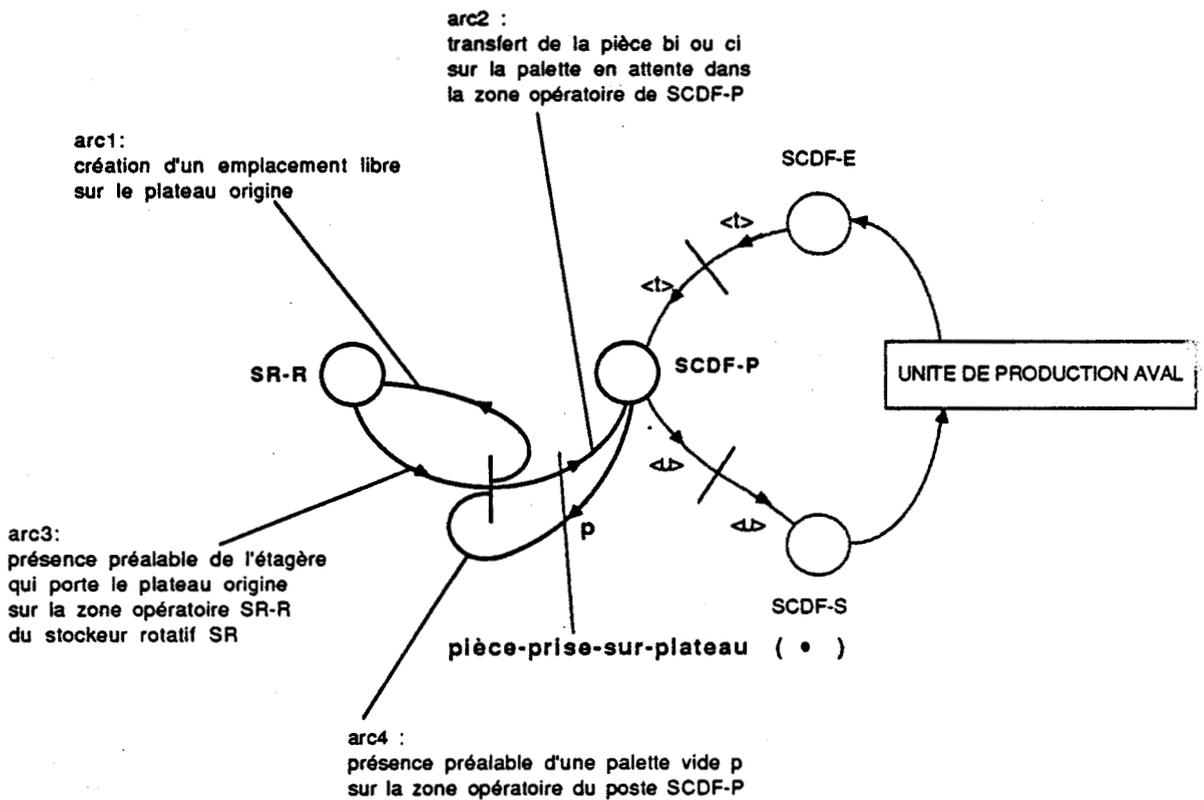


Figure III.57

La fonction pièce-prise-sur-plateau permet de générer une marque parmi les marques  $pbi$  et  $pci$  suivant le choix effectué par le Niveau Hiérarchique.

La marque  $p$  représente la présence d'une palette  $p$  sur la zone opératoire du poste SCDF-P.

Comme précédemment, ces marques colorées sont utiles au fonctionnement du RdP représentant la Partie Commande de l'unité de production desservie par le poste de palettisation-dépalettisation.

### Transition positionnement stockeur SR

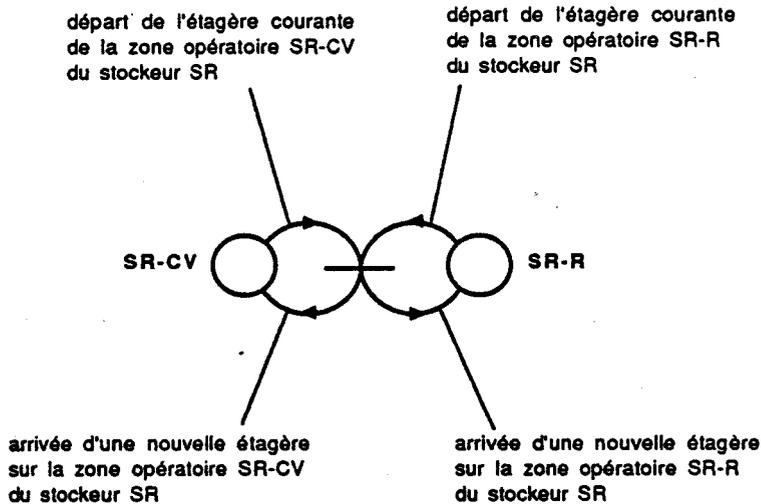


Figure III.58

Pour chacune des zones opératoires SR-CV et SR-R, la transition représente le transfert ou plutôt le repositionnement externe (au sens où cela a été vu, Chapitre III, Partie I) d'une étagère du stockeur rotatif SR. Les deux repositionnements sont bien sûr synchronisés sur le mouvement de rotation d'ensemble des étagères du stockeur rotatif. Le paramétrage de la commande de rotation est effectué par le Niveau Hiérarchique. La structure du Prégraphe est suffisante pour construire les contraintes de coordination et de séquentialité sur le RdP de la Partie Commande.

(Remarque : On trouvera en **annexe 1** une représentation colorée possible du Prégraphe et une discussion introduisant, à partir de ce modèle coloré, le modèle qui vient d'être présenté).

### IL5 GRAPHE DEVELOPPE

Le graphe RdP développé obtenu à partir du Prégraphe après l'étape de structuration est donné sur la figure d'ensemble III.59. C'est le modèle de la Partie Commande. On reconnaît les processus relatifs à chacune des places du Prégraphe ainsi que les liaisons entre ces processus.

- Les commandes ① ROTATION, ② POSE, ③ PRISE, ④ DEPOT, ⑤ SAISIE, ⑥ SAISIE-M et ⑦ DEPOT-M sont paramétrées par le Niveau Hiérarchique qui fournit leurs arguments (ex. numéro d'étagère + emplacement sur l'étagère, position dans un plateau, angle de rotation).



- On constate que les indéterminismes directionnels qui apparaissent sur le prégraphe se retrouvent ici au niveau des places TEST-OUT du graphe développé RdP. Il appartient au Niveau Hiérarchique de régler ces indéterminismes en bloquant ou en débloquant les transitions  $tP$ ,  $tD$ ,  $tE$ ,  $t1R$ ,  $t1CV$ ,  $t2$ ,  $t3$ ,  $t4$ ,  $t5$ ,  $t6$ ,  $t7$ ,  $t8CV$ , ... au moyen d'arcs adaptatifs ou de l'interprétation de ces transitions.
- On remarque, pour terminer, que le résultat retourné par certaines fonctions telles que "pièce-prise-sur-plateau" est également déterminé par le Niveau Décisionnel.

Un échantillon du Niveau Hiérarchique pour cet exemple est donné à titre indicatif dans l'annexe 2.

### IL.6 DISCUSSION D'UN PETIT EXEMPLE POUVANT CONCERNER EVENTUELLEMENT LES RESEAUX A PREDICATS

Rappelons que dans le cas des ateliers flexibles apparaissent un certain nombre d'indéterminismes de routage des pièces qui, pour être résolus par le Niveau Hiérarchique, nécessitent de plus en plus une reconnaissance individuelle des pièces par le Système de Commande. On a montré dans l'exemple précédent du stockeur rotatif que l'image du procédé exigée par le niveau Hiérarchique est parfois si fine qu'il n'est pas envisageable de la faire porter par le graphe de la Partie Commande. Il est alors nécessaire de construire une représentation spécifique. Il existe toutefois des cas intermédiaires dans lesquels la construction d'une image opérative hors du graphe de la Partie Commande est peut être superflue et pourrait être portée par le graphe RdP moyennant en particulier des extensions de la coloration. Cela mérite d'être discuté. C'est dans ce cadre que se situe l'utilisation éventuelle des réseaux à prédicats.

Prenons le fragment de prégraphe ci-dessous :

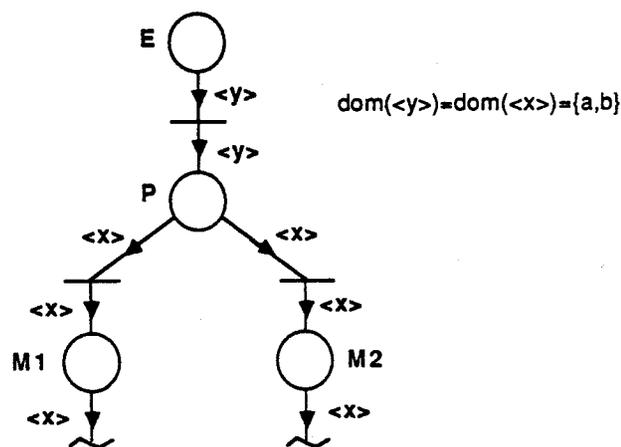


Figure III.60

Les pièces a et b peuvent suivre au sein de leurs gammes opératoires deux routages possibles à partir du site P : un routage vers M1 ou un routage vers M2.

Supposons qu'une "préférence" soit donnée à un routage plutôt qu'à l'autre à partir de la place E mais que le choix de routage définitif sera fait par le Niveau Hiérarchique seulement au niveau de la place P en tenant compte également d'autres données telles que l'encombrement des machines M1, M2.

Il est possible de faire porter l'information supplémentaire concernant "l'option préférentielle" de routage sur les marques qui circulent dans le graphe de la Partie Commande si on ne désire pas développer une image fine du procédé à l'extérieur du graphe RdP.

On peut utiliser alors des marques qui ont la forme d'un doublet ( $\langle x \rangle$ ,  $\langle \text{préférence} \rangle$ ) où  $\text{dom}(\langle \text{préférence} \rangle) = \{ M1, M2 \}$ . L'information  $\langle \text{préférence} \rangle$  ne sert qu'au Niveau Hiérarchique qui, suivant la complexité de la décision, peut être éventuellement décentralisé localement sur la Partie Commande.

Dans le cas présent, l'information "préférence" appartient à un domaine fini. Il est envisageable de travailler sur des informations appartenant à un domaine infini, comme la date, le numéro d'entrée ou d'identification.

Ces informations rapportées concernent des problèmes locaux de décision dont la résolution n'est pas fixée de façon définitive et rigide. Elles ne remettent pas en cause la construction classique du Prégraphe coloré. Il ne s'agit que d'adjoindre des informations à un Prégraphe existant.

Il est donc possible d'utiliser toute une gamme de possibilités quant à la représentation opérative au sein du système de commande. Notre choix jusqu'à maintenant a été de ne pas compliquer le modèle RdP de la Partie Commande afin de pouvoir continuer à en faire la construction systématique et afin d'en faciliter l'implantation sur les outils industriels existants.

#### IL.7 REMARQUE SUR LE MAINTIEN DE LA COHERENCE ENTRE LES DIFFERENTES IMAGES DU PROCEDE

Nous avons vu comment le Système de Commande peut comporter deux représentations du procédé ou images opératives. Le problème qui se pose,

bien entendu, est de savoir comment faire la correspondance entre les marques qui circulent dans le modèle RdP de la Partie Commande et l'image du procédé plus fine, extérieure au graphe RdP, qui est utilisée pour des problèmes de décision complexes du Niveau Hiérarchique. Il serait bien sûr possible d'associer une étiquette aux marques colorées de la même façon que celle que nous avons examinée dans le paragraphe II.6. Cette étiquette permettrait d'établir les correspondances exactes par l'intermédiaire d'une table de correspondance (éventuellement, elle permettrait également d'aller consulter par adressage une base de données image du procédé commune au Niveau Hiérarchique et à la Partie Commande). Toutefois, pour le moment, nous préférons utiliser une propriété des réseaux structurés sur lesquels nous travaillons : les processus sont saufs, ils ne contiennent qu'une seule marque à tout instant. Cette marque correspond à un objet réel unique situé sur la zone opératoire physique à laquelle est associé le processus. Cette correspondance simple qui repose sur le fait qu'il n'y a qu'une seule pièce sur chaque zone opératoire à un instant donné est toutefois très suffisante pour le moment.

## **II.8 CONCLUSIONS**

La modélisation du système de commande est actuellement décomposée en deux niveaux principaux :

- Le niveau bas ou Partie Commande chargé du séquençement simple et de la coordination des commandes à envoyer au Procédé.
- Le niveau haut ou Niveau Hiérarchique lui-même décomposé en étages de planification et de gestion, chargé de prendre des décisions et de paramétrer la Partie Commande.

Suivant le cas, l'un de ces deux niveaux peut être plus développé que l'autre. A l'extrême, la situation peut être celle d'une simple machine séquentielle classique sans aspect décisionnel ou, au contraire, celle du pilotage direct par un système à base de connaissances ( système expert). La plupart des problèmes de commande de systèmes de production flexibles se situent entre ces deux pôles. (Les exemples de pilotage direct par système expert reprennent souvent en fait une "Partie RdP" sous forme de règles de production alors que leur intérêt est de travailler sur des problèmes décisionnels dont le raisonnement n'a pas un séquençement figé). La frontière sera déterminée principalement par la finesse de l'image du procédé nécessaire à la commande. L'idée qui peut être

tirée de l'exemple qui précède est qu'il faudra répartir la complexité de l'image du procédé sur les deux niveaux de la façon la plus adaptée en fonction de la situation rencontrée. Le graphe de la Partie Commande pourra ainsi être décomposé en plusieurs sous-ensembles dans lesquels le rapport de complexité Partie Commande/Niveau Hiérarchique sera différent et adapté au problème de chaque sous-ensemble. Ainsi, la Partie Commande sera parfois simplifiée (ex. RdP sans coloration) pour n'en faire qu'un simple exécutant paramétré du Niveau Décisionnel. Parfois, le Niveau Hiérarchique local, installé sous forme de règles simples ou même sur le Modèle de la Partie Commande sous forme RdP adaptatif ou interprété autonome, utilisera une complexité du modèle de la Partie Commande qui devra intégrer une image plus fine du procédé. Ces aspects de modélisation doivent être pris en compte dès la conception du Prégraphe. C'est pour cette raison qu'il faut considérer la possibilité d'étendre par enrichissement le Prégraphe aux Réseaux à Prédicats. La coloration s'est avérée toutefois suffisante jusqu'à maintenant dans de très nombreux exemples (voir en annexe 3 une discussion illustrée à partir d'un exemple industriel).

L'adaptation de la modélisation pour chaque sous-ensemble permettra également une augmentation des performances de simulation. Il sera en effet possible d'utiliser une simulation multi-processeurs comprenant sur chaque processeur un simulateur spécialisé adapté au traitement d'un type de modèle RdP accompagné éventuellement de son niveau hiérarchique local (voir [CAS 88]).

Retenons simplement, pour terminer cette partie, que le Prégraphe constitue le squelette du modèle Réseau de Petri et que les choix de modélisation faits sur le Prégraphe auront une conséquence sur la modélisation de la Partie Commande, et donc sur le Niveau Hiérarchique. Ainsi, dans l'exemple précédent, la modélisation de la Partie Commande du stockeur rotatif introduit dès le Prégraphe un certain nombre de spécifications relatives au Niveau Hiérarchique et au paramétrage qu'il doit fournir au graphe RdP. La conception du Prégraphe a donc une grande importance.

### AVANT PROPOS A LA PARTIE III

Le prégraphe a été créé initialement et essentiellement pour construire le modèle de la Partie Commande des unités de production considérées. C'est principalement autour de cette utilisation que se centre le propos du mémoire. Toutefois, il s'avère qu'une version étendue du Prégraphe peut servir à aborder également des problèmes de prédimensionnement et de préétude de comportement dynamique. Bien que ces développements soient actuellement en cours d'étude, nous allons toutefois présenter dans la partie qui va suivre les idées sur lesquelles ils reposent.

### III.

#### LES PROBLEMES DE PREDIMENSIONNEMENT DE L'ARCHITECTURE DES UNITES DE PRODUCTION ET DE PREETUDE SIMPLIFIEE DE LEUR COMPORTEMENT : LE ROLE POSSIBLE DU PREGRAPHE.

##### III.1 DEMARCHES ANALYTIQUES ET SIMULATION ; PRESENTATION GENERALE DU PROBLEME.

La conception et la validation du comportement d'un système de production discret sont extrêmement complexes à maîtriser.

La première étape est la construction d'un ou plusieurs modèles de représentation.

La deuxième étape est l'étude des modèles par l'intermédiaire d'outils adaptés qui sont de deux types :

- Les démarches analytiques
- La simulation

(N.B. : Un inventaire précis et actuel de ces méthodes et de leur contexte d'utilisation, présentant l'état de l'art, des applications industrielles et des renseignements pratiques a été fait dans deux ouvrages écrits par Jean-Marie PROTH (INRIA) et Jean-Bernard CAVAILLE (CERT-DERA) à l'occasion des conférences SIPRODIS 1987 sur la simulation en production discrète (ICAV 87a|ICAV 87b)).

- Les démarches analytiques travaillent à partir de modèles mathématiques sur lesquels un calcul formel direct ou au moins une résolution numérique sont envisageables. Il arrive que l'on ne souhaite observer qu'un seul critère (problème mono-critère). Cette formulation autorise la recherche de la meilleure commande (ou commande optimale) au sens du critère retenu. Il arrive que cette recherche fasse appel à des résultats théoriques connus et puisse être menée de manière totalement analytique. Sinon on utilise des heuristiques plus ou moins rigoureuses. Dans le cas où le problème posé par le modèle est trop complexe pour être résolu, même numériquement et par des voies heuristiques, ou lorsque le problème est par essence multi-critères et que la pondération est difficile, on procède par essai-erreur.

Les avantages des démarches analytiques sont :

- qu'elles représentent une approche active en ce sens qu'elles permettent d'obtenir une commande optimale ou au moins proche de l'optimum,
- que l'implantation informatique requiert des temps de calcul raisonnables,
- qu'elles conduisent à un résultat compact, c'est-à-dire à un résultat que l'utilisateur peut évaluer avec un effort de synthèse et d'interprétation minimal.

Le défaut crucial de ces méthodes est qu'elles exigent très souvent des simplifications et approximations importantes pour obtenir un modèle que l'on peut traiter à l'aide des outils mathématiques disponibles. Ceci conduit parfois à créer des modèles simplifiés indépendants pour pouvoir obtenir une idée de certains aspects du comportement, ce qui peut mener, en l'absence de prudence, à des résultats très éloignés de la réalité. Théoriquement, un bon modèle doit être suffisamment fin pour décrire au plus près un fonctionnement et suffisamment global pour ne pas perdre la cohérence de l'ensemble des caractéristiques du système réel.

- La simulation travaille sur des modèles qui se composent :
    - d'un ensemble de paramètres qui constitue une image souvent très fine du système étudié et qu'on appelle état du système.
    - d'un ensemble d'instructions d'évolution.
    - éventuellement d'un ensemble de critères.
    - et enfin du processus qui fait évoluer l'état du système en fonction des instructions et des critères lorsque l'intervalle de temps sur lequel s'effectue l'étude est donné.
- Ce processus est en général un programme informatique.

Les avantages de la simulation sont :

- que les modèles sont beaucoup plus proches de la réalité et autorisent des représentations beaucoup plus complexes que les modèles destinés aux démarches analytiques.
- qu'elle permet une expérimentation d'un comportement pour des phénomènes qu'il est impossible d'évaluer avec les techniques

mathématiques disponibles (c'est le cas, en particulier, lorsque les instructions d'évolution sont des règles de gestion complexes. Des outils formels d'étude de l'occurrence de certains états comme la logique temporelle restent encore très limités dans le cas général).

La simulation permet en particulier la détection d'anomalies de fonctionnement et l'expérimentation de régimes transitoires ou non permanents.

Les inconvénients majeurs sont :

- que l'évaluation et l'interprétation des résultats de simulation sont très délicates.

Soit le résultat de la simulation correspond à l'attente du concepteur, soit des phénomènes imprévus apparaissent, et il est alors souvent difficile de faire l'interprétation des résultats. Les informations sont dispersées et locales. La signification des résultats est en général difficile à rattacher au fonctionnement global du système étudié. L'interprétation de ces données requiert une grande expertise.

- en outre, la simulation ne peut assurer l'optimisation du fonctionnement. Elle assure tout au plus dans le meilleur des cas un bon comportement.
- la simulation ne peut également garantir un bon fonctionnement que dans un intervalle de temps borné. Elle ne donne pas de résultats généraux sauf dans le cas de phénomènes périodiques ou stationnaires. En particulier, pour les vrais systèmes de production flexibles, elle ne peut garantir un comportement qu'en production normale sur une séquence d'entrée de pièces définie et limitée ou que sur les cas simulés de passage en mode dégradé.
- enfin la simulation est très gourmande en temps machine (CPU). Ceci est dû à la finesse de la modélisation qui accroît la taille des modèles, et à la durée de l'intervalle de simulation nécessaire pour obtenir la réalisation des critères ou au moins des résultats consistants.

Dans le cas où le système étudié fonctionne en régime permanent, la simulation donne des résultats généraux à défaut d'être optimaux. Elle est donc d'un grand intérêt. La simulation est alors effectuée ici une fois pour toutes. Le temps de simulation est alors moins gênant. A titre d'information, prenons l'exemple de l'évaluation de la productivité d'une ligne de production réelle. Il s'agit d'une ligne d'usinage de petites boîtes de vitesse étudiée à la division recherche de la RNUR.

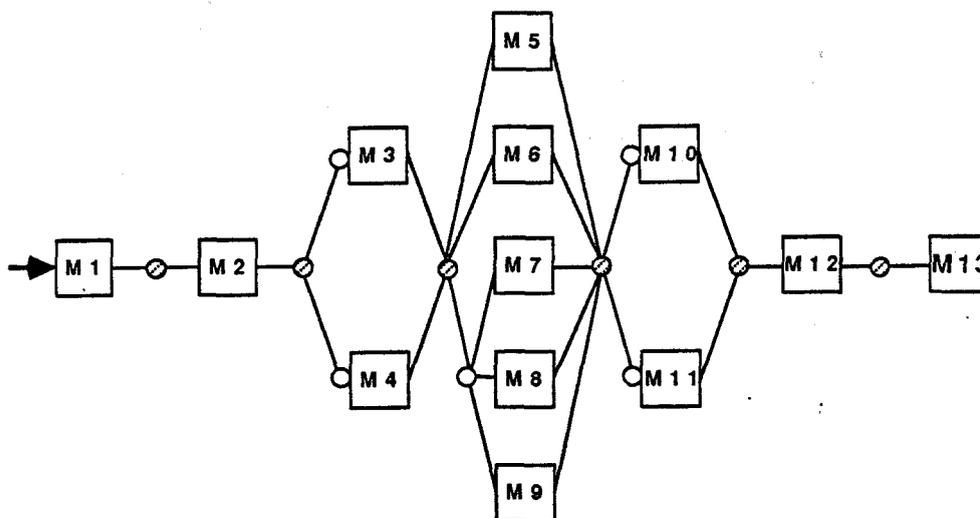


Figure III.61

Les rectangles représentent les machines et les cercles des stocks tampons. La productivité de la ligne a été calculée analytiquement (logiciel CALIF) en régime stationnaire et en faisant une approximation importante sur la configuration des stocks intermédiaires. L'évaluation a retourné une productivité située dans un intervalle de 3,21 à 3,22 pièces/unité temps. La ligne a également été simulée en utilisant le langage SLAM. L'horizon de simulation a dû être choisi assez long pour que les lois statistiques sur les durées soient vérifiées. Pour se stabiliser à une production moyenne de 3,182 pièces/unité de temps, le simulateur a demandé 126 h de temps CPU sur DEC/VAX 750.

Les démarches analytiques et les techniques de simulation s'avèrent être deux outils à la fois complémentaires et limités. Les techniques analytiques permettent d'obtenir rapidement de premières évaluations moyennant certaines approximations importantes. La simulation qui nécessite une charge élevée en temps machine permet de contrôler les hypothèses simplificatrices de modélisation et d'affiner les résultats. L'utilisation préalable de méthodes analytiques évite des tâtonnements trop importants au cours de la simulation.

Dans le domaine de la productique, les meilleurs résultats obtenus conjointement par ces deux outils sont liés aux systèmes de production fonctionnant en régime stationnaire. Il s'agit essentiellement de la production

sur chaînes transferts et en grande série. Des résultats intéressants ont été obtenus également pour des ateliers flexibles moyenne série pour lesquels on peut définir une notion de régime permanent. Les limites imposées par les modèles analytiques sont comblées dans une certaine mesure par la simulation. Le cas des ateliers flexibles pour petites séries et pour le panachage petite-moyenne série reste, quant à lui, très problématique. Ces ateliers travaillent en effet très souvent en régime non-permanent.

## III.2 LES PROBLEMES LIES AU PROJET CASPAIM : EBAUCHE D'UNE DEMARCHE DE DIMENSIONNEMENT

### III.2.1. Introduction

Le projet CASPAIM permet d'aborder la modélisation d'un grand nombre d'unités de production pour lesquelles des outils classiques de prédimensionnement sont efficaces. Dans le cas des cellules flexibles, les problèmes liés à la présence d'un grand nombre de fonctionnements en mode transitoire ou non-permanent nous avaient d'abord conduit à penser que les outils de dimensionnement classiques ne pouvaient pas être utilisés et que seule la simulation directe des modèles fins (Niveau Hiérarchique, Partie Commande et Procédé) pouvait permettre d'aboutir à des résultats à partir de séquences d'entrée de pièces définies. L'utilisation nous a montré que si, moyennant une augmentation des performances du simulateur, une simulation finale des modèles était la seule garantie relative d'un fonctionnement correct, il était par contre hors de question de mener toute la démarche de dimensionnement de l'architecture matérielle et de la commande de l'atelier uniquement à partir de la simulation des modèles finaux.

L'idée d'une démarche de dimensionnement progressive utilisant la coopération des techniques analytiques et de simulation sur des modèles intermédiaires est alors apparue.

Les études de prédimensionnement sont actuellement à leurs débuts dans l'équipe du L.A.I.I. Seule une présentation simpliste va en être donnée ici, elle va permettre toutefois de montrer l'importance que le PREGRAPHE peut être amené à jouer dans ce cadre.

### III.2.2. Les essais actuels

L'idée de départ est de commencer le dimensionnement en se plaçant dans l'hypothèse du régime permanent.

La notion intuitive qui est sous-jacente est la suivante : même si une unité de production ne fonctionne que suivant un régime assimilable à un régime transitoire, on ne peut en attendre des performances que relatives à un régime permanent correspondant. Cette notion est fondamentale et classique dans d'autres domaines scientifiques que la Productique, par exemple en Automatique des processus continus, en Electronique ou en Mécanique Analytique. Elle est le point d'entrée de l'étude de la régulation des processus dynamiques (étude des points d'équilibre absolus et relatifs et comportement en régime permanent, stabilité relative, vitesse de réponse transitoire).

Pour les ateliers flexibles, l'idée est de dire simplement que les performances optimales seraient obtenues pour un fonctionnement en régime permanent asymptotique (par exemple, production en grande série des mêmes ratios de types de pièces), et en mode de production normale, c'est-à-dire non dégradée. La recherche des performances en régime non permanent s'apparente alors aux notions de recherche de stabilité du régime permanent (c'est-à-dire la capacité du système à ne pas s'éloigner de son régime permanent à cause de phénomènes de blocage principalement) et de vitesse de réponse transitoire (cette vitesse est la rapidité avec laquelle le système tend à rejoindre ou installer son régime permanent).

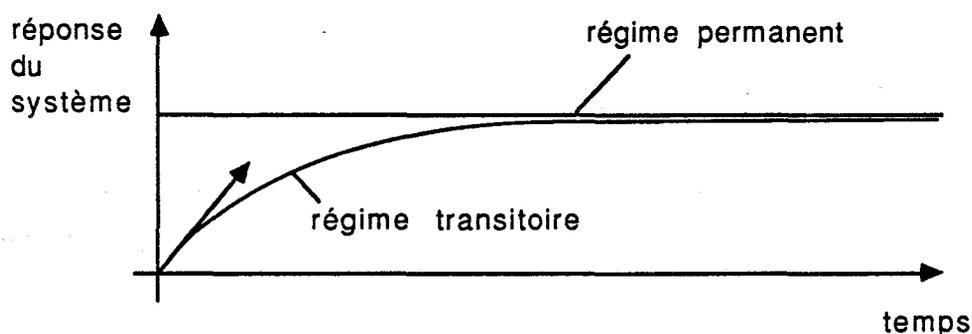


Figure III.62

La régulation doit permettre d'assurer la précision, la stabilité et la vitesse de la réponse à une commande en vue d'obtenir une consigne de régime permanent (ex. des PID).

Pour notre étude concernant les ateliers flexibles, le schéma est le suivant :

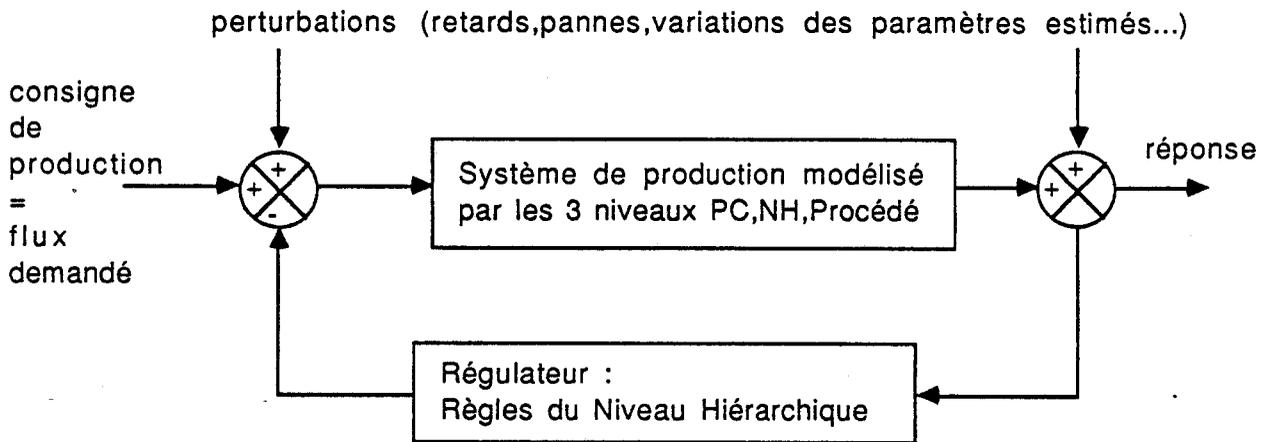


Figure III.63

La régulation ne doit intervenir qu'après le dimensionnement du régime permanent qui consiste à définir l'architecture matérielle et la commande autorisant le débit maximal stationnaire.

Elle devrait permettre alors de faire tendre les régimes "transitoires" le plus vite possible en direction des performances des régimes permanents tout en évitant une instabilité de la réponse due aux écarts du régime de fonctionnement avec le régime stationnaire.

a) Etude du régime permanent

Un éventail d'outils important peut être utilisé. Le problème actuel est de déterminer quel est l'outil qui est le plus adapté au problème à traiter.

a.1) Calcul des débits moyens en régime stationnaire sans files d'attente

Reprenons l'exemple de la cellule décrite au chapitre I constituée des deux machines M1 et M2, d'un banc d'entrée E et d'un banc de sortie S, et

d'un bras robot central R assurant les transferts. Introduisons une modification par rapport à l'exemple décrit au chapitre I afin de rendre l'exemple illustratif pour ce qui nous intéresse ici. Les pièces de type m2 sont remplacées ici par des pièces de type m12 qui peuvent être usinées indifféremment sur M1 ou M2.

Le prégraphe destiné à la structuration est ici obtenu de façon élémentaire :

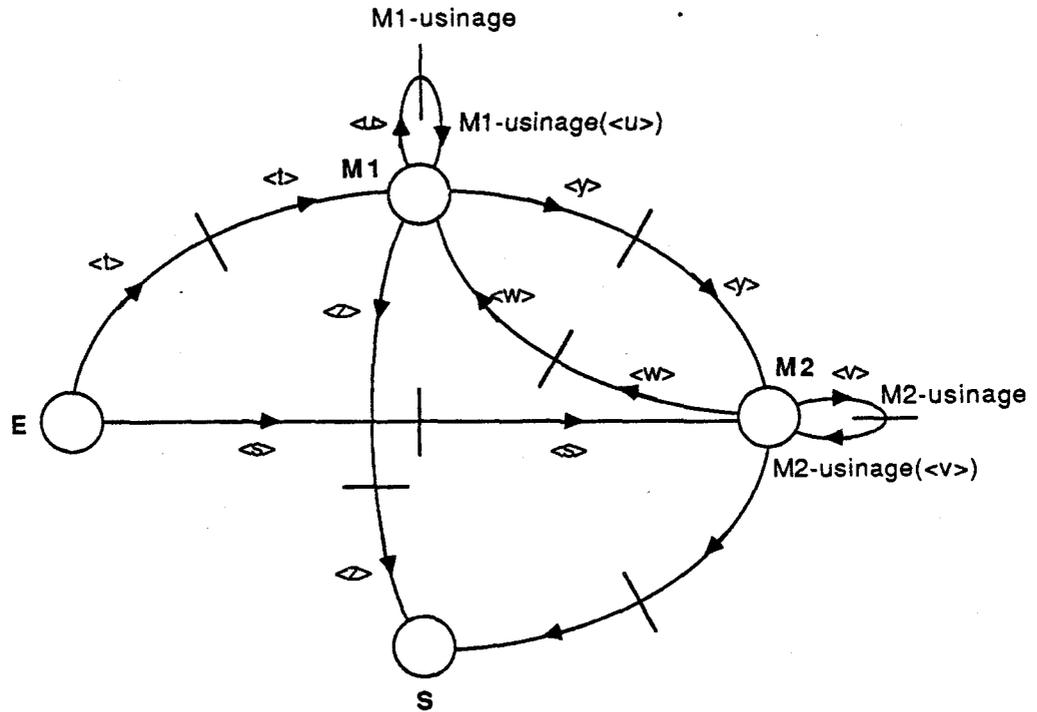


Figure III.64

avec $\text{dom}(s) = \{m12-, m1-m2-\}$	et M1-usinage : $\langle u \rangle \rightarrow M1\text{-usinage}(\langle u \rangle)$
$\text{dom}(t) = \{m1-, m12-, m1-m2-\}$	$m1- \rightarrow m1+$
$\text{dom}(u) = \{m1-, m12-, m1-m2-, m1-m2+\}$	$m12- \rightarrow m12+$
$\text{dom}(v) = \{m12-, m1-m2-, m1+m2-\}$	$m1-m2- \rightarrow m1+m2-$
$\text{dom}(w) = \{m1-m2+\}$	$m1-m2+ \rightarrow m1+m2+$
$\text{dom}(x) = \{m12+, m1+m2+\}$	M2-usinage : $\langle v \rangle \rightarrow M2\text{-usinage}(\langle v \rangle)$
$\text{dom}(y) = \{m1+m2-\}$	$m12- \rightarrow m12+$
$\text{dom}(z) = \{m1+, m12+, m1+m2+\}$	$m1-m2- \rightarrow m1-m2+$
	$m1+m2- \rightarrow m1+m2+$

Considérons une séquence d'entrée de 50 pièces dont

- 22 pièces de type m1-m2-
- 14 pièces de type m1-
- 14 pièces de type m12-

ce qui correspond à des ratios de production de

$$\frac{11}{25} \text{ pour } m1-m2-$$

$$\frac{7}{25} \text{ pour } m1-$$

$$\frac{7}{25} \text{ pour } m12-$$

Intéressons-nous aux flux moyens suivants :

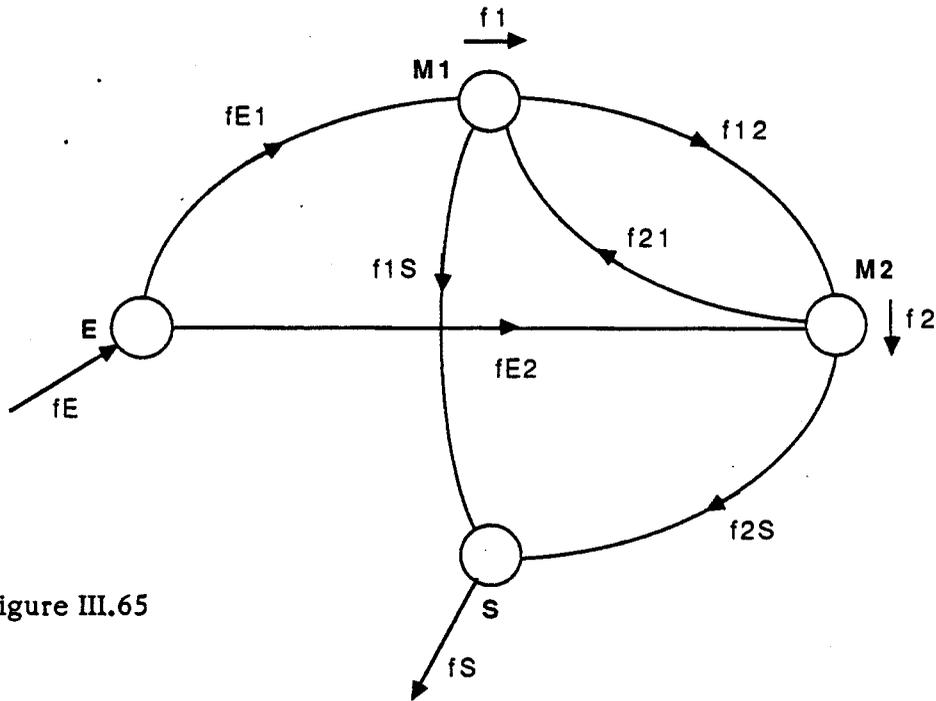


Figure III.65

Plaçons-nous dans l'hypothèse du régime permanent. On peut appliquer la loi de conservation des flux.

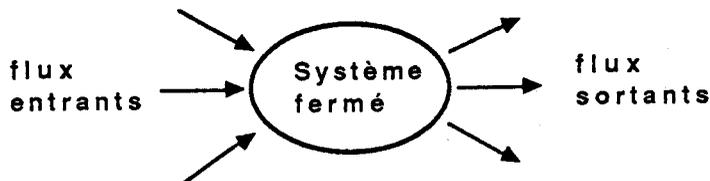


Figure III.66

$$\sum \text{flux entrants} = \sum \text{flux sortants}$$

En distinguant la composition de chacun des flux on a alors :

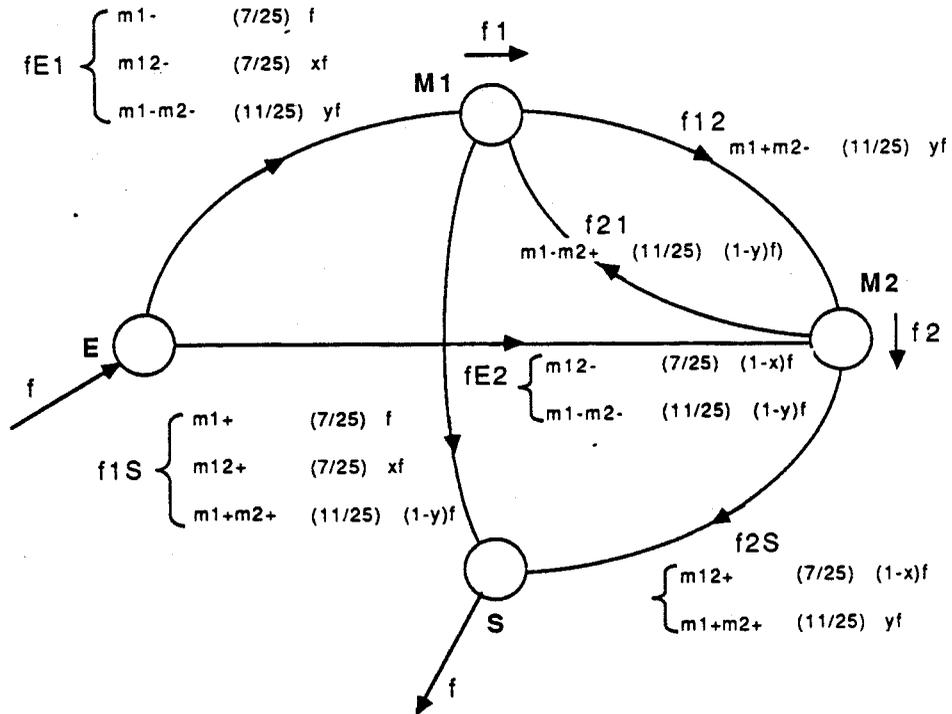


Figure III.67

où  $f$  est le débit global de la cellule.

Au niveau de l'entrée E apparaît un indéterminisme directionnel pour les pièces  $m12-$  et  $m1-m2-$  qui peuvent aller indifféremment vers M1 ou M2. On traduit la résolution de l'indéterminisme par les coefficients respectifs  $x$  et  $y$  ( $x \in [0,1]$ ,  $y \in [0,1]$ ).

Le flux total  $f1$  passant à travers la machine M1 est donné par la formule

$$f1 = fE1 + f21 = f1S + f12 = \frac{1}{25} (18 + 7x) f$$

Le flux total  $f2$  passant à travers la machine M2 est obtenu de la même façon :

$$f2 = f12 + fE2 = f21 + f2S = \frac{1}{25} (18 - 7x) f$$

Si on développe les places M1 et M2 pour faire apparaître les places tampons  $Ei$  et  $Si, i \in [1,2]$ , si on remplace le graphe coloré par un graphe non coloré étiqueté par les flux, si on supprime les transitions et si on étiquette les arcs par des temporisations on obtient le graphe ci-dessous :

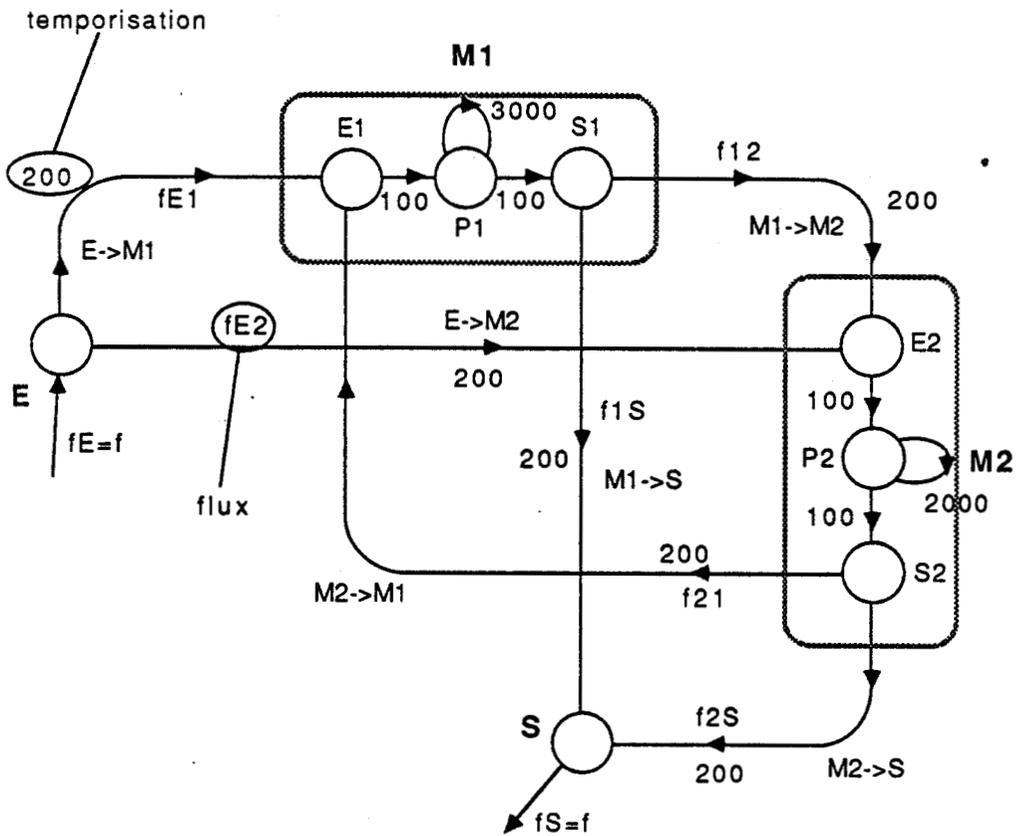


Figure III.68

avec

$$f_E = f_S = f$$

$$f_{E1} = \frac{1}{25} (7 + 7x + 11y) f$$

$$f_{E2} = \frac{1}{25} (18 - 7x - 11y) f$$

$$f_{12} = \frac{11}{25} y f$$

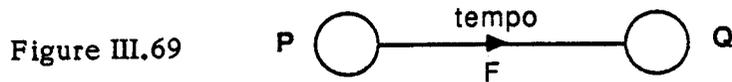
$$f_{21} = \frac{11}{25} (1 - y) f$$

$$f_{1S} = \frac{1}{25} (18 + 7x - 11y) f$$

$$f_{2S} = \frac{1}{25} (7 - 7x + 11y) f$$

Considérons une temporisation "tempo".

"tempo" représente le temps minimal séparant le transfert de deux pièces consécutives, à ne pas confondre dans le cas général avec le temps nécessaire au transfert d'une pièce de P vers Q



L'arc (P,Q) est également étiqueté par un flux F.

L'inverse de la temporisation représente le nombre de pièces maximal par unité de temps qui peuvent circuler suivant l'arc.

$$\frac{1}{\text{tempo}} = n \text{ pièces/unité de temps.}$$

Le flux F est donc majoré par le flux maximal admissible  $n = \frac{1}{\text{tempo}}$

$$f \leq \frac{1}{\text{tempo}}$$

On construit alors les contraintes sur les flux.

Les contraintes les plus fortes sont les trois contraintes ci-dessous :

Robot R : La somme des flux des transferts doit être inférieure ou égale au débit maximal  $f_R$  du robot R.

$$f_R = f_{E1} + f_{E2} + f_{12} + f_{21} + f_{1S} + f_{2S} \leq \frac{1}{200}$$

$$\text{soit } \frac{61}{25} f \leq \frac{1}{200}$$

ou encore

$$f \leq \frac{1}{488}$$

Machine M1 : Le temps de service unitaire est de  $100 + 3\,000 + 100 = 3\,200$  unités de temps

on a donc la contrainte  $f_1 = \frac{1}{25} (18 + 7x) f \leq \frac{1}{3200}$

Soit  $f \leq \frac{1}{128(18+7x)}$

Machine M2 : La contrainte est ici  $f_2 = \frac{1}{25} (18 - 7x) f \leq \frac{1}{2200}$

Soit  $f \leq \frac{1}{88(18-7x)}$

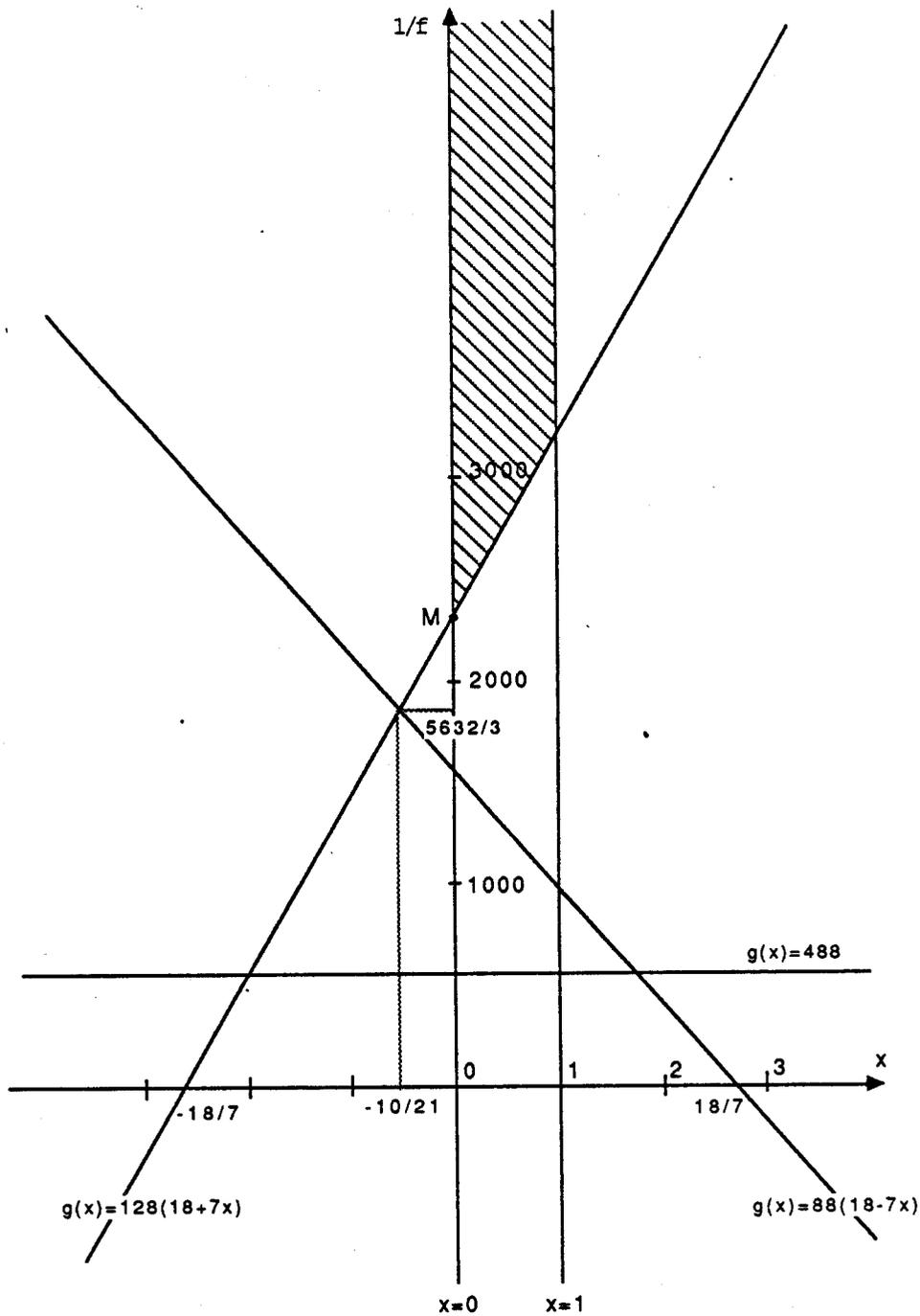


Figure III.70

Les contraintes imposent au point  $\begin{pmatrix} x \\ 1/f \end{pmatrix}$  d'être situé dans la zone hachurée.

Le flux maximal sera obtenu au point M.  $f_{\max} = 1 \text{ pièce}/2304 \text{ unités de temps}$

$x=0$

$y$  peut être quelconque dans  $[0,1]$  ce qui est tout à fait normal étant donné la symétrie des parcours possibles des pièces  $m_1$ - $m_2$ -

il est alors possible de calculer numériquement tous les flux. En particulier :

$$f1 = \frac{1}{25} (18 + 7 x) \quad f = \frac{1}{3200} \quad \text{pour la machine M1}$$

$$f2 = \frac{1}{25} (18 - 7 x) \quad f = \frac{1}{3200} \quad \text{pour M2}$$

et

$$fR = \frac{61}{25} \quad f = \frac{1}{686} \quad \text{pour le robot R}$$

Les débits maximaux pour M1, M2 et R étant

$$D1 = \frac{1}{3200}$$

$$D2 = \frac{1}{2200}$$

$$DR = \frac{1}{200}$$

On obtient les taux d'engagements suivants pour les différents matériels :

$$M1 : \quad t1 = \frac{f1}{D1} = \frac{3200}{3200} = 1$$

$$M2 : \quad t2 = \frac{f2}{D2} = \frac{2200}{3200} \approx 0,69$$

$$R : \quad tR = \frac{fR}{DR} = \frac{200}{686} \approx 0,29$$

On en déduit bien évidemment que M1 est le goulet d'étranglement du système. Il est donc nécessaire de le reconfigurer.

### La programmation linéaire

Dans le cas général, l'introduction des coefficients de résolution des indéterminismes tels que  $x$  et  $y$  peut conduire à des contraintes multivariées. Moyennant éventuellement un changement de variables fractionnaire, suivant la façon dont on formule le fractionnement des flux, le problème peut être ramené à un problème qui est du ressort de la programmation linéaire ([FAU 79]).

Les contraintes, si elles sont compatibles, délimitent dans un espace à  $n+1$  dimensions ( $n$ =nombre de variables) un hypervolume convexe à l'intérieur ou à la périphérie duquel se trouve(nt) le (ou les) point(s) dont les coordonnées  $(x_1, x_2, \dots, x_n, \frac{1}{f})$  satisfont aux contraintes.

La programmation linéaire a alors pour objet de résoudre le problème d'optimisation qui consiste à maximiser  $f$  dans le domaine ainsi défini.

### Réseau monoclasse ou multiclassé

La représentation qui a été utilisée pour les calculs précédents a confondu tous les objets circulant dans le système de production en une seule classe d'objets. Les services sont alors considérés comme étant identiques pour tous les types d'objets (par exemple le temps d'usinage sur M1 est supposé être unique pour les objets de types  $m1-$ ,  $m12-$ , et  $m1-m2-$ ). Le routage des objets est défini de façon probabiliste. Les flux sont calculés en fonction des ratios de production fixés pour chacune des gammes. On dit que le réseau est monoclasse.

Il est bien entendu possible de construire une étude similaire en distinguant les types d'objet et les services relatifs à ces objets (on distingue par exemple les temps d'usinage et de transfert pour chacune des gammes). Ceci conduit à accroître le nombre des contraintes. (NB : si  $T_{m1}$ ,  $T_{m12}$  et  $T_{m1m2}$  sont les temps de service de la machine M1 pour des pièces de type  $m1-$ ,  $m12-$ ,  $m1-m2*$  on aura maintenant la contrainte

$$f < \frac{1}{\frac{1}{25}(7.T_{m1}+11.T_{m1m2}+7.T_{m12}.x)}$$

en effet, pendant un intervalle de temps  $T$

$$\frac{7}{25} f.T \text{ pièces } m1- + \frac{7}{25} x f.T \text{ pièces } m12- + \frac{11}{25} f.T \text{ pièces } m1-m2*$$

arrivent à la machine M1

le temps nécessaire pour les usiner est alors donné par :

$$\frac{7}{25} f.T.T_{m1} + \frac{7}{25} x f.T.T_{m12} + \frac{11}{25} f.T.T_{m1m2}$$

qui doit bien sûr être inférieur ou égal à  $T$  pour qu'il n'y ait pas engorgement).

## La contribution du Prégraphe

L'étude précédente a montré l'importance de l'outil que représente le Prégraphe en tant que graphe de flux dans la construction des calculs. Cette démarche peut être tout à fait informatisée à partir du modèle Prégraphe et d'un ensemble de questions posées à l'utilisateur.

Nous allons aborder maintenant un autre outil pour lequel le Prégraphe va pouvoir servir également de base de construction et qui introduit une modélisation du régime permanent plus fine que celle qui vient d'être examinée.

### a.2) Les files d'attente

La notion d'attente était absente du schéma précédent. Les modèles basés sur les files d'attente permettent de l'intégrer.

### Principes généraux

De façon générale, une file d'attente est constituée par un centre de service pouvant comprendre plusieurs serveurs, et une zone d'attente dans laquelle les objets clients de ces services attendent lorsque tous les serveurs du centre de service sont occupés.

Les paramètres qu'il faut fixer pour caractériser une file d'attente sont donnés ci-dessous :

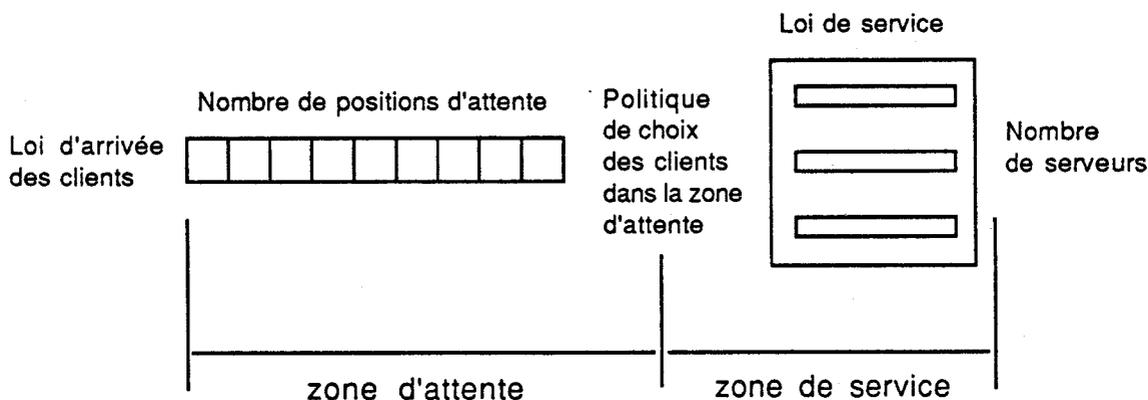


Figure III.71

Un réseau de files d'attente, comme le montre la figure suivante, est un ensemble de N files d'attente parcourues par des clients selon un cheminement défini de façon soit probabiliste (probabilité, à la sortie d'une file d'attente, d'aller dans une autre file d'attente ou de sortir du réseau), soit déterministe (succession des files d'attente parcourues par un type de client).

Les lois de routage des clients à la sortie des files d'attente, ainsi qu'à l'arrivée dans le réseau, sont ainsi à rajouter aux paramètres déjà définis précédemment pour caractériser un réseau de files d'attente.

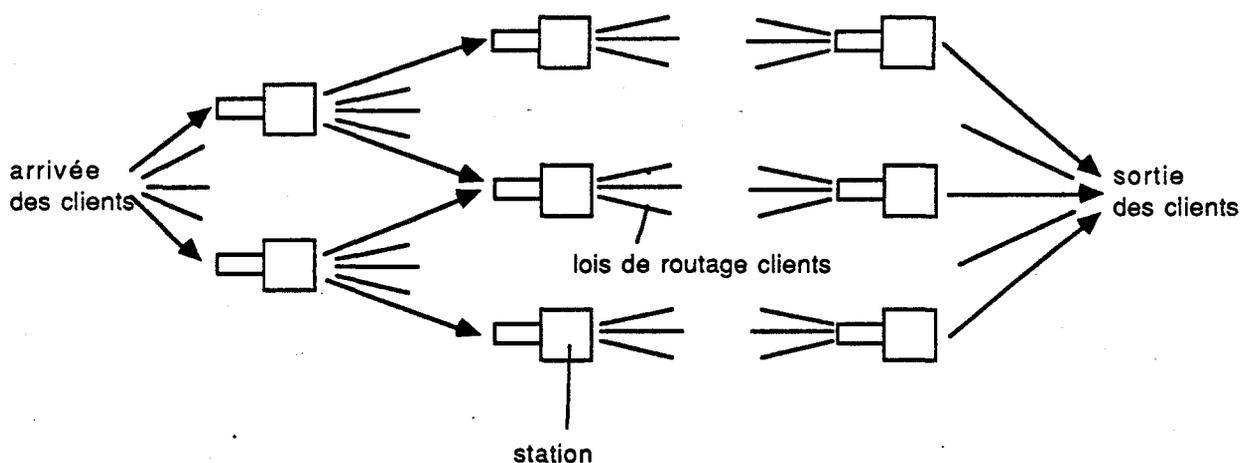


Figure III.72

Les résultats que l'on peut attendre de la théorie des files d'attente sont essentiellement des valeurs moyennes de paramètres de fonctionnement en régime permanent.

Le Prégraphe peut également ici être d'une grande utilité dans la construction du modèle du réseau à files d'attente. Toutefois le problème est ici complexe. Les hypothèses simplificatrices sur lesquelles reposent les outils d'étude exploitables ne permettent de travailler que dans un cadre très restrictif. La construction d'un modèle à files d'attente nécessite une certaine expérience pour obtenir une précision et un niveau de détail intéressant, compte tenu des outils de résolution qui existent. C'est un domaine où l'introduction des systèmes experts donnera probablement des résultats intéressants, comme cela se voit déjà en Automatique des processus continus pour les méthodes de modélisation (identification) et d'optimisation de la commande. Les propriétés de l'architecture du Prégraphe seront importantes dans le déroulement de l'expertise.

## Méthodes

Il n'est pas question de développer ici les méthodes de résolution qui existent. Ce paragraphe n'a pour but que de fournir un certain nombre de références d'étude.

Un survol de présentation donnant une bonne idée des méthodes disponibles est fait dans [CAV 87]. Une introduction aux calculs est donnée dans le Précis de Recherche Opérationnelle de Robert FAURE ([FAU 79]), dans le Guide de la Recherche Opérationnelle de ALJ et FAURE ([ALJ 86]) et dans ([ROS 87]).

Les ouvrages de référence que nous utilisons sont essentiellement ([GEL 81] [GEL 82] et [KLE 76])

### a.3) Etude de stratégies de production en régime permanent

Nous avons vu dans les deux paragraphes précédents des méthodes qui permettent de détecter par exemple les goulets d'étranglement et qui donnent une aide pour configurer l'architecture matérielle et donc la partie Commande dans l'hypothèse d'un régime permanent.

L'introduction des coefficients de résolution d'indéterminismes tels que  $x$  et  $y$  (cf § a.1) se réfère, elle, à l'étude de stratégies de production par l'intermédiaire du routage des pièces.

Citons une étude très intéressante concernant la construction d'une stratégie de coopération entre cellules flexibles de production en régime permanent ([BER 86]). Les auteurs présentent un exemple dans lequel un ensemble de cellules flexibles possède un parc de chariots filoguidés qui sont répartis entre les cellules mais qui peuvent être partagés en cas de surcharge du potentiel de transport de l'une d'elles. L'étude cherche à déterminer quelle stratégie de coopération permet d'obtenir les temps de réponse minimaux de la part du système de transport. Un modèle analytique complexe est construit de façon à pouvoir être résolu par une technique d'optimisation en programmation linéaire. La construction des calculs est faite à partir d'un graphe de flux.

Citons enfin, en marge de l'étude des régimes permanents, des travaux relatifs à la Technologie de Groupe et permettant de construire une

architecture de production favorisant l'installation d'un régime permanent. Ces techniques ([NAD 85]) qui ont pour but de regrouper les gammes à l'intérieur de classes en fonction de leur ressemblance (suivant des critères) permettent de construire des cheminements privilégiés pour chacune des classes dans une unité de production. Ceci permet de ne plus travailler sur des gammes trop précises mais sur des classes. L'établissement de cheminements privilégiés permet en outre de faire passer sur des zones déterminées de l'unité de production des gammes qui ne nécessitent pas, du fait de leur proximité, des changements d'outils trop fréquents et des temps de préparation des machines importants entre chaque pièce.

b) Etude des régimes non permanents

b.1) Affinement par simulation de la détermination des limites d'un régime permanent

Les outils analytiques d'étude des régimes permanents font appel à un certain nombre d'hypothèses de simplification qui permettent de mener les calculs de résolution. Par exemple, pour les files d'attente, l'hypothèse stochastique (ou Markovienne) de comportement des lois d'arrivée et de service est générale (exception faite de certains cas très particuliers et de l'analyse opérationnelle ([DAL 86])). Egalement, sauf dans quelques cas limités d'utilisation des méthodes de conservation ([LEM 77] [LEM 78] [KRA 73]) ou dans quelques exemples particuliers ([JOH 85]), la capacité des files d'attente est supposée être infinie.

Ces écarts de la modélisation avec la réalité font que certains problèmes d'engorgement sont négligés et qu'il est d'abord nécessaire de réajuster l'évaluation des limites du régime permanent par simulation.

Un exemple donné dans ([BEN 85]) montre comment un circuit de chariots automoteurs, servant à la manutention de colis dans un magasin, a dû être complètement redessiné après simulation à cause d'un blocage par engorgement qui n'avait pas été détecté par les méthodes analytiques.

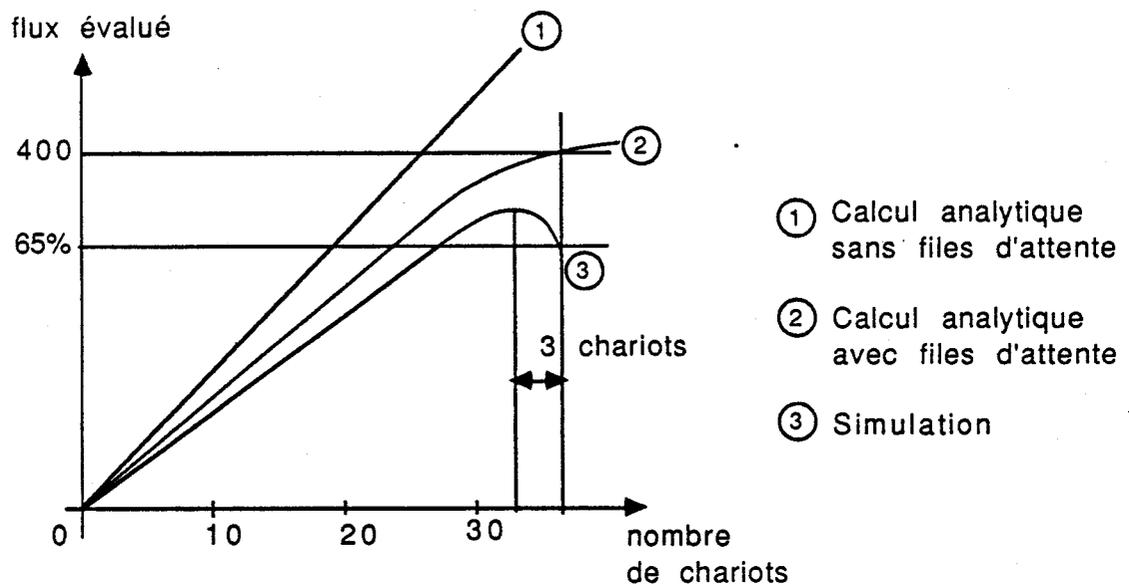


Figure III.73

### b.2) Les unités de production flexibles

La simulation va avoir maintenant un but supplémentaire. Il s'agit de construire, grâce aux résultats, des règles du niveau hiérarchique qui permettent de ne pas trop s'éloigner du fonctionnement qu'aurait un régime permanent reprenant par exemple les mêmes ratios de production mais en grande série. Les règles devront, dans la mesure du possible, permettre de respecter la résolution des indéterminismes de routage déterminée en régime permanent.

On peut distinguer ici deux types d'unités flexibles :

- les ateliers flexibles moyenne série.
- les ateliers flexibles petite série.

#### —> Ateliers moyenne série

Ces ateliers ont des gammes très variées mais fixées pour une période de temps importante.

L'étude du régime permanent et l'ajustement du régime réel par simulation sont tout à fait justifiés. La simulation permettra de lever un certain nombre de cas de blocage et d'engorgement.

Les outils d'étude du régime permanent peuvent donner dans ce type de cas des résultats très justes même en présence d'approximations.

→ Ateliers petite série

Ils sont d'étude très délicate.

Les gammes sont très mouvantes et en tout cas fixées pour un temps très court.

Deux situations peuvent encore être envisagées :

- i) Les ordres de lancement des nouvelles gammes attendent la fin de la série précédente.

Il est dans ce cas possible d'établir des ratios qui permettent d'étudier un régime permanent asymptotique et de déterminer des coefficients de résolution des indéterminismes directionnels de routage. Une simulation plus ou moins simplifiée peut intervenir pour faire l'ajustement du niveau hiérarchique. Cela exige bien sûr d'avoir du temps. C'est possible si les ordres de lancement font l'objet d'une gestion prévisionnelle stricte et si la démarche est complètement informatisée. Les améliorations viendront de l'augmentation des performances des outils informatiques.

- ii) La deuxième situation est la plus délicate. Il s'agit du cas où les ordres de lancement d'une série de gammes sont donnés alors que la série précédente n'est pas encore terminée.

On est dans ce cas dans un régime tout à fait transitoire.

Le prédimensionnement et l'ajustement ne peuvent être universels.

La seule solution est alors de prévoir une architecture de production qualitativement performante (avec dérivations, itinéraires multiples, et un minimum de conflits d'accès à des ressources) pour éviter dans le cas le plus général les blocages les plus élémentaires.

L'étude de prédimensionnement consiste alors à mener une étude expérimentale discrétisée, en faisant varier les ratios, les coefficients de résolution d'indéterminismes, c'est-à-dire en essayant le maximum de routages compte tenu des grandes classes de gammes à envisager, pour localiser le maximum de blocages.

### b.3) Les outils de simulation

De nombreux outils de simulation connus permettent de travailler sur des modèles basés sur les files d'attente. Citons QNAP 2 (Queing Network Analysis Package 2) qui dispose également de solveurs analytiques permettant l'étude préalable du régime permanent (QNAP 2 est développé par la Société SIMULOG). Citons RESQ (Research Queueing Package) issu d'une refonte complète de GPSS et développé par le Centre de recherche I.B.M. à Yorktown, NY, U.S.A. Citons également SLAM II, SIMAN, SIMSCRIPT. Tous ces outils de simulation sont répertoriés dans (ICAV 87I).

Pour le moment, une adaptation du simulateur développé par Emmanuel CASTELAIN a été réalisée (IHEI 88bI) pour travailler sur un modèle dérivé du Prégraphe. La version épurée et spécialisée du simulateur est très rapide. Le modèle simulable dérivé du Prégraphe comprend un certain nombre d'extensions par rapport au modèle de base. En particulier :

- Des capacités sont attribuées aux places du Prégraphe et une zone d'attente leur est associée, gérée avec divers systèmes de priorité : FIFO, ...
- Des temporisations sont placées sur les arcs qui sont en amont et en aval des transitions.

Par exemple :

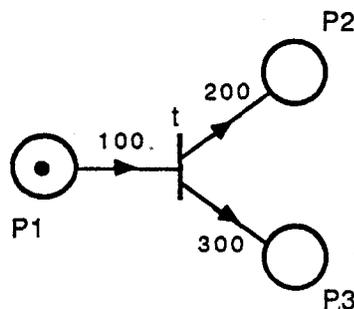


Figure III.74

Supposons qu'à l'instant 0 on ait la configuration décrite ci-dessus. La transition t est déclenchable, elle est tirée.

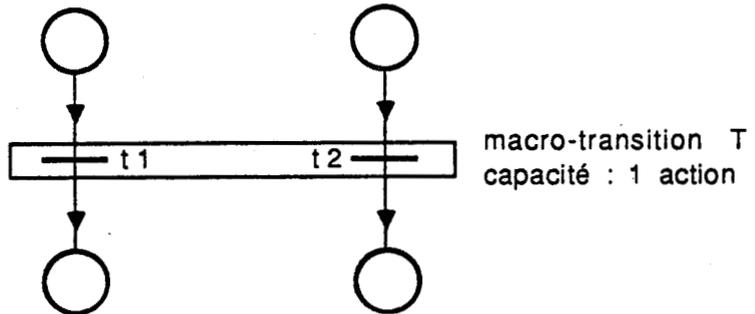
La marque qui n'est plus disponible pour d'autres tirs subsiste dans la place P1 jusqu'à l'instant 100, date à laquelle elle disparaît de cette place.

Une marque apparaît dans la place P2 à l'instant 200 et dans la place P3 à l'instant 300.

- Les transitions peuvent être regroupées en "macro-transitions" auxquelles sont également associées des capacités. On peut ainsi représenter la limitation des ressources qui assurent l'exécution des transitions composantes des macro-transitions.

Par exemple :

Figure III.75



Si la capacité de la macro-transition est d'une transition tirable à la fois, on aura alors exclusion mutuelle entre t1 et t2. Le choix sera fait par l'intermédiaire d'une fonction probabiliste basée sur le calcul des flux en régime permanent (voir calcul similaire dans l'alinéa suivant).

- Lorsque des indéterminismes directionnels se présentent, le tir s'effectue dans une des directions en fonction d'une fonction aléatoire respectant les ratios des flux calculés en régime permanent. Considérons par exemple la place P ci-dessous :

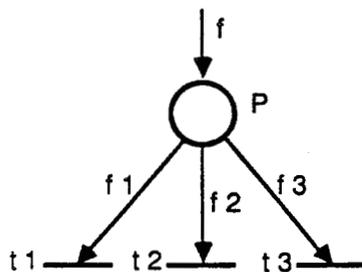


Figure III.76

$$f_1 = xf, \quad f_2 = yf, \quad f_3 = (1-x-y)f$$

Si les flux suivant les trois arcs (P,t1), (P,t2) et (P,t3) sont respectivement  $f_1$ ,  $f_2$  et  $f_3$ , alors la probabilité de tir suivant chacun de ces arcs est respectivement

$$\frac{f_1}{f_1+f_2+f_3} = x, \quad \frac{f_2}{f_1+f_2+f_3} = y \quad \text{et} \quad \frac{f_3}{f_1+f_2+f_3} = 1-x-y$$

- Enfin, les entrées de pièces sont déterminées par des fonctions aléatoires basées sur des lois statistiques. Ces lois sont déterminées par l'étude d'histogrammes fournies par la production qui est en amont de l'unité étudiée, ou sont construites autour des calculs de flux effectués dans l'hypothèse du régime permanent. Si on connaît les taux moyens d'arrivée et les écarts types pour chacun des types de pièces, il est alors possible de générer l'échéancier d'entrée. Les techniques en simulation de génération des entrées suivant des lois statistiques sont décrites dans [PRI 84].

L'outil de simulation présenté permet une détection efficace de situations de blocage qui sont autrement très difficiles à prévoir. Il doit permettre également de dimensionner les tampons de stockage destinés à absorber les fluctuations génératrices d'engorgements.

Une première étape de conception du Niveau Hiérarchique sera également possible en remplaçant les lois décrites de résolution des indéterminismes par des lois plus nuancées.

## CONCLUSION

A l'issue de ce dernier chapitre, il est clair que le PREGRAPHE est un outil central dans la chaîne de conception des cellules de production et bien entendu, en particulier, des cellules flexibles. La raison en est simple. Le prégraphe définit les actions élémentaires fondamentales que l'architecture matérielle et le système de commande devront autoriser. Il peut être utilisé aussi bien comme graphe de flux pour le prédimensionnement de l'architecture matérielle que comme squelette du système de coordination et de séquençement des commandes potentielles applicables au Procédé.

Le rôle du Prégraphe est encore destiné à s'accroître avec l'évolution du projet CASPAIM. Une plus grande importance est maintenant donnée aux étapes de dimensionnement et de validation de comportement réalisées à partir du modèle Prégraphe enrichi et simulable, moins lourd à manipuler que les modèles fins. La démarche d'enrichissement progressif du Prégraphe initial afin d'obtenir le modèle Prégraphe simulable amène à poser dès ce niveau des questions qui étaient jusqu'alors réservées à la phase de développement structuré et au modèle développé résultant. Il s'agit en particulier de la réduction des parallélismes (exclusions mutuelles, synchronisations), de la capacité des ressources et lieux physiques, de la temporisation des actions, et également de l'étude des indéterminismes et des blocages permettant d'aborder la construction du Niveau Hiérarchique. La phase de développement structuré tend à se limiter à une simple étape de compilation finale, le modèle résultant ne nécessitant plus d'ajustements majeurs ni de retours-arrière. Les figures ci-dessous résument l'évolution (figures III.77 et III.78).

Enfin, étant donné l'enrichissement du modèle Prégraphe simulable, il est pratiquement envisageable de faire de la commande macroscopique à partir de ce modèle ou de s'en servir en simulation prédictive, comme outil d'aide à la décision du Niveau Hiérarchique au sein même de la commande et non plus seulement en phase de préétude. Toutes ces utilisations constituent des champs d'investigation qu'il faudra développer.

**LE PROJET CASPAIM HIER**

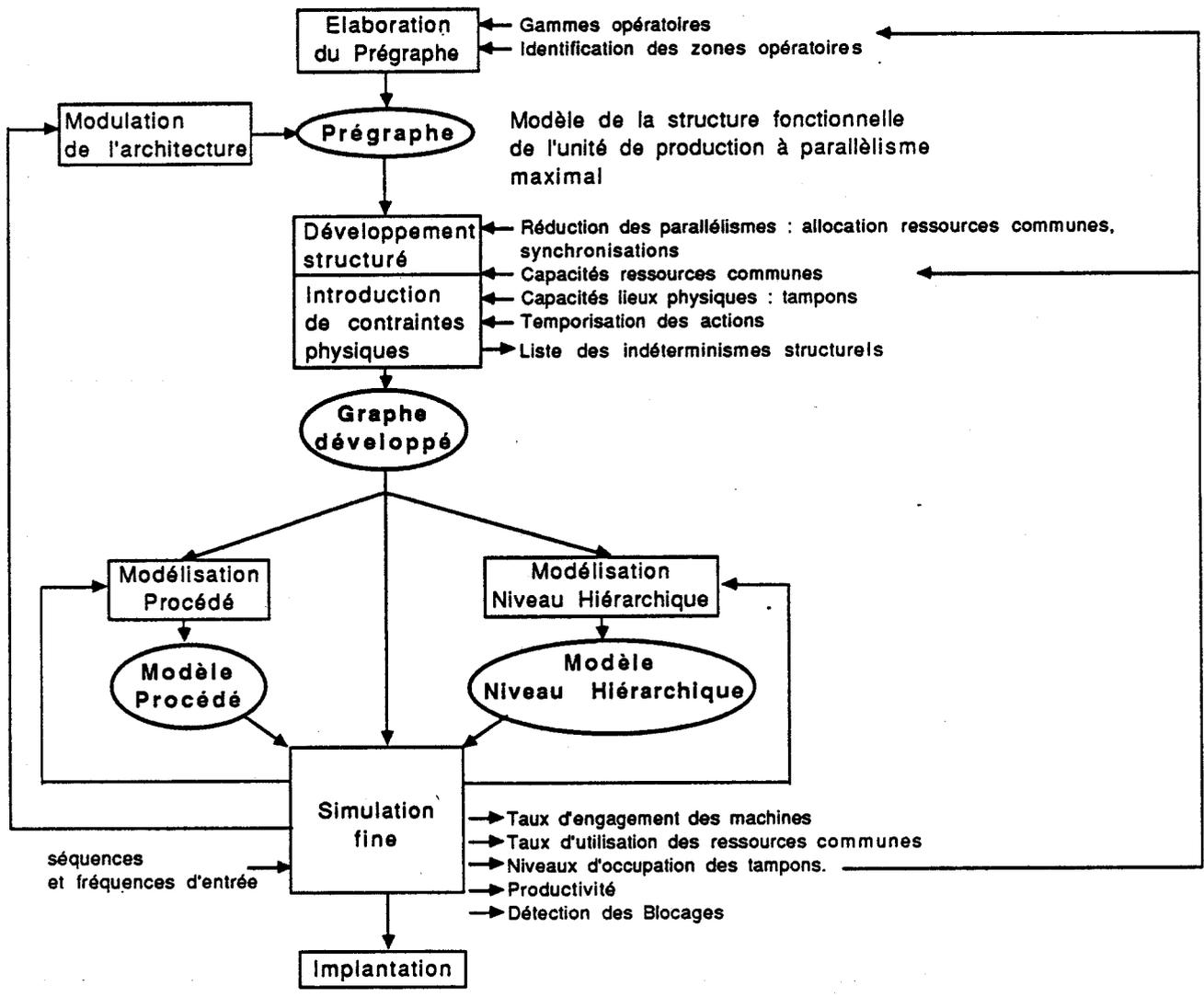


Figure III.77



**CONCLUSION GENERALE**

## 1) A PROPOS DU PREGRAPHE

Nous avons présenté dans ce mémoire un outil essentiel faisant partie de la chaîne de conception du système de commande de cellules flexibles de production.

A partir de la spécification des cheminements et des actions élémentaires que le système de commande doit assurer, cet outil fournit le schéma de la construction ultérieure systématique de la Partie Commande. Celle-ci est chargée d'assurer le séquençement et la coordination des commandes applicables au Procédé. Elle doit, de plus, être rapidement reconfigurable.

Ce schéma de construction constitue une base de travail solide - cohérente, complète et concise - et permet d'éviter des erreurs de conception dues à des spécifications trop hâtives, entâchées d'erreurs et de lacunes. Rappelons que la méthode invoquée repose d'abord sur une description formalisée des actions élémentaires et des zones opératoires associées assistée par un logiciel d'analyse de cohérence et de complétude d'enchaînement logique. Une aide à la conception des routages est également apportée par une approche de la combinatoire des gammes. Une traduction et une agrégation des descriptions des gammes et des routages sont alors effectuées pour obtenir un graphe RdP Coloré portant les informations de base nécessaires à l'obtention du graphe RdP Structuré, Adaptatif, Coloré de la Partie Commande. Des étapes de modulation viennent compléter la démarche.

Nous avons montré comment la Partie Commande est obtenue à partir du Prégraphe par une étape pouvant être assimilée à une compilation avec enrichissement. Les limites et l'évolution de la modélisation de la Partie Commande ont été discutées.

Nous avons également montré que, par son essence même, le Prégraphe pouvait, moyennant quelques enrichissements, servir de base à l'étude de prédimensionnement de l'architecture matérielle et à une préétude du comportement dynamique et des stratégies de routage.

Le Prégraphe est bien destiné à devenir un outil central de la conception globale du comportement des unités flexibles et du système de commande chargé d'assurer ce comportement.

Le Prégraphe fait à ce titre penser aux graphes de flux qui sont à la base des schémas directeurs des méthodes de conception et de développement des systèmes d'information.

Les développements qui sont en cours maintenant autour du Prégraphe portent essentiellement sur trois points :

- 1) Le premier travail consiste à faire une étude systématique des problèmes limites de modélisation qui ont été évoqués au chapitre III Partie II. Pour cela, seule une étude concrète sur de nombreux exemples industriels peut apporter des réponses pragmatiques.

Cette démarche est nécessaire pour ne pas s'éloigner des préoccupations industrielles réelles et pour tester efficacement la méthode.

- 2) Le deuxième travail porte sur l'utilisation du modèle dérivé et simulable du Prégraphe pour le prédimensionnement et la préeétude du comportement dynamique. Des résultats intéressants ont déjà été obtenus sur l'analyse des phénomènes de blocage.
- 3) Enfin, le troisième travail va consister à rendre la construction du Prégraphe encore plus conviviale et plus rapide.

En particulier, l'utilisation de possibilités graphiques est à l'étude pour assister la saisie des règles de description en amont de l'analyse de cohérence et de complétude et pour donner une vision plus globale de l'évolution dynamique de cette analyse.

Un des buts recherchés est en particulier la possibilité d'introduire très rapidement de nouvelles gammes qui utilisent les potentialités de routage de l'architecture matérielle et du système de commande. Après analyse de cohérence et de complétude, seuls les domaines des variables libres utilisées par les routages nécessaires à la nouvelle gamme doivent être reconfigurés, ce qui peut être fait très rapidement sur les dispositifs d'implantation de la Partie Commande. Il est en effet très intéressant de pouvoir valider très vite de nouvelles gammes. Cela permet de diminuer considérablement le temps de préparation d'un nouvel ordre de fabrication (qu'on appelle aussi temps de set-up).

## 2) A PROPOS DU SYSTEME DE COMMANDE

La chaîne de conception de la Partie Commande qui est décomposée en deux volets est maintenant achevée par la jonction des deux logiciels qui forment un ensemble opérationnel. Les exigences de structuration et d'intégration dont on a parlé dans l'introduction générale sont respectées par le modèle et la méthode de conception. La Partie Commande représente le niveau de coordination et de séquençement des commandes applicables au Procédé. Il reste maintenant à s'intéresser à la conception du Niveau Hiérarchique qui regroupe les niveaux de décision et qui est construit jusqu'à maintenant de façon empirique. La conception du Niveau Hiérarchique utilisera à la fois une analyse descendante des niveaux de supervision et un regroupement par analyse montante des problèmes d'indéterminismes et de blocages de la Partie Commande.

D'autres équipes de recherche avec lesquelles le Laboratoire d'Automatique et d'Informatique Industrielle de l'I.D.N. a des échanges suivis travaillent dans une direction proche de la nôtre et ont fait des choix de modélisation tout à fait comparables (Réseaux de Petri pour les niveaux de coordination et de séquençement + Règles de production pour les niveaux de décision) (LAAS TOULOUSE |VAL 88|, Université de SARAGOSSE |MAR 88|). Citons également les travaux du LAG GRENOBLE |MER 88|. Cette concentration de travaux allant dans un sens commun tout en présentant des apports originaux laisse bon espoir de converger rapidement vers une bonne maîtrise des problèmes suscités par la conception et la réalisation des systèmes de commande des unités flexibles de production.

**BIBLIOGRAPHIE**

## BIBLIOGRAPHIE

### INTRODUCTION GENERALE

- [ATA 87a] ATABAKHCHE H.,SIMONETTI-BARBALHO D.,VALETTE R.,COURVOISIER M.  
"Commande d'ateliers : un compromis est-il possible entre une approche graphique et une approche intelligence artificielle ?"  
Revue RAIRO APII,Vol.21,n°4,pp 377-394,1987.
- [ATA 87b] ATABAKHCHE H.  
"Utilisation conjointe de l'intelligence artificielle et des réseaux de Petri : Application au contrôle d'exécution d'un plan de fabrication"  
Thèse de Doctorat de l'Université Paul Sabatier,TOULOUSE,18 Décembre 1987.
- [AUG 87] AUGEZ P.  
"Une méthodologie intégrée d'aide à la conception des systèmes de fabrication"  
Thèse de Docteur Ingénieur,I.D.N., LILLE, 9 Juin 1987.
- [BEN 86] BENASSY J.,BENCHIMOL G.,BLOCH G.,FERRE A.,PHILIP C.,ROSTAN G.,SAUVAGE L.,VAYSSIERE P.  
"L'usine intégrée par ordinateur"  
Editions HERMES,PARIS 1986.
- [BEN 87] BENASSY J.  
"La gestion de production"  
Editions HERMES,PARIS 1987.
- [BLA 86] BLANCHARD M.  
"Comprendre,maîtriser et appliquer le GRAFCET"  
Editions CEPADUES-TOULOUSE 1986.
- [BON 85] BONETTO R.  
"Les Ateliers Flexibles de Production"  
Editions HERMES,PARIS 1985.
- [BOU 88] BOUREY J.P.  
"Structuration de la partie procédurale du système de commande de cellules de production flexibles dans l'industrie manufacturière"  
Thèse de Doctorat de l'Université, LILLE, 23 Mars 1988.
- [BOU 87] BOUTEILLE D.,BOUTEILLE N.,CHANTREUIL S.,COLLOT R.,FRACHET J.P.,LE GRAS H.,MERLAUD C.,SELOSSE J.,SFAR A.  
"Les Automatismes Programmables"  
Editions CEPADUES-TOULOUSE 1987.
- [BRA 79] BRARD P.,BOSSY J.C.,MERLAUD C.,FAUGERE P.  
"Le GRAFCET,sa pratique et ses applications"  
Editions EDUCALIVRE,1979.
- [CAS 87] CASTELAIN E.  
"Modélisation et simulation interactive de cellules de production flexibles dans l'industrie manufacturière"  
Thèse de Doctorat de l'Université, LILLE, 26 Février 1987.
- [COR 86] CORIAT B.,BOYER R.  
"Technical Flexibility and Macro-Stabilisation"  
Colloque de VENISE (ITALIE) ,Avril 1986.

- [DEF 85] DEFAUX M.,LOREAL A.  
"Atelier flexible : commencez petit"  
Revue L'Usine Nouvelle,pp 47-59, Avril 1985.
- [FRO 84] FROMENT B.,LESAGE J.J.  
"Productique : les techniques de l'usinage flexible"  
Editions DUNOD,PARIS 1984, Série Génie Mécanique.
- [GER 83] GERWIN D.  
"A framework for analysing the flexibility of manufacturing processes"  
School of Business and Administration  
University of WISCONSIN,MILWAUKEE USA,1983.
- [GRE 85] GREPA (Groupe de travail réuni à l'ADEPA)  
"Le GRAFCET,de nouveaux concepts"  
Editions CEPADUES,1985.
- [ING 88] INGERSOLLS ENGINEERS  
"L'usine intégrée"  
Editions HERMES,PARIS 1988.
- [KAR 87] KARFIA A.  
"Présentation d'une base de connaissances adaptée à la modélisation par réseaux de Petri structurés du contrôle des processus de production discrétisés"  
Thèse de Doctorat de l'Université, LILLE, 9 Juillet 1987.
- [LAM 87] LAMY P.  
"Ordonnancement et gestion de production"  
Editions HERMES,PARIS 1987.
- [LHO 88] LHOSTE P.,TIXADOR J.M.,MOREL G.,ROESCH M.  
"Outils de Conception des Systèmes Automatisés de Production.  
Tools for the Design of Manufacturing Control Systems"  
Congrès AFCET Automatique 1988 "Quelle Automatique dans les Industries Manufacturières",Proc.pp 331-340,  
GRENOBLE (FRANCE),10-12 Octobre 1988
- [MAR 87] MARTINEZ J.,MURO P.R.,SILVA M.  
"Modelling,validation and software implementation of production systems using high level Petri-Nets"  
IEEE International conference on Robotics and Automation  
Proc.pp 1180-1185,RALEIGH (USA),31 Mars-3 Avril 1987.
- [MAR 88] MARTINEZ J.,MURO P.R.,SILVA M.,SMITH S.F.,VILLARROEL J.L.  
"Merging Artificial Intelligence Techniques and Petri-Nets for Real Time Scheduling and Control of Production Systems"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 528-531,Vol.3  
PARIS,18-22 Juillet 1988.
- [MER 88] MERCIER DES ROCHETTES R.,DESCOTES-GENON B.,LADET P.  
"Modelling and FMS Control Implementation"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 559-561,Vol.3  
PARIS,18-22 Juillet 1988.

- [PRU 87a] PRUNET F.,CAZALOT C.,STURLESE J.L.  
"De PIASTRE à AMI100 : Le GRAFCET en simulation"  
Conférences SIPRODIS sur la simulation en production discontinue  
Actes du séminaire pp 53-63,PARIS, 2-3 Juin 1987  
Editions EC2 Edition,Colloques & Conseil,Collection NOVOTIQUE,1987.
- [PRU 87b] PRUNET F., STURLESE J.L.,CAZALOT C.,GINESTET E.,PANAGET D.,  
DECHENAUX G.,LLORCA P.  
"Méthodologie et Implantation Automatique de commande d'automatisme à l'aide  
de la chaîne PIASTRE"  
Revue RAIRO APII,Vol.21,n°4,pp 299-321,1987.
- [SAH 87] SAHRAOUI A.  
"Contribution à la surveillance et à la commande d'atelier"  
Thèse de Doctorat de l'Université Paul SABATIER, TOULOUSE ,21 Octobre 1987
- [SLA 88] SLACK N.  
"Manufacturing systems flexibility : an assessment procedure"  
Revue Computer-Integrated Manufacturing Systems,  
Vol. 1,n° 1,pp 25-31,Février 1988,Butterworth & Co (Publishers) Ltd.
- [STO 79] STOFFAES C.  
"L'avenir de l'industrie automobile"  
1979.
- [THE 82a] THELLIEZ S.,TOULOTTE J.M.  
"GRAFCET et logique industrielle programmée"  
Editions EYROLLES-PARIS 1982.
- [THE 82b] THELLIEZ S.,TOULOTTE J.M.  
"Applications industrielles du GRAFCET"  
Editions EYROLLES-PARIS 1982.
- [VAL 88] VALETTE R.,CARDOSO J.,ATABAKHCHE H.,COURVOISIER M.,LEMAIRE T.  
"Petri-Nets and Production rules for Decision levels in FMS control"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 522-524,Vol.3  
PARIS,18-22 Juillet 1988.
- [VIL 88] VILLARROEL J.L.,MARTINEZ J.,SILVA M.  
"GRAMAN : A Graphic System for Manufacturing System Design"  
IMACS SMS '88 International Symposium on System Modelling and Simulation  
Proc. pp 79-84,18-21 Septembre 1988,CETRARO(CS),ITALIE.

## CHAPITRE 1

- [ALA 86] ALANCHE P.,SALVI P.,LHOSTE P.,MOREL G.,ROCHE M.,SALIM M.  
"Application de la modélisation de la Partie Opérative à la structuration de la commande"  
dans "Méthodes et outils modernes de conception et d'exploitation de la commande de procédés discontinus complexes"  
Journées AFCET de MONTPELLIER (FRANCE),6-7 Mars 1986.
- [ALL 86] ALLA H.,LADET P.,MARTIN F.,MARTINEZ J.,SILVA M.  
"Spécification de la commande d'ateliers flexibles à l'aide de Réseaux de Pétri colorés"  
dans "Méthodes et outils modernes de conception et d'exploitation de la commande de procédés discontinus complexes"  
Journées AFCET de MONTPELLIER (FRANCE),6-7 Mars 1986.
- [ANA 87] ANAKOK Y.  
"Une méthode d'analyse et de synthèse de la partie commande des machines et des systèmes automatisés de production"  
Thèse de Docteur Ingénieur en Automatique,ISMCM,S<sup>t</sup> OUEN,3 Juillet 1987.
- [AUG 87] AUGEZ P.  
"Une méthodologie intégrée d'aide à la conception des systèmes de fabrication"  
Thèse de Docteur Ingénieur,I.D.N., LILLE, 9 Juin 1987.
- [BEA 87] BEAUBOUCHER N.  
"Implantation répartie du graphe d'un système de commande sur des automates programmables reliés par réseau local"  
D.E.A. de Productique,L.A.I.I, I.D.N.,22 Juin 1987.
- [BEL 86] BEL G.,BERRADA M.,CORREGE M.,DUBOIS D.  
"SIMULISP: Un banc d'essai de systèmes experts pour la commande des ateliers de production"  
AFCET, Congrès Automatique, Proc.pp 463-475, TOULOUSE, Octobre 1985.
- [BEN 85] Groupe de prospective "Des usines pour demain"  
Coordinateur Guy BENCHIMOL  
"La conception des usines de demain"  
Editions HERMES,PARIS 1985.
- [BLA 86] BLANCHARD M.  
"Comprendre,maîtriser et appliquer le GRAFCET"  
Editions CEPADUES-TOULOUSE 1986.
- [BOI 88] BOIS S.  
"Implantation répartie du graphe d'un système de commande sur des automates programmables reliés par réseau local,et implantation de son niveau hiérarchique"  
D.E.A. de Productique,L.A.I.I, I.D.N.,29 Juin 1988.
- [BON 86] BONNEMAY A.  
"SYSTEL: Un logiciel de 4<sup>ème</sup> génération pour la conduite d'installations industrielles"  
Moyens d'automatisation dans les Industries Manufacturières,  
Convention Automatique Productique, Proc.pp 65-69,PARIS,Mai 1986.

- [BOU 86] BOUREY J.P.,CORBEEL D.,CRAYE E.,GENTINA J.C.  
"Adaptive and coloured structured Petri Nets for description,analysis and synthesis of hierarchical control and reliability of flexible cells in manufacturing systems"  
1<sup>st</sup> European Workshop on Fault Diagnostics,Reliability and Related Knowledge -based Approaches. Ile de RHODES (GRECE), Septembre 1986.  
Vol.1 ,pp 281-295,D.Reidel Publ.Comp.1987.
- [BOU 87a] BOUTEILLE D.,BOUTEILLE N.,CHANTREUIL S.,COLLOT R.,FRACHET J.P.,  
LE GRAS H.,MERLAUD C.,SELOSSE J.,SFAR A.  
"Les Automatismes Programmables"  
Editions CEPADUES-TOULOUSE 1987.
- [BOU 87b] BOUREY J.P.,CORBEEL D.,CRAYE E.,GENTINA J.C.  
"Utilisation des Réseaux de Petri structurés adaptatifs colorés dans l'analyse et la synthèse du contrôle hiérarchisé de processus discontinus. Partie A : Les modèles de description"  
Revue RAIRO APII,Vol.21,n°4,pp 343-362,1987.
- [BOU 87c] BOUREY J.P.,GENTINA J.C.  
"Computer aided design for structuration and representation of control of flexible manufacturing systems"  
8<sup>th</sup> European Workshop on application and theory of Petri Nets,  
Proc. pp 117-135, SARAGOSSE (ESPAGNE) ,24-26 Juin 1987.
- [BOU 88a] BOUREY J.P., GENTINA J.C.,KAPUSTA M.  
"Coloured Adaptive Structured Petri Nets : A tool for the automatic synthesis of hierarchical control of flexible manufacturing systems"  
Revue Computer-Integrated Manufacturing Systems,  
Vol. 1,n° 1,pp 39-47,Février 1988,Butterworth & Co (Publishers) Ltd.
- [BOU 88b] BOUREY J.P.  
"Structuration de la partie procédurale du système de commande de cellules de production flexibles dans l'industrie manufacturière"  
Thèse de Doctorat de l'Université, LILLE, 23 Mars 1988.
- [BOU 88c] BOUREY J.P.,CASTELAIN E.,GENTINA J.C.,KAPUSTA M.  
"C.A.S.P.A.I.M.: a computer aided design of the control system of F.M.S."  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 517-521,Vol.3  
PARIS,18-22 Juillet 1988.
- [BRA 83a] G.W.BRAMS  
"Réseaux de Pétri :Théorie et pratique"  
T.1 Théorie et analyse  
Editions MASSON,PARIS 1983.
- [BRA 83b] G.W.BRAMS  
"Réseaux de Pétri :Théorie et pratique"  
T.2 Modélisation et applications  
Editions MASSON,PARIS 1983.
- [BRA 79] BRARD P.,BOSSY J.C.,MERLAUD C.,FAUGERE P.  
"Le GRAFCET,sa pratique et ses applications"  
Editions EDUCALIVRE,1979.
- [CAS 85] CASTELAIN E.,CORBEEL D.,GENTINA J.C.  
"Comparative simulations of control processes described by Petri Nets"  
IEEE MONTECH'85-COMPINT'85,Proc. pp  
MONTREAL (CANADA),Septembre 1985.

- [CAS 87a] CASTELAIN E., CORBEEL D., GENTINA J.C.  
 "Utilisation des Réseaux de Petri structurés adaptatifs colorés dans l'analyse et la synthèse du contrôle hiérarchisé de processus discontinus. Partie B : Apports de l'Intelligence Artificielle dans la réalisation d'un simulateur de Réseaux de Petri adaptatifs colorés : Application à la conception et à la validation de la commande des processus industriels"  
 Revue RAIRO APII, Vol.2, n°4, pp 363-375, 1987.
- [CAS 87b] CASTELAIN E.  
 "Modélisation et simulation interactive de cellules de production flexibles dans l'industrie manufacturière"  
 Thèse de Doctorat de l'Université, LILLE, 26 Février 1987.
- [CAS 87c] CASTELAIN E., GENTINA J.C.  
 "Description of manufacturing processes by means of a type-frame language"  
 IMACS International Symposium on Artificial Intelligence, Expert Systems and languages in modelling and simulation", IMACS 1987, BARCELONE (ESPAGNE), Proc. pp 179-183, 2-4 Juin 1987.
- [CAV 87] CAVAILLE J.B., PROTH J.M.  
 "Pratique de la simulation en production discontinue"  
 Conférences SIPRODIS  
 Editions EC2 Edition, Colloques & Conseil, Collection NOVOTIQUE, 1987.
- [CER 82] Rapport du CERT (Centre d'Etudes et de Recherches de TOULOUSE)  
 "Théorie des réseaux à files d'attente et algorithmes pour l'évaluation des performances d'un atelier de production"  
 Rapport n°2, 1/7278, CERT-DERA, 1982.
- [COR 79] CORBEEL D.  
 "Schéma de cablage et schéma de contrôle. Application à la simulation et à la gestion des processus industriels"  
 Thèse de Doctorat de Spécialité, LILLE, 1979.
- [COR 80] CORBEEL D.,  
 "Formal description of processes systems and exception handling"  
 Mini & Micro, Proc. pp 335-339, BUDAPEST (HONGRIE), 9-11 Septembre 1980.
- [COR 81] CORBEEL D., GENTINA J.C., VERCAUTER C.  
 "Méthodologie de description des systèmes de processus et de gestion d'erreurs"  
 IASTED Modelling, Identification and control, DAVOS, 1981.
- [COR 83a] CORBEEL D., VERCAUTER C., GENTINA J.C.  
 "Spécification et conception des systèmes de conduite de processus temps réel"  
 Proc. of MIMI'83, LUGANO (SUISSE), pp 36-39.
- [COR 83b] CORBEEL D., VERCAUTER C., GENTINA J.C.  
 "Réduction et propriétés structurelles des réseaux de Petri structurés"  
 Proc. of MIMI'83, LUGANO (SUISSE), pp 40-43.
- [COR 83c] CORBEEL D., VERCAUTER C., GENTINA J.C.  
 "Généralisation des réseaux de Petri"  
 A.I.83, LILLE, Actes Vol.III, pp 201-206.
- [COR 84] CORBEEL D., GENTINA J.C., VERCAUTER C.  
 "Adaptive Petri Nets for real time applications"  
 IMACS Digiteh'84, PATRAS (GRECE), 1984.

- [COR 85a] CORBEEL D.,GENTINA J.C.,VERCAUTER C.  
 "Application of an extension of Petri Nets to modelization of control and production processes"  
 6<sup>th</sup> European workshop on application and theory of Petri Nets,  
 Proc. pp 53-74, ESPOO (FINLANDE),26-28 Juin 1985.
- [COR 85b] CORBEEL D.,VERCAUTER C., GENTINA J.C.  
 "Modélisation homogène du graphe de contrôle d'un système de conduite de processus industriels"  
 AFCET, Congrès Automatique 1985, TOULOUSE, pp 517-533.
- [COU 86] COURVOISIER M.,VALETTE R.  
 "Commande des procédés discontinus . Logique séquentielle"  
 Editions DUNOD Université/BORDAS-PARIS 1986.
- [CRA 88] CRAYE E.,BARBEZ J.M.,MAYET J.,GENTINA J.C.  
 "Hierarchical Level and Implementation for Analysis and Synthesis of Control and Reliability of Flexible Manufacturing Systems"  
 IMACS 88, 12<sup>ème</sup> Congrès Mondial IMACS, Proc. pp 552-558,Vol.3  
 PARIS, 18-22 juillet 1988.
- [CRU 87] CRUETTE D.  
 "Implantation répartie du graphe d'un système de commande sur des automates programmables reliés par réseau local,et implantation d'un niveau hiérarchique"  
 D.E.A. de Productique,L.A.I.I, I.D.N.,7 Septembre 1987.
- [DOL 83] DOLLE F.  
 "Méthode de Conception des systèmes automatiques pour la production discontinue"  
 Convention ADI n° 81/543, 1983.
- [FAL 85] FALLER B.  
 "L'ordinateur et les jeux de l'esprit"  
 Revue La Recherche ,Vol.16,n°170,pp 1164-1174,Octobre 1985  
 ou dans La Recherche en Intelligence Artificielle ,pp 51-76  
 Editions du SEUIL/La Recherche ,Collection POINTS.
- [FRA 87] FRACHET J.P.  
 "Concepts et outils généraux pour la manipulation interactive des graphismes technologiques utilisés en simulation et conception de systèmes automatisés"  
 Conférences SIPRODIS sur la simulation en production discontinue  
 Actes du séminaire pp 94-104 ,PARIS, 2-3 Juin 1987  
 Editions EC2 Edition,Colloques & Conseil,Collection NOVOTIQUE,1987.
- [GAS 88] GASNIER B.,VERCAUTER C.  
 "An interactive tool for computer-aided manufacturing sequence specification"  
 IFAC International Symposium, Trends in Control and Measurement Education  
 Proc. pp 297-300, SWANSEA (UK), 11-13 juillet 1988.
- [GEN 79] GENRICH H.,LAUTENBACH K.,THIAGARAJAN P.  
 "Elements of General Net Theory"  
 Proc. of the Advanced Course on General Net Theory of Processes and Systems"  
 HAMBURG (R.F.A.), 1979.

- [GEN 87a] GENTINA J.C.,CORBEEL D.  
 "Coloured adaptive structured Petri Nets : a tool for the automatic synthesis of hierarchical control of flexible manufacturing systems"  
 IEEE International Conference on Robotics and Automatism ,  
 Proc. pp 1166-1173, RALEIGH (USA) ,31 Mars-3 Avril 1987.
- [GEN 87b] GENTINA J.C.,BOUREY J.P.,KAPUSTA M.  
 "Coloured adaptive structured Petri Nets : a tool for the aided synthesis of the control system of flexible manufacturing systems (F.M.S.)"  
 Conférence IBM-International Education Centre  
 The Second Acis Studies Conference  
 Centre IBM de LA HULPE à BRUXELLES,14-16 Octobre 1987.
- / [GIR 84] GIRODC.  
 "La conception et la réalisation d'un logiciel de simulation de réseaux de Pétri"  
 Thèse de Docteur Ingénieur,I.D.N., LILLE ,25 Septembre 1984.
- [GRE 85] GREPA (Groupe de travail réuni à l'ADEPA)  
 "Le GRAFCET,de nouveaux concepts"  
 Editions CEPADUES,1985.
- [HAC 75a] HACK M.  
 "Petri-Net Languages"  
 Computation Structures Group Memo 124, Project MAC.  
 M.I.T., Cambridge (Mass.), June 1975.
- [HAC 75b] HACK M.  
 "Decision problem for Petri-Nets and vector addition systems"  
 Project MAC., TR 59  
 M.I.T., Cambridge (Mass.), March 1975.
- [KAP 87] KAPUSTA M.,GENTINA J.C.  
 "Introduction to a first step of the aided design of the control system of flexible manufacturing cells"  
 IEEE MONTECH'87-COMPINT'87,Proc. pp 258-262,  
 MONTREAL (CANADA),9-11 Novembre 1987.
- / [KAR 87] KARFIA A.  
 "Présentation d'une base de connaissances adaptée à la modélisation par réseaux de Petri structurés du contrôle des processus de production discrétisés"  
 Thèse de Doctorat de l'Université, LILLE, 9 Juillet 1987.
- / [LAU 86] LAURIERE J.L.  
 "Intelligence Artificielle : résolution de problèmes par l'homme et la machine"  
 Chap.VI,pp 255-297  
 Editions EYROLLES-PARIS 1986.
- [MIC 87] BITARD C.  
 "Assembleur 8086 : premiers pas vers la norme"  
 Revue MICRO-SYSTEMES, n°75, Mai 1987, pp 152-157
- [MOA 85] MOALLA M.  
 "Réseaux de Petri interprétés et Grafcet"  
 Revue TSI Technique et Science Informatique, Vol.4, n°1, pp 17-30  
 Janvier-Février 1985, Editions DUNOD/AFCEC-BORDAS.

- [PEC 87] PECQUET P.  
"Maquette de pilotage d'un atelier reconfigurable"  
IEEE MONTECH'87-COMPINT'87, Proc. pp 275-278,  
MONTREAL (CANADA), 9-11 Novembre 1987.
- [PRI 79] PRITSKER A. A. B., PEGDEN C.D.  
"Introduction to Simulation and SLAM I"  
Editions John WILEY , NEW-YORK 1979.
- [PRU 87a] PRUNET F., CAZALOT C., STURLESE J.L.  
"De PIASTRE à AMI100 : Le GRAFCET en simulation"  
Conférences SIPRODIS sur la simulation en production discontinue  
Actes du séminaire pp 53-63, PARIS, 2-3 Juin 1987  
Editions EC2 Edition, Colloques & Conseil, Collection NOVOTIQUE, 1987.
- [PRU 87b] PRUNET F., STURLESE J.L., CAZALOT C., GINESTET E., PANAGET D.,  
DECHENAUX G., LLORCA P.  
"Méthodologie et Implantation Automatique de commande d'automatisme à l'aide  
de la chaîne PIASTRE"  
Revue RAIRO APII, Vol.21, n°4, pp 299-321, 1987.
- [SAH 87] SAHRAOUI A.  
"Contribution à la surveillance et à la commande d'atelier"  
Thèse de Doctorat de l'Université Paul SABATIER, TOULOUSE , 21 Octobre 1987
- [SCH 84] SCHIPER A., CORAY G., HIRSBRUNNER B.  
"Une structure de contrôle à deux niveaux pour la programmation heuristique  
parallèle"  
Revue TSI Technique et science informatique, Vol. 3, n°1, pp 33-43,  
Janvier-Février 1984, Editions DUNOD / AFCET-BORDAS.
- [SOL 80] SOLBERG J.  
"The optimal planning of computerized manufacturing systems"  
1980.
- [TAH 87] TAHON C., SALLEZ Y., SOENEN R.  
"Systèmes automatisés de production : un outil de simulation intégré utilisant  
le langage ADA"  
IEEE MONTECH'87-COMPINT'87, Proc. pp 283-286,  
MONTREAL (CANADA), 9-11 Novembre 1987.
- [TEL 87] BOUREY J.P., CASTELAIN E., CRAYE E., GENTINA J.C., KAPUSTA M.  
Contrat d'Etude TELEMECANIQUE-IDN  
Rapport n°2, 6 Novembre 1987.
- [THE 82a] THELLIEZ S., TOULOTTE J.M.  
"GRAFCET et logique industrielle programmée"  
Editions EYROLLES-PARIS 1982.
- [THE 82b] THELLIEZ S., TOULOTTE J.M.  
"Applications industrielles du GRAFCET"  
Editions EYROLLES-PARIS 1982.
- [VAL 77] VALK R.  
"Self-modifying Nets"  
Bericht 34, Fachbereich Informatik, HAMBURG 1977.

[VAL 78] VALK R.  
"Self-modifying Nets, a natural extension of Petri-Nets"  
Lecture notes in Computer Science, Vol.62,pp 464-476  
Springer Verlag, BERLIN 1978.

[VER 82] VERCAUTER C.  
"Sur un ensemble d'outils d'aide à la spécification et à la  
conception des systèmes industriels"  
Thèse de Doctorat de 3<sup>ème</sup> Cycle, LILLE, 5 Juillet 1982.

## CHAPITRE 2

- [AUG 87] AUGEZ P.  
"Une méthodologie intégrée d'aide à la conception des systèmes de fabrication"  
Thèse de Docteur Ingénieur, I.D.N., LILLE, 9 Juin 1987.
- [AYE 86] AYEL M., PIPARD E., ROUSSET M.C.  
"Le contrôle de cohérence dans les bases de connaissances"  
Journées Nationales sur l'Intelligence Artificielle  
organisées par le PRC-GRECO IA  
Proc.pp 171-185.
- [BAP 84] BAPTISTE P., FAVREL J.  
"Base de données pour la gestion d'un atelier flexible"  
Colloque international sur la productique et la robotique,  
BORDEAUX (FRANCE), 6-8 Mars 1984.
- [BEN 86] BENASSY J., BENCHIMOL G., BLOCH G., FERRE A., PHILIP C., ROSTAN G.,  
SAUVAGE L., VAYSSIÈRE P.  
"L'usine intégrée par ordinateur"  
Editions HERMES, PARIS 1986.
- [BOU 84] BOURJAUULT A.  
"Contribution à une approche méthodologique de l'assemblage automatisé :  
Elaboration automatique des séquences opératoires"  
Thèse d'Etat, BESANCON, 12 Novembre 1984.
- [BOU 86a] BOURJAUULT A., LHOTE F.  
"Approche méthodologique de la conception d'un système d'assemblage en  
production automatisée"  
Le Point en Productique, pp 43-54, Technique et Documentation,  
Lavoisier, PARIS 1986.
- [BOU 86b] BOURJAUULT A.  
"Modélisation d'un processus d'assemblage"  
Revue RAIRO APII, Vol.20, n°2, pp 183-198, 1986.
- [BOU 87a] BOURJAUULT A., HENRIOUD J.M.  
"Détermination des sous-assemblages d'un produit à partir des séquences  
temporelles d'assemblage"  
Revue RAIRO APII, Vol.21, n°2, pp 117-127, 1987.
- ✓ [BOU 87b] BOURJAUULT A., CHAPPE D., HENRIOUD J.M.  
"Elaboration automatique des gammes d'assemblage à l'aide de Réseaux de Pétri"  
Revue RAIRO APII, Vol.21, n°4, pp 323-342, 1987.
- [BOU 88] BOUREY J.P.  
"Structuration de la partie procédurale du système de commande de cellules de  
production flexibles dans l'industrie manufacturière"  
Thèse de Doctorat de l'Université, LILLE, 23 Mars 1988.
- [CAM 84] CAMPAGNE J.P., FAVREL J.  
"Elaboration automatique de gammes d'assemblage : proposition d'une  
méthodologie d'approche"  
Revue L'Onde Electrique, mars-avril 1984, pp 142-146.

- [CAM 85] CAMPAGNE J.P.,PEYRON J.,TEMANI M.  
"Structuration de bases de données techniques autour de nomenclatures et gammes mères en vue de l'élaboration automatique des gammes"  
Revue RAIRO APIL,Vol.19 ,n°4,pp 343-357,1985.
- [CAM 87] CAMPAGNE J.P.,CAPLAT C.  
"Elaboration automatique des gammes d'assemblage"  
2<sup>ième</sup> Conférence internationale INRIA  
Systèmes de production  
Proc.pp 537-550,6-10 Avril 1987,PARIS.
- / [CHA 88] CHAPPE D.,BOURJALUT A.  
"Utilisation des Réseaux de Pétri temporisés pour la conception et l'évaluation de systèmes d'assemblage automatisés"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 538-542,Vol.3  
PARIS,18-22 Juillet 1988.
- [FAV 84] FAVREL J.,CAMPAGNE J.P.  
"Génération automatique de gammes d'assemblage"  
Revue française de gestion industrielle  
Vol.3,n°1-2,1984,pp 39-47.
- / [LE 86] LE BEUX P.,FONTAINE D.  
"Un système d'acquisition de connaissances pour systèmes experts."  
Revue TSI Technique et science informatique,Vol. 5,n°1,pp 7-20,  
Janvier-Février 1986,Editions DUNOD /AFCET-BORDAS.
- [TEM 85] TEMANI M.  
"Contribution à la modélisation des processus d'assemblage : élaboration automatique des gammes"  
Thèse de 3<sup>ième</sup> cycle,Janvier 1985.

### CHAPITRE 3

- [ALJ 86] ALJ A.,FAURE R.  
"Guide de la Recherche Opérationnelle"  
Editions MASSON,PARIS 1986.
- [BEN 85] Groupe de prospective "Des usines pour demain"  
Coordinateur Guy BENCHIMOL  
"La conception des usines de demain"  
Editions HERMES,PARIS 1985.
- [BER 86] BERMAN O.,MAIMON O.  
"Cooperation among Flexible Manufacturing Systems"  
IEEE Journal of Robotics and Automation,Vol.RA-2,n°1,March 1986.
- [BOU 88] BOUREY J.P.  
"Structuration de la partie procédurale du système de commande de cellules de production flexibles dans l'industrie manufacturière"  
Thèse de Doctorat de l'Université, LILLE, 23 Mars 1988.
- [CAS 88] CASTELAIN E.,GENTINA J.C.  
"Petri-Nets and Artificial Intelligence in the context of simulation and modelling of manufacturing systems"  
IMACS SMS '88 International Symposium on System Modelling and Simulation  
Proc. pp 28-33,18-21 Septembre 1988,CETRARO(CS),ITALIE.
- [CAV 87] CAVAILLE J.B.,PROTH J.M.  
"Pratique de la simulation en production discontinue"  
Conférences SIPRODIS  
Editions EC2 Edition,Colloques & Conseil ,Collection NOVOTIQUE,1987.
- [DAL 86] DALLERY Y.,DUBOIS D.  
"L'analyse opérationnelle : une approche non stochastique des systèmes de files d'attente"  
Revue RAIRO APII,Vol.20,n°1,pp 43-86,1986.
- [ENG 81] ENGELBERGER J.,F.  
"Les Robots Industriels : Applications,Gestion et Pratique"  
Editions HERMES,PARIS 1981.
- [FAU 79] FAURE R.  
"Précis de Recherche Opérationnelle"  
Editions DUNOD/BORDAS,PARIS 1979.
- [GEL 81] GELENBE E.,LABETOUILLE J.,MARIE R.,METIVIER M.,PUJOLLE G.,STEWART W.  
"Réseaux de files d'attente : modélisation et traitement numérique"  
Monographie informatique de l'AFCEC,  
Editions Hommes et Techniques,PARIS 1981.
- [GEL 82] GELENBE E.,PUJOLLE G.  
"Introduction aux réseaux de files d'attente"  
Editions EYROLLES et CNET-ENST,PARIS 1982.
- [HEB 88] HEBRARD A.  
"Réalisation d'un logiciel permettant la visualisation d'un prégraphe"  
D.E.A. de Productique,L.A.I.I, I.D.N.,28 Juin 1988.

- [HEI 88a] HEIDEL S.  
 "Réalisation d'un outil d'aide à la visualisation graphique du prégraphe d'une cellule flexible"  
 Diplomarbeit, Fachhochschule KOLN, Abteilung Gummersbach, Fachbereich Informatik/ Institut Industriel du Nord LILLE, Juin 1988.
- [HEI 88b] HEINZERLING T.  
 "Adaptation of the Pregraph Petri Net defined in the CASPAIM Project in order to make the prototypage and the preliminary evaluation of the parallel architecture of the control in the field of C.I.M."  
 Masters Thesis, Technical University MUNICH / Institut Industriel du Nord LILLE, Novembre 1988.
- [JOH 85] JOHRI P.K., LIPPER E. H., SENGUPTA B.  
 "Modeling and analysis of a production line with finite buffers and machines subject to breakdown"  
 Revue RAIRO AII, Vol.19, n°5, pp 471-483, 1985.
- [KLE 76] KLEINROCK L.  
 "Queueing Systems,  
 Vol.1 : Theory  
 Vol.2 : Computer Applications"  
 Editions John WILEY, NEW YORK 1976.
- [KRA 73] KRAKOWSKI M.  
 "Conservation Methods in Queueing Theory"  
 Revue RAIRO Recherche Opérationnelle, V-1, pp 63-84, mars 1973.
- [LEM 77] LEMAIRE B.  
 "Méthode de conservation et blocage dans les files d'attente"  
 Revue RAIRO Recherche Opérationnelle, Vol.11, n°4, pp 363-377, nov.1977.
- [LEM 78] LEMAIRE B.  
 "Théorème de conservation des clients dans les files d'attente"  
 Revue RAIRO Recherche Opérationnelle, Vol.12, n°4, pp 395-399, nov.1978.
- [NAD 85] NADIF A., COSTANTINI M., MUTEL B.  
 "Mesures de ressemblance de gammes de fabrication"  
 Revue RAIRO AII, Vol.19, n°5, pp 455-470, 1985.
- [PRE 87] PREAT J.F.  
 "La représentation graphique des réseaux de Petri, un outil d'aide à la conception assistée des systèmes flexibles de production"  
 D.E.A. de Productique, L.A.I.I., I.D.N., 7 Septembre 1987.
- [PRI 84] PRITSKER A. A. B.  
 "Introduction to Simulation and SLAM II"  
 Editions John WILEY, NEW YORK 1984.
- [ROS 87] ROSEAUX  
 "Exercices et Problèmes résolus de Recherche Opérationnelle,  
 Tome 2 : Phénomènes aléatoires en Recherche Opérationnelle"  
 Editions MASSON, PARIS 1987.
- [TAI 87] TAILLEZ F.  
 "Une interface graphique entre une base de données de réseaux de Petri structurés et un simulateur de réseaux de Petri structurés adaptatifs colorés"  
 D.E.A. de Productique, L.A.I.I., I.D.N., 7 Septembre 1987.

## CONCLUSION GENERALE

- [MAR 88] MARTINEZ J.,MURO P.R.,SILVA M.,SMITH S.F.,VILLARROEL J.L.  
"Merging Artificial Intelligence Techniques and Petri-Nets for Real Time  
Scheduling and Control of Production Systems"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 528-531,Vol.3  
PARIS,18-22 Juillet 1988.
- [MER 88] MERCIER DES ROCHETTES R.,DESCOTES-GENON B.,LADET P.  
"Modelling and FMS Control Implementation"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 559-561,Vol.3  
PARIS,18-22 Juillet 1988.
- [VAL 88] VALETTE R.,CARDOSO J.,ATABAKHCHE H.,COURVOISIER M.,LEMAIRE T.  
"Petri-Nets and Production Rules for Decision Levels in FMS Control"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 522-524,Vol.3  
PARIS,18-22 Juillet 1988.

## ANNEXES

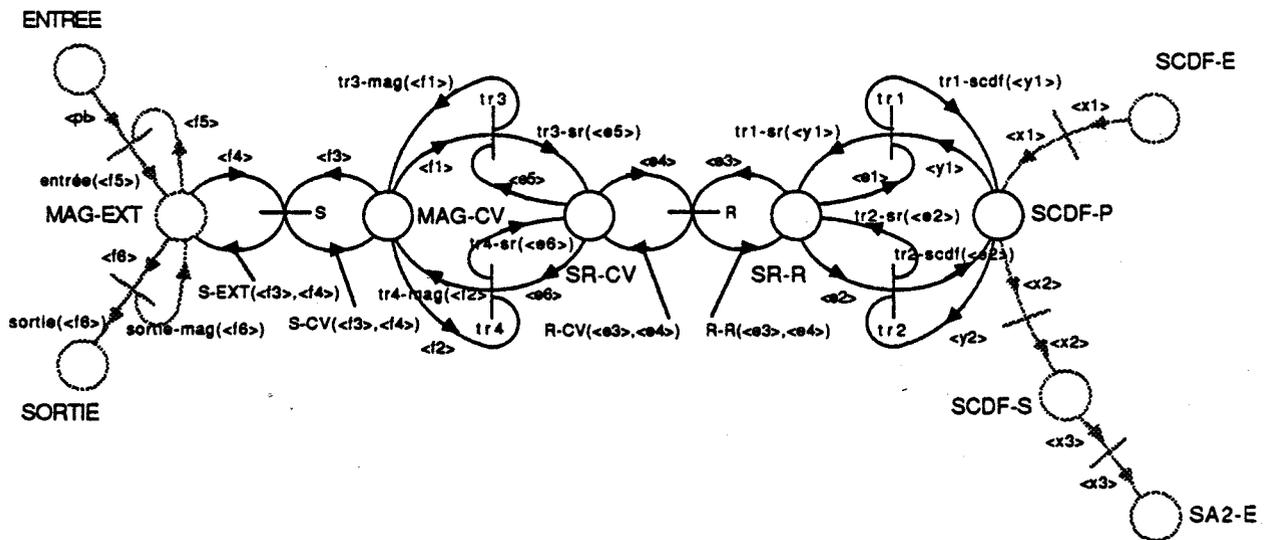
- [CAS 87] CASTELAIN E.  
"Modélisation et simulation interactive de cellules de production flexibles dans  
l'industrie manufacturière"  
Thèse de Doctorat de l'Université, LILLE, 26 Février 1987.
- [FRA 87] FRACHET J.P.  
"Concepts et outils généraux pour la manipulation interactive des graphismes  
technologiques utilisés en simulation et conception de systèmes automatisés"  
Conférences SIPRODIS sur la simulation en production discontinuée  
Actes du séminaire pp 94-104 ,PARIS, 2-3 Juin 1987  
Editions EC2 Edition,Colloques & Conseil,Collection NOVOTIQUE,1987.
- [FRA 88] FRACHET J.P.,POUGET J.P.,SAIDI A.  
"Object-oriented model for the specification of an integrated set of tools for  
aiding design and exploitation of automated production systems"  
IMACS 88,12<sup>ème</sup> Congrès Mondial IMACS,Proc. pp 546-551,Vol.3  
PARIS,18-22 Juillet 1988.
- [MOA 85] MOALLA M.  
"Réseaux de Petri interprétés et Grafcet"  
Revue TSI Technique et Science Informatique, Vol.4, n°1, pp 17-30  
Janvier-Février 1985, Editions DUNOD/AFCEC-BORDAS.
- [RHE 88] RHEMES G.  
"Automaticien,voici votre atelier de Génie logiciel"  
Revue Electronique Industrielle,n°142,15 Avril 1988,pp 49-53.
- [VER 88] VERWEE F.  
"Maquette d'illustration du traitement de la cohérence dans un dossier  
d'affaire"  
D.E.A. de Productique,L.A.I.I, I.D.N.,Septembre 1988.

**ANNEXES**

## **ANNEXE 1**

**Informations complémentaires relatives au choix du modèle Prégraphe  
du paragraphe II.4, Chapitre III Partie II**

Reprenons l'exemple du Chapitre III Partie II. A la place du prégraphe présenté au paragraphe II.4, il était a priori possible d'imaginer le prégraphe ci-dessous :



On prend le parti ici de représenter sur le modèle RdP de la Partie Commande les étagères des stockeurs rotatifs par des marques colorées. Une marque "SR-pl-v" désignera par exemple une étagère provenant du stockeur rotatif SR, et dont l'emplacement gauche contient un plateau pl et dont l'emplacement droit est vide (symbole v). Une marque "MAG-pl-pl" fait par exemple référence à une étagère du stockeur MAGASIN qui porte 2 plateaux pl.

On a alors :

$$\text{dom}(\langle e1 \rangle) = \{ \text{SR-pl-v}, \text{SR-v-pl}, \text{SR-pl-pl} \}$$

$$\text{dom}(\langle e2 \rangle) = \{ \text{SR-pl-v}, \text{SR-v-pl}, \text{SR-pl-pl} \}$$

$$\text{dom}(\langle e3 \rangle) = \{ \text{SR-v-v}, \text{SR-pl-v}, \text{SR-v-pl}, \text{SR-pl-pl} \}$$

$$\text{dom}(\langle e4 \rangle) = \{ \text{SR-v-v}, \text{SR-pl-v}, \text{SR-v-pl}, \text{SR-pl-pl} \}$$

$$\text{dom}(\langle e5 \rangle) = \{ \text{SR-v-v}, \text{SR-pl-v}, \text{SR-v-pl} \}$$

$$\text{dom}(\langle e6 \rangle) = \{ \text{SR-pl-v}, \text{SR-v-pl}, \text{SR-pl-pl} \}$$

$$\text{dom}(\langle f1 \rangle) = \{ \text{MAG-pl-v}, \text{MAG-v-pl}, \text{MAG-pl-pl} \}$$

$$\text{dom}(\langle f2 \rangle) = \{ \text{MAG-v-v}, \text{MAG-pl-v}, \text{MAG-v-pl} \}$$

$$\text{dom}(\langle f3 \rangle) = \{ \text{MAG-v-v}, \text{MAG-pl-v}, \text{MAG-v-pl}, \text{MAG-pl-pl} \}$$

$$\text{dom}(\langle f4 \rangle) = \{ \text{MAG-v-v}, \text{MAG-pl-v}, \text{MAG-v-pl}, \text{MAG-pl-pl} \}$$

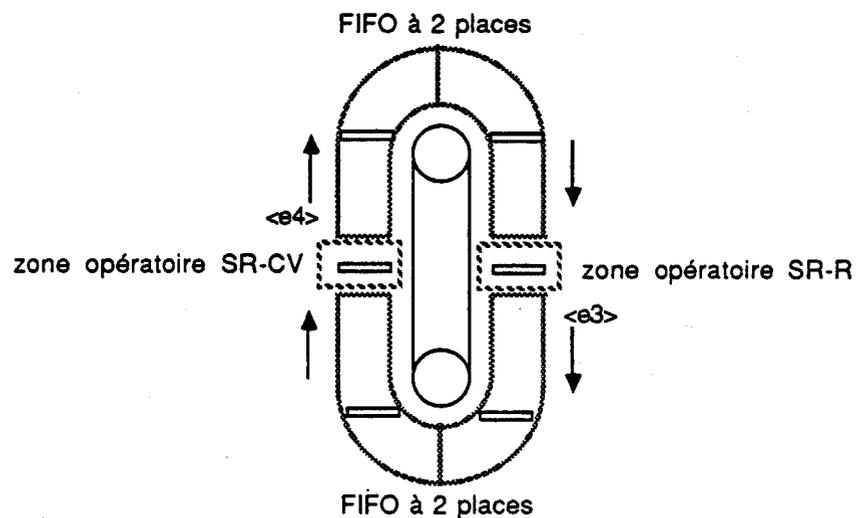
tR1 :	<e1> SR-pl-v SR-v-pl SR-pl-pl	tR1(<e1>) SR-pl-v SR-v-pl SR-pl-pl		
tR2-SR :	<e2> SR-pl-v SR-v-pl SR-pl-pl	tR2-SR(<e2>) SR-pl-v SR-v-pl SR-pl-pl	tR2-SCDF : <e2> SR-pl-v SR-v-pl SR-pl-pl	tR2-SCDF(<e2>) bi ou ci bi ou ci bi ou ci
tR3-SR :	<e5> SR-v-v SR-pl-v SR-v-pl	tR3-SR(<e5>) SR-pl-v ou SR-v-pl SR-pl-pl SR-pl-pl	tR3-MAG : <f1> MAG-pl-v MAG-v-pl MAG-pl-pl	tR3-MAG(<f1>) MAG-v-v MAG-v-v MAG-v-pl ou MAG-pl-v
tR4-SR :	<e6> SR-pl-v SR-v-pl SR-pl-pl	tR4-SR(<e6>) SR-v-v SR-v-v SR-v-pl ou SR-pl-v	tR4-MAG : <f2> MAG-v-v MAG-pl-v MAG-v-pl	tR4-MAG(<f2>) MAG-pl-v ou MAG-v-pl MAG-pl-pl MAG-pl-pl

R-CV et R-SR : Ces fonctions doivent générer des marques identifiant des étagères à l'issue d'une rotation R du stockeur rotatif.

Ici se pose un problème qui mérite discussion.

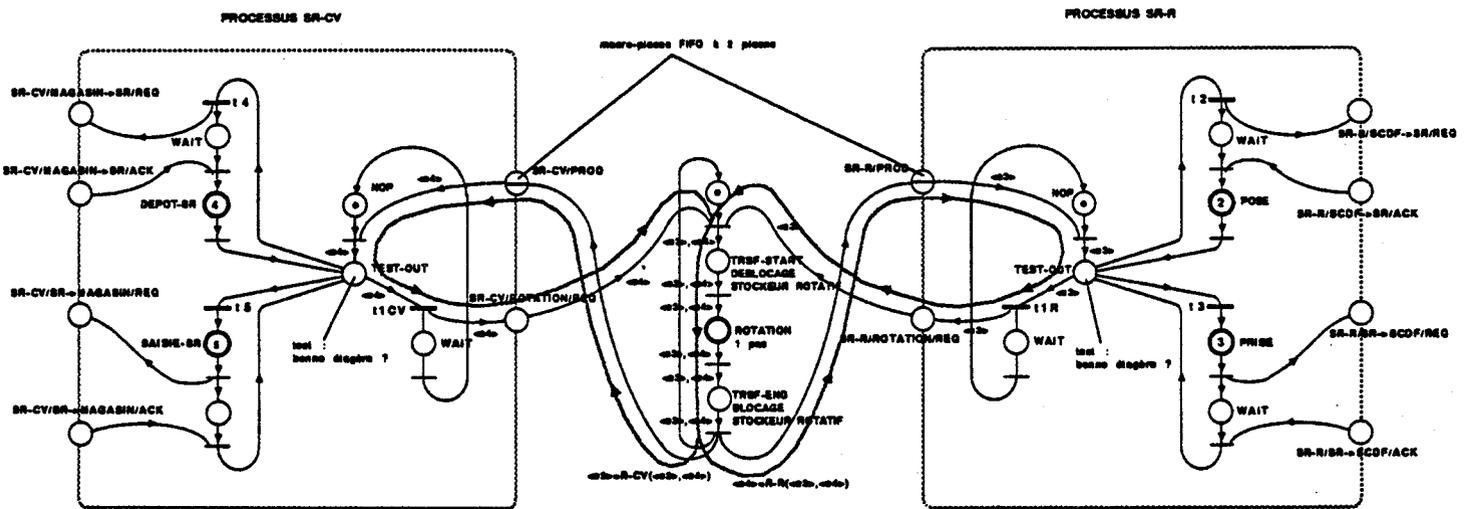
a) Supposons que  $R-CV(\langle e3 \rangle, \langle e4 \rangle) = \langle e3 \rangle$  et  $R-R(\langle e3 \rangle, \langle e4 \rangle) = \langle e4 \rangle$ .

Cela signifie que la rotation du stockeur rotatif fait rentrer l'étagère <e3> qui était sur la zone opératoire SR-R dans la file d'attente à 2 places de la zone opératoire SR-CV, et l'étagère <e4> qui était sur la zone opératoire SR-CV dans la file d'attente à 2 places de la zone opératoire SR-R.



La rotation R n'est alors que d'un pas et la commande doit être réitérée jusqu'à ce que les bonnes étagères soient identifiées sur les zones opératoires.

Ceci correspond, pour chaque pas de rotation, à un parcours très lourd des marques dans le graphe RdP de la Partie Commande :



Ce choix de représentation n'est donc pas intéressant.

b) Supposons maintenant que les valeurs retournées par les fonctions R-CV et R-R soient déterminées par le Niveau Hiérarchique sur une image plus fine du stockeur rotatif, à partir du choix de la rotation devant être effectuée.

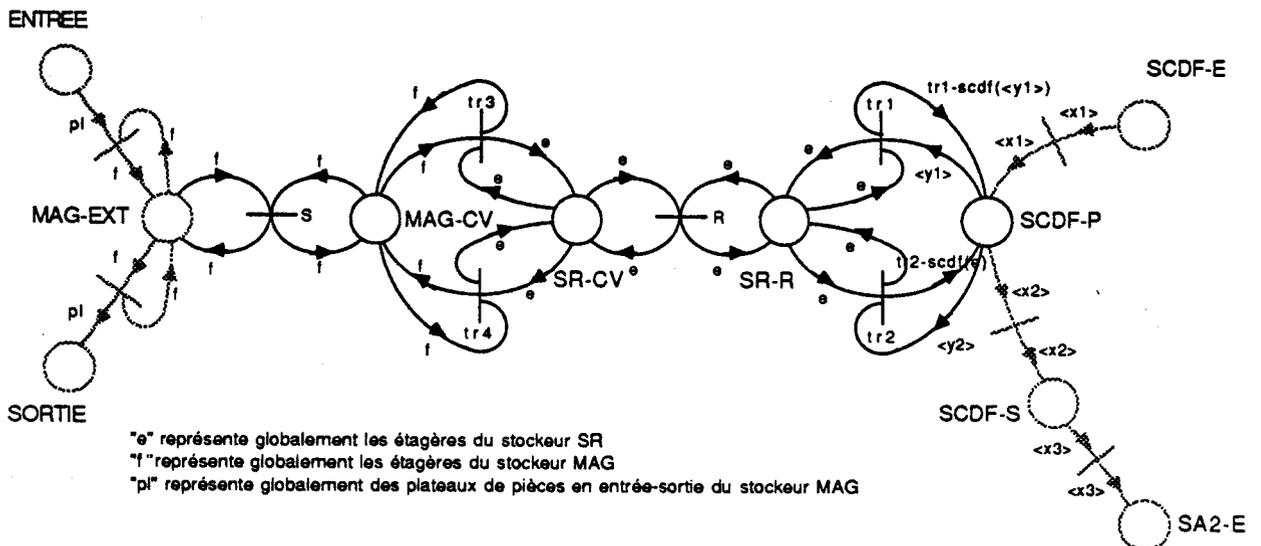
La rotation peut alors être d'un nombre quelconque de pas.

Les FIFOs des zones opératoires SR-R et SR-CV ne sont plus gérées au niveau du modèle de la Partie Commande. Les places SR-CV/PROD et SR-R/PROD deviennent des places simples.

De la même façon que pour les fonctions R-CV et R-R, le Niveau Hiérarchique impose certains des résultats retournés par les autres fonctions.

On s'aperçoit dès lors que la coloration ne sert plus vraiment sur cette portion du modèle de la Partie Commande puisque son rôle est repris de façon plus pointue par l'image plus fine du procédé qui est utilisée par le Niveau Hiérarchique.

Ceci permet donc de justifier la construction de l'exemple du Chapitre III Partie II. Il est toutefois possible de laisser une coloration résiduelle afin de faciliter la lecture et la construction du modèle prégraphe, comme l'illustre la figure ci-dessous.



"e" représente globalement les étagères du stockeur SR  
 "f" représente globalement les étagères du stockeur MAG  
 "pl" représente globalement des plateaux de pièces en entrée-sortie du stockeur MAG

Le modèle de la Partie Commande se limite alors au simple rôle de coordination et de séquençement des commandes applicables au procédé. Le paramétrage de ces commandes, la levée des indéterminismes sont réalisés par un véritable système à base de connaissances.

La question qui se pose est alors d'évaluer l'importance de la partie coordination et séquençement pour savoir s'il ne serait pas possible de l'intégrer au système à base de connaissances qui assure pour le moment les fonctions de décision et de paramétrage. On arrive ainsi à la notion de pilotage direct par système à base de connaissances (= système expert). La réponse n'est pas facile et c'est principalement l'expérience et l'évolution technique qui permettront de justifier ce genre de choix.

## **ANNEXE 2**

**Exemples de fonctions du Niveau Hiérarchique  
pour l'exemple du Chapitre III Partie II**

## COMMANDES DE LA PARTIE COMMANDE PARAMETREES PAR LE NIVEAU HIERARCHIQUE

### Place ① ROTATION

L'amplitude (ou l'angle) <a> de la rotation ayant été déterminée par le Niveau Hiérarchique, la commande ROTATION d'amplitude <a> est envoyée par la Partie Commande au moteur du stockeur rotatif SR.

### Place ② POSE

Les informations Etagère <n> et Position <m> étant construites (n = numéro d'étagère + symbole D (droite) ou G (gauche), m = code de la position dans le plateau qui se trouve à l'emplacement <n>), la commande POSER la pièce sur la position libre <m> du plateau de l'étagère <n> destination est envoyée au bras manipulateur R.

### Place ③ PRISE

Les informations Etagère <n> et Position <m> étant construites, la commande PRENDRE la pièce qui se trouve sur la position <m> du plateau de l'étagère <n> origine est envoyée au bras manipulateur R.

### Places ④ DEPOT-SR et ⑥ SAISIE-M

Les informations Etagère <n> origine et Etagère <n> destination étant construites, les commandes SAISIE-M du plateau qui se trouve sur l'emplacement <n> origine du stockeur MAGASIN et DEPOT-SR sur l'emplacement <n> destination du stockeur SR peuvent être envoyées au convoyeur CV quand les places seront marquées.

### Places ⑤ SAISIE-SR et ⑦ DEPOT-M

De même, les informations Etagère <n> origine et Etagère <n> destination étant construites, les commandes SAISIE-R du plateau qui se trouve sur l'emplacement <n> origine du stockeur SR et DEPOT-M sur l'emplacement <n> destination du stockeur MAGASIN peuvent être envoyées au convoyeur CV quand les places seront marquées.

### Règle 1 TRANSFERT SCDF-P->SR-R : REQUETE

Si -il y a une palette portant une pièce assemblée sur la zone opératoire SCDF-P (marque pbf ou pcf  $\in$  dom(<y>) sur la place SCDF-P/TEST-OUT)

-il y a au moins un plateau dans le stockeur rotatif pouvant accueillir cette pièce (on établit la liste des plateaux différents des plateaux rebuts qui ont au moins un emplacement vide et qui contiennent des objets de la famille de la pièce : bi et bf pour une pièce bf, ci et cf pour une pièce cf)

-la zone opératoire ZO-R du stockeur SR n'est pas l'objet d'une opération en cours (on a une marque sur la place SR-R/TEST-OUT)

alors -une requête est effectuée pour réaliser un transfert SCDF-P->SR-R (on place une marque "SCDF-P->SR-R" dans la liste REQUETES)

### Règle 2 TRANSFERT SR-R->SCDF-P : REQUETE

Si -il y a une palette vide sur la zone opératoire SCDF-P (marque p sur la place SCDF-P/TEST-OUT)

-il y a au moins un plateau dans le stockeur rotatif pouvant fournir une pièce non assemblée bi ou ci (on établit la liste des plateaux contenant au moins une pièce bi ou ci. On effectue une distinction entre les plateaux rebuts et les autres suivant la provenance de la palette vide)

-la zone opératoire ZO-R du stockeur SR n'est pas l'objet d'une opération en cours (on a une marque sur la place SR-R/TEST-OUT)

alors -une requête est effectuée pour réaliser un transfert SR-R->SCDF-P (on place une marque "SR-R->SCDF-P" dans la liste REQUETES)

### Règle 3 TRANSFERT MAGASIN-CV->SR-CV : REQUETE

Si -les zones opératoires ZO-MAGASIN-CV et ZO-SR-CV ne sont pas l'objet d'opérations de transfert en cours (marques sur les places MAGASIN-CV/TEST-OUT et SR-CV/TEST-OUT)

-il y a au moins un plateau sur le stockeur MAGASIN plein de pièces identiques bi ou ci

-il y a au moins un emplacement vide sur le stockeur SR

alors -une requête est effectuée pour réaliser un transfert MAGASIN-CV->SR-CV (on place une marque "MAGASIN-CV->SR-CV" dans la liste REQUETES)

### Règle 4 TRANSFERT SR-CV->MAGASIN-CV : REQUETE

Si -les zones opératoires ZO-SR-CV et ZO-MAGASIN-CV ne sont pas l'objet d'opérations de transfert en cours (marques sur les places SR-CV/TEST-OUT et MAGASIN-CV/TEST-OUT)

-il y a au moins un plateau sur le stockeur SR vide ou plein de pièces identiques bf ou cf

-il y a au moins un emplacement vide sur le stockeur MAGASIN

alors -une requête est effectuée pour réaliser un transfert SR-CV->MAGASIN-CV (on place une marque "SR-CV->MAGASIN-CV" dans la liste REQUETES)

## Règle 5 PRIORITES ENTRE LES REQUETES DE TRANSFERT : CHOIX

Si -les 4 règles de requête de transfert ont été passées en revue

-la priorité n'a pas encore été réglée

-les priorités suivantes ont été établies pour le cycle d'inférence :

"SR-CV->MAGASIN-CV"	priorité x
"MAGASIN-CV->SR-CV"	priorité y
"SCDF-P->SR-R"	
"SR-R->SCDF-P"	priorité z

avec (x,y,z) appartenant à l'ensemble des permutations de {1,2,3}.

alors -on place dans la variable "TRANSFERTS-AUTORISES" la marque du transfert prioritaire.

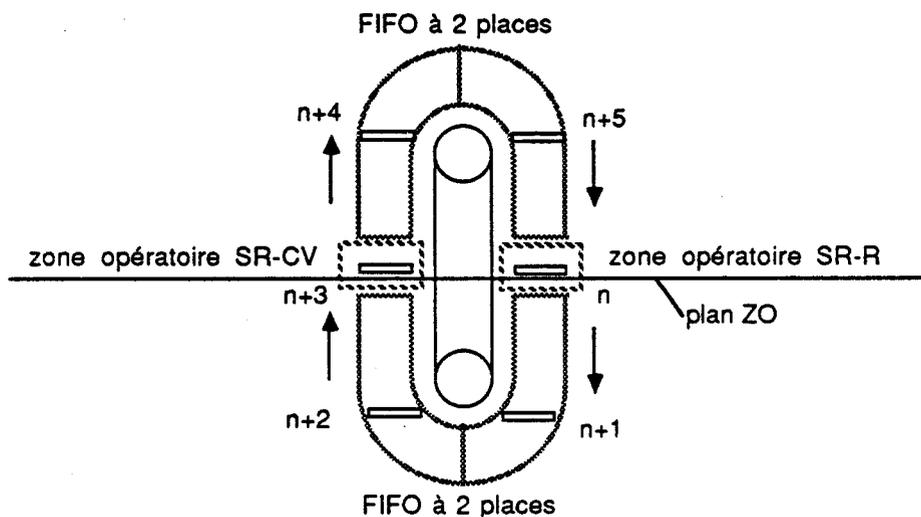
Au cas où un des autres transfert serait compatible(\*) avec un transfert prioritaire, on place également sa marque dans la variable "TRANSFERTS-AUTORISES".

-la priorité est réglée.

(\*) Remarque : Le cas de compatibilité peut apparaître entre un transfert utilisant la zone opératoire ZO-R du stockeur rotatif SR et un transfert utilisant la zone opératoire ZO-CV.

Si une coïncidence fait que les étagères sollicitées par chacun des deux transferts peuvent se trouver simultanément dans les zones opératoires respectives (ZO-R pour l'une et ZO-CV pour l'autre), les deux transferts peuvent être exécutés en même temps.

Cette coïncidence se traduit par le fait que la différence des numéros des étagères pouvant être appelées par les transferts est égale à une valeur entière particulière qui leur permet d'être simultanément dans le plan ZO. Par exemple ici  $n_2 - n_1 = 3$ .



## DETERMINATION DES PARAMETRES DES TRANSFERTS AUTORISES

### Règle 6 TRANSFERT SCDF-P->SR-R SEUL : CHOIX DE L'EMPLACEMENT LIBRE POUR LA POSE D'UNE PIECE SUR UN PLATEAU

Si -la priorité a été réglée entre les requêtes de transfert

-le transfert SCDF-P->SR-R figure seul dans la liste "TRANSFERTS-AUTORISES"

alors -choisir le plateau dans la liste des plateaux possibles et l'emplacement dans le plateau (la règle de choix construit l'information Etagère <n> où n = numéro d'étagère + symbole D (droite) ou G (gauche) et l'information Position <m> où m est le code de la position, ex : coordonnées cartésiennes relatives)

-le choix des paramètres est réglé.

### Règle 7 TRANSFERT SR-R->SCDF-P SEUL : CHOIX DE L'EMPLACEMENT SUR UN PLATEAU OU EFFECTUER LA PRISE D'UNE PIECE

Si -la priorité a été réglée entre les requêtes de transfert

-le transfert SR-R->SCDF-P figure seul dans la liste "TRANSFERTS-AUTORISES"

alors -choisir le plateau dans la liste des plateaux possibles et l'emplacement de la pièce dans le plateau (la règle de choix construit l'information Etagère <n> où n = numéro d'étagère + symbole D ou G et l'information Position <m> où m est le code de la position. La fonction "pièce-prise-sur-plateau" du graphe de la Partie Commande fournira une marque bi ou ci en fonction du choix effectué)

-le choix des paramètres est réglé.

Règle 8 TRANSFERT MAGASIN-CV->SR-CV SEUL : CHOIX DU PLATEAU ET DE SA DESTINATION

Si -la priorité a été réglée entre les requêtes de transfert

-le transfert MAGASIN-CV->SR-CV figure seul dans la liste "TRANSFERTS-AUTORISES"

alors -choisir le plateau dans la liste des plateaux possibles du stockeur MAGASIN et l'emplacement libre destination du stockeur SR (la règle de choix construit les informations Etagère <n> origine et Etagère <n> destination)

-le choix des paramètres est réglé.

Règle 9 TRANSFERT SR-CV->MAGASIN-CV SEUL : CHOIX DU PLATEAU ET DE SA DESTINATION

Si -la priorité a été réglée entre les requêtes de transfert

-le transfert SR-CV->MAGASIN-CV figure seul dans la liste "TRANSFERTS-AUTORISES"

alors -choisir le plateau dans la liste des plateaux possibles du stockeur SR et l'emplacement libre destination du stockeur MAGASIN (la règle de choix construit les informations Etagère <n> origine et Etagère <n> destination)

-le choix des paramètres est réglé.

### Règle10 TRANSFERTS COMPATIBLES : CHOIX DES PARAMETRES

Si -la priorité a été réglée entre les requêtes de transfert

-la liste "TRANSFERTS-AUTORISES" comprend 2 transferts T1 et T2 qui peuvent être compatibles

alors -choisir pour les transferts T1 et T2 le même type de paramètres que dans les règles 6, 7, 8 et 9 mais parmi les possibilités qui respectent la contrainte de compatibilité entre T1 et T2. Cette contrainte impose que les étagères du stockeur rotatif relatives à T1 et T2 puissent se trouver simultanément sur les zones opératoires respectives SR-R et SR-CV ( $n2 - n1 = 3$ )

-le choix des paramètres est réglé.

### Règle11 ROTATION DU STOCKEUR ROTATIF

Si -le choix des paramètres est réglé

-les étagères qui ont été choisies ne sont pas sur les zones opératoires

alors -déterminer la rotation <a> (=nombre de pas) à imposer au stockeur rotatif pour positionner les étagères choisies dans les bonnes zones opératoires

-débloquer les transitions t1R et t1CV

### Règle12 TRANSFERTS EFFECTIFS : DEBLOCAGE DES TRANSITIONS

Si -le choix des paramètres est réglé

-les étagères qui ont été choisies sont sur les zones opératoires

alors-débloquer les transitions relatives aux transferts de la liste des "TRANSFERTS-AUTORISES", soit :

- tD et t2 pour SCDF-P->SR-R
- tP et t3 pour SR-R->SCDF-P
- t4 et t6 pour MAGASIN-CV->SR-CV
- t5 et t7 pour SR-CV->MAGASIN-CV

-réinitialiser les variables du cycle d'inférence (REQUETES, TRANSFERTS-AUTORISES, priorité-réglée, choix-des-paramètres-réglé,...)

**N.B :**

1) Le reblocage des transitions peut être pris en charge par diverses techniques qui dépendent de l'installation des règles pour la simulation des modèles ou pour l'implantation réelle. Le simulateur construit par Emmanuel CASTELAIN ([CAS 87]) permet de représenter le Niveau Hiérarchique sous forme de règles de surveillance et/ou sous forme de règles de requête qui correspondent respectivement au cycle par activité et au cycle par événement du pas de simulation. Le lecteur trouvera dans la thèse mentionnée les précisions nécessaires. Par ailleurs, le lecteur trouvera dans la thèse à venir (janvier 1989, LILLE) d'Etienne CRAYE "De la modélisation à l'implantation automatisée de la commande hiérarchisée de cellules de production flexibles dans l'industrie manufacturière" les informations utiles concernant l'implantation des modèles.

2) Les règles qui élaborent les priorités x, y, z entre les transferts, ou qui président au choix effectif des paramètres des transferts parmi toutes les possibilités, paramètrent elles-mêmes les règles précédentes. Elles font partie d'un niveau plus élevé du Niveau Hiérarchique qui se rapproche du Niveau de Planification/Ordonnancement. Nous ne les développerons pas. Ceci nous entraînerait trop loin ici.

## **ANNEXE 3**

**Discussion d'un petit exemple industriel**

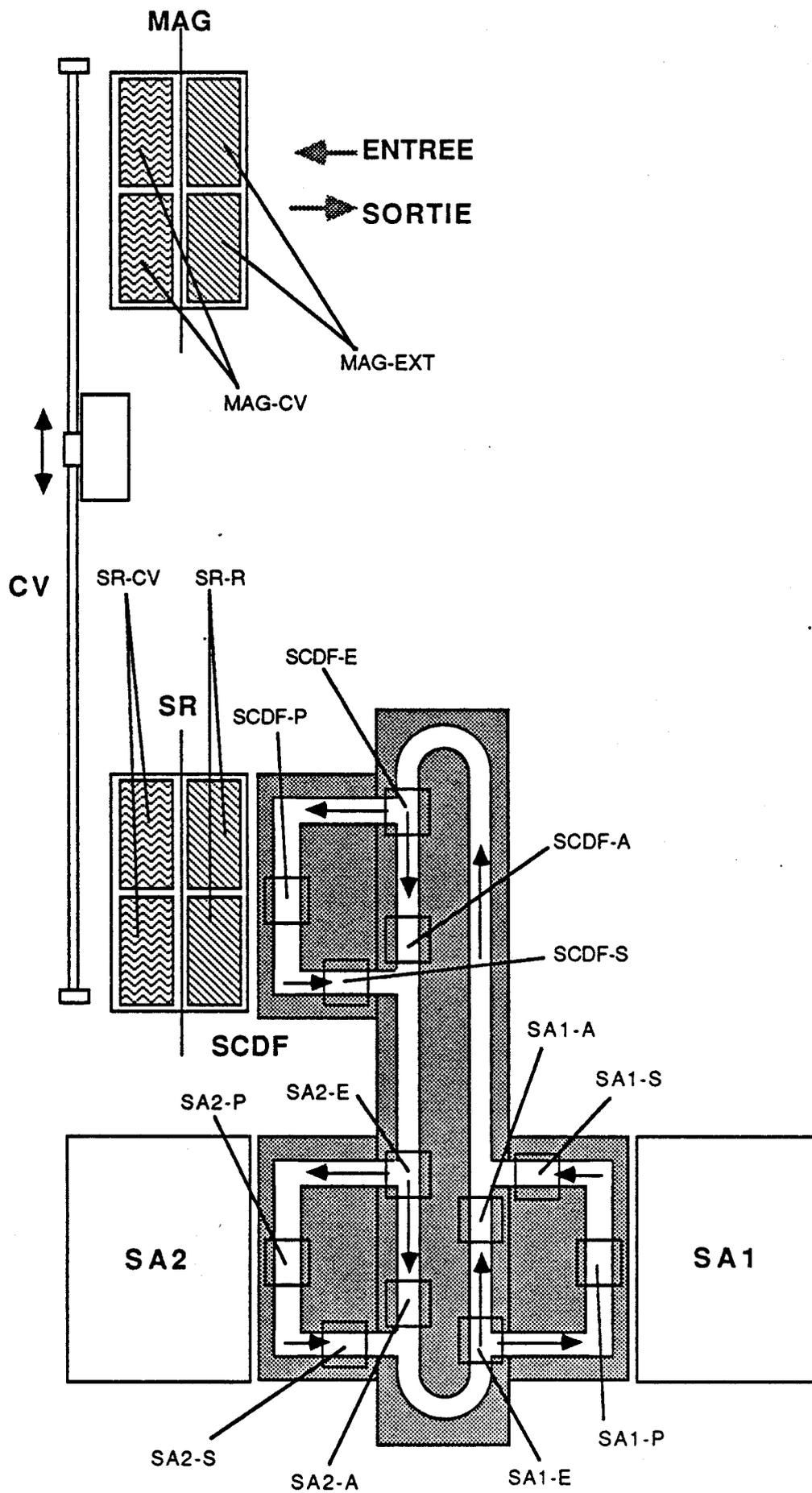


Figure Annexe 3.1

Considérons la cellule flexible présentée sur la figure ci-contre :

## ARCHITECTURE

Cette cellule, destinée à l'assemblage et au contrôle des pièces assemblées, a une structure modulaire caractéristique de ce type d'applications.

-La partie représentée en gris est constituée d'un ensemble de 3 modules de dérivation desservis par une ligne de convoyage centrale L assurant le transport des pièces par l'intermédiaire de palettes p.

La description de la structure des modules de dérivation a déjà été faite au Chapitre II Partie I exemple 6.

SA1 et SA2 constituent les 2 stations d'assemblage. Les assemblages sont effectués par ajout de composants extérieurs sur des pièces brutes convoyées chacune par une palette. Ces opérations sont assurées par des robots d'assemblage sur les zones opératoires SA1-P et SA2-P.

La station SA1 comporte également un système de vision qui permet de contrôler la bonne réalisation de l'assemblage. Les pièces en défaut sont placées dans un rebut.

La palettisation des pièces brutes et la dépalettisation des pièces assemblées et contrôlées sont assurées au niveau de la station de chargement-déchargement flexible SCDF. Les pièces brutes proviennent de plateaux placés sur les étagères d'un stockeur rotatif SR (2 plateaux par étagère, 10 étagères). Les pièces assemblées sont redéposées dans les emplacements vides des plateaux. Les transferts entre les plateaux et le poste de palettisation-dépalettisation SCDF-P sont réalisés par un manipulateur programmable R (2 axes).

-Le stockeur SR sert de tampon de stockage de plateaux au niveau de la station SCDF. Un stockeur MAG identique au stockeur SR assure la même fonction vis à vis d'une zone qu'on ne détaillera pas (flèches ENTREE-SORTIE).

Un convoyeur constitué d'un monorail motorisé véhicule les plateaux entre le stockeur MAG et le stockeur SR à l'aide d'une balancelle passive.

Le transfert des plateaux entre les stockeurs rotatifs et la balancelle est réalisé par un manipulateur pneumatique à cycle fixe.

## GAMMES

Les gammes devant être réalisées par cette cellule sont les suivantes :

-Contact électrique auxiliaire :

Cette assemblage est réalisé à partir d'un boîtier b posé sur une palette p.

Des constituants extérieurs sont alors rapportés au boîtier dans l'ordre suivant :

- > écrou e
- > plage fixe f
- > vis étrier v

L'assemblage du contact auxiliaire est réalisé sur la station SA1 au niveau du poste SA1-P.

-Cartes électroniques :

2 types de cartes électroniques sont assemblées à partir d'un circuit imprimé commun c :

1) cartes 1 :

Les composants suivants sont insérés dans un ordre quelconque sur le circuit c posé sur une palette p :

- > un ensemble r de 2 relais miniatures
- > un ensemble t de 2 transistors

2) cartes 2 :

Les composants suivants sont insérés dans un ordre quelconque sur le circuit c posé sur une palette p :

- > un ensemble r de 2 relais miniatures
- > un ensemble t de 2 transistors
- > une résistance surélevée s

Notation :

une carte **c** palettisée sera représentée par la couleur **pc**.

une carte **c** palettisée sur laquelle **r** et **t** ont déjà été insérés sera représentée par la couleur **pcrt**.

On en déduit le sens des autres notations.

L'assemblage des cartes est réalisé essentiellement sur la station **SA2** au niveau du poste **SA2-P**. L'ensemble **r** des 2 résistances peut également être inséré sur la station **SA1** au niveau du poste **SA1-P**.

Un système de vision assure au niveau du poste **SA1-P** l'inspection de toutes les pièces palettisées dont l'assemblage est terminé, c'est-à-dire **pcrt**, **pcrts** et **pbefv**. Les pièces contrôlées et correctes sont notées **pcrt+**, **pcrts+** et **pbefv+**. Les pièces palettisées qui présentent des défauts (**pbefv-**, **pcrt-**, **pcrts-**) sont dépalettisées au niveau du poste **SA1-P** et placées dans un rebut. La palette correspondante recevra au niveau du poste **SCDF-P** une pièce brute identique à la pièce de base de l'ensemble rebuté. Cette pièce remplaçante provient d'un plateau spécial destiné au remplacement des rebuts.

Chaque plateau contient un seul type de pièces.

Il est possible de proposer pour cette cellule plusieurs Prégraphes de base légèrement différents. Les exemples traités au cours du mémoire permettent d'obtenir sans difficulté la structure fondamentale. On ne reviendra donc pas sur la construction. Etudions par contre quelques variantes liées à l'image du procédé portée par le Prégraphe.

Rappelons que le Système de commande, constitué d'un niveau de coordination (Partie Commande) et d'un niveau de décision (Niveau Hiérarchique), a besoin d'une représentation du procédé pour élaborer sa commande. L'image du procédé peut être supportée de diverses façons en fonction des possibilités du dispositif technologique d'implantation des modèles et des limites des modèles eux-mêmes (Notons qu'afin de faciliter l'implantation, les possibilités des modèles sont en général choisies en fonction des limites des techniques d'implantation).

La figure Annexe 3.3 n°1 représente le Prégraphe sur lequel on a placé la coloration maximale. Un exemple de coloration finie est donnée à la page qui suit la figure. On constate que les fonctions **S-EXT**, **S-CV**, **SR-CV**, **SR-R**, **tr2-scdf**, entrée et sortie-mag génèrent des marques en fonction d'informations portées par une représentation plus fine du procédé qui ne peut plus être portée sur la Partie Commande et qui, en outre, est plutôt utilisée par le Niveau Décisionnel. Nous avons déjà discuté ces problèmes dans l'Annexe 1 et au Chapitre III Partie II. On est ainsi amené à s'intéresser plutôt au Prégraphe de la figure 2 sur lequel une partie de la coloration a été supprimée.

Supposons que la coloration de la figure Annexe 3.3 n°1 soit maintenant infinie. Les variables peuvent par exemple prendre leurs valeurs dans l'ensemble des entiers naturels. Il est ainsi possible d'identifier par exemple par un numéro les objets correspondant aux marques circulant dans le modèle RdP de la Partie Commande. Une table de correspondance peut être établie à partir de cet adressage. Il est ainsi possible d'accéder à des fiches regroupant les connaissances utiles sur chacun des objets repérés par un numéro. Le gros inconvénient de cette méthode est, compte tenu des dispositifs technologiques industriels existants, de nécessiter beaucoup de communications externes qui ralentissent le fonctionnement. Ce problème disparaîtra avec l'accroissement des possibilités des automates qui pourront intégrer davantage de connaissances.

La figure Annexe 3.3 n°2 représente le Prégraphe sur lequel on a supprimé la partie de la coloration relative aux stockeurs rotatifs. L'image du procédé pour cette partie est complètement portée à l'extérieur du Prégraphe et donc à l'extérieur de la Partie Commande. Pour la partie concernant les modules d'assemblage et de contrôle organisés autour de la ligne centrale de convoyage, la coloration a été maintenue. Elle permet de supporter facilement une image du procédé qui peut être suffisante ici. Il est possible de rapporter à la coloration d'autres

informations pour former un n-uplet du type ( $\langle x \rangle, n_1, n_2, \dots$ ) où  $\langle x \rangle$  représente la coloration initiale et où  $n_1, n_2, \dots$  représentent des informations utilisées plus localement ou moins fréquemment.  $n_1, n_2, \dots$  pourraient à la limite représenter des adresses de renvoi à des tables plus complexes.

Les possibilités de représentation du procédé au sein du Système de commande sont véritablement multiples. Tout dépend des possibilités d'implantation (voir figure Annexe 3.2) et des limites imposées par les outils de validation des modèles RdP (ex:simulation).

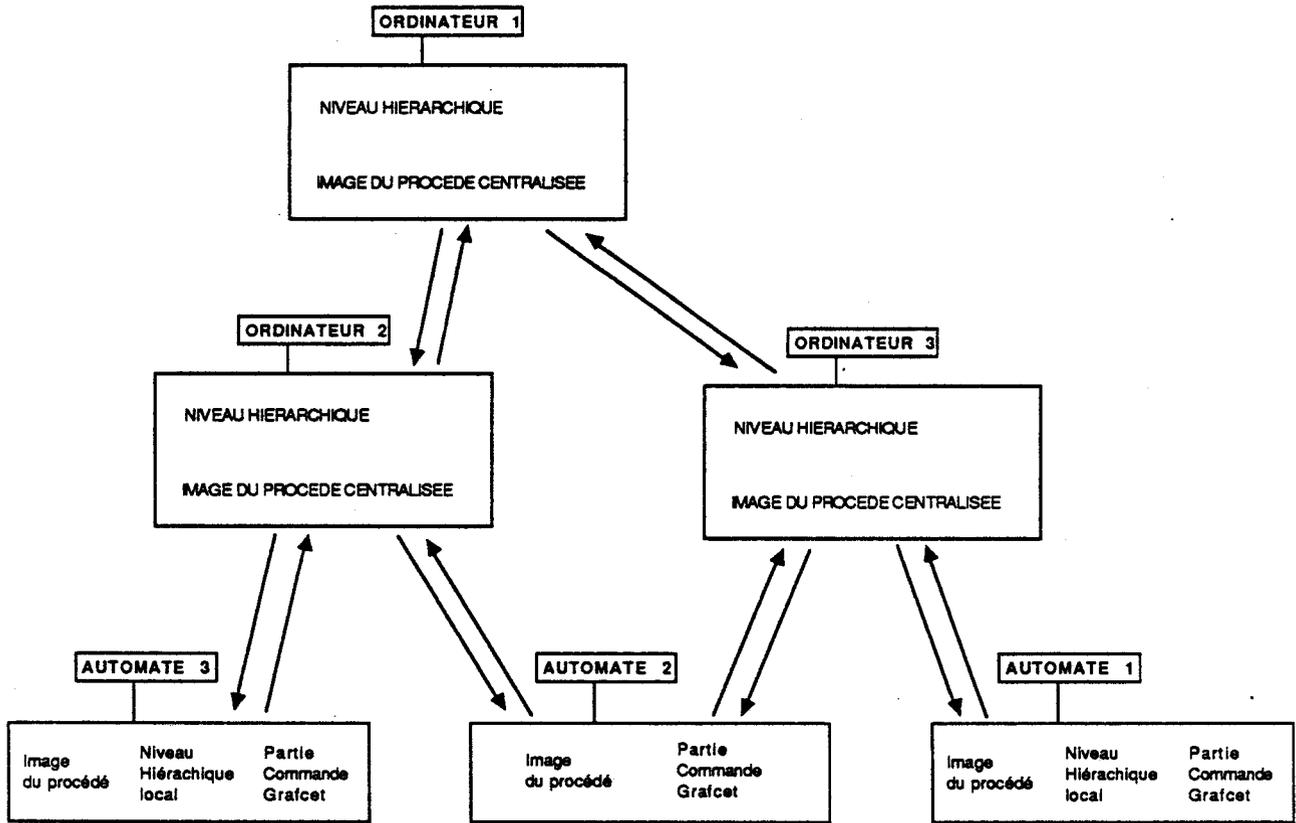


Figure Annexe 3.2

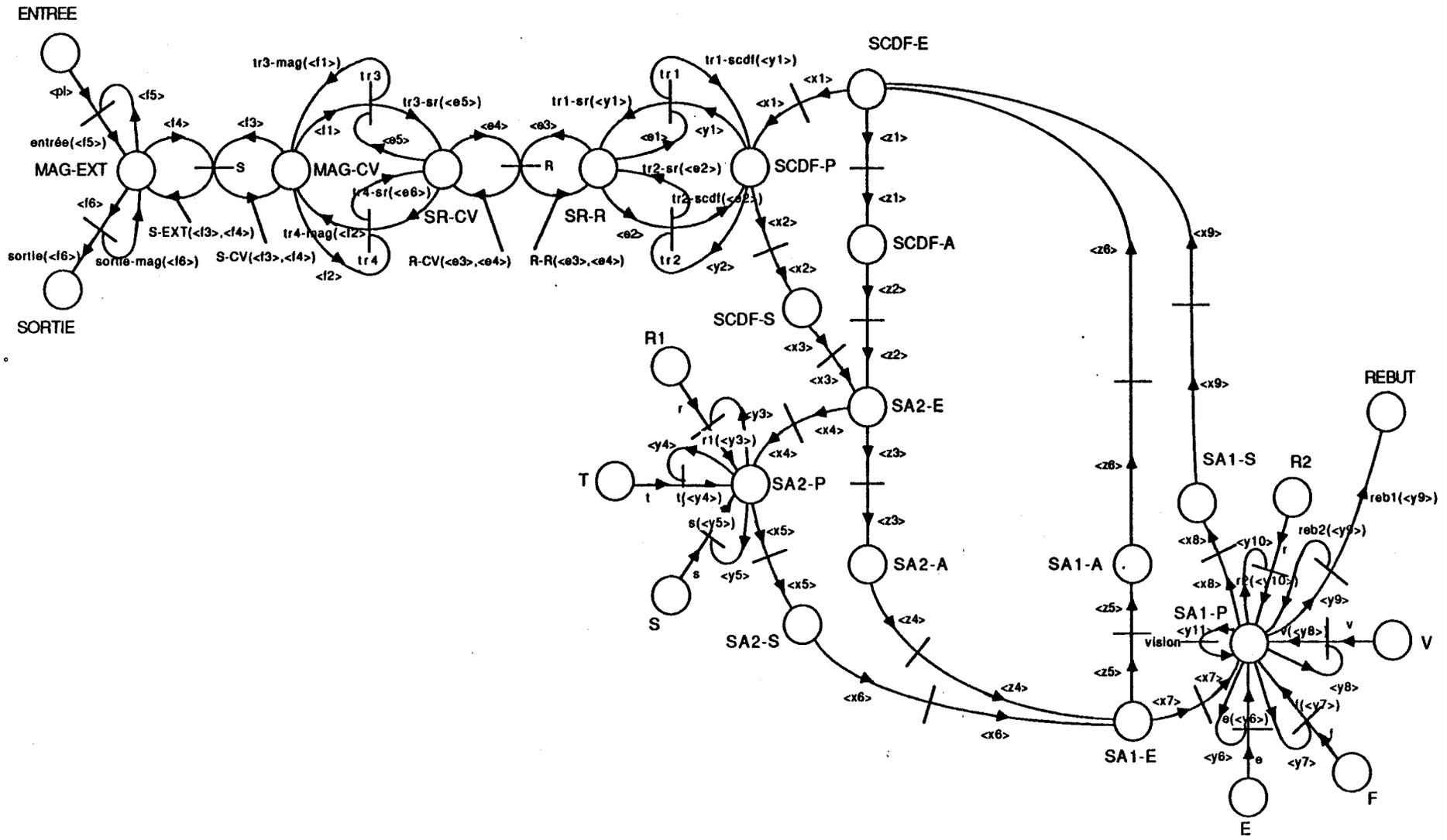


Figure Annexe 3.3 n°1

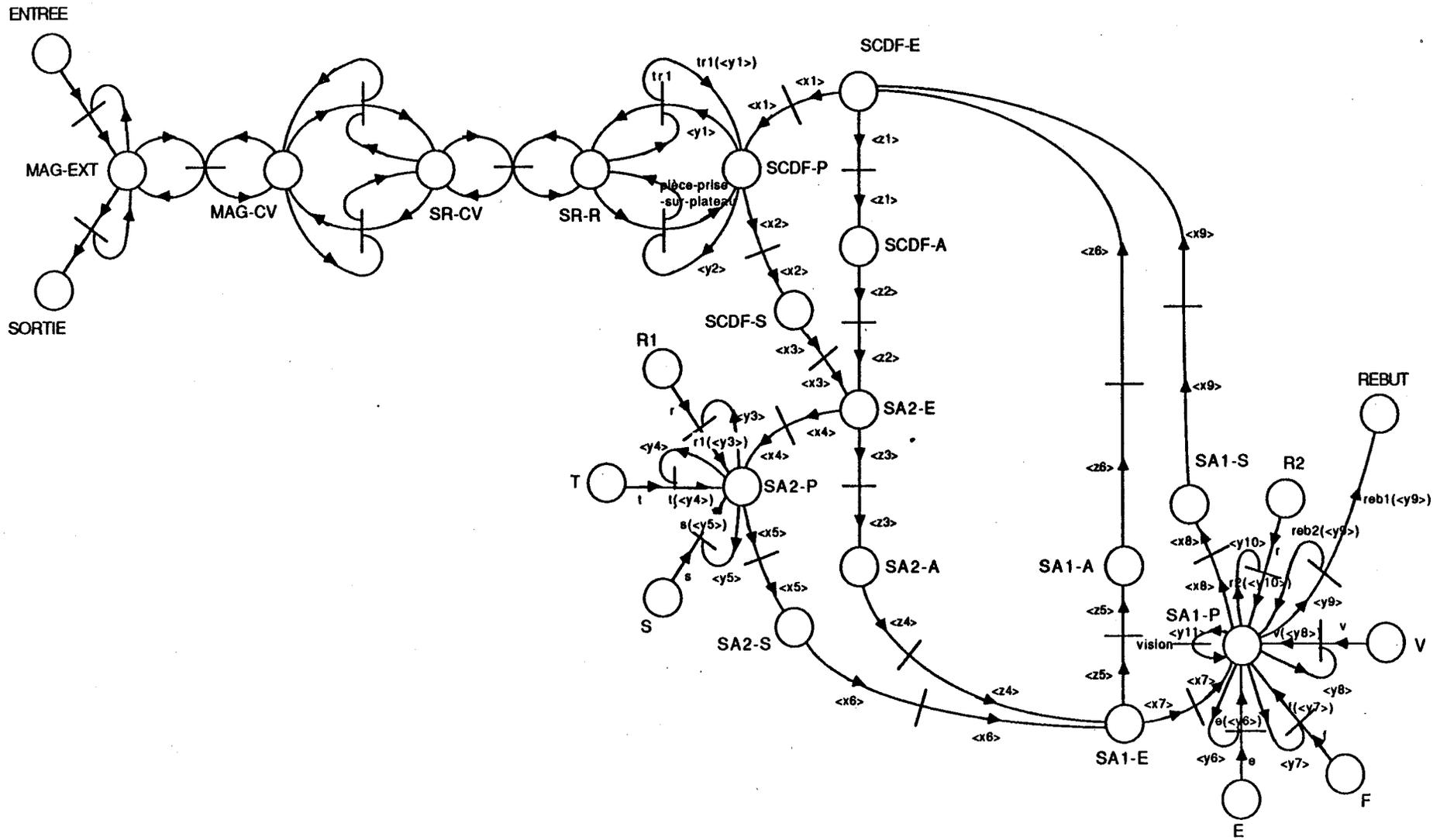


Figure Annexe 3.3 n°2

**Figure Annexe 3.3 n°2**  
**Domaines de couleurs, possibilité n°1**

Définition du fonctionnement représenté :

La première possibilité de coloration détaillée ici représente un mode de fonctionnement dans lequel un certain ordonnancement est imposé aux pièces dès leur palettisation au niveau du poste SCDF-P. En effet, l'ordre dans lequel les stations SA1 et SA2 devront intervenir, ainsi que les opérations qui devront être effectuées sur ces stations sont imposés aux pièces dès le départ. Ces informations sont portées par les palettes : p représente une palette neutre, p10, p30, p31, p32, p50, p51, p52 représentent des palettes portant les informations relatives aux gammes respectives 10, 30, 31, 32, 50, 51 et 52.

Les gammes sont définies ci dessous :

numéro	Stations (dans l'ordre d'intervention)	Travaux associés
10	SA1	montage écrous e,plages fixes f et vis v
30	SA2	montage relais r et transistors t
31	SA1 SA2	montage relais r montage transistors t
32	SA2 SA1	montage transistors t montage relais r
50	SA2	montage relais r montage transistors t montage résistances s
51	SA1 SA2	montage relais r montage transistors t et résistances s
52	SA2 SA1	montage transistors t et résistances s montage relais r

Coloration au sein des modules :

**SCDF**

$\text{dom}(\langle x1 \rangle) = \{p, p10, p30, p31, p32, p50, p51, p52, p10b, p30c, p31c, p32c, p50c, p51c, p52c\}$

$\text{dom}(\langle x2 \rangle) = \{p10b, p30c, p31c, p32c, p50c, p51c, p52c\}$

$\text{dom}(\langle x3 \rangle) = \{p10b, p30c, p31c, p32c, p50c, p51c, p52c\}$

dom(<y1>) = {p10befv+,p30crt+,p31crt+,p32crt+,p50crts+,p51crt+,p52crts+}

dom(<y2>) = {p,p10,p30,p31,p32,p33,p50,p51,p52 }

tr1 : <y1>	tr1(<y1>)
p10befv+	p
p30crt+	p
p31crt+	p
p32crt+	p
p50crts+	p
p51crts+	p
p52crts+	p

la fonction "pièce-prise-sur-plateau" génère une marque appartenant à l'ensemble : {p10b,p30c,p31c,p32c,p50c,p51c,p52c} en fonction du choix fait par le Niveau Hiérarchique sur l'image dont il dispose du stockeur rotatif SR et en fonction du type de palette présent en SCDF-P. Les palettes provenant de rebuts (p10,p30,p31,p32,p50,p51,p52) recevront une pièce dont la gamme correspond à la gamme de la pièce qui a été rebutée (respectivement 10b,30c,31c,32c,50c,51c,52c).

## SA2

dom(<x4>) = {p30c,p31cr,p32c,p50c,p51cr,p52c}

dom(<x5>) = {p30crt,p31crt,p32ct,p50crts,p51crts,p52cts}

dom(<x6>) = {p30crt,p31crt,p32ct,p50crts,p51crts,p52cts}

dom(<y3>) = {p30c,p30ct,p50c,p50ct,p50cs,p50cts}

dom(<y4>) = {p30c,p30cr,p31cr,p50c,p50cr,p50cs,p50rs,p51cr,p52c,p52cs}

dom(<y5>) = {p50c,p50cr,p50ct,p50rt,p52c,p52ct}

r :	<y3>	r(<y3>)
	p30c	p30cr
	p30ct	p30crt
	p50c	p50cr
	p50ct	p50crt
	p50cs	p50crs
	p50cts	p50crts

t :	<y4>	t(<y4>)
	p30c	p30ct
	p30cr	p30crt
	p31cr	p31crt
	p32c	p32ct
	p50c	p50ct
	p50cr	p50crt
	p50cs	p50cts
	p50crs	p50crts
	p51cr	p51crt
	p51crs	p51crts
	p52c	p52ct
	p52cs	p52cts

s :	<y5>	s(<y5>)
	p50c	p50cs
	p50cr	p50crs
	p50ct	p50cts
	p50crt	p50crts
	p51cr	p51crs
	p51crt	p51crts
	p52c	p52cs
	p52ct	p52cts

**SA1**

dom(<x7>) = {p10b,p30crt,p31crt,p31c,p32ct,p50crts,p51crts,p51c,p52cts}

dom(<x8>) = {p10,p30,p31,p32,p50,p51,p52,p10befv+,p30crt+,p31crt+,p32crt+,  
p50crts+,p51crts+,p52crts+,p31cr,p51cr}

dom(<x9>)= {p10,p30,p31,p32,p50,p51,p52,p10befv+,p30crt+,p31crt+,p32crt+,  
p50crts+,p51crts+,p52crts+,p31cr,p51cr}

dom(<y6>) = {p10b}

dom(<y7>) = {p10be}

dom(<y8>) = {p10bef}

dom(<y9>) = {p10befv-,p30crt-,p31crt-,p32crt-,p50crts-,p51crts-,p52crts-}

dom(<y10>)= {p31c,p32ct,p51c,p52cts}

dom(<y11>)= {p10befv,p30crt,p31crt,p32crt,p50crts,p51crts,p52crts}

r2 :	<y10>	r2(<y10>)
	p31c	p31cr
	p32ct	p32crt
	p51c	p51cr
	p52cts	p52crts

e :	<y6>	e(<y6>)
	p10b	p10be

f :	<y7>	f(<y7>)
	p10be	p10bef

v :	<y8>	v(<y8>)
	p10bef	p10befv

reb1:	<y9>	reb1(<y9>)
	p10befv-	befv-
	p30crt-	crt-

p31crt-	crt-
p32crt-	crt-
p50crts-	crts-
p51crts-	crts-
p52crts-	crts-

reb2:<y9>	reb2(<y9>)
p10befv-	p10
p30crt-	p30
p31crt-	p31
p32crt-	p32
p50crts-	p50
p51crts-	p51
p52crts-	p52

vision :<y11>	vision(<y11>)
p10befv	(p10befv-,p10befv+)
p30crt	(p30crt-,p30crt+)
p31crt	(p31crt-,p31crt+)
p32crt	(p32crt-,p32crt+)
p50crts	(p50crts-,p50crts+)
p51crts	(p51crts-,p51crts+)
p52crts	(p52crts-,p52crts+)

## CONVOYEUR CENTRAL

dom(<z1>) = {p,p10,p30,p31,p32,p50,p51,p52,p10b,p30c,p30crt,p31c,p31cr,p31crt,  
p32c,p32ct,p50c,p50crts,p51c,p51cr,p51crts,p52c,p52cts,  
p10befv+,p30crt+,p31crt+,p32crt+,p50crts+,p51crts+,p52crts+}

dom(<z2>) = {p,p10,p30,p31,p32,p50,p51,p52,p10b,p30c,p30crt,p31c,p31cr,p31crt,  
p32c,p32ct,p50c,p50crts,p51c,p51cr,p51crts,p52c,p52cts,  
p10befv+,p30crt+,p31crt+,p32crt+,p50crts+,p51crts+,p52crts+}

dom(<z3>) = {p,p10,p30,p31,p32,p50,p51,p52,p10b,p30c,p30crt,p31c,p31cr,p31crt,  
p32c,p32ct,p50c,p50crts,p51c,p51cr,p51crts,p52c,p52cts,  
p10befv+,p30crt+,p31crt+,p32crt+,p50crts+,p51crts+,p52crts+}

dom(<z4>) = {p,p10,p30,p31,p32,p50,p51,p52,p10b,p30c,p30crt,p31c,p31cr,p31crt,  
p32c,p32ct,p50c,p50crts,p51c,p51cr,p51crts,p52c,p52cts,  
p10befv+,p30crt+,p31crt+,p32crt+,p50crts+,p51crts+,p52crts+}

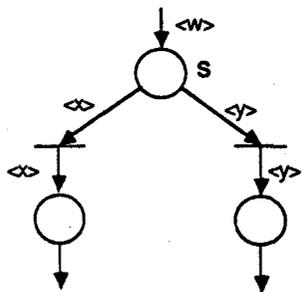
dom(<z5>) = {p,p10,p30,p31,p32,p50,p51,p52,p10b,p30c,p30crt,p31c,p31cr,p31crt,  
p32c,p32ct,p50c,p50crts,p51c,p51cr,p51crts,p52c,p52cts,  
p10befv+,p30crt+,p31crt+,p32crt+,p50crts+,p51crts+,p52crts+}

dom(<z6>) = {p,p10,p30,p31,p32,p50,p51,p52,p10b,p30c,p30crt,p31c,p31cr,p31crt,  
p32c,p32ct,p50c,p50crts,p51c,p51cr,p51crts,p52c,p52cts,  
p10befv+,p30crt+,p31crt+,p32crt+,p50crts+,p51crts+,p52crts+}

Figure Annexe 3.3 n°2  
 Domaines de couleurs, possibilité n°2

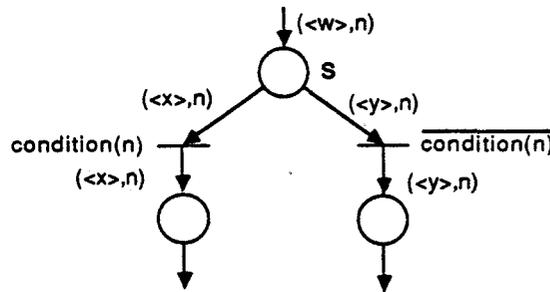
Définition du fonctionnement représenté :

Une autre possibilité de coloration donnée ici laisse beaucoup plus de souplesse de circulation aux pièces traitées. Elle engendre en contrepartie un plus grand nombre d'indéterminismes directionnels. Pour résoudre ces indéterminismes, un supplément d'information concernant le procédé est nécessaire. Ces informations peuvent être portées ou adressées par les marques en utilisant les réseaux à prédicats. La coloration impose un cheminement rigide des marques sur le graphe de commande. Les informations annexes  $n_1, n_2, \dots$ , rapportées à la coloration  $\langle x \rangle$  dans les  $n$ -uplets  $(\langle x \rangle, n_1, n_2, \dots)$  supportés par les marques des réseaux à prédicats, peuvent être utilisées alors par des conditions d'orientation des pièces établies de manière dynamique (par un Niveau Décisionnel éventuellement local). On ne peut pas établir ces conditions systématiquement dès le départ sur le graphe de commande. C'est la raison pour laquelle, en dehors de quelques cas simples, seule la coloration rigide peut être examinée efficacement lors de l'analyse de cohérence et de complétude qui conduit au Prégraphe. Les variables représentant les informations complémentaires sont alors à rajouter par la suite sur le modèle Prégraphe qui porte déjà la coloration rigide.



$\text{dom}(\langle w \rangle) = \{p\text{-type1}, p\text{-type2}\}$   
 $\text{dom}(\langle x \rangle) = \{p\text{-type1}\}$   
 $\text{dom}(\langle y \rangle) = \{p\text{-type2}\}$

L'orientation des marques est très rigide.



$\text{dom}(\langle w \rangle, n) = \{(p, \text{type1}), (p, \text{type2})\}$   
 $\text{dom}(\langle x \rangle, n) = \{(p, \text{type1}), (p, \text{type2})\}$   
 $\text{dom}(\langle y \rangle, n) = \{(p, \text{type1}), (p, \text{type2})\}$

"condition(n)" est déterminée de façon dynamique par le Niveau Hiérarchique (éventuellement local).

$n$  peut être une propriété identificatrice de la pièce correspondant à la marque, ou une adresse vers une table de description de l'image utile de la pièce.

L'orientation des pièces peut être ainsi déterminée de façon très évolutive, contrairement à ce qui est permis par la coloration simple.

L'exemple donné ci-dessous comporte encore un certain niveau de coloration. Si on veut avoir une souplesse de fonctionnement maximale, il est possible de supprimer toute coloration du graphe de coordination de la commande. On établit alors le maximum d'indéterminismes qui devront être résolus par des conditions établies dynamiquement à partir des informations données ou adressées par  $n_1, n_2, \dots$  ou consultables par un autre moyen.

Coloration au sein des modules :

### SCDF

$\text{dom}(\langle x1 \rangle) = \{p, \text{pbefv+}, \text{pcrt+}, \text{pcrts+}\}$

$\text{dom}(\langle x2 \rangle) = \{p, \text{pb}, \text{pc}\}$

$\text{dom}(\langle x3 \rangle) = \{p, \text{pb}, \text{pc}\}$

$\text{dom}(\langle y1 \rangle) = \{\text{pbefv+}, \text{pcrt+}, \text{pcrts+}\}$

$\text{dom}(\langle y2 \rangle) = \{p\}$

$\text{tr1} : \langle y1 \rangle$	$\text{tr1}(\langle y1 \rangle)$
pbefv+	p
pcrt+	p
pcrts+	p

la fonction "pièce-prise-sur-plateau" génère une marque appartenant à l'ensemble : {pb,pc} en fonction du choix fait par le Niveau Hiérarchique sur l'image dont il dispose du stockeur rotatif SR et du poste SCDF-P.

### SA2

$\text{dom}(\langle x4 \rangle) = \{\text{pc}, \text{pcr}, \text{pct}, \text{pcs}, \text{pcrt}, \text{pcts}\}$

$\text{dom}(\langle x5 \rangle) = \{\text{pc}, \text{pcr}, \text{pct}, \text{pcs}, \text{pcrt}, \text{pcts}, \text{pcrts}\}$

$\text{dom}(\langle x6 \rangle) = \{\text{pc}, \text{pcr}, \text{pct}, \text{pcs}, \text{pcrt}, \text{pcts}, \text{pcrts}\}$

$\text{dom}(\langle y3 \rangle) = \{\text{pc}, \text{pct}, \text{pcs}, \text{pcts}\}$

$\text{dom}(\langle y4 \rangle) = \{\text{pc}, \text{pcr}, \text{pcs}, \text{pcrs}\}$

$\text{dom}(\langle y5 \rangle) = \{\text{pc}, \text{pcr}, \text{pct}, \text{pcrt}\}$

$r : \langle y3 \rangle$	$r(\langle y3 \rangle)$
pc	pcr
pct	pcrt
pcs	pcrs
pcts	pcrts

$t : \langle y4 \rangle$	$t(\langle y4 \rangle)$
pc	pct
pcr	pcrt
pcs	pcts
pcrs	pcrts

s :	<y5>	s(<y5>)
	pc	pcs
	pcr	pcrs
	pct	pcts
	pcrt	pcrts

### SA1

dom(<x7>) = {pb,pbe,pbef,pbefv,pc,pct,pcs,pcts,pcrt,pcrts}

dom(<x8>) = { pb,pbe,pbef,pbefv,pbefv+,pc,pct,pcs,pcrt,pcts,pcrts,pcrt+,pcrts+}

dom(<x9>) = {pb,pbe,pbef,pbefv,pbefv+,pc,pct,pcs,pcrt,pcts,pcrts,pcrt+,pcrts+}

dom(<y6>) = {pb}

dom(<y7>) = {pbe}

dom(<y8>) = {pbef}

dom(<y9>) = {pbefv-,pcrt-,pcrts-}

dom(<y10>)={pc,pct,pcs,pcts}

dom(<y11>)={pbefv,pcrt,pcrts}

r2 :	<y10>	r2(<y10>)
	pc	pcr
	pct	pcrt
	pcs	pcrs
	pcts	pcrts

e :	<y6>	e(<y6>)
	pb	pbe

f :	<y7>	f(<y7>)
	pbe	pbef

v :	<y8>	v(<y8>)
	pbef	pbefv

reb1:	<y9>	reb1(<y9>)
	pbefv-	befv-
	pcrt-	crt-
	pcrts-	crts-

reb2:	<y9>	reb2(<y9>)
	pbefv-	p
	pcrt-	p
	pcrts-	p

vision :<y11>  
pbefv  
pcrt  
pcrts

vision(<y11>  
(pbefv-,pbefv+)  
(pcrt-,pcrt+)  
(pcrts-,pcrts+)

## CONVOYEUR CENTRAL

dom(<z1>) = {p,pb,pbe,pbef,pbefv,pbefv+,pc,pcr,pct,pcs,pcrt,pcrs,pcts,pcrts,pcrt+,pcrts+}

dom(<z2>) = {p,pb,pbe,pbef,pbefv,pbefv+,pc,pcr,pct,pcs,pcrt,pcrs,pcts,pcrts,pcrt+,pcrts+}

dom(<z3>) = {p,pb,pbe,pbef,pbefv,pbefv+,pc,pcr,pct,pcs,pcrt,pcrs,pcts,pcrts,pcrt+,pcrts+}

dom(<z4>) = {p,pb,pbe,pbef,pbefv,pbefv+,pc,pcr,pct,pcs,pcrt,pcrs,pcts,pcrts,pcrt+,pcrts+}

dom(<z5>) = {p,pb,pbe,pbef,pbefv,pbefv+,pc,pcr,pct,pcs,pcrt,pcrs,pcts,pcrts,pcrt+,pcrts+}

dom(<z6>) = {p,pb,pbe,pbef,pbefv,pbefv+,pc,pcr,pct,pcs,pcrt,pcrs,pcts,pcrts,pcrt+,pcrts+}

## **ANNEXE 4**

**Notes sur le projet PTA**

Par deux fois dans ce mémoire, une référence à un projet PTA (Poste de Travail pour l'Automaticien) a été faite. Peu de références bibliographiques sont disponibles sur ce projet pour le moment. Citons essentiellement [FRA 87],[FRA 88],[VER 88], ainsi qu'un article de vulgarisation [RHE 88]. Nous allons brièvement donner quelques précisions ici.

La composante essentielle de ce projet est le projet BASEPTA dont le but est de définir un modèle de données représentant une application automatisée industrielle. Ce modèle doit permettre de supporter les différents traitements de conception, d'exploitation, de maintenance... qui utiliseront, enrichiront, modifieront la base de données de l'application.

De ce modèle, on déduira le cahier des charges d'un standard d'échange des données de systèmes de C.A.O d'automatismes.

La désignation du projet (BASEPTA signifie Base Application et Standard d'Echange du Poste de Travail de l'Automaticien) fait clairement apparaître les deux axes de l'étude.

Ce projet est né en 1984 d'un club utilisateurs réunissant un ensemble de grands industriels français utilisateurs d'automatismes. Il a été redéfini en 1987.

Une équipe technique constituée de sept personnes détachées de différentes entreprises et organismes travaille à la définition du modèle.

La construction des traitements que devra supporter le modèle fait l'objet de divers appels d'offres. Parmi eux, le projet TECHNO-X ([FRA 87],[VER 88]) se veut la partie édition du dossier de l'application dans PTA.

Le financement est à la fois assuré par l'Etat et par les entreprises membres du projet.

Une thèse de doctorat, présentée à l'Université de LILLE par H.PERRIEN, ingénieur à la TELEMÉCANIQUE, prévue pour le premier semestre 1989, permettra de faire le point sur ces travaux.

## **ANNEXE 5**

**Echantillon des règles de transposition RdPSAC->GRAFCET**

## Règles de transposition RdPSAC->GRAFCET

### -Codage des couleurs

Le codage des couleurs se fait par numérotation : à une couleur est associée une valeur numérique codée sur un certain nombre d'octets (2 octets pour les exemples qui ont été abordés au LAII-IDN).

### -Transposition des structures RdP en Grafcet

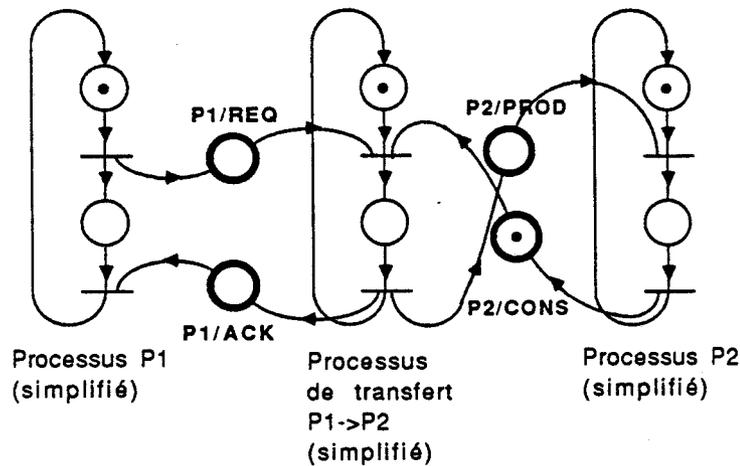
#### 1) les processus :

Les graphes de commande sont structurés en processus qui, par construction, sont saufs. A un instant donné, chaque processus ne peut donc contenir qu'une marque colorée. Ainsi, la transposition des processus en Grafcet est simple. A chaque place du processus est associé une étape Grafcet et à chaque processus est associé un mot interne  $W_p$  contenant le code de la marque colorée transitant par ce processus.

Arrivée de la marque "a" dans le processus :  $W_p \leftarrow \text{code}(a)$   
 Départ de la marque du processus :  $W_p \leftarrow \text{vide}$

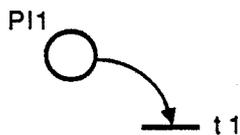
#### 2) les places de liaison :

Les places de liaison entre processus RdP (figure ci-dessous) sont représentées par des mots internes.

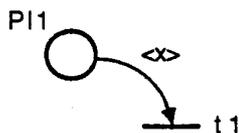


Le test du marquage de la place se fait en testant la valeur du mot interne.

Notons  $PI_1$  une place de liaison et  $WI_1$  le mot interne associé  
 Notons  $t_1$  une transition, ainsi que sa réceptivité.



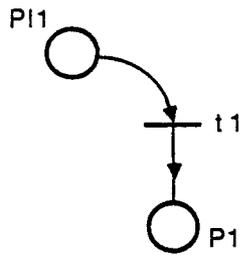
se transpose en  $\text{---} t_1 \ \& \ (WI_1 \neq 0)$



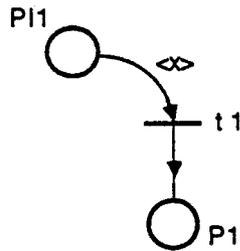
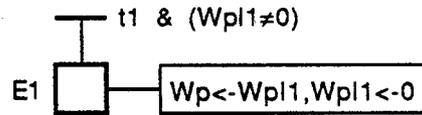
se transpose en  $\text{---} t_1 \ \& \ (WI_1 \in \{\text{code}(a), \text{code}(b)\})$

$\text{dom}(\langle x \rangle) = \{a, b\}$

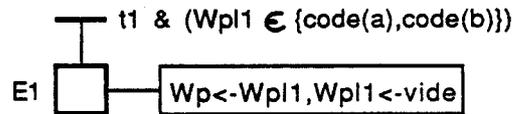
Le tir de la transition doit enlever une marque de la place de liaison.  
 Soit P1 une place d'un processus P et soit Wp le mot interne associé au processus P.



se transpose en

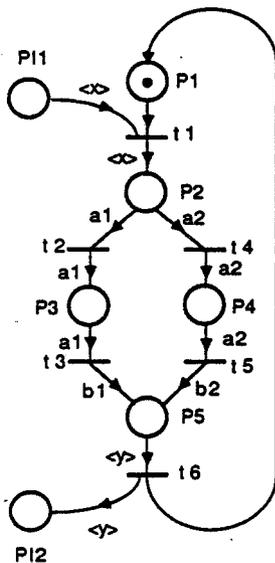


se transpose en

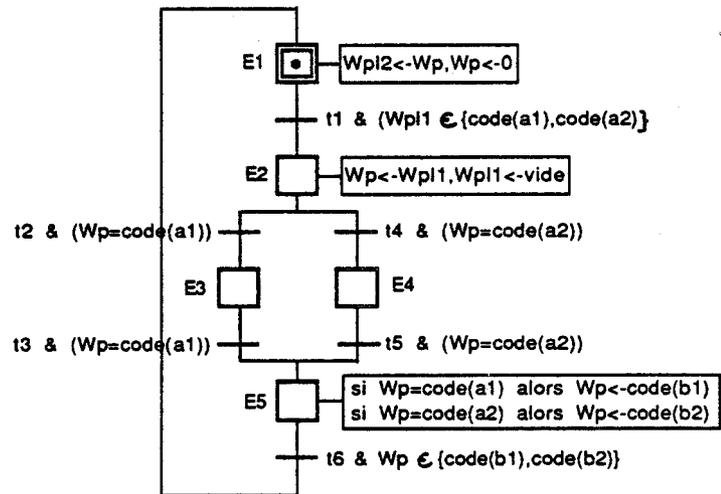


dom(<x>)= {a,b}

3) Exemple :



se transpose sous la forme



dom(<x>)= {a1,a2}  
 dom(<y>)= {b1,b2}

4) Le caractère adaptatif :

La structuration et la coloration étant traduites, il reste à traduire le caractère adaptatif des réseaux de Petri. Ce caractère adaptatif peut être également représenté sous la forme des réseaux de Petri interprétés. On trouvera dans [MOA 85] la correspondance complète entre Grafcet et RdP interprétés.



## RESUME

Nous présentons dans ce mémoire un outil essentiel à la réalisation d'une chaîne de conception assistée du modèle de la Partie Commande de cellules flexibles de production.

Cet outil fournit un graphe appelé Prégraphe qui représente le modèle fonctionnel minimal de l'unité de production. Il constitue une base de description cohérente, complète et concise permettant d'obtenir de manière systématique un modèle structuré de la Partie Commande, chargée d'assurer la coordination et le séquençement des commandes applicables au procédé.

La méthode d'obtention du Prégraphe repose sur la description formalisée sous forme de règles opératoires des actions élémentaires et des zones opératoires associées, assistée par un logiciel d'analyse de cohérence et de complétude d'enchaînement logique qui permet d'éviter des oublis et des lacunes de conception. Une traduction et une agrégation des descriptions des gammes et des routages sont alors effectuées pour obtenir un Prégraphe sous la forme d'un réseau de Petri coloré portant les informations de base nécessaires à l'obtention du modèle réseau de Petri structuré, adaptatif et coloré de la Partie Commande. Des étapes de modulation viennent compléter la démarche.

Les limites et l'évolution du modèle Prégraphe et de la modélisation de la Partie Commande sont discutées, ainsi que l'utilisation possible d'une version enrichie du Prégraphe pour des étapes de prédimensionnement de l'architecture matérielle et de préétude de comportement dynamique des unités de production.

Le mémoire est illustré par un grand nombre d'exemples industriels.

## MOTS-CLEFS

METHODOLOGIE DE CONCEPTION

SYSTEMES DE PRODUCTION FLEXIBLES

MODELISATION

PARTIE COMMANDE

RESEAUX DE PETRI

STRUCTURATION

GRAPHE FONCTIONNEL PRELIMINAIRE

COHERENCE ET COMPLETUE LOGIQUES