

T H E S E

présentée à

L ' UNIVERSITE LYON I CLAUDE BERNARD

pour l'obtention

DU DIPLOME DE DOCTORAT

par

Béatrix BESOMBES

UN SYSTEME D'AIDE A LA CONDUITE D'ATELIER FLEXIBLE

BASE RESEAUX DE PETRI COLORES

soutenue le 8 Novembre 1990 devant le jury :

Mr J. DIMNET Rapporteur

Mr J.M PROTH Rapporteur

Mr G. BEL

Mr D. CAUCHY

Mr P. LADET

Mr A. MATHON

T H E S E

présentée à

L ' UNIVERSITE LYON I CLAUDE BERNARD

pour l'obtention

DU DIPLOME DE DOCTORAT

par

Béatrix BESOMBES



UN SYSTEME D'AIDE A LA CONDUITE D'ATELIER FLEXIBLE

BASE RESEAUX DE PETRI COLORES

soutenu le 8 Novembre 1990 devant le jury :

Mr J. DIMNET Rapporteur

Mr J.M PROTH Rapporteur

Mr G. BEL

Mr D. CAUCHY

Mr P. LADET

Mr A. MATHON

Cette thèse a été préparée au Département Stratégie du Développement
de l'Ecole des Mines de St Etienne, 458 Cours Fauriel, 42000 ST ETIENNE.

634 PL 12

UNIVERSITE CLAUDE BERNARD - LYON I

Président de l'Université
1er Vice-Président Fédération Santé
1er Vice-Président Fédération Sciences
Président du Comité de Coordination
des Etudes Médicales
Secrétaire Général

M. le Professeur P. ZECH
M. le Professeur L.M. PATRICOT
M. le Professeur UZAN

M. le Professeur P. ZECH
M. F. MARIANI

FEDERATION SANTE

- UFR de Médecine GRANGE-BLANCHE
- UFR de Médecine ALEXIS-CARREL
- UFR de Médecine LYON-NORD
- UFR de Médecine LYON-SUD
- Institut des Sciences Biologiques et Pharmaceutiques
- UFR d'Odontologie
- Institut des Techniques de Réadaptation
- Département d'Innovation et de Coordination Pédagogique

Directeur: Mme le Pr. PELLET
Directeur: M. le Pr. EVREUX
Directeur: M. le Pr. PATRICOT
Directeur: M. le Pr. DEJOUR

Directeur: M. le Pr. VILLARD

Directeur: M. le Pr. MAGLOIRE

Directeur: M. le Pr. LLORCA

FEDERATION SCIENCES

- Institut des Sciences de la Matière
- Institut des Sciences de l'Ingénierie et du Développement Technologique
- Institut de Chimie et Biologie

- Institut d'Analyse des Systèmes Biologiques et Socio-Economiques
- Institut des Sciences de la Terre, de l'Océan, de l'Atmosphère, de l'Espace et de l'Environnement
- UFR des Activités Physiques et Sportives
- I.U.T. A
- I.U.T. B
- Département de 1er Cycle Pluridisciplinaire Sciences
- Département de 2eme Cycle
 - Sciences de la Vie et de la Terre
 - Sciences pour l'Ingénieur
 - Sciences de l'Analyse et de la matière

Directeur: M. le Pr. ELBAZ

Directeur: M. le Pr. FONTAINE
Directeur: Mme VARAGNAT
(M.C.)

Directeur: M. le Pr. LEGAY

Directeur: M. le Pr. ELMY
Directeur: M. le Pr. CAMY
Directeur: M. le Pr. GIELLY
Directeur: M. le Pr. PIVOT
Directeur: M. PONCET
(M.C.)

Directeur: M. le Pr. BLANCHET
Directeur: M. le Pr. BETHOUX

Directeur: M. le Pr. VIALLE

***A mes parents
avec toute mon affection***

***A mon compagnon ERIC
et à notre fils THOMAS
en témoignage de mon amour***

REMERCIEMENTS

Les travaux présentés dans ce mémoire ont été réalisés au laboratoire du Département Stratégie du Développement de l'Ecole des Mines de Saint-Etienne.

Je tiens à remercier ;

Monsieur A. MATHON, Professeur à l'E.N.S.M.E, qui m'a accueillie dans le laboratoire dont il est responsable, pour la qualité des moyens mis à ma disposition.

Monsieur P. LADET, Professeur à l'E.N.S.M.E puis à l'I.N.P.G de Grenoble, qui a dirigé mes travaux, pour le soutien et la qualité de l'encadrement qu'il m'a accordés.

J'exprime ma gratitude à ;

Monsieur J.M PROTH, Directeur de Recherche à l'INRIA à Metz,

Monsieur J. DIMNET, Professeur à l'Université Lyon I Claude Bernard,

Monsieur G. BEL, Ingénieur de Recherche au CERT/DERA à Toulouse,

Monsieur D. CAUCHY, Ingénieur à l'A.D.E.P.A de Saint Etienne,

Monsieur P. LADET, Professeur à l' Ecole d'Ingénieurs Electriciens de Grenoble,

Monsieur A. MATHON, Professeur à l'Ecole des Mines de Saint Etienne,

qui ont apporté leur compétence pour l'évaluation de ce travail et accepté de participer au jury de thèse.

Je suis très reconnaissante aux membres de la société E.D.A ; F. MARTIN et B. PICARD D'ESTELAIN pour le soutien logistique qu'ils m'ont apporté dans la première partie de mes travaux.

Enfin, je tiens à remercier tous les membres du département ; Bernadette, Bertrand, Christine, Didier, François-Noël, Lucien, Marie-Agnès, Partha, Patrick, Philippe, Sahid, Zahia, pour l'amitié dont ils m'ont entourée et la sympathie que j'ai pu trouver en eux.

Que ceux qui ont participé de près ou de loin à la réalisation de ce mémoire trouvent ici le témoignage de ma reconnaissance.

SOMMAIRE

INTRODUCTION

CHAPITRE I : POSITION DU PROBLEME - LA CONDUITE DES ATELIERS FLEXIBLES

I - 1 LES ATELIERS FLEXIBLES

I - 1 - 1 Définition

I - 1 - 2 Intérêts des ateliers flexibles

I - 1 - 3 Limites des ateliers flexibles

I - 2 LA GESTION DE PRODUCTION

I - 2 - 1 Définition

I - 2 - 2 Structure hiérarchique du système de gestion de production

I - 3 LA CONDUITE DES ATELIERS FLEXIBLES

I - 3 - 1 Définition

I - 3 - 2 Hypothèses

I - 3 - 3 Objectifs

I - 3 - 4 Critères d'évaluation

I - 4 CONCLUSION

CHAPITRE II : LA SIMULATION - UTILISATION DES RESEAUX DE PETRI COLORES
TEMPORISES

II - 1 LES OUTILS DE SIMULATION

II - 1 - 1 Définitions de modèles à événement discrets

II - 1 - 2 Les différentes approches de la modélisation

II - 1 - 2 - a Approche par réseaux de file d'attente

II - 1 - 2 - b Approche par l'algèbre des dioïdes

II - 1 - 2 - c Approche par les Réseaux de Petri (RdP)

II - 1 - 3 Les différentes approches de la simulation

II - 1 - 3 - a Description par événements

II - 1 - 3 - b Description par activités

II - 1 - 3 - c Description par processus

II - 1 - 4 Conclusion

II - 2 MODELISATION PAR RESEAUX DE PETRI COLORES

II - 2 - 1 Les Réseaux de Petri Colorés

II - 2 - 1 - a Définition informelle

II - 2 - 1 - b Définition formelle

II - 2 - 1 - c La validation des RdPC

II - 2 - 2 Les Réseaux de Petri Colorés Temporisés

II - 2 - 3 Importance de la notion de conflit

II - 3 APPROCHE METHODOLOGIQUE DE LA MODELISATION

II - 3 - 1 Ou'est-ce que l'on modélise ?

II - 3 - 1 - a L'approche gamme

II - 3 - 1 - b L'approche atelier

II - 3 - 1 - c Comparaison des deux approches

II - 3 - 2 Jusqu'où colorer ?

II - 3 - 2 - a Simplicité des fonctions

II - 3 - 2 - b Présentation du conflit

II - 3 - 3 Conclusion

II - 4 L'INTERPRETEUR DE RdPCT

II - 4 - 1 La notion de cycle de simulation

II - 4 - 2 La notion de transition à décision

II - 4 - 3 Les temporisations aléatoires

II - 4 - 4 Le forçage d'un marquage

II - 4 - 5 L'algorithme de l'interpréteur

II - 4 - 6 Résultats de la simulation

CHAPITRE III : LES METHODES D'ORDONNANCEMENT -

APPORT DE L'INTELLIGENCE ARTIFICIELLE

III - 1 LES METHODES D'ORDONNANCEMENT

III - 1 - 1 Les méthodes de résolution du problème statique

III - 1 - 1 - a Les méthodes exactes

III - 1 - 1 - b Les méthodes heuristiques

III - 1 - 2 Les méthodes de résolution du problème dynamique

III - 1 - 2 - a Les méthodes de la théorie des graphes

III - 1 - 2 - b Les méthodes simulatoires

III - 1 - 3 Conclusion

III - 2 APPORT DE L'INTELLIGENCE ARTIFICIELLE

III - 2 - 1 Différence entre approche procédurale

et approche intelligence artificielle

III - 2 - 2 Les systèmes experts

III - 2 - 2 - a La base de connaissance

III - 2 - 2 - b Le moteur d'inférence

III - 2 - 3 Les outils d'aide à la décision en ordonnancement

III - 3 APPROCHE RETENUE

III - 3 - 1 Les langages objets

III - 3 - 2 Organisation de la connaissance

III - 3 - 3 Traitement de l'incertain

III - 3 - 4 Conclusions et Perspectives

CHAPITRE IV : EXEMPLES D'APPLICATIONS - VALIDATION INDUSTRIELLE

IV - 1 EXEMPLE 1 : PROBLEME DES GAMMES MULTIPLES

IV - 1 - 1 Conflit jeton

IV - 1 - 2 Conflit transition

IV - 2 EXEMPLE 2 : PROBLEME DE PILOTAGE

IV - 2 - 1 Conflit jeton

IV - 2 - 2 Conflit transition

IV - 3 EXEMPLE 3 : PROBLEME DE PANNES

IV - 3 - 1 Conflit jeton

IV - 3 - 2 Conflit transition

IV - 4 CONCLUSIONS

CONCLUSIONS

BIBLIOGRAPHIE

INTRODUCTION

Les années 80 ont vu l'apparition d'une nouvelle terminologie pour désigner l'ensemble des sciences et techniques de la production et de la conduite des processus manufacturiers ; on parle maintenant de productique. La productique n'est pas une technologie mais plutôt une approche nouvelle de l'organisation industrielle manufacturière.

Elle peut se définir comme l'intégration de l'ensemble des sciences en "ique"; mécanique, robotique, automatique, électronique, pneumatique,...., autour d'un outil commun; l'informatique, afin d'améliorer la productivité de l'entreprise, d'où le label "product".

Ce nouveau concept d'intégration concerne aujourd'hui l'entreprise dans sa totalité, ce que les anglo-saxons illustrent fort bien par l'appellation Computer Integrated Manufacturing (CIM) qui est l'équivalent du terme de productique.

La fluctuation des marchés et de la demande ont rendu indispensable la notion de flexibilité à tous les niveaux de l'entreprise :

- dans la phase de conception des produits (bureau d'étude et des méthodes), grâce aux outils de CAO et de DAO qui permettent de créer de nouveaux produits et de définir automatiquement les gammes de fabrication correspondantes,
- dans la phase commerciale, de suivi des commandes, de l'approvisionnement et de lancements, grâce à des outils logiciels de gestion de production qui permettent de suivre la vie d'un produit dans l'entreprise de façon à ce que le client puisse le recevoir à la date prévue,
- comme dans la phase de production proprement dite, grâce à l'introduction de l'automatisation et des ateliers flexibles, qui permettent de fabriquer simultanément plusieurs types de produits dans le même atelier et de s'adapter rapidement à toute modification de la demande tant en quantité qu'en variété.

Rançon de cette flexibilité, l'accroissement du nombre de paramètres et la difficile appréhension de leurs interactions ont rendu de plus en plus complexe la prise de décision dans de tels systèmes.

Faire de la productique dans une entreprise, ce n'est pas être un spécialiste de chacun des domaines rencontrés, mais plutôt développer une stratégie qui permette aux différents experts de l'entreprise de se rencontrer, de mettre en commun leurs connaissances, leurs outils propres, afin de les aider à prendre une décision. L'outil informatique est alors le support et le garant de la cohérence des différents niveaux de décision entre eux.

La décomposition hiérarchique de la prise de décision dans l'entreprise fait apparaître trois niveaux :

- la planification des fabrications,
- l'ordonnancement court terme,
- la commande même du système de production.

Le cadre de notre étude se place essentiellement dans l'interaction des deux derniers niveaux de cette décomposition, avec l'introduction des problèmes décisionnels de conduite des systèmes de production ou ateliers flexibles.

Parcequ'elle permet de prévoir le comportement dynamique de l'atelier, la simulation est vite apparue comme un outil indispensable d'aide à la décision dans les problèmes de conception, puis de conduite des ateliers flexibles.

Parallèlement, les systèmes de Gestion de Production Assistée par Ordinateur (GPAO), basés sur des heuristiques apportent une aide indispensable à l'ordonnancement des tâches et à la planification de la production.

Chacun de ces outils se trouve confronté à un type de problème bien particulier (la représentation des systèmes séquentiels pour les outils de simulation, l'explosion combinatoire des solutions pour les outils d'ordonnancement et de gestion) pour lesquels on a développé des méthodes ou des modèles adaptés.

Pourtant, chacun de ces outils trouve sa limite dans l'interaction entre les différents niveaux de décision.

Que faire d'un outil de simulation qui ne permette pas de représenter des règles de gestion et d'ordonnancement ?

Que faire d'un outil de gestion de production qui ne tient pas compte de l'état réel de l'atelier à tout instant et dont le plan d'ordonnancement se trouve impraticable dès lors qu'une panne ou qu'une demande urgente survient ?

Notre contribution porte sur la construction d'un système d'aide à la conduite des ateliers flexibles. Le terme de conduite englobe ici les notions de commande et d'ordonnancement. Nous proposons d'utiliser conjointement les approches graphiques du type Réseaux de Petri Colorés Temporisés et les techniques d'Intelligence Artificielle selon leurs potentialités respectives. Le but est alors de conduire en temps réel l'atelier, de façon à contrôler l'exécution du plan de fabrication et de s'adapter aux aléas; pannes ou demandes urgentes.

L'originalité de notre approche tient essentiellement dans le fait que nous faisons remonter la décision, de l'atelier modélisé vers le système de gestion de production, de façon à être plus proches de la configuration réelle de la prise de décision. Elle répond aux questions du chef de l'atelier qui doit réorganiser son plan de production devant une panne ou une demande urgente.

Le présent mémoire est découpé en quatre chapitres.

La première partie introduit le problème de la conduite d'atelier flexible, en redéfinissant la notion même d'atelier flexible, puis en positionnant sa problématique au sein de la démarche de gestion de production de l'entreprise.

La deuxième partie est consacrée aux différents outils de modélisation et de simulation d'ateliers flexibles. La présentation des Réseaux de Petri Colorés, puis d'une approche méthodologique de modélisation nous conduit à présenter l'interpréteur de RdPC développé pour notre application avec toutes les spécificités nécessaires pour un système d'aide à la conduite.

La troisième partie présente les méthodes d'ordonnancement et leurs limitations. L'intérêt d'une approche basée sur les techniques de l'intelligence artificielle nous permet d'introduire l'approche que nous avons retenue pour le système SAGASSE.

La quatrième partie illustre trois exemples d'applications du système SAGASSE avec le développement de bases de connaissances plus appropriées aux problèmes d'ordonnancement, de routage, et de pannes. Le dernier exemple nous a permis de valider le système proposé sur une étude industrielle.

CHAPITRE I

POSITION DU PROBLEME

LA CONDUITE DES ATELIERS FLEXIBLES

I - 1 LES ATELIERS FLEXIBLES

Face aux nouveaux enjeux économiques, l'amélioration de la productivité des industries manufacturières est devenue une véritable nécessité pour lutter contre la concurrence. Par ailleurs, l'exigence de qualité des produits et l'évolution du contexte social contribuent à rechercher une amélioration des conditions de travail en supprimant les tâches répétitives et fastidieuses.

Au cours de ces dernières années, l'informatisation et l'automatisation ont permis l'introduction de nouveaux schémas d'organisation des moyens de production. De l'organisation Taylorienne (ligne de transfert produisant un seul type de produit à cadence fixée), on est passé au concept d'atelier Flow-Shop (flux de produits suivant des itinéraires plus ou moins rigides), puis d'ilôts de fabrication (regroupement de produits voisins mettant en oeuvre des moyens semblables).

L'industrie manufacturière se caractérise aujourd'hui par une production de petites et moyennes séries dans lesquelles les produits peuvent répondre à une demande et à des gammes très diversifiées, mettant en oeuvre des moyens non spécifiques. Une même machine est capable d'effectuer plusieurs tâches et une même tâche peut être effectuée sur plusieurs machines. Les schémas présentés précédemment; lignes transfert, ilôts, se montrent alors trop rigides. L'automatisation se heurte à la recherche d'un équilibre entre l'amélioration de la productivité d'une part et la nécessité de laisser une grande souplesse d'adaptation aux moyens de production d'autre part.

Les progrès faits dans différents domaines, tels l'informatique industrielle, la robotique, les machines à commande numérique, ont permis l'introduction d'une nouvelle organisation des moyens de production appelée atelier flexible.

I - 1 - 1 Définition

Un atelier flexible est caractérisé par "le regroupement de moyens de fabrication les plus polyvalents possible reliés par un système de transport automatisé, le tout commandé par un système de contrôle informatisé". [BEHRTUL - 85]

Nous désignons par ressources l'ensemble des moyens mis en oeuvre lors de la fabrication. Elles comprennent :

- les machines conventionnelles ou à commande numérique, les groupes de machines ou même les cellules flexibles,
- les outils avec leurs magasins,
- les moyens de stockage,
- les moyens de transport ; chariots filoguidés, convoyeurs, ponts roulants, robots,...
- les supports nécessaires au transport des pièces, palettes, crochets, pinces,...

Le niveau d'automatisation des différentes ressources, et leur capacité à s'adapter rapidement à une modification de la demande en termes de nature ou de quantité de produits est le garant de la flexibilité de l'atelier.

Nous désignons par gamme la description de l'ensemble des opérations que doit subir un produit lors de sa transformation. Une opération ou tâche est elle-même définie par la ou les ressources concernées et par sa durée, caractérisant le temps pendant lequel la ressource machine ou moyen de transport est indisponible. Lorsque l'ordre des machines est identique pour toutes les gammes, quel que soit le produit, quelle que soit l'opération, on parle de production en flux de produits ou de Flow-Shop. Nous nous intéresserons au cas plus général, où l'ordre des opérations est différent pour chaque produit, on parle alors de production par tâches ou Job-Shop.

L'exigence de productivité induit la notion de flexibilité de l'outil de production qui nous conduit à rencontrer des opérations mettant en oeuvre une même ressource, mais on peut aussi rencontrer une opération pouvant être réalisée sur plusieurs ressources avec des performances identiques ou dégradées. On parle alors de "gamme-multiple". Cette pratique est très courante dans l'industrie manufacturière car elle permet de réorienter un produit vers une machine alternative en cas de panne ou de situation de blocage. Cette notion apporte une plus grande flexibilité au mode opératoire.

Nous désignons par routage, la description du circuit physique parcouru par le produit lors de sa fabrication. Il précise l'ensemble des ressources de transport nécessaires. De la même façon que pour la gamme, nous introduisons la notion de routage multiple lorsque le circuit n'est pas défini a priori.

De par la gestion en temps réel de l'ensemble des moyens de production, et la flexibilité de l'atelier, l'objectif est de s'adapter rapidement à l'évolution de la demande, tant en nature du produit, qu'en quantité.

I - 1 - 2 Intérêts des ateliers flexibles

Le nombre d'ateliers d'usinage, dits flexibles, en service dans le monde est évalué à environ 200, dont une trentaine en France. (Rapport sur l'état de l'informatisation en France). Malgré leur coût élevé, on peut parler d'un véritable essor des ateliers flexibles. Leur nombre, en France, a doublé entre les années 1984 et 1986.

Cet engouement s'explique aisément par les intérêts mis en jeu dans une telle implantation.

L'atelier flexible permet :

- l'adaptation rapide à des variations de la demande ou du marché en terme de :
 - variété du produit,
 - quantité à produire,
 - modification du produit.
- l'amélioration des conditions de travail et de sécurité du personnel.
- l'amélioration de la qualité des produits par la diminution des tâches fastidieuses et répétitives sources de défaillance humaine.
- l'augmentation du taux d'engagement des machines, grâce à :
 - la diminution du personnel et la possibilité de travailler en équipe 7 jours sur 7 et 24 heures sur 24,
 - l'utilisation de machines à commande numérique polyvalentes, réduisant le nombre de types de machines distinctes,
 - la diminution des temps improductifs (réglage, changement d'outil),
 - la réduction des temps de manutention pour le chargement, le déchargement.

- la diminution des en-cours et des stocks de produits finis, grâce à :
 - la réduction des temps de manutention pour le transfert des pièces,
 - l'amélioration de la disponibilité des machines,
 - la diminution du nombre de pièces défectueuses.
- la diminution des délais de livraison client, grâce à :
 - la diminution du temps de cycle des produits à travers une diminution des temps non opératoires (transport, stockage), grâce à la possibilité de réaliser des opérations multiples sur un même centre d'usinage.
- l'amélioration du contrôle de la production grâce au suivi informatique des produits dans l'atelier.

Cette liste est non exhaustive, elle regroupe les principaux résultats observés lors de l'implantation des premiers ateliers flexibles. [LEARY - 83] [GINTY - 81]

I - 1 - 3 Limites des ateliers flexibles

Si les ateliers flexibles sont un outil performant pour améliorer la productivité des entreprises, afin de lutter contre la concurrence et s'adapter à l'évolution des marchés, ils ne sauraient être une solution standard pour toutes les entreprises. [PROTH - 86]

Leur conception, leur implantation, puis leur commande ne sont pas choses aisées. Avant la décision de mettre en place un atelier flexible, le chef d'entreprise doit mesurer et prendre en compte les limites et difficultés liées à un tel système.

Les principales limites des ateliers flexibles tiennent à :

- leur coût très élevé. L'investissement est deux à trois fois plus important que pour un système de production traditionnel, tandis que leur rentabilité est très difficile à évaluer (coût de la qualité, coût des retards,...),
- l'intégration et la concertation de toutes les fonctions de l'entreprise est indispensable. En amont, le bureau d'étude, le bureau des méthodes, le service de maintenance, les services de planification et de commande doivent être consultés pour une meilleure définition des entrées et des besoins de l'atelier. En aval, tous ces services doivent prendre en compte les résultats des suivis informatisés pour mettre à jour leurs propres informations. L'implantation d'un atelier flexible s'inscrit dans une démarche productique qui passe inmanquablement par l'informatisation des différents services de l'entreprise.
- la difficile acceptation par le personnel de l'atelier. L'atelier flexible est en effet très souvent perçu comme synonyme de licenciement et de suppression d'emploi. Il semble en fait, qu'il entraîne davantage un changement des habitudes et une redistribution du pouvoir avec une restructuration des ressources humaines justifiée par la nécessité de changement de qualification. Un gros effort d'information puis de formation du personnel est donc indispensable.

- la complexité rencontrée lors de leur conception. De part l'exigence de flexibilité, un grand nombre de paramètres sont variables et liés entre eux. Il est alors très difficile de dimensionner les ressources ; capacité des stocks, nombre de machines, choix du moyen de transport,... L'ajustement de ces différents paramètres est le garant du juste équilibre entre productivité et flexibilité.
- la difficulté d'automatiser un certain nombre de fonctions, comme l'élimination des copeaux, le contrôle de l'état d'usure des outils.. La robotique, l'automatisation de la gestion des outils, la reconnaissance des pièces défectueuses sont encore des sujets de recherche dans le domaine de l'automatique et de l'informatique. Des progrès sont faits ou restent à faire, mais se pose alors le problème de la difficile modification de l'atelier flexible.
- la complexité de leur commande. De la même façon que pour leur conception, la conduite, d'un atelier flexible nécessite une parfaite connaissance des différents paramètres permettant de contrôler le parcours des produits, la gestion des stocks, la gestion du réseau de transport,... Ces différents éléments justifient une période de mise au point et d'installation très longue, nécessitant par ailleurs un personnel très qualifié.
- la difficulté de réagir à un certain nombre de modes de fonctionnement aléatoires comme les pannes, les casses d'outils, l'absentéisme du personnel...

Le grand nombre de degrés de liberté des ateliers flexibles rend très complexe tant leur conception que leur conduite. Il devient alors nécessaire d'appréhender le système de production dans son ensemble, en insérant l'atelier flexible dans la démarche productive de l'entreprise et en le resituant par rapport aux outils de gestion de production en amont.

I - 2 LA GESTION DE PRODUCTION

I - 2 - 1 Définition

Parallèlement à l'introduction des ateliers flexibles, qui augmentent l'efficacité des moyens de production, l'amélioration de la gestion des flux de produits, à l'intérieur de l'entreprise, semble indispensable pour un gain de productivité. La gestion de production cherche à répondre à la question : comment organiser la production en tenant compte des contraintes techniques, des contraintes dues à la limitation des moyens dont on dispose dans l'atelier, et des objectifs fixés, comme par exemple, des dates de livraison ? [THOMAS - 80]

La gestion de production est généralement définie comme l'ensemble des méthodes et des moyens visant :

- à produire à moindre coût,
- en utilisant les moyens de production à leur plein régime,
- et en réduisant les délais de fabrication afin de satisfaire au plus vite et au mieux la clientèle.

I - 2 - 2 Structure hiérarchique du système de gestion de production

Divers modèles de gestion de production qui font apparaître une structure hiérarchisée ont été développés ces dernières années. [MELESE - 72] [DOUMEINGTS - 79] [ERSCHLER - 86].

Le schéma de la figure I-1 est un exemple de structure où la décomposition hiérarchique fait apparaître les différents horizons de décision. Pour chacun de ces niveaux, nous précisons la nature des décisions prises et les informations nécessaires à celles-ci.

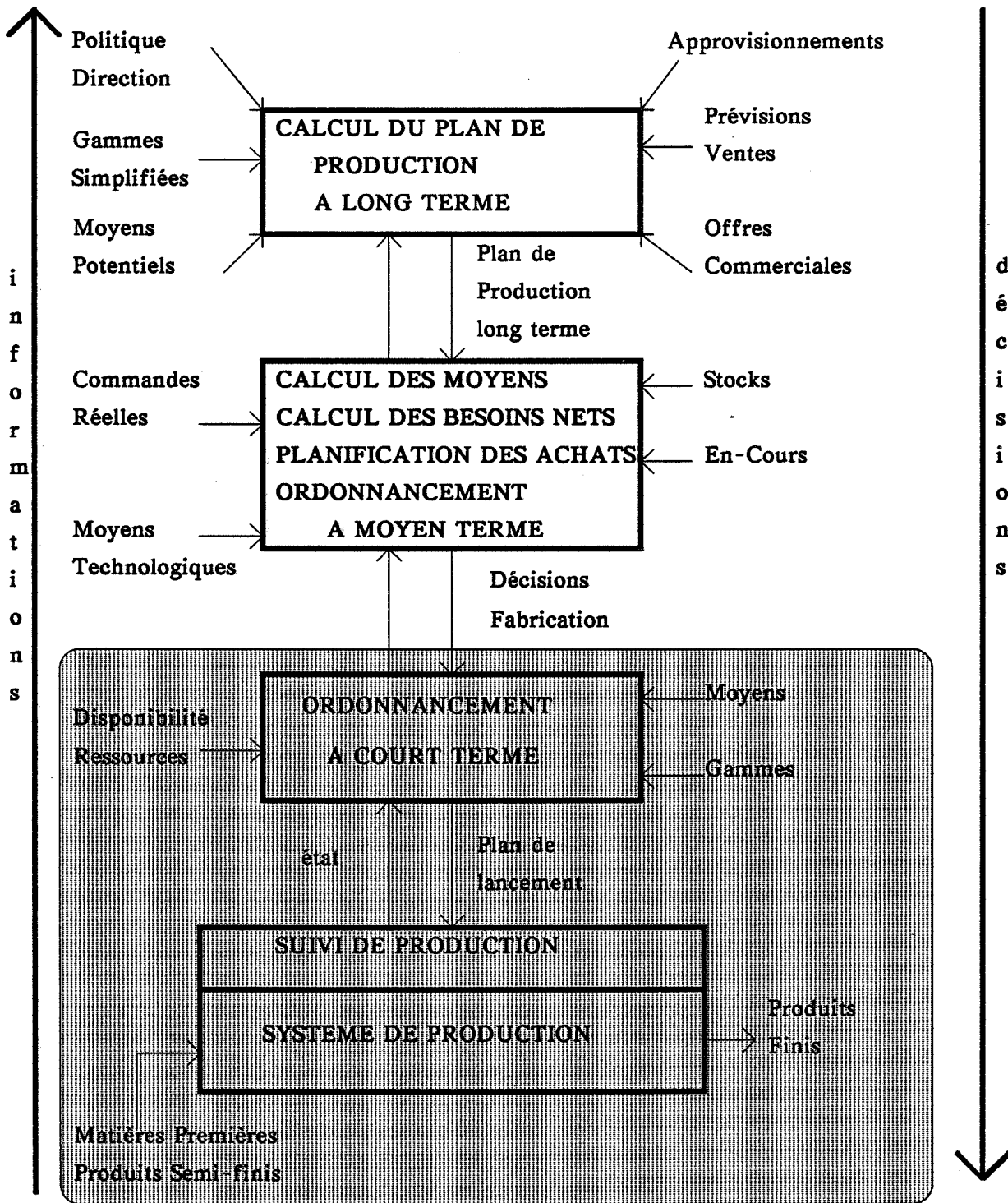


Figure I-1: Décomposition hiérarchique de la gestion de production

Le niveau de calcul de plan de production à long terme :

Il regroupe l'ensemble des choix stratégiques de l'entreprise. A partir des objectifs de la direction générale, de la capacité des moyens de production, des possibilités d'approvisionnement des matières, on détermine un plan de production. On distingue alors divers types de plan de production : travail sur stock, travail à la commande avec groupement par lots (flux poussé) ou travail à la demande (flux tiré). Il permet la définition des produits et des quantités à lancer dans l'atelier, il détermine aussi la charge induite par le carnet de commande sur les moyens de production.

Le niveau d'ordonnancement à moyen terme :

Il permet de déterminer la planification des approvisionnements et la planification de la fabrication, de façon à synchroniser les programmes d'atelier et les arrivées des ressources. Il se décompose en quatre étapes : le calcul des moyens bruts, le calcul des besoins nets, la planification des approvisionnements et l'ordonnancement à moyen terme. Cet ordonnancement à moyen terme détermine les dates de début de mise en fabrication des produits dans l'atelier, de façon à satisfaire les objectifs du plan de production.

L'ordonnancement à court terme :

Ce niveau ordonnance les opérations élémentaires sur les machines en tenant compte des contraintes techniques définies par les gammes et des contraintes de disponibilité des ressources produits et machines. Il permet la construction du plan de lancement qui définit l'ordre et les dates de passage des pièces sur les machines.

Le suivi de la production :

Ce niveau collecte l'ensemble des informations sur l'état de l'atelier, et donne en permanence une photographie instantanée de l'état d'avancement de la production.

La principale caractéristique de cette décomposition hiérarchique est la grande interaction de ces différents niveaux entre eux. La prise en compte des retours d'informations permet la remise en question des décisions prises, pour une meilleure réponse au problème de la gestion de production.

Le cadre de notre étude se situe dans l'interaction entre les deux niveaux inférieurs de ce schéma. Le problème de conduite des ateliers flexibles se trouve à l'intersection des problèmes d'ordonnancement d'une part et de commande du système physique d'autre part.

Pour chacun de ces problèmes, il existe des outils appropriés que nous développerons dans les chapitres suivants. Nous proposons alors un système d'aide à la décision pour la régulation du flux physique de production.

L'objectif de notre étude est d'isoler chacun des points de décision rencontrés au niveau de l'atelier flexible, pour faire remonter l'information, puis la prise de décision au niveau hiérarchique supérieur, celui de l'ordonnancement court terme. La simulation nous permettra de mettre en évidence les différents points de décision de l'atelier qui devront être résolus par le niveau décisionnel réalisant l'ordonnancement. Lorsque le niveau supérieur a construit une solution au problème décisionnel, la simulation permet de tester en "temps-réel" les effets de cette décision.

L'originalité de notre approche tient au fait que nous avons abordé le problème de la gestion de production "par le bas", par le niveau opérationnel et non pas par le niveau décisionnel. Cette approche présente l'intérêt d'être plus proche du problème réel, car elle permet la prise en compte de l'état réel, instantané de l'atelier. Elle répond localement aux problèmes du responsable d'atelier qui doit prendre une décision : quel produit, quelle machine, quelle direction pour le système de transport, lorsque l'atelier ne répond plus aux conditions normales de fonctionnement prévues par le système d'ordonnancement.

I - 3 LA CONDUITE DES ATELIERS FLEXIBLES

I - 3 - 1 Définition

La conduite d'atelier flexible peut se définir comme la fonction décisionnelle "regroupant toutes les décisions prises au cours du cheminement des pièces dans l'atelier et prenant en compte l'état réel, instantané de l'atelier". [ATABAKHCHE - 87]

La fonction conduite se caractérise donc par une parfaite connaissance de l'état de l'atelier afin de tenir compte des multiples aléas pouvant se produire, mais aussi par une gestion des problèmes d'ordonnement qui s'avère très complexe dans le cas d'un atelier de type job-shop.

Dans le cas d'un atelier flexible, la complexité du problème de conduite est due :

- à la complexité de l'organisation physique des moyens de production, voire des îlots de fabrication, reliés entre eux par le système de transport,
- à la complexité des flux de produits entraînant des difficultés à contrôler la circulation des produits à travers l'atelier,
- à la diversité des caractéristiques opérationnelles et fonctionnelles des différentes ressources,
- au grand nombre des combinaisons possibles (affectation des ressources, planification desancements) pour résoudre le problème d'ordonnement,
- au caractère aléatoire des perturbations au cours de la fabrication qui peuvent amener à modifier l'ordonnement initialement prévu (panne, demande urgente,...),
- la difficulté d'estimer les conséquences réelles des décisions prises et des actions correctrices alors engagées.

L'efficacité du système de conduite est donc liée à :

- la qualité du système de suivi, chargé de la mise à jour de la connaissance de l'état de l'atelier,
- l'adéquation des décisions puis des actions engendrées vis à vis de l'ordonnancement et des critères de gestion de l'entreprise.

Le problème de conduite introduit donc une composante dynamique au problème classique d'ordonnancement qui justifie l'approche du problème par simulation.

De nombreux travaux de recherche ont été réalisés pour tester les performances de politiques d'ordonnancement par la simulation de systèmes fictifs. La simulation déroule le temps en modifiant à chaque événement l'état du système modélisé selon des règles d'ordonnancement et des lois de commande choisies. L'expérimentation consiste alors à faire varier la taille de l'atelier (par exemple le nombre de machines), la fréquence d'arrivée des commandes, le jeu des dates de fin attendues et les données opératoires des gammes (nombre d'opérations et/ou durées opératoires) pour tirer des observations statistiques sur le comportement des règles de conduite que l'on souhaite valider.

Les méthodes efficaces pour un atelier fictif, manquent toutefois de souplesse vis à vis de l'environnement réel de l'atelier. Il ne s'agit plus alors de mesurer les performances, mais de réaliser les objectifs dans un environnement perturbé.

Il paraît alors préférable de concevoir la fonction ordonnancement comme un processus de décision, basé sur un simulateur, et dont le but est d'organiser en temps réel, l'utilisation des moyens afin de réaliser au mieux la fabrication des produits dans l'atelier, connaissant le plan et les objectifs fixés par le plan à moyen terme. Il ne s'agit pas alors de décrire un ordonnancement optimal, mais plutôt de réagir au mieux aux perturbations survenant dans l'atelier : pannes, demandes urgentes,... pour satisfaire les objectifs de production.

I - 3 - 2 Hypothèses

L'objectif est, à terme, de relier directement l'outil de simulation au système physique pour donner une aide à la décision en temps réel pour la conduite de l'atelier. Mais, dans un premier temps, nous utiliserons la simulation comme outil de validation ; du modèle de représentation de l'atelier d'une part et des règles de pilotage et d'ordonnancement d'autre part.

Le choix d'un système réel nous contraint toutefois à figer un certain nombre de données liées à la structure de l'atelier :

- Le nombre de ressources machines et moyens de transport est connu.
- Les stocks ont une capacité limitée par leur taille.
- L'organisation physique des ressources entre elles est déterminée.

Nous ferons par ailleurs un certain nombre d'hypothèses afin de préciser les données du problème.

- De plus en plus d'ateliers flexibles intègrent des groupes de machines similaires pouvant indifféremment exécuter les mêmes opérations. Nous les avons donc intégrés dans notre étude en temps que degré de liberté supplémentaire.
- Pour rendre compte des problèmes aléatoires dus aux pannes, nous génèrons l'occurrence d'une panne sur les ressources machines et systèmes de transport, par des lois statistiques (loi uniforme, loi exponentielle, loi de Gauss...) autour de la valeur moyenne de la durée de bon fonctionnement MTBF. De même, la durée de la panne suivra des lois statistiques autour d'une valeur moyenne ou encore être supposée constante pour la ressource concernée.
- Les gammes sont connues en début de simulation. L'affectation des opérations aux machines et les durées opératoires correspondantes ne sont toutefois pas complètement déterminées puisque nous prenons en compte des gammes-multiples.
- Les temps de réglage propres à chaque machine sont certains et inclus dans la durée des opérations.
- Les temps de transfert entre machines et/ou stocks sont connus et dépendent de la ressource de transport nécessaire.
- Chaque opération commencée ne peut plus être interrompue. L'occurrence d'une panne sur une ressource qui travaille ne sera prise en compte qu'à la fin de l'opération, pour une machine, ou du transfert, pour un système de transport. Elle pourra avoir pour effet la mise au rebut de la pièce en cours de fabrication.

Nous supposons qu'un ordonnancement "classique" de la production a été fait en amont, déterminant l'ordre et la fréquence d'arrivée des produits dans l'atelier, de façon à ce que la charge induite soit admissible, et que les dates de fin désirée puissent être respectées. Cette hypothèse nous replace dans le schéma hiérarchique de la gestion de production présenté précédemment, en précisant le plan de lancement et les objectifs fixés par l'ordonnancement moyen terme.

I - 3 - 3 Objectifs

Les objectifs du système de conduite sont aussi les critères appliqués au système de gestion de production. Ils doivent répondre aux exigences de l'entreprise :

- satisfaire le client en terme de coût, de délai, de qualité,
- dégager des bénéfices financiers par une gestion optimum de l'outil de production.

Nous avons élaboré une représentation graphique multicritère de ces différents objectifs à travers le schéma de la figure I-2.

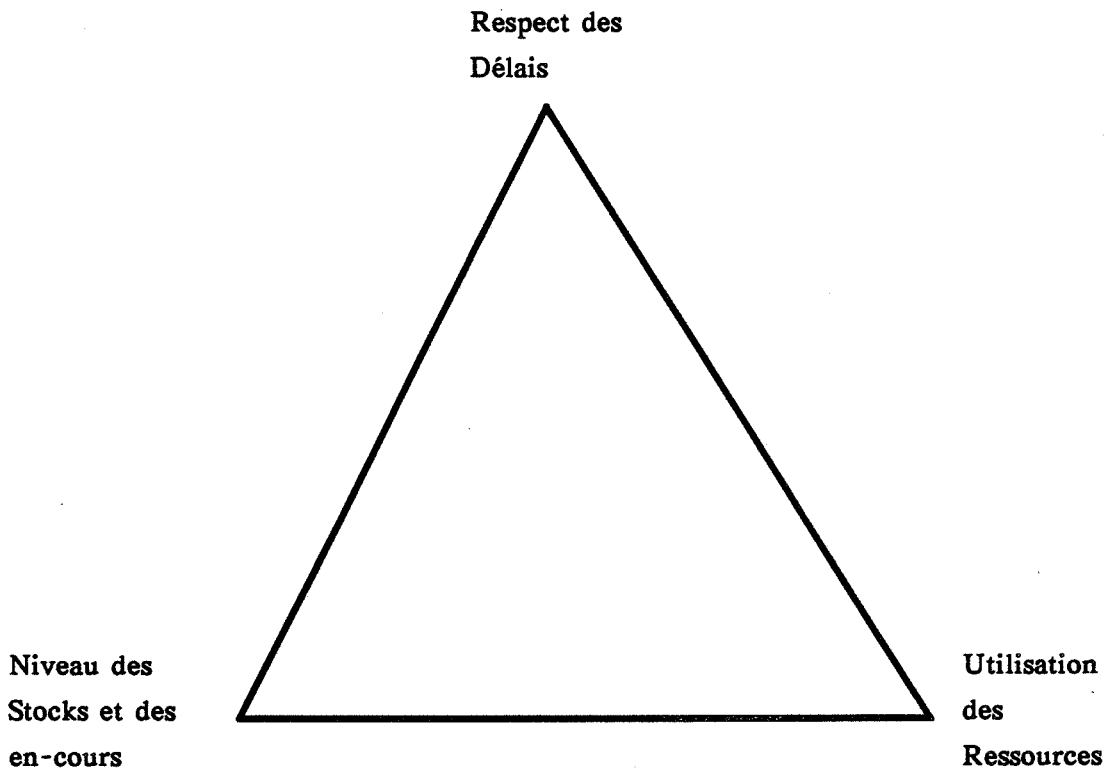


Figure I-2 : Les objectifs d'un système de conduite

Nous définirons chacun des éléments avant de montrer l'interaction des objectifs entre eux.

- La diminution des stocks de produits finis et des en-cours. Nous désignons par en-cours l'ensemble des produits présents dans l'atelier, en cours de fabrication. Il est évident qu'ils représentent une immobilisation financière importante pour l'entreprise. Par ailleurs, les coûts de gestion associés aux stocks ne peuvent être négligés.

- Le respect des délais de réalisation des ordres de fabrication. Cet objectif apparait au niveau de l'atelier sous la forme d'une date de fin désirée, calculée par le système de gestion de production. C'est elle qui permettra de livrer le produit à la date de livraison annoncée au client. Fournir un produit après la date limite a des conséquences commerciales et financières néfastes pour l'entreprise ; cela accroît les stocks et les en-cours et discrédite l'entreprise vis à vis de son client quant à la qualité et à l'efficacité de son outil de production.

- L'optimisation de l'utilisation des ressources. Le coût de plus en plus élevé des ressources engagées dans les ateliers flexibles ; machines à commande numérique, chariots filoguidés... rend indispensable un amortissement le plus rapide possible.

La difficulté pour atteindre ces objectifs est liée au fait qu'ils sont fortement contradictoires ; l'optimisation de l'un des éléments dégrade les performances de l'atelier vis à vis des deux autres.

- L'optimisation de l'utilisation des ressources tend à utiliser les ressources au maximum de leur capacité, mais a pour effet un engorgement de l'atelier avec des niveaux d'en-cours très importants, apparition de goulots d'étranglement, générant ainsi des temps de passage des produits dans l'atelier prohibitifs.

- La diminution des stocks de produits finis et des en-cours a donné naissance à la méthode de gestion appelée KANBAN, d'origine japonaise. Basée sur une production dite à flux tendu, elle consiste à "produire le produit demandé, au moment où il est demandé, dans la quantité demandée". [COURTOIS - 89] Si elle paraît simple à définir, cette méthode est en fait difficile à implanter dans une entreprise ; elle nécessite une très grande fluidité de l'écoulement des produits et est très sensible à la présence de perturbations. Tandis que les stocks ont tendance à amortir les effets des perturbations, l'utilisation de la méthode Kanban a plutôt pour effet de les amplifier. Son implantation dans l'entreprise doit donc être précédée par une politique de qualité visant le "zéro défaut" au niveau du produit, et par une politique de maintenance visant le "zéro aléa" au niveau des moyens de production.

- Le respect des délais de réalisation des ordres de fabrication implique une diminution des en-cours ou une duplication des ressources afin de supprimer les temps d'attente des produits. Cette politique induit une augmentation des temps morts pour les machines et une baisse de la productivité de l'atelier. Par ailleurs, cela pose le problème de l'introduction de produits dits "urgents" qui ont pour effet l'engorgement des stocks par les produits moins prioritaires et la sous-exploitation des moyens de production non directement concernés par cette fabrication urgente.

Il semble donc que la réalisation des objectifs de conduite passe par la recherche d'un équilibre entre ces trois critères antagonistes. Si l'on considère que chacun de ces critères peut être évalué par une fonction coût qui lui est associée (coût du retard, coût de stockage, coût d'amortissement), la position d'équilibre est alors le barycentre de la pyramide ainsi construite. La figure I-3 illustre cette construction.

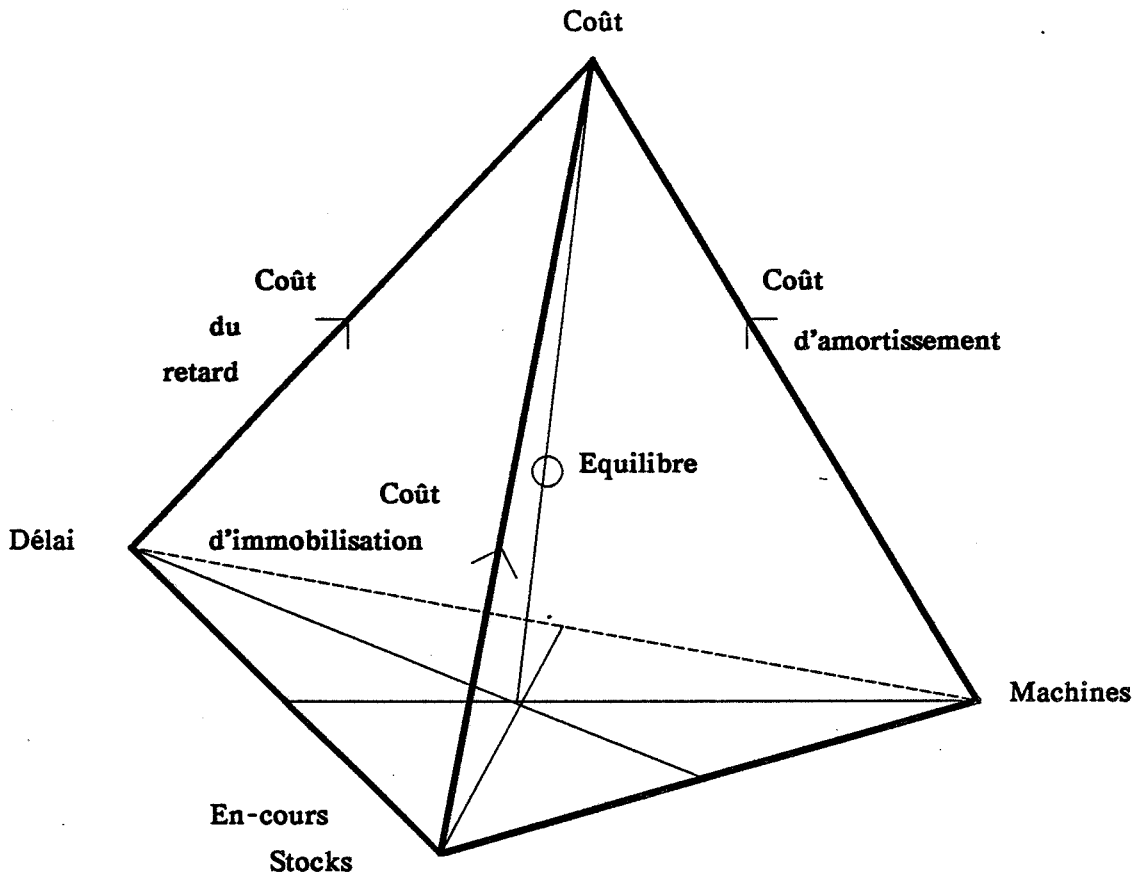


Figure I-3: Pyramide des coûts

Positionner l'atelier à l'intérieur de cette pyramide permet de visualiser les performances du système de conduite à un instant donné. Cette construction apporte une base d'informations indispensable qui aide l'opérateur humain ou le système de conduite à prendre une décision, le choix d'une action étant guidé par la volonté de ramener l'atelier le plus proche possible de la position barycentre de la pyramide.

Une dernière difficulté subsiste toutefois, liée à l'environnement dynamique et perturbé du problème de conduite ; chacun des indicateurs est variable dans le temps, fonction de l'état de l'atelier. Il est alors important de définir quelles sont les variables d'observation qui permettront d'évaluer les performances du système vis à vis des objectifs de conduite.

I - 3 - 4 Critères d'évaluation

Nous distinguerons deux types de critères ; des critères globaux caractérisant les performances moyennes de l'atelier, des critères locaux donnant des informations court terme sur l'état de l'atelier.

La bibliographie regroupe essentiellement des critères globaux concernant le respect des délais. Russel utilise les valeurs moyenne et maximum du temps de passage dans l'atelier et du retard d'un ordre de fabrication. [RUSSEL - 87]

Pour notre part, nous utiliserons des critères propres à chacun des points de la pyramide présentée précédemment :

- Pour le respect des délais :
 - le retard moyen par produit,
 - le retard moyen tous produits confondus,
 - la variance du retard par produit,
 - la variance du retard tous produits confondus,
 - le retard maximum par produit,
 - taux de produits achevés en avance par produit,
 - taux de produits achevés en retard par produit.

- Pour la minimisation des stocks et des en-cours :
 - le nombre total de produits entrés et sortis de l'atelier sur une période donnée,
 - le nombre moyen de produits dans le stock amont d'une machine par produits,
 - le nombre moyen de produits dans le stock amont d'une machine, tous produits confondus,
 - la valeur moyenne en heures de travail, des opérations en attente dans le stock amont d'une machine,
 - la durée moyenne d'achèvement de chacun des produits,
 - la durée moyenne d'achèvement des produits tous produits confondus,
 - la variance de la durée d'achèvement de chacun des produits,
 - la variance de la durée d'achèvement des produits tous produits confondus.

- Pour l'optimisation de l'utilisation des ressources :
 - taux de blocage par machine ; la machine a terminé son opération mais ne peut être déchargée en aval,
 - taux d'inutilisation par machine,
 - taux de panne par machine,
 - taux d'occupation par machine,
 - taux de blocage par ressource de transport ; la ressource de transport ne peut être déchargée,
 - taux d'inutilisation par ressource de transport,
 - taux d'occupation par ressource de transport,
 - taux de panne par ressource de transport.

La difficulté rencontrée avec les critères globaux réside dans le fait que ces indicateurs traduisent la qualité de la conduite en fin de simulation ; ce sont des données historiques. On peut se poser la question de savoir quelle signification ils peuvent avoir en cours de simulation, lorsque l' "ordonnanceur" ou le "piloteur" de l'atelier doit prendre sa décision.

La simulation nous donne un grand nombre d'informations caractérisant l'état réel de l'atelier à un instant donné. Cet ensemble d'informations concerne les produits (valeur de la marge, nombre d'opérations restant à faire, durée opératoire restant à faire,...), les stocks (nombre de produits en attente, équivalence en temps opératoire des produits en attente, date d'entrée dans le stock par produit,...), les systèmes de transport et les machines (état de la ressource, date de libération,...). La difficulté est alors de définir parmi cette abondance d'informations, un ensemble de variables d'observations suffisamment pertinentes pour aider le décideur. Nous citerons ici les travaux de C. HERITIER qui, dans le cadre de sa thèse, utilise les techniques d'analyse de données pour trouver les corrélations entre ces différentes variables afin d'extraire les plus caractéristiques. [HERITIER - 90]

Mais nous avons également introduit la notion de critères locaux qui nous permet à un instant donné de la simulation, de situer la production vis à vis de ce qui est attendu.

L'hypothèse qu'un ordonnancement "classique" de la production a été fait en amont, nous donne un ensemble d'informations prédictives caractérisant la production et les objectifs à atteindre tels qu'ils ont été calculés pour un fonctionnement sans aléas :

- La date de fin attendue par produit,
- La durée du cycle de fabrication attendue par produit,
- La charge induite par le lancement qui détermine le taux d'occupation attendu par machine,

Nous mesurons ainsi la distance, à un instant donné, entre l'objectif et les performances réelles de l'atelier simulé.

- L'état d'avancement du produit dans sa gamme et la somme des durées opératoires restant à réaliser permettront, à un instant donné, d'évaluer le retard ou l'avance R_i du produit i , vis à vis de la date de fin attendue. ($i = \{1, \dots, n\}$ et n le nombre de produits)
- La date d'entrée dans l'atelier et la somme des durées opératoires restant à réaliser permettront, à un instant donné, d'évaluer la durée réelle du cycle de fabrication. Un produit qui a une durée de cycle supérieure à celle attendue va augmenter le niveau des en-cours, et engorger les stocks. Nous noterons EDC_i l'écart entre la durée de cycle attendue et la durée de cycle calculée à partir de la simulation.
- Le taux d'occupation de la machine ou du système de transport, calculé depuis le début de la simulation, permet, à un instant donné, d'évaluer l'incidence des aléas sur l'utilisation des ressources. Nous noterons ETO_j l'écart entre le taux d'occupation attendu et le taux d'occupation calculé à partir de la simulation. ($j = \{1, \dots, m\}$ et m nombre de machines).

Afin de visualiser les différentes composantes nécessaires lors de la prise de décision, nous placerons l'atelier et sa production à l'intérieur de la pyramide, à l'aide de formules simplifiées de coût liées aux différents critères. Voir figure I - 4.

Soit CR_i le coût estimé du retard par unité de temps pour le produit i

$$\text{Coût retard} = \text{Somme} (CR_i * R_i)$$
$$i = 1 \dots n$$

Soit CS_i le coût estimé de stockage par unité de temps pour le produit i

$$\text{Coût en-cours} = \text{Somme} (CS_i * EDC_i)$$
$$i = 1 \dots n$$

Soit CA_j le coût estimé d'amortissement de la machine ou du système de transport j

$$\text{Coût ressource} = \text{Somme} (CA_j * ETO_j)$$
$$j = 1 \dots m$$

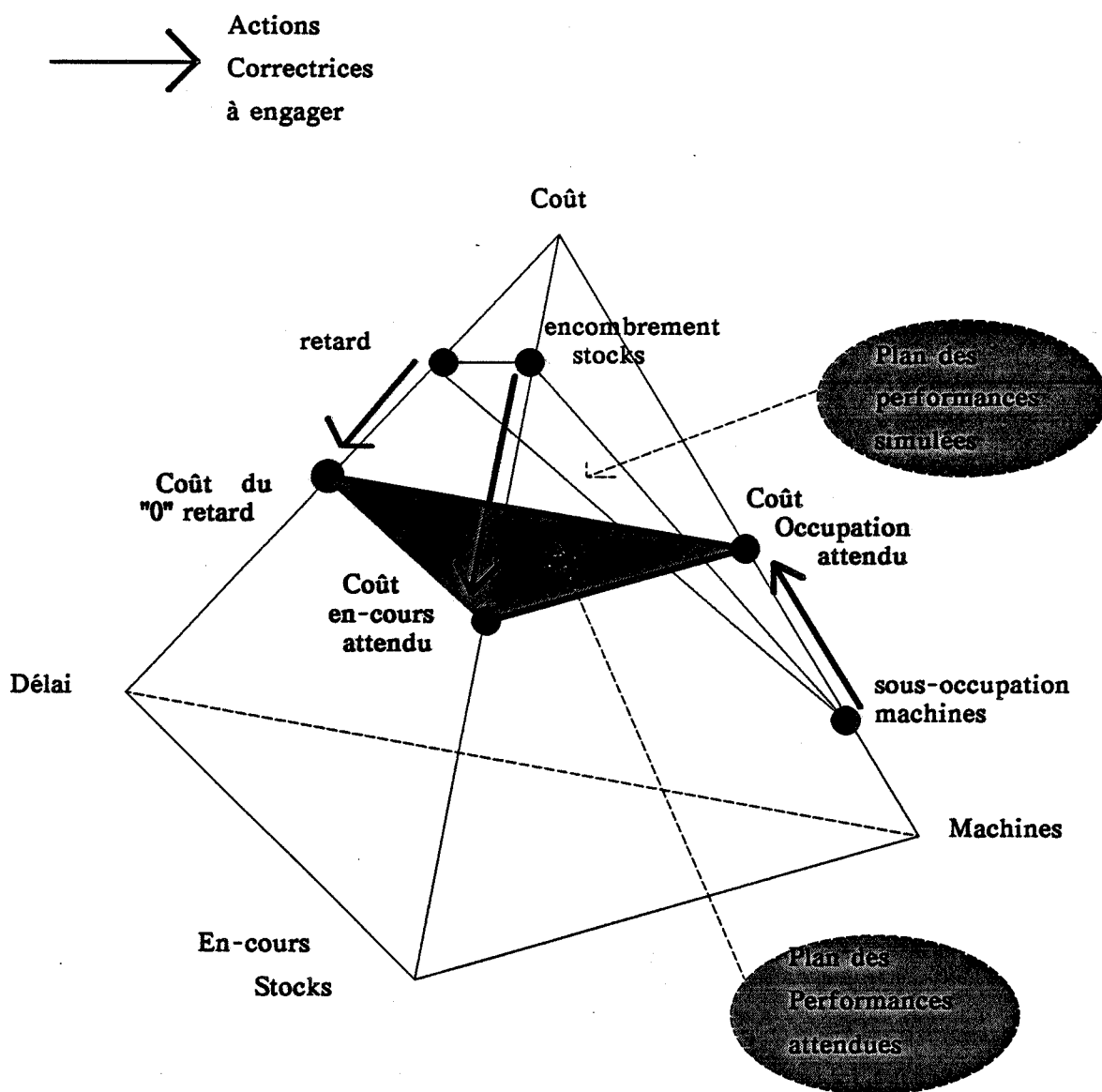


Figure I-4: Pyramide d'évaluation de la conduite

Le plan ainsi formé à l'intérieur de la pyramide aide le décideur à visualiser quelles sont la ou les composantes pénalisantes pour le système de production et donc de rechercher une action correctrice plus adaptée. Nous développerons dans les chapitres suivants les techniques et outils alors utilisés.

I - 4 CONCLUSION

Comme nous avons pu le voir dans ce chapitre, la conduite d'ateliers flexibles travaillant en Job-Shop n'est pas chose aisée, de par le grand nombre de paramètres à prendre en compte.

Les systèmes actuels sont plus des systèmes d'informations que des systèmes de conduite proprement dite, incluant à la fois la commande et l'ordonnancement. Par ailleurs, la recherche théorique ne permet pas de prendre en compte l'aspect "temps réel" de la conduite d'atelier lorsque le plan d'ordonnancement calculé par les méthodes d'optimisation se révèle impraticable par l'apparition d'événements aléatoires tels que les pannes. Dans la pratique, la conduite est alors réalisée de façon intuitive par les responsables d'atelier, parfois sans tenir compte des objectifs du système de production.

Il paraît donc important de donner à l'entreprise un outil d'aide à la décision pour la conduite temps réel de l'atelier.

Ce système doit mettre en évidence toutes les étapes du problème de la conduite :

- La définition des critères et des objectifs du système de gestion de production.
- La connaissance de l'état de l'atelier, et des points physiques, voire des instants, où il y a des décisions à prendre pour la conduite.
- L'utilisation d'un outil de simulation permettant de prévoir et tester l'évolution du système.
- La construction d'un ensemble de méthodes ou d'actions qui ont été au préalable testées par simulation, à proposer au décideur afin de l'aider dans sa prise de décision.

Nous développerons les deux principales composantes d'un tel système dans les chapitres suivants ; d'une part le choix d'un outil de simulation adapté, et d'autre part, les techniques d'ordonnancement à mettre en place pour la décision. Nous proposons un système d'aide à la gestion et au pilotage d'atelier, ou aide à la conduite, basé sur l'interaction dynamique de ces deux types d'outils.

CHAPITRE II

LA SIMULATION

UTILISATION DES RESEAUX DE PETRI COLORES TEMPORISES

II - 1 LES OUTILS DE SIMULATION

Face à la flexibilité accrue des moyens de production, on a vu apparaître une grande difficulté tant dans la phase de conception, que dans la phase de commande puis de conduite des ateliers flexibles, liée au grand nombre de paramètres et à l'interaction des différents composants du système de production.

On a alors cherché à élaborer des outils permettant de prévoir le fonctionnement de tels systèmes, en construisant des modèles logico-mathématiques de leur dynamique. La description du système ne pouvant se faire par des équations différentielles d'état, comme pour les systèmes continus, on a développé des modèles de description basés sur la logique des changements d'état, permettant de passer d'un état à un autre, appelés modèles à événements discrets.

La simulation, implantation informatique de ces modèles à événements discrets, constitue une aide pouvant intervenir à plusieurs niveaux :

- lors de la conception : elle permet de comparer les différents projets d'implantation des ressources en évaluant grossièrement leur fonctionnement, et en donnant des renseignements sur la production qu'ils autorisent.
- lors du dimensionnement : elle permet une spécification plus précise du projet retenu, en ajustant les paramètres structuraux comme les dimensions des stocks, les dimensions des systèmes de transport,...
- lors de l'implantation : elle permet de mettre au point les règles de conduite de l'atelier; règles de gestion des stocks, règles de priorité des systèmes de transport,...
- lors du pilotage : elle permet une aide à la conduite temps réel par évaluation des conséquences de certaines décisions lors de fonctionnement en mode perturbé (pannes, stock saturé, machine bloquée...).

II - 1 - 1 Définitions de modèles à événement discrets

Suivant la terminologie de la théorie des systèmes, un modèle à événements discrets se caractérise par : [BEL - 85] [CANALS - 86]

- des objets ou entités ; les pièces à fabriquer, les machines, les systèmes de transport,...
- des relations entre ces objets ; gamme de fabrication.

Un objet est caractérisé par un ou plusieurs attributs auxquels des valeurs peuvent être affectées. Les attributs fixes, associés à un objet, définissent la nature de l'objet, son type : machine, stock, ..., et ses caractéristiques : taille du stock, taux de panne de la machine, ... tandis que les valeurs des attributs variables définissent, à un instant donné, l'état de l'objet concerné à cet instant : état d'occupation d'une machine, position d'une pièce dans l'atelier, ...

L'ensemble de tous les états de tous les objets du système permet de définir l'état instantané du système.

On introduit également des notions décrivant l'évolution du système :

- activité : lorsqu'au cours d'un intervalle de temps, l'état d'un objet ne change pas, il est engagé dans une activité ; travail, panne, maintenance, attente. Un objet ne peut donc être engagé que dans une seule activité à la fois,
- événement : dès que l'état d'un objet varie, il y a présence d'un événement qui initialise une nouvelle activité,
- processus : succession d'un nombre fini d'activités qui concernent l'objet.

Une fois décrit l'état instantané du processus, nous décrivons la dynamique du système, c'est à dire les mécanismes qui gouvernent les changements d'état, en définissant pour chaque événement, les différents objets concernés et dans quelle activité ils vont s'engager.

On définit ainsi les contraintes de précédence entre les différentes activités : trajectoire décrite par les objets devant respecter ces contraintes.

Dans le cas des systèmes de production, les mécanismes définissant la logique de changement d'état sont de deux types ;

- les règles opératoires ou contraintes technologiques qui ne peuvent être modifiées dans le modèle car elle caractérisent le processus de fabrication du produit (ex : gamme). Elles limitent l'ensemble des transitions possibles entre états.
- les règles de gestion et de pilotage qui sont inconnues a priori, et que l'on va chercher à déterminer pour améliorer la fluidité des produits dans l'atelier et ainsi obtenir de meilleures performances du système de production. Elles permettent de choisir l'orientation à prendre à un instant donné, définissant ainsi les trajectoires des objets.

Les caractéristiques principales de la dynamique des modèles à événements discrets sont :

- le parallélisme des activités ; plusieurs activités peuvent avoir lieu en même temps
- la synchronisation ; une activité ne peut commencer que lorsque les objets concernés sont disponibles.
- le non déterminisme lié à la nécessité d'ajouter des règles de décision au modèle pour traduire certains changements d'état.

II - 1 - 2 Les différentes approches de la modélisation

Les différentes façons de traduire les mécanismes de changement d'état ont donné naissance à différents types de modèles, que nous décrivons sommairement.

II - 1 - 2 - a Approche par réseaux de files d'attente.

Ce type de modèle est basé sur une représentation du système de production par un réseau de files d'attente, où les machines et les systèmes de transport sont des serveurs, tandis que les pièces sont les clients. [GELENBE - 82]

L'état instantané est décrit par le nombre de clients en attente ou en fonctionnant devant chaque serveur.

La logique du changement d'état se traduit par l'affectation de probabilités de transition à chaque événement possible. Les probabilités de transition se calculent à partir des durées moyennes des activités, des règles opératoires, et des règles de pilotage des machines.

Dans l'état actuel des recherches, on ne sait trouver des solutions analytiques que si le réseau est à forme produit, soit décomposable sous la forme [CAVILLE - 82] :

$$P (n_1, \dots, n_M) = K * P_1 (n_1) * \dots * P_M (n_M)$$

avec n_i nombre de clients devant la machine i

M nombre de machines

$P (n_1, \dots, n_M)$ probabilité de l'état (n_1, \dots, n_M)

$P_i (n_i)$ probabilité marginale d'avoir n_i clients à la machine i

Ces modèles, basés sur l'analyse stochastique, donnent des valeurs d'indices de performance moyens calculés sur une période de fonctionnement suffisamment longue pour que les fréquences observées soient stables.

L'hypothèse de la non préemption, interdiction d'interrompre l'usinage d'une pièce au profit d'une autre pièce, commune aux systèmes de production, implique des conditions restrictives au niveau du modèle :

- Les durées opératoires doivent être exponentielles et indépendantes du type de pièce considéré. Cette condition nécessite un effort d'agrégation au niveau des données et est souvent peu réaliste.
- Les règles de gestion des files d'attente sont obligatoirement du type FIFO (premier arrivé premier sorti). Cette condition laisse peu de liberté dans l'étude de la conduite.

De plus, ce type de modélisation ne permet de prendre en compte, ni les temps de préparation et de réglage des machines entre deux opérations, ni les pannes, ni les blocages dus à des stocks saturés.

Pour des productions de petite et moyenne séries, ce type de modélisation paraît inadapté, de par la variation de la production sur la période étudiée qui rend non pertinente la notion de taux de production.

Cette approche semble toutefois intéressante lors de la première évaluation des différents projets d'atelier, lorsqu'il s'agit d'effectuer des comparaisons pour lesquelles les modèles à une seule classe de clients peuvent suffire.

II - 1 - 2 - b Approche par l'algèbre des dioïdes.

Ce type de modèle suppose une logique de changement d'état déterministe pour laquelle on peut définir, pour chaque ressource, la suite des activités dans laquelle elle sera engagée. [GONDRAN - 87]. Ceci a pour conséquence que l'on connaît, pour chaque pièce, la gamme spécifiant la route qu'elle suivra dans l'atelier, l'ordre dans lequel les pièces passeront sur les machines, et l'ordre de tous les déplacements des ressources de transport. Cette méthode est appliquée à des processus de production répétitifs ; c'est à dire que lorsqu'une ressource a participé à toutes les activités la concernant, elle est à nouveau disponible pour la première de ces activités [COHEN - 83].

On construit alors un graphe potentiel-tâches, cyclique, représentant le fonctionnement du système.

On traduit la condition de début d'une activité par l'équation matricielle :

$$X = X \otimes A \oplus U \otimes B$$

avec X vecteur des dates au plus tôt des activités, de composante x_j pour l'activité j ,

A matrice carrée des durées opératoires obtenue à partir du graphe potentiel-tâche,

$A_{ij} = d_i$ durée opératoire de l'activité i , si l'activité i précède l'activité j dans le graphe,

$A_{ij} = -\infty$ sinon.

U vecteur des dates de disponibilité des ressources de composante u_r pour la ressource r

B matrice rectangulaire,

$B_{ri} = 0$ si la ressource r commence son cycle d'activité par l'activité i ,

$B_{ri} = -\infty$ sinon.

et où l'opération \oplus est l'opération maximum

\otimes est un produit matriciel de type $(\max,+)$ de l'algèbre des dioïdes :

$$(M \otimes N)_{ij} = \max_k (M_{ik} + N_{kj})$$

La résolution de cette équation nécessite la recherche du plus long chemin entre activités qui permet la construction de la matrice A^* associée aux durées des plus longs chemins. L'obtention des dates de début au plus tôt de chaque activité x_i s'exprime sous la forme matricielle par l'équation :

$$X = U \otimes B \otimes A^*$$

Par ailleurs, l'expression récurrente relative au n -ième cycle d'activité permet d'obtenir les dates de disponibilité des ressources, fonction des dates de début au plus tôt des activités du cycle précédent.

$$U_n = X_{n-1} \otimes C$$

avec $C_{ir} = d_i$ si i est la dernière activité effectuée à l'aide de la ressource r

$$C_{ir} = -\infty \text{ sinon}$$

D'où l'expression récurrente qui définit l'évolution des dates de disponibilité des ressources, traduisant ainsi la dynamique du procédé :

$$U_n = U_{n-1} \otimes B \otimes A^* \otimes C$$

avec U_0 l'état initial du système décrit par la disponibilité des ressources.

Par la nécessité de définir complètement la logique de changement d'état sous forme de règles opératoires déterministes, ce type d'approche devient vite irréaliste. Comment planifier à l'avance, tous les déplacements d'un chariot filoguidé ?

L'intégration des différents emplacements disponibles d'un stock dimensionné en tant que ressources du système permet de mettre en évidence les blocages, mais présente l'inconvénient d'alourdir considérablement le modèle.

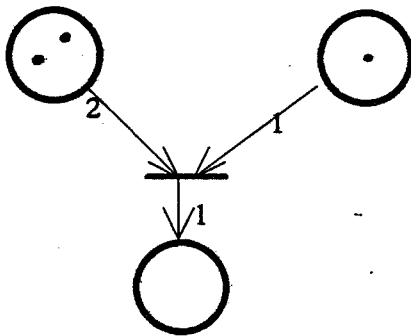
Contrairement aux modèles à file d'attente, ce type de modèle permet de prendre en compte dans la matrice A , des temps non opératifs comme : les temps de préparation sur les machines, les durées de transport...

Cette méthode permet par ailleurs, une première analyse des problèmes de conduite de l'atelier, à travers un choix d'ordonnancement au niveau de chaque machine par l'introduction de règles opératoires particulières.

II - 1 - 2 - c Approche par les Réseaux de Petri (RdP)

Un système de production manufacturier est, de par sa structure et son fonctionnement, est un système discret ou séquentiel.

Les RdP sont un outil particulier de modélisation pour systèmes discrets. Ce type de modèle est basé sur une représentation graphique, sous forme de réseau à deux types de noeuds, places et transitions qui composent la structure du modèle.



Structure du modèle RdP d'une opération d'assemblage

Par la suite, des conventions propres à la modélisation des systèmes de production permettent de réaliser l'interprétation du modèle pour décrire l'évolution du système discret :

- les activités sont représentées par des transitions. L'ensemble des transitions caractérise l'ensemble des événements susceptibles de faire évoluer le système.
- Les places du réseau représentent les états du système. Elles décrivent ainsi les conditions permettant à une activité de démarrer.
- les jetons permettent de représenter les ressources. La présence d'un jeton dans une place définit l'état de la ressource concernée. Un jeton dans une place peut signifier "machine occupée", "machine en panne", "machine libre", selon la place correspondante, traduisant ainsi la disponibilité de la ressource machine.

- les arcs sont la liaison logique entre une place et une transition ou entre une transition et une place. Une transition est reliée à une ou plusieurs places, par des arcs en amont pour les places d'entrées traduisant ainsi les conditions nécessaires de début de l'activité, et par des arcs en aval pour les places de sortie traduisant les effets de cette activité.
- Chaque arc est valué par un entier positif, appelé poids de l'arc, traduisant le nombre de jetons nécessaires à l'activité.

L'évolution du système se fait par le franchissement des transitions. Une transition est dite validée si chacune des places amont contient un nombre supérieur ou égal au poids de l'arc joignant cette place à la transition. Le franchissement de la transition validée consiste à retirer de chaque place d'entrée un nombre de jetons égal au poids de l'arc joignant cette place à la transition, et à déposer dans chaque place aval un nombre de jetons égal au poids de l'arc joignant la transition à la place.

Exemple de modélisation d'un stock géré suivant la politique FIFO :

Soit un stock de 5 emplacements, pouvant transporter 2 types de pièces différentes. Plusieurs producteurs alimentent ce stock en produits différents, tandis qu'à la sortie des consommateurs attendent ces produits. Les emplacements e1 et e4 sont libres, e2 et e5 sont occupés par un produit P1, tandis que e3 est occupé par un produit P2. La figure II-1 illustre le stock, tandis que la figure II-2 illustre le modèle RdP correspondant.

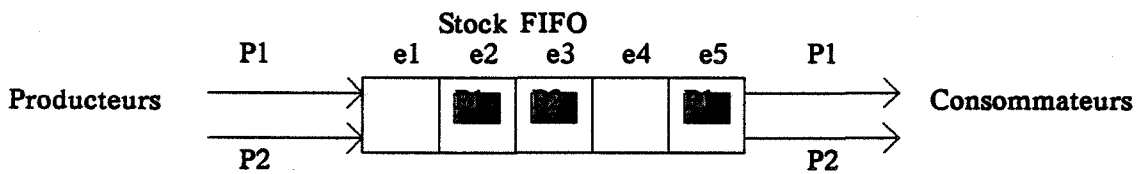


Figure II-1: Stock géré en FIFO

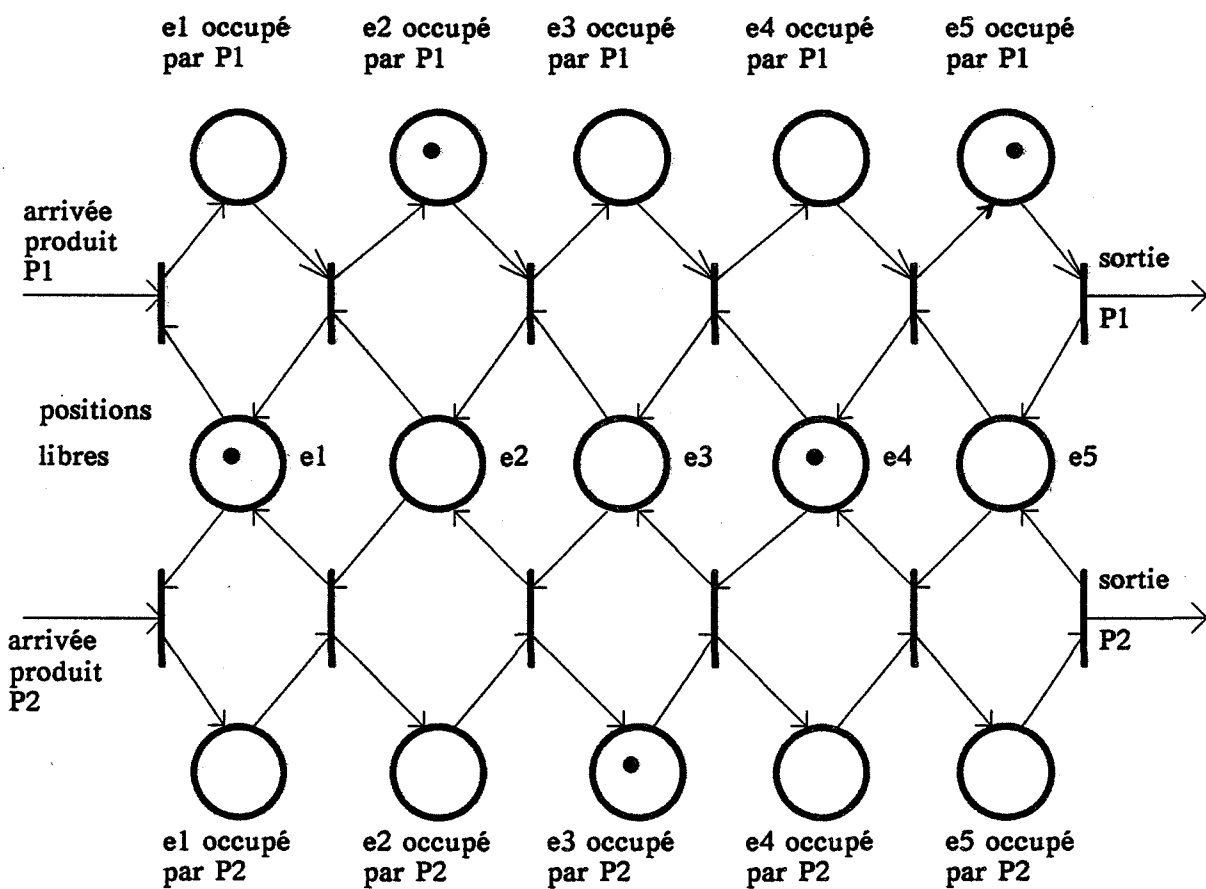


Figure II-2: modèle RdP du stock FIFO

Un produit P1 ne pourra entrer à l'emplacement e1 que si celui-ci est libre. Auquel cas, il occupera la position e1. De même, il ne pourra évoluer de la position e1 à la position e2 que si cette dernière est libre, ce qui nécessite l'évolution du produit P1 de e2 vers e3. Dans ce cas, l'occupation de e2 se traduit par la libération de e1 et par l'attribution de e2 au produit P1. On raisonne de la même façon pour le produit P2.

Sur cet exemple, nous pouvons visualiser les principaux avantages des RdP :

- la synchronisation,
- la représentation du parallélisme des activités.

La représentation par Réseau de Petri est non déterministe, au sens où elle ne résoud pas les indéterminations du modèle. La notion de conflit, cas où certaines places ont accès à plusieurs transitions, met en évidence un choix non encore résolu par le modèle.

Une solution est alors d'exprimer la règle de décision qui sera utilisée pour le choix de la transition, à travers une modélisation de la règle. L'exemple précédent illustre un exemple de modélisation de règle de décision en FIFO, mais il est beaucoup plus difficile de représenter des règles plus complexes comme le choix suivant la date de fin désirée EDD, ou le choix suivant la plus courte durée opératoire SPT. Par ailleurs, cette modélisation alourdit considérablement le réseau, pénalisant la lisibilité du modèle.

L'intérêt graphique de cette approche est complété par la possibilité d'une validation formelle issue des propriétés mathématiques du graphe associé.

On définit [BRAMS - 83] un réseau autonome par l'ensemble $R = (P, T, \text{Pré}, \text{Post}, M_0)$

- P est l'ensemble fini des places tel que $\text{Card}(P) = n$
- T est l'ensemble fini des transitions tel que $\text{Card}(T) = m$ et $P \cap T = \emptyset$
- Pré est la fonction incidence avant, $\text{Pré}: P * T \rightarrow [N$

L'application Pré donne le poids des arcs placés en amont des transitions.

- Post est la fonction incidence arrière, $\text{Post}: P * T \rightarrow [N$

L'application Post donne le poids des arcs placés en aval des transitions.

Les applications Pré et Post sont des matrices $n*m$ (n nombre de places, m nombre de transitions), dont les éléments sont des termes $\text{Pré}(P_i, T_j)$ et $\text{Post}(P_i, T_j)$.

- M_0 est le marquage initial, $M_0: P \rightarrow [N$

Il définit le contenu de chacune des places à l'état initial.

Soit la matrice d'indice W associée au graphe telle que :

$$\forall T_j \in T, \forall P_i \in P, W(P_i, T_j) = \text{Post}(P_i, T_j) - \text{Pré}(P_i, T_j)$$

Une transition T_j est validée pour un marquage M si et seulement si :

$$\forall P_i \in P, M(P_i) \geq \text{Pré}(P_i, T_j)$$

Si une seule transition T_j est validée pour le marquage M , le vecteur s^* (dim m), exprimant la séquence de tir de transitions à partir du marquage M est de la forme $(0, 0, \dots, 0, T_j, 0, \dots, 0)$. Le franchissement de T_j permet d'obtenir le nouveau marquage M' :

$$M' = M + W \cdot s^*$$

Cette équation fonde l'analyse des réseaux par les techniques d'algèbre linéaire. Nous notons une propriété particulière dont nous illustrerons plus loin une application.

Soit le vecteur de dim n , $X = U$ une des solutions de l'équation : $X^T \cdot W = 0$

en multipliant l'équation d'évolution du marquage par U^T on obtient :

$$U^T \cdot M' = U^T \cdot M + U^T \cdot W \cdot s^*$$

d'où l'invariant linéaire de place : $U^T \cdot M' = U^T \cdot M$

On montre que l'on peut valider un réseau c'est à dire vérifier que le graphe obtenu correspond bien aux spécifications du système.

La validation permet l'analyse des propriétés du système :

- Le RdP est-il vivant ? Cette propriété permet de vérifier que le modèle obtenu est sans blocage, donnant ainsi des informations sur la viabilité du système lui-même.
- Le RdP est-il borné ? Cette propriété permet de connaître le nombre maximum de jetons contenus dans une place lors du fonctionnement, limite du système lui-même. Cette propriété peut être utilisée lors du dimensionnement d'un atelier, pour déterminer le nombre de palettes du système de transport, la taille d'un stock...
- Le RdP est-il propre ? Cette propriété permet de vérifier que le modèle est cyclique et donc qu'il peut retrouver son état initial à partir de n'importe quelle configuration, assurant ainsi l'autonomie du système.

Nous citerons trois techniques de validation :

- par énumération des marquages avec la construction du graphe des marquages accessibles. Cette méthode présente l'inconvénient d'être vite très lourde à implanter, dès lors que le nombre de transitions devient important ; on assiste alors à une explosion combinatoire du graphe.
- par réduction suivant des règles bien définies qui conservent les propriétés du réseau : élimination des places implicites, agglomération des séquences, suppression des transitions identiques.
- par l'interprétation des invariants linéaires de marque qui permettent de déterminer le caractère borné de certaines places et le caractère vivant du réseau.

L'intérêt de ces techniques de validation est leur caractère formel qui permet de tirer des propriétés du système par simple observation du graphe, avant même toute autre analyse.

Toutefois, nous devons noter que les RdP deviennent inutilisables dès lors que le nombre de places et de transitions devient trop grand, ce qui est le cas dans la modélisation des systèmes de production. Sur l'exemple simple présenté précédemment, on constate en effet que le nombre de positions du stock et le nombre de produits pouvant être stockés, font vite exploser le graphe en le rendant illisible.

Les Réseaux de Petri généralisés ou Réseaux de haut niveau remédient à ces limites, nous citerons :

- les Réseaux Places/Transitions [REISIG - 86] pour lesquels les places sont des compteurs,
- les Réseaux à Prédicat [GENRICH - 86] pour lesquels on associe une information à chaque jeton et un prédicat, condition logique, à chaque transition,
- les Réseaux Colorés [JENSEN - 80] pour lesquels on associe des informations à chaque jeton appelées couleur, et des fonctions aux arcs. Nous développerons plus en détail cette extension des réseaux de Petri dans les paragraphes suivants, puisque c'est l'approche que nous avons décidé d'utiliser.
- les Réseaux à Structure de Données [SIBERTIN - 85] pour lesquels les jetons sont remplacés par des structures de données.
- les Réseaux à Objets [SIBERTIN - 88] pour lesquels les jetons sont remplacés par des objets et les transitions par des structures comprenant les conditions, les actions, et les règles d'émission des objets.

Les réseaux dits généralisés présentent alors l'inconvénient d'être beaucoup plus difficiles à valider, de par la complexité des structures d'informations mises en place. Les réseaux de Petri colorés ont toutefois fait l'objet de travaux sur leur validation par la recherche des invariants, ce qui justifie en partie notre choix ultérieur [ALLA - 84].

Par ailleurs, les RdP autonomes présentent l'inconvénient de ne pas représenter la notion de temps dans l'évolution du système, puisque seule la notion d'état intervient. Il devient alors difficile de prendre en compte le comportement dynamique du système lié aux durées des opérations ou aux temps de transport par exemple.

Les réseaux de Petri Temporisés apportent une réponse à ce problème en ajoutant la notion de temporisation sur les places qui rend indisponible le jeton dans la place considéré pendant toute la durée de la temporisation. Une approche duale existe, qui place les temporisations sur les transitions pour traduire la durée de l'activité correspondante. Mais là encore, cette extension ne permet plus de valider le modèle par les techniques formelles [CHRETIENNE - 83].

Le fonctionnement du réseau devient alors non-autonome, conditionné et synchronisé par le comportement temporel des différentes ressources qu'il représente.

Il semble donc que les réseaux de Petri puissent être utilisés dans un premier temps, lors d'une première phase d'étude, pour l'aide au dimensionnement par les techniques de validation formelle, puis implantés informatiquement comme outil de simulation sous réserve d'utiliser les réseaux de Petri Temporisés.

Cette dernière constatation nous conduit directement aux outils de simulation permettant justement un suivi temporel de la dynamique des systèmes de production.

II - 1 - 3 Les différentes approches de la simulation

Face aux limites des méthodes analytiques présentés précédemment, on a cherché à reproduire pas à pas, événement par événement, l'évolution de l'état du système dans le temps, en traduisant la logique des changements d'état [CAVILLE - 88]. On a alors développé un ensemble de modèles à événements discrets pour la simulation.

On distingue alors trois façons de décrire la logique de changements d'état [BEL - 85] :

- par événements,
- par activités,
- par processus.

II - 1 - 3 - a Description par événements

Cette description se décompose en deux étapes :

- l'énumération des différents types d'événements ou changements d'état pouvant être rencontrés au cours de l'évolution du système,
- la modélisation de la logique de changements d'état, par la définition, pour chaque type d'événements, des conditions sur l'état qui conduisent à l'occurrence de l'événement, et des changements d'état correspondants.

L'étude du comportement dynamique du système consiste alors à exécuter dans l'ordre chronologique les logiques de changements d'état associées à chaque événement, à la date à laquelle celui-ci se produira.

L'évolution de la simulation se fait par "propagation d'événements" ; chaque événement engendre un ou plusieurs événements fils, auxquels on associe une date d'occurrence. Les événements sont rangés par ordre d'occurrence dans un échéancier. L'avance du temps se fait par saut d'une date d'événement à la date de l'événement suivant.

La manipulation et l'entretien de l'échéancier se font par un ensemble de programmes et de structures de données appelé noyau de synchronisation, qui permet la remise en cause de l'ordre préétabli à chaque apparition et traitement d'un nouvel événement.

Nous citerons des logiciels de simulation comme SIMSCRIPT II ou GPSS FORTRAN qui utilisent cette description.

II - 1 - 3 - b Description par activités

Cette description est duale de la précédente ; au lieu de répertorier tous les types d'événements, on répertorie les types d'activités rencontrés dans le fonctionnement du système. La modélisation du changement d'état se fait en précisant les conditions sur l'état du système, nécessaires au début et à la fin d'une activité.

L'avance du temps se fait en général à l'aide d'une horloge : à chaque pas de temps, on examine, pour chaque activité, si les conditions sont remplies et permettent de changer d'activité.

Ceci implique que l'on est obligé de tester toutes les activités à chaque pas de temps, ce qui peut être très pénalisant pour les temps de calcul, si le pas de temps est mal adapté. De plus, l'ordre dans lequel les activités sont testées peut fortement influencer l'évolution du système.

Cette description a été peu utilisée pour le développement de logiciels de simulation, Nous citerons le progiciel TZAR qui semble être basé sur ce principe.

II - 1 - 3 - c Description par processus

Cette description ne décrit plus la logique de changements d'état relative à chaque événement ou à chaque activité, mais plutôt la logique de changement d'état d'une séquence d'événements prédéterminés ou processus.

Le choix des séquences d'événements doit être suffisamment caractéristique pour que celles-ci puissent être reproduites un grand nombre de fois, à différents instants, pour différents objets. Un exemple de processus utilisé peut être la séquence d'événements correspondants à l'évolution d'une pièce dans une file d'attente, limitée par les événements arrivée d'une pièce dans la file d'attente et demande d'usinage.

La modélisation nécessite donc la définition des différents processus, et pour chacun d'eux, la description de la logique de changement d'état associée décrivant le cheminement, événement par événement, de l'entrée à la sortie du processus. La connexion des différents processus entre eux, et la spécification de leurs interactions donnent ensuite la description du système complet. Les conflits ou la synchronisation entre processus peuvent être pris en compte par des règles du type "attendre jusqu'à un événement extérieur au processus". Chaque processus peut ainsi commander des sous-processus, attendre, ou faire d'autres processus jusqu'à ce que certaines conditions soient remplies.

Cette description a été la plus utilisée pour le développement de logiciels de simulation. La plupart des simulateurs, comme QNAP2, SLAM, PAWS, SIMAN, utilisent cette description et proposent des processus pré-programmés sous forme de primitives standards, afin de faciliter la modélisation et la programmation. Par contre, la modélisation des lois de gestion ou de fonctionnements particuliers se fait dans tous les cas par des sous-programmes Fortran, Pascal ou C .

II - 1 - 4 Conclusion

Comme nous avons pu le voir, les méthodes de modélisation analytique comme les méthodes simulatoires permettent de représenter le fonctionnement du système physique, avec la prise en compte des règles opératoires, du parallélisme et des problèmes de synchronisation.

Tant dans la phase de conception que dans la phase de dimensionnement, elles apportent une aide à la décision, en apportant un grand nombre d'informations sur le comportement dynamique de l'atelier, face à un plan de lancement donné.

Toutefois, force est de constater qu'il est beaucoup plus difficile de représenter les règles de gestion qui très souvent font appel à des modules procéduraux externes, où à des représentations particulières qui alourdissent le modèle. De même, tous les outils de simulation ne permettent pas la prise en compte des pannes et de leur traitement par des règles de gestion appropriées.

L'utilisation de la simulation pour apporter une aide au pilotage et à l'ordonnancement d'atelier flexible nécessite une représentation souple, facilement modifiable par l'utilisateur, des règles de décision et de gestion qu'il veut évaluer.

Trop souvent l'implémentation informatique des simulateurs masque des points de décision, par le traitement séquentiel de l'information, et ne permet pas d'en modifier le traitement. Les règles de priorité et les règles de gestion des stocks doivent être parfaitement paramétrables pour permettre à l'utilisateur de tester les différentes solutions qui s'offrent à lui.

Nous avons donc cherché à développer un outil de modélisation puis de simulation qui permette de mettre en évidence les différents points de décision du système, sans les résoudre. Ces décisions relèvent en effet, d'un niveau décisionnel supérieur où les objectifs et critères de gestion sont à prendre en compte.

L'idée est donc de séparer ces deux niveaux en utilisant, pour chacun d'eux, les outils appropriés ; la simulation pour représenter le fonctionnement de l'atelier et mettre en évidence les points de décision, un module décisionnel pour prendre ces décisions.

II - 2 MODELISATION PAR RESEAUX DE PETRI COLORES

Dans le monde industriel, la commande des différents systèmes automatisés ; machines à commande numérique, chariots filoguidés, convoyeurs, par des automates programmables a contribué à l'uniformisation des langages de programmation.

Le Grafcet est désormais largement utilisé dans le monde industriel, pour décrire les fonctions locales de commande de ces différentes ressources. Il permet d'obtenir un modèle très lisible, qui met en évidence le parallélisme et les synchronisations.

Le marquage d'un grafcet est de type booléen, une étape est active ou inactive, une étape active ré-activée par l'arrivée d'un nouveau jeton reste active mais voit les deux jetons fusionner. Un tel marquage permet de représenter à tout instant l'état atteint, mais ne permet pas d'associer une signification physique au jeton. L'information portée étant trop pauvre, on ne peut pas, par exemple, traduire le suivi d'une pièce dans l'atelier. On ne peut donc envisager d'utiliser le Grafcet comme langage de modélisation des systèmes à événements discrets.

L'intérêt principal d'une modélisation par Réseaux de Petri se justifie alors par le partage des mêmes concepts et principes que le Grafcet, puisque historiquement, le Grafcet est issu de l'approche par RdP. Tandis que la simulation par RdP peut être utilisée en temps réel pour apporter une aide à la décision, l'utilisation du grafcet semble dédiée à la commande proprement dite du système séquentiel. La volonté d'homogénéisation des outils de modélisation, entre le niveau commande et le niveau simulation, nous a donc conduit à choisir l'approche par RdP, pour représenter le comportement dynamique des systèmes de production étudiés. De plus, le réseaux de Petri peut, à terme, être modèle de commande, au sens de MERCIER des ROCHETTES [MERCIER - 88], ce qui renforce l'intérêt d'une telle approche en augmentant la cohérence entre les niveaux de simulation et de commande.

II - 2 - 1 Les Réseaux de Petri Colorés

Mais l'utilisation des Réseaux de Petri pour la modélisation des ateliers flexibles se heurte à deux contraintes importantes : la généralité et la concision. La généralité du modèle obtenu doit permettre la prise en compte de la diversité des machines, des pièces, des outils et de leurs caractéristiques sans qu'il soit besoin de reprendre sa structure chaque fois qu'une modification est apportée au système étudié. Cette propriété permettra par exemple, d'utiliser une même structure graphique du modèle lors de la phase de dimensionnement. La concision est le garant de la lisibilité du modèle obtenu, sans supprimer d'informations.

Pour répondre à ces deux contraintes, nous avons choisi d'utiliser les Réseaux de Petri Colorés (RdPC) qui sont une abréviation et une généralisation des RdP.

Nous présenterons dans un premier temps une définition informelle qui permet de comprendre les grands principes de façon intuitive, puis la définition formelle de l'outil de modélisation.

II - 2 - 1 - a Définition informelle

L'idée intuitive des RdPC peut être illustrée par la volonté de réduire la taille du modèle RdP "classique" de l'application, par "pliage" suivant un plan d'information particulier, tout en veillant à la conservation de la totalité de l'information.

Considérons pour cela le modèle RdP d'un stock géré suivant la politique de gestion FIFO, présenté au chapitre II dans la figure II - 2. Nous avons constaté que la taille du stock et le nombre de produits distincts pouvant être stockés, étaient vite pénalisants pour la lisibilité du réseau. Soit e le nombre de positions du stock, n le nombre de produits distincts, le nombre de places du réseau p vérifie : $p = (n+1) \cdot e$

Une première solution consiste à plier le graphe suivant le plan horizontal qui distingue les différents produits. La figure II - 3 donne le réseau obtenu :

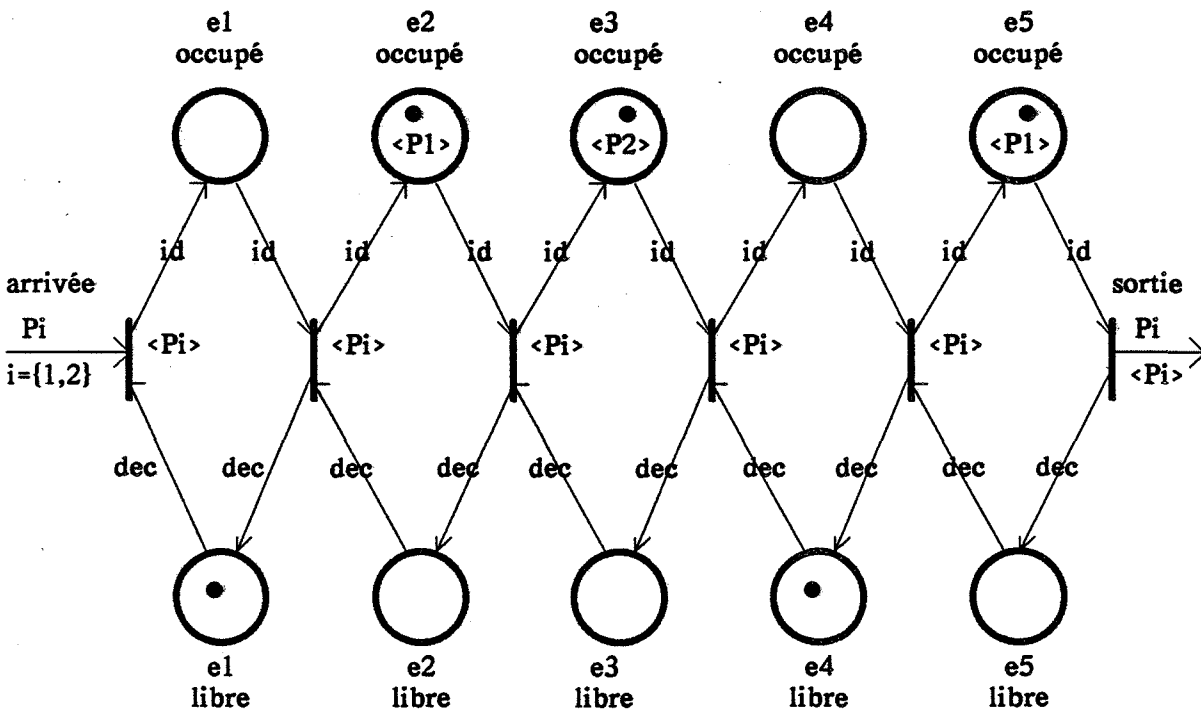


figure II-3: RdP du stock FIFO coloré suivant les produits

L'information sur la nature du produit (n° de série) pour être conservée, est alors portée par un indice de couleur caractérisant le jeton dans les places Stock Occupé.

Un jeton de la place e_i occupé peut avoir plusieurs valeurs d'indice de nature du produit pour distinguer celui qui occupe effectivement l'emplacement i du stock. La transition permettant l'évolution du produit est alors tirable pour plusieurs valeurs d'indice de nature du produit différents. Pour traduire les différentes possibilités de franchissement de la transition, on introduit la notion d'ensemble de couleurs associé à la transition.

Mais il faut alors indiquer comment va évoluer cet indice lors du franchissement de la transition, caractérisant ainsi l'évolution du produit d'un emplacement à l'emplacement suivant du stock. Pour cela, on introduit la notion de fonction associée à l'arc.

Une transition du réseau est validée et tirable par rapport à chacune des couleurs contenues dans la place e_i occupée pourvu que la place e_{i+1} libre soit marquée. L'information "couleur" est dans ce cas, conservée dans le tir de la transition, ce que l'on traduit par une fonction identité notée id .

Ici, il est important de conserver de façon identique, cette information tout au cours de l'évolution, la fonction Identité est donc associée aux arcs décrivant l'avancée du produit dans le stock. Mais il pourrait également être nécessaire de changer la couleur d'un jeton au cours d'un franchissement de façon à représenter une transformation de l'information. La fonction décoloration, notée dec , permet de supprimer l'information portée par la couleur lorsque celle-ci n'a plus de signification physique $dec(\langle pi \rangle) = \langle . \rangle$. On introduit alors des fonctions plus élaborées dont nous donnerons des exemples ultérieurement.

De la même façon, on pourrait construire un pliage suivant le plan vertical caractérisant la position dans le stock.

En résumé, on peut dire que les éléments nouveaux apportés par l'outil RdPC sont les suivants [ALLA - 87]:

- les jetons sont identifiés par des couleurs. L'information n'est plus portée uniquement par les places, mais par l'association place-couleur, afin de réduire la taille des modèles obtenus. Un jeton à l'intérieur d'une place est caractérisé par un n-uplet regroupant les informations attributs le définissant. Par exemple, au jeton déposé dans une place caractérisant l'état d'occupation d'un stock, peut être associé le doublet (numéro pièce, emplacement dans le stock).
- un ensemble de couleurs est associé à chaque transition indiquant les différentes possibilités de tir.
- une fonction est associée à chaque arc indiquant le cas échéant, la transformation de couleur appliquée à la couleur de tir. La notion de poids de l'arc est conservée pour indiquer le nombre de jetons identiques nécessaires à la validation de la transition puis engagés dans le tir.

Les fonctions traduisent la dépendance fonctionnelle entre les places et les transitions.

La représentation des RdPC est la suivante:

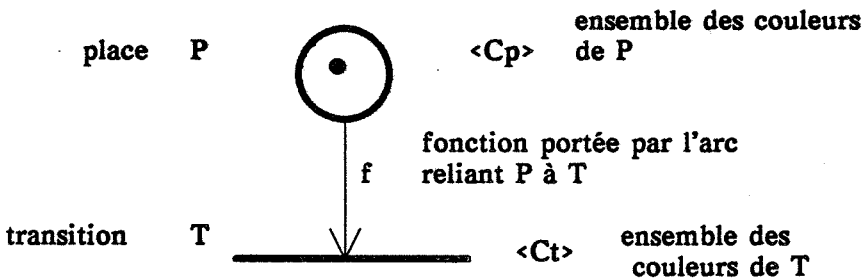


figure II-4: Représentation des RdPC

Une transition est validée par rapport aux différentes couleurs de l'ensemble C_t . Une transition validée est franchie. Le franchissement conduit à une transformation du marquage initial et des couleurs dans les places concernées, par le biais de la fonction $f(C_t) = C_p$. Cette évolution traduit le changement d'état du système décrit.

Si l'on reprend l'exemple du stock géré en FIFO, le réseau obtenu par coloration complète est décrit par la figure II - 5 :

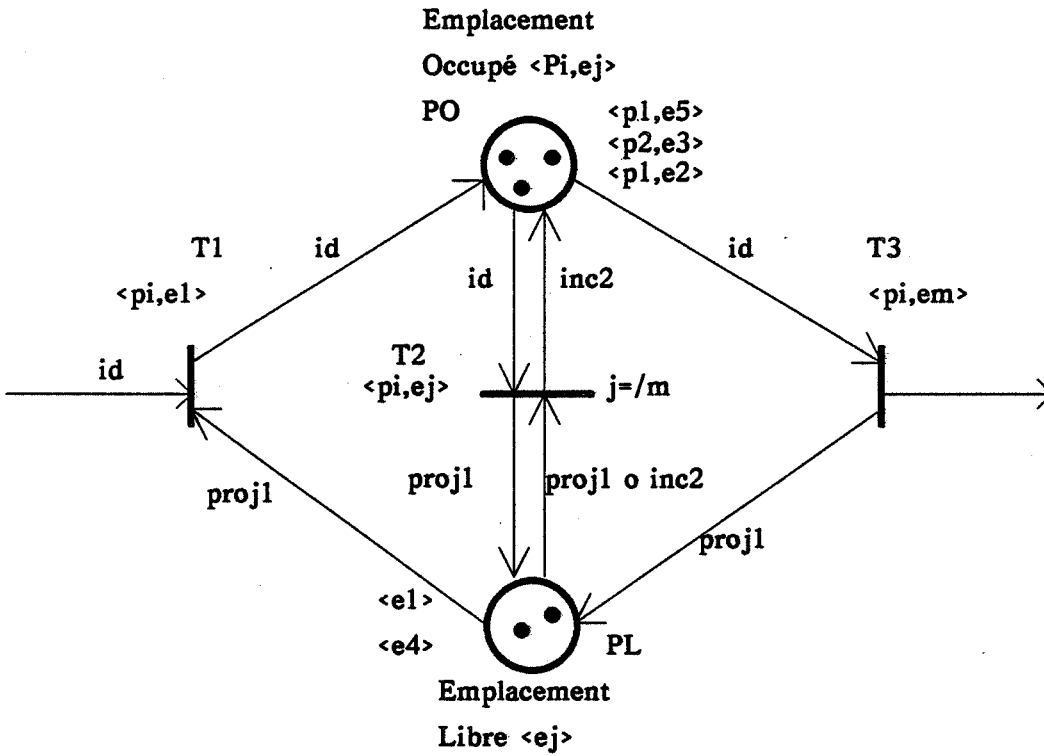


figure II-5: Modèle RdPC du stock FIFO

Dans ce RdPC, les couleurs sont des doublets $\langle pi, ej \rangle$ qui représentent l'emplacement ej du stock occupé par le produit pi avec :

$i \in \{1, \dots, n\}$ n nombre de produits, $n \in \mathbb{N}$, dans notre exemple $n = 2$

et $j \in \{1, \dots, m\}$ m nombre d'emplacements du stock, $m \in \mathbb{N}$, dans notre exemple $m = 5$

La transition T1 : entrée d'un produit i quelconque dans l'emplacement 1 du stock, est définie par l'ensemble de couleurs $\langle pi, e1 \rangle$.

La transition T2 : évolution d'un produit i quelconque de l'emplacement j occupé à la emplacement suivant, est définie par l'ensemble de couleurs $\langle pi, ej \rangle$ avec $j \neq m$.

La transition T3 : sortie d'un produit i quelconque venant de l'emplacement m du stock, est définie par l'ensemble de couleurs $\langle pi, em \rangle$.

La fonction projection $\text{proj}^1 (\langle p_i, e_j \rangle) = \langle e_j \rangle$ permet une projection sur la 1ère composante.

La fonction incrément $\text{inc}^2 (\langle p_i, e_j \rangle) = \langle p_i, e_{j+1} \rangle$ permet de trouver le successeur de la 2ème composante modulo m .

L'utilisation de fonctions composées permet de réduire l'ensemble des fonctions du réseau et de leur garder une signification physique. Par ailleurs, elle permet des simplifications dans le calcul des invariants, lors de l'étape de validation formelle du modèle.

Le marquage initial caractérise l'état initial du système. Dans l'état initial décrit par la figure II-5, la place Emplacement stock Occupé contient trois jetons de couleur $\langle p_1, e_2 \rangle$, $\langle p_1, e_5 \rangle$ et $\langle p_2, e_3 \rangle$ qui traduit le fait que les emplacements e_2 et e_5 sont occupés par des produits 1 et que l'emplacement e_3 est occupé par un produit 2. La place Emplacement stock Libre contient deux jetons $\langle e_1 \rangle$ et $\langle e_4 \rangle$ qui indiquent que les emplacements 1 et 4 du stock sont libres.

A titre d'exemple, nous constatons que la transition T2 est validée par rapport à la couleur $\langle p_2, e_3 \rangle$ car d'une part, la place Emplacement stock Occupé contient la couleur

$$\text{Id} (\langle p_2, e_3 \rangle) = \langle p_2, e_3 \rangle ,$$

traduisant le fait que le produit p_2 occupe l'emplacement 3 du stock,

et d'autre part, la place emplacement Stock Libre contient la couleur

$$\text{Proj}^1 . \text{Inc}^2 (\langle p_2, e_3 \rangle) = \text{Proj}^1 (\langle p_2, e_4 \rangle) = \langle e_4 \rangle ,$$

traduisant le fait que l'emplacement 4 du stock est disponible.

Le tir de T2 dépose un jeton de couleur $\text{Inc}^2 (\langle p_2, e_3 \rangle) = \langle p_2, e_4 \rangle$ dans la place Emplacement stock Occupé et retire un jeton $\text{Proj}^1 (\langle p_2, e_3 \rangle) = \langle e_3 \rangle$ de la place Emplacement stock Libre. Le produit p_2 occupe maintenant l'emplacement 4 du stock, tandis que l'emplacement 3 est libre.

Nous avons ainsi représenté l'évolution d'un produit dans un stock FIFO par un réseau de Petri comprenant 2 places et 3 transitions. Le cas général du stock à m emplacements et pouvant accepter n produits de types différents conserve la même structure graphique du modèle ; nombre de places, nombre de transitions. Seuls les ensembles de couleurs des transitions et les domaines de définition des fonctions sont modifiés pour tenir compte du changement de n ou de m .

Grâce à la concision de l'information, la généralisation du modèle est donc possible sans changer la structure du graphe, ce qui facilite l'exploitation et la lisibilité du réseau.

II - 2 - 1 - b Définition formelle [JENSEN - 82]

Un réseau de Petri coloré (RdPC) est un sextuplet

$$\text{RDPC} = \{ P, T, C, \text{Pré}, \text{Post}, M_0 \}$$

où P est un ensemble fini de places avec $\text{Card}(P) = n$

T est un ensemble fini de transitions avec $\text{Card}(T) = m$ et $P \cap T = \emptyset$

C est la fonction couleur

$C(P_i)$ caractérise l'ensemble des couleurs de la place P_i , nous noterons C_i

$C(T_j)$ caractérise l'ensemble des couleurs de la transition T_j , nous noterons C_j

Pré est la fonction incidence avant définie sur $P \times T$ telle que :

$$\forall P_i, T_j \text{ Pré}(P_i, T_j) \text{ est une fonction de } C_j \rightarrow C_i[N]$$

avec $C_i[N]$ ensemble de toutes les combinaisons linéaires de C_i dans l'ensemble des entiers naturels

Post est la fonction incidence arrière définie sur $P \times T$ telle que :

$$\forall P_i, T_j \text{ Post}(P_i, T_j) \text{ est une fonction de } C_j \rightarrow C_i[N]$$

avec $C_i[N]$ ensemble de toutes les combinaisons linéaires de C_i dans l'ensemble des entiers naturels

M_0 marquage initial est une fonction définie sur P telle que

$$\forall P_i \text{ } M_0(P_i) : C_i \rightarrow [N, M_0(P_i) \in C_i[N]$$

le marquage $M(P_i)$ donne le nombre de jetons de chaque couleur dans la place P_i

Les fonctions Pré et Post sont des transformations linéaires, elles peuvent être mises sous la forme de matrices dont les éléments sont les fonctions matricielles portées par les arcs.

Dans l'exemple du stock FIFO vu précédemment on a :

$$\text{Pré} \begin{array}{l} \text{PO} \\ \text{PL} \end{array} \begin{array}{ccc} \text{T1} & \text{T2} & \text{T3} \\ \left[\begin{array}{ccc} 0 & \text{id} & \text{id} \\ \text{proj}^1 & \text{proj}^1 \cdot \text{inc}^2 & 0 \end{array} \right] \end{array}$$

$$\text{Post} \begin{array}{l} \text{PO} \\ \text{PL} \end{array} \begin{array}{ccc} \text{T1} & \text{T2} & \text{T3} \\ \left[\begin{array}{ccc} \text{id} & \text{inc}^2 & 0 \\ 0 & \text{proj}^1 & \text{proj}^1 \end{array} \right] \end{array}$$

avec 0 la fonction nulle

id la fonction identité , à titre d'exemple illustratif pour $n = 2$ et $m = 5$:

$$\text{Pré}(\text{PO}, \text{T3}) \begin{array}{ccc} & \langle p1, e5 \rangle & \langle p2, e5 \rangle \\ \begin{array}{l} \langle p1, e1 \rangle \\ \langle p2, e1 \rangle \\ \dots \\ \langle p1, e4 \rangle \\ \langle p2, e4 \rangle \\ \langle p1, e5 \rangle \\ \langle p2, e5 \rangle \end{array} & \left[\begin{array}{cc} 0 & 0 \\ 0 & 0 \\ \dots & \dots \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{array} \right] \end{array}$$

proj^1 la fonction projection, à titre d'exemple pour $n = 2$ et $m = 5$:

$$\text{Pré}(\text{PL}, \text{T1}) \begin{array}{ccc} & \langle p1, e1 \rangle & \langle p2, e1 \rangle \\ \begin{array}{l} \langle e1 \rangle \\ \langle e2 \rangle \\ \langle e3 \rangle \\ \langle e4 \rangle \\ \langle e5 \rangle \end{array} & \left[\begin{array}{cc} 1 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{array} \right] \end{array}$$

inc² la fonction incrément, à titre d'exemple pour n= 2 et m = 5 :

	$\langle p1,e2 \rangle$	$\langle p2,e2 \rangle$	$\langle p1,e3 \rangle$	$\langle p2,e3 \rangle$	$\langle p1,e4 \rangle$	$\langle p2,e4 \rangle$
$\langle p1,e1 \rangle$	0	0	0	0	0	0
...						
Post(PO,T2) $\langle p2,e2 \rangle$	0	0	0	0	0	0
$\langle p1,e3 \rangle$	1	0	0	0	0	0
$\langle p2,e3 \rangle$	0	1	0	0	0	0
$\langle p1,e4 \rangle$	0	0	1	0	0	0
$\langle p2,e4 \rangle$	0	0	0	1	0	0
$\langle p1,e5 \rangle$	0	0	0	0	1	0
$\langle p2,e5 \rangle$	0	0	0	0	0	1

Une transition Tj est validée par rapport à une couleur $\langle ck \rangle$ de son ensemble de couleurs associé Cj si le marquage courant M vérifie :

$$\forall Pi \text{ place d'entrée de } Tj, M(Pi) (\langle ck \rangle) \geq \text{Pré} (Pi, Tj) (\langle ck \rangle)$$

Une transition validée est franchie. Le franchissement se traduit par le retrait de $\text{Pré}(Pi, Tj) (\langle ck \rangle)$ jetons de chaque place Pi amont de Tj et par le dépôt de $\text{Post}(Pi, Tj) (\langle ck \rangle)$ jetons dans chaque place aval Pi de Tj.

Soit M'(Pi) le marquage obtenu à partir de M(Pi) après le franchissement de la transition Tj, on a :

$$M'(Pi) = M(Pi) + (\text{Post} (Pi, Tj) - \text{Pré} (Pi, Tj))$$

II - 2 - 1 - c La validation des RdPC

La validation permet de vérifier que le réseau répond correctement aux spécifications du système étudié, elle donne aussi des informations sur le comportement du système et ses propriétés.

La validation des RdPC peut se faire soit par construction du graphe des marquages accessibles, soit par dépliage du graphe et étude après réduction du graphe RdP obtenu. Nous citons les travaux de MERCIER DES ROCHETTES qui a construit un algorithme de dépliage pour la validation [MERCIER - 88] et nous renvoyons aux travaux de ALLA [ALLA - 87] sur la validation formelle par les techniques d'interprétation des invariants linéaires de places.

Nous illustrerons cette partie par la validation du stock FIFO présenté précédemment.

Soit $W = \text{Post} - \text{Pré}$ la matrice d'incidence associé au réseau de la figure II - 5 :

$$W = \begin{matrix} & & \begin{matrix} T1 & & T2 & & T3 \end{matrix} \\ \begin{matrix} PO \\ PL \end{matrix} & \left[\begin{array}{ccc} id & inc^2 - id & -id \\ -proj^1 & proj^1 - proj^1 \cdot inc^2 & proj^1 \end{array} \right] \end{matrix}$$

La matrice d'incidence W est factorisable sous la forme $W = F \cdot A \cdot G$

$$W = \begin{bmatrix} id & 0 \\ 0 & proj^1 \end{bmatrix} \cdot \begin{bmatrix} id & id & -id \\ -id & -id & id \end{bmatrix} \cdot \begin{bmatrix} id & 0 & 0 \\ 0 & inc^2 - id & 0 \\ 0 & 0 & id \end{bmatrix}$$

La décomposition de la matrice d'incidence fait apparaître des matrices diagonales qui facilitent la recherche des solutions de $X^T \cdot W = 0$.

Soit $U = \begin{bmatrix} \text{proj}^1 \\ \text{id} \end{bmatrix}$ unique solution de $X^T \cdot W = 0$

On en tire la relation d'invariance du marquage de place :

$$(\text{proj}^1, \text{id}) \cdot \begin{bmatrix} M(\text{PO}) \\ M(\text{PL}) \end{bmatrix} = (\text{proj}^1, \text{id}) \cdot \begin{bmatrix} M_0(\text{PO}) \\ M_0(\text{PL}) \end{bmatrix}$$

avec M_0 décrit dans la figure II - 5 on obtient :

$$\text{proj}^1[M(\text{PO})] + M(\text{PL}) = \text{proj}^1[\langle p_1, e_5 \rangle] + \text{proj}^1[\langle p_1, e_2 \rangle] + \text{proj}^1[\langle p_2, e_3 \rangle] + \langle e_1 \rangle + \langle e_4 \rangle$$

soit encore

$$\text{proj}^1[M(\text{PO})] + M(\text{PL}) = \langle e_5 \rangle + \langle e_2 \rangle + \langle e_3 \rangle + \langle e_1 \rangle + \langle e_4 \rangle$$

L'interprétation de cette relation d'invariance permet de retrouver les spécifications du système :

- Si la place PL contient un jeton $\langle e_k \rangle$ alors il ne peut pas y avoir de jeton $\langle p_i, e_k \rangle$ dans la place PO. Il y a exclusion mutuelle entre les places PO et PL par rapport à la couleur $\langle e_k \rangle$.
- Le partage d'un emplacement par deux entités $\langle p_i, e_k \rangle$ et $\langle p_i', e_k \rangle$ donnerait $2\langle e_k \rangle$ dans le second membre après transformation par la fonction projection. Le partage de la ressource est donc bien représenté.

Cet invariant permet de déduire également que le réseau est borné, en effet, pour tout marquage accessible M à partir de M_0 , on a :

$$M(\text{PO}) \leq \sum_{j=1}^m \langle p_i, e_j \rangle \quad \text{et} \quad M(\text{PL}) \leq \sum_{j=1}^m \langle e_j \rangle \quad \text{avec } m = 5$$

Cette constatation permet de vérifier que la capacité limitée du stock est bien représentée par le modèle.

De même, le réseau est sans blocage :

si le stock est vide : $M(PL) = \langle e1 \rangle + \langle e2 \rangle + \langle e3 \rangle + \langle e4 \rangle + \langle e5 \rangle$

la transition T1 sera toujours valide par rapport à n'importe quelle couleur $\langle pi, e1 \rangle$, on peut entrer n'importe quel produit.

m

si le stock est plein : $M(PL) = 0$ d'où $M(PO) = \text{Somme } \langle pi, ej \rangle$ avec $m=5$

j = 1

la transition T3 sera valide, on peut toujours sortir le produit du dernier emplacement.

si le stock n'est ni vide ni plein : cas du marquage initial M0 de l'exemple

si l'emplacement m est occupé, la transition T3 est validée, on peut sortir le produit du stock.

sinon, il existe au moins un emplacement k tel que l'emplacement k+1 soit libre, ce qui valide la transition T2.

Les travaux de ALLA sur la validation formelle des RdPC ont permis de déterminer des algorithmes de décomposition de la matrice d'incidence dans le cas où celle-ci n'est pas facilement décomposable en matrices diagonales [ALLA - 84].

II - 2 - 2 Les Réseaux de Petri Colorés Temporisés

La modélisation par Réseaux de Petri Colorés permet de représenter l'évolution du système sans intervention d'informations extérieures, sans interprétation, on parle de modèle autonome. Mais il est nécessaire, dans le contexte d'une simulation d'ateliers flexibles de tenir compte de la notion temporelle liée aux durées de transfert ou de traitement L'introduction de la notion de temps comme celle d'événement extérieur issu de capteurs placés sur le système physique rend le modèle non-autonome.

Plusieurs approches sont proposées pour représenter la notion de temps :

- temporisations sur les places
- temporisations sur les transitions
- temporisations sur les arcs

Nous avons opté pour une représentation du temps liée aux places, on parle de RdPC Places-Temporisés.

Soit le RdPC = { P, T, C, Pré, Post, M0}

et D la fonction temporisation de P dans [R

A la place P_i , on associe un vecteur de temporisation D_i , de dimension Card $[C(P_i)]$.

Chaque composante dk de ce vecteur représente la durée d'indisponibilité du jeton de couleur $\langle dk \rangle$ dans la place P_i . Un jeton dans une place est soit disponible soit indisponible. Le jeton $\langle ck \rangle$, déposé à l'instant t dans la place P_i , ne pourra intervenir dans le tir de la transition dont P_i est amont, qu'à l'instant $t + dk$.

Cette temporisation ou durée d'indisponibilité de la ressource dans la place permet de représenter les durées opératoires (lorsque la place concernée traduit l'état machine en travail), et les temps de transfert (lorsque la place concernée traduit l'état ressource de transport occupée).

Dans le cas d'une temporisation nulle, le jeton ne passe pas par une phase d'indisponibilité. Toutes les places ne sont pas temporisées, car elle ne traduisent pas toutes une activité. Une temporisation placée sur une place machine libre peut toutefois permettre de représenter une durée de maintenance préventive sur une machine entre deux phases opératoires.

Les règles d'évolution du RdPCT sont identiques à celles du RdPC associé, avec la contrainte que seuls les jetons disponibles à l'instant t peuvent intervenir dans le tir d'une transition. Le marquage M se décompose alors en deux marquages distincts M_d , M_i , constitués respectivement des jetons disponibles et des jetons indisponibles $M = M_d + M_i$.

Dans le modèle du stock FIFO présenté jusqu'alors, on peut envisager d'introduire une temporisation D constante, de valeur d , quelque soit la couleur $\langle p_i, e_j \rangle$ du jeton de la place PO , caractérisant la vitesse de transfert d'une position à la position suivante du stock.

Le principal intérêt d'une temporisation sur les places se justifie dès lors que l'on veut utiliser l'information portée par les différents indices de couleur de la place pour déterminer des valeurs de temporisations plus élaborées. On peut imaginer une vitesse de transport qui soit fonction du type de produit, ou encore fonction de l'emplacement du stock concerné. La structure vectorielle de la temporisation attachée à la place permet alors toutes ces variantes.

L'introduction de la temporisation dans les RdPC ne rend plus possible l'étape de validation formelle. La validation formelle du RdPC permet une préétude des propriétés du système avant simulation.

La notion de Réseaux de Petri Temporisés conduit directement à des implémentations informatiques qui permettent une simulation du fonctionnement temporel du système modélisé.

De nombreux logiciels de simulation à base de RdP ont été réalisés dans les laboratoires de recherche, nous citerons ceux qui sont véritablement dédiés à la simulation d'ateliers flexibles :

- SEDRIC Simulateur de RdP interprétés, colorés temporisés développé au Laboratoire d'Automatique et d'Analyse des Systèmes (LAAS) de l'Université Paul Sabatier-Toulouse [VALETTE - 85].
- LORIC Simulateur de RdP colorés temporisés développé par l'Institut National de Recherche en Informatique et en Automatique (INRIA) [LEOPOULOS -85].
- ARCANTIEL Simulateur de RdP colorés Temporisés développé à l'Institut National Polytechnique de Grenoble (LAG - INPG) [MARTIN - 87].

II - 2 - 3 Importance de la notion de conflit

La difficulté d'un système d'aide à la conduite est de tenir compte de l'état réel de l'atelier et de faire apparaître chacun des points de décision du système. Trop souvent, les simulateurs masquent un grand nombre de décisions par leur implantation informatique qui résolvent à priori des problèmes décisionnels sans les faire apparaître, en choisissant le premier d'une liste par exemple.

Le modèle RdP du système nous permet de connaître, l'état de chacune des ressources du système. Cette information est disponible à chaque instant, pour être prise en compte par le système de conduite, afin de prendre une décision plus pertinente face à l'occurrence d'un événement imprévu. Cette information nous permettra de tenir compte de l'état de l'atelier lors de l'occurrence d'un événement comme l'arrivée d'une demande urgente par exemple, qui bouleverse l'ordonnancement initialement établi. De même, la connaissance de l'état de chacune des ressources pourra permettre d'ajuster des règles de conduite plus élaborées, comme le routage vers une machine secondaire, lors de l'occurrence d'un événement panne.

La connaissance du comportement de l'atelier par simulation fait apparaître des états dits critiques qui conduisent à des situations de blocage irréversibles du système de production. Un système de conduite doit tenir compte de ces états prédéfinis pour améliorer ses règles de pilotage en anticipant sur l'état critique.

Pour notre application à l'aide à la conduite, le principal intérêt des RdP est lié au caractère non déterministe de ce type de modèle qui apparaît sous la forme de conflit dans le réseau. La notion de conflit correspond à une indétermination du mode de fonctionnement du procédé lui-même. Elle traduit la notion de choix et donc de décision à prendre par le système de conduite.

Nous distinguerons deux types de conflits dans les RdPC :

- Le conflit transition
- Le conflit jeton.

Le conflit transition se définit par le fait que plusieurs transitions sont franchissables simultanément tandis qu'elles engagent une même ressource non partageable. Nous illustrons ce type de conflit par une décision d'aiguillage ou de routage entre la sortie d'un stock aval de machine et le système de transport. figure II - 6. Le jeton <e> de la place Position Intersection Libre valide simultanément les transitions évacuation stock aval pour un produit P1 et évolution du système de transport pour le produit P2. Comme le jeton est non partageable, il y a conflit entre ces deux transitions et choix décisionnel à prendre par le système de conduite.

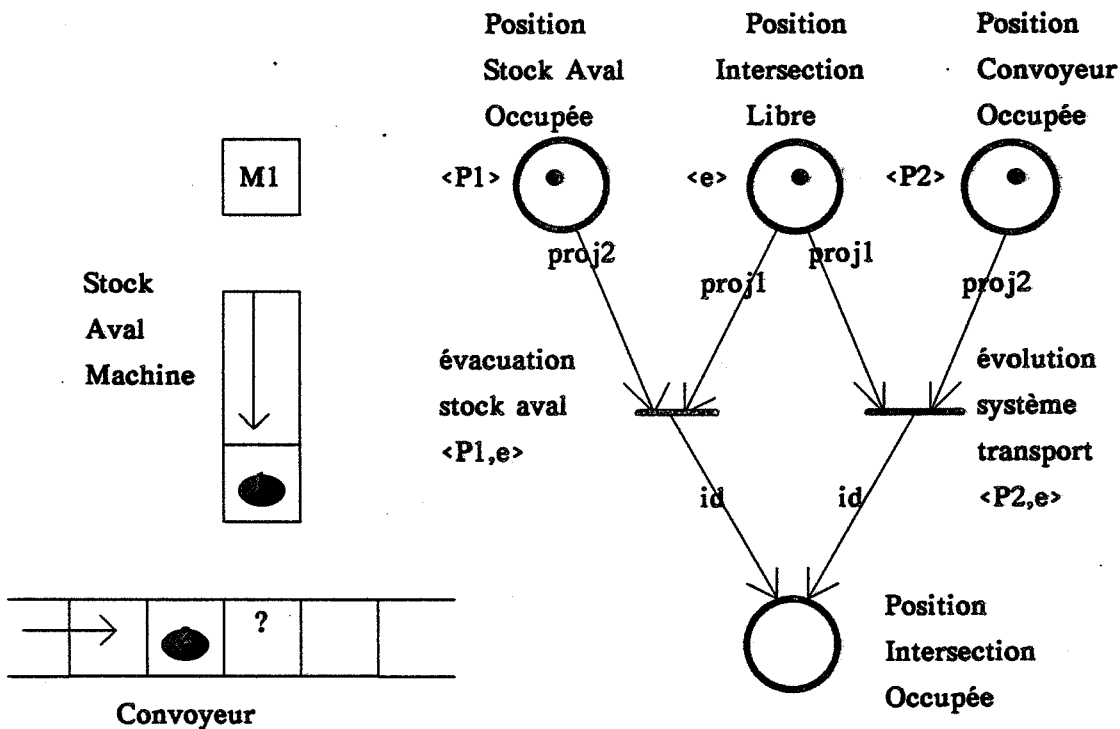


figure II-6: Exemple de conflit transition

Le conflit jeton se définit par le fait que plusieurs jetons colorés valident simultanément une même transition. Nous illustrons ce type de conflit par le choix d'une pièce dans le stock amont d'une machine dans la figure II - 7. Le jeton de la place machine libre, peut être engagé dans le tir de la transition chargement machine par plusieurs couleurs de tir de la transition; soit <1> soit <3> soit <4>, suivant le produit que l'on choisit de traiter prioritairement. Comme ce jeton est non partageable, il y a conflit entre ces différentes possibilités de tir de la transition et choix décisionnel à prendre pour le système de conduite.

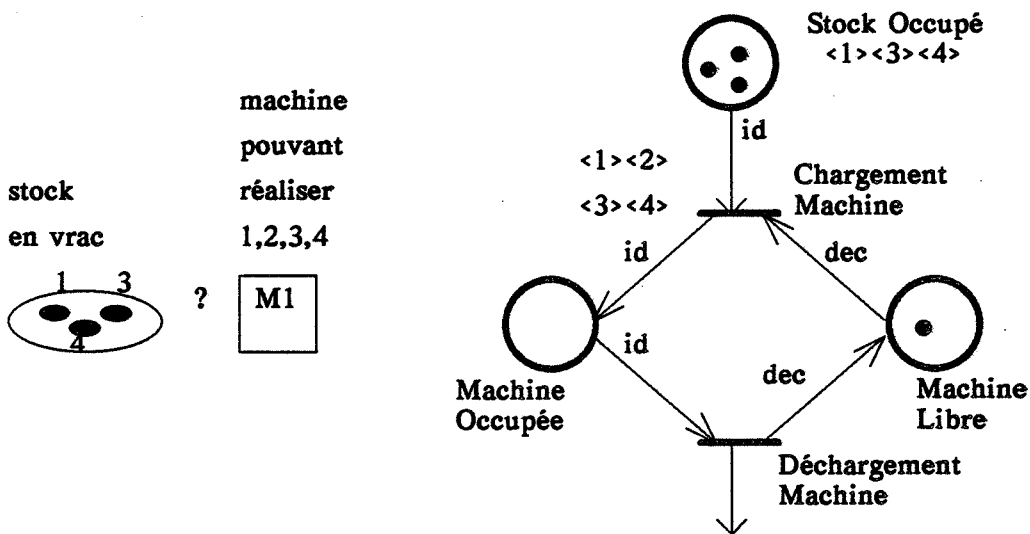


figure II-7: Exemple de conflit jeton

On désigne aussi le conflit transition sous le terme de conflit structurel parce que décrit par la structure même du réseau. En ce qui concerne le conflit jeton, on parle d'indéterminisme d'agrégation car le conflit jeton est propre à l'agrégation des réseaux colorés obtenu lors du "pliage" du RdP associé. Suivant le niveau de coloration choisi, la décision se présentera soit sous la forme d'un conflit transition, soit sous la forme d'un conflit jeton. Nous verrons dans le paragraphe suivant une proposition de méthodologie de modélisation visant à déterminer le niveau de coloration fonction de la décision à prendre.

La plupart des simulateurs basés RdP, et plus généralement tous les simulateurs, masquent cette notion de conflit par l'implémentation informatique. Les conflits sont résolus soit implicitement par le déroulement séquentiel du programme qui prend donc le premier élément de la liste, soit par l'affectation de priorité aux ressources (jetons) et/ou aux activités (transitions). La règle de décision est alors figée et ne permet pas la prise en compte de l'état de l'atelier.

Une autre solution consiste à modéliser de façon plus approfondie le système en incorporant les règles de gestion et de décision permettant la conduite. Modéliser un stock géré suivant la politique de gestion FIFO est, nous l'avons vu, possible avec les RdPC, mais il devient beaucoup plus difficile de modéliser des règles de décision plus complexes. De plus cette solution présente l'inconvénient de figer la règle de décision et d'alourdir considérablement le réseau.

Dans les deux cas, la décision est prise par le simulateur qui intègre donc à la fois des entités de description du fonctionnement du système et des entités de gestion. Cela va à l'encontre des méthodes de hiérarchisation des systèmes qui ont l'habitude de distinguer le niveau commande et le niveau décision.

Répondre à un conflit revient à faire des choix qui relèvent du niveau décisionnel. Ces choix dépendent à la fois de l'état global de l'atelier et des politiques de gestion et de stratégie de l'entreprise. Cette décision ne doit être prise ni par le modèle graphique, ni par l'interpréteur au cours de la simulation, elle doit déclencher la consultation d'un expert en pilotage d'atelier ou d'un responsable d'ordonnancement.

II - 3 APPROCHE METHODOLOGIQUE DE LA MODELISATION

Les paragraphes précédents ont montré que les RdPC étaient particulièrement adaptés pour décrire les problèmes de synchronisation et de parallélisme des activités de systèmes à événements discrets comme les ateliers flexibles. Les notions de couleurs et de fonctions permettent d'obtenir des modèles généraux caractérisés par une grande concision de l'information.

Mais un problème de méthode se pose alors, pour définir :

- Qu'est-ce que l'on modélise ?
- Jusqu'où colorer sans perdre la lisibilité du réseau ou la correspondance physique des états ?

II - 3 - 1 Ou'est-ce que l'on modélise ?

Il existe en effet deux approches d'utilisation des Réseaux de Petri dans la modélisation du comportement des ateliers flexibles :

- une approche gamme, qui décrit les différentes phases opératoires et les ressources engagées pour chaque gamme.
- une approche atelier, qui décrit la structure de l'atelier et les interactions entre les différentes ressources, lorsque les produits circulent dans celui-ci suivant leur gamme opératoire.

Nous illustrons ces deux approches à travers un exemple commun très simple.

Soit un atelier d'usinage, constitué de trois machines ; M1, M2 et M3, pouvant réaliser deux produits différents P1, P2 dont le processus opératoire est décrit par trois gammes distinctes. Une opération peut être réalisée sur deux machines alternatives décrites par une gamme-multiple.

Le tableau suivant décrit la matrice d'affectation des ressources pour chaque opération de chaque gamme, avec les durées opératoires correspondantes :

	op1	op2
P1	M1 15	M3 } ou } 25 M2 }
P2	M1 30	

II - 3 - 1 - a L'approche gamme

L'approche gamme est illustrée par le RdP de la figure II - 8. Chaque gamme est représentée par un sous-réseau, enchaînement de places qui représentent soit l'attente d'une opération, soit une opération en-cours. Les sous-réseaux sont reliés entre-eux par une place représentant la ressource machine M1 partagée entre les opérations op1 de la gamme 1 et op1 de la gamme 2.

S'il existe un jeton dans la place A-op1-g1 (pièce P1 en attente de l'opération op1 de sa gamme sur M1) et un jeton dans la place A-op1-g2 (pièce P2 en attente de l'opération op1 de sa gamme sur M1), de façon simultanée, et que la machine M1 est disponible, il y a alors un conflit entre les transitions T1 et T7. Ce conflit traduit la décision d'ordonnancement lorsque deux pièces sont candidates sur une même ressource. De même, l'opération op2 ne pourra commencer que lorsqu'il y aura un jeton dans la place M2 ou un jeton dans la place M3, si les deux existent simultanément, il y aura un second conflit entre les transitions T3 et T4 qui représente le choix de routage d'une pièce sur deux machines distinctes.

Cet exemple illustre la représentation des contraintes de succession des gammes et du partage de ressource. La levée des conflits, donc le choix entre pièces ou entre machines n'est pas exprimé par le Réseau de Petri, il fait appel à un niveau de décision supérieur, choix d'ordonnancement du produit prioritaire entre T1 et T7, choix de routage par rapport à l'utilisation des ressources entre T3 et T4.

Les temporisations sont associées aux places opérations pour traduire la durée de chacune des opérations.

GAMME 1

GAMME 2

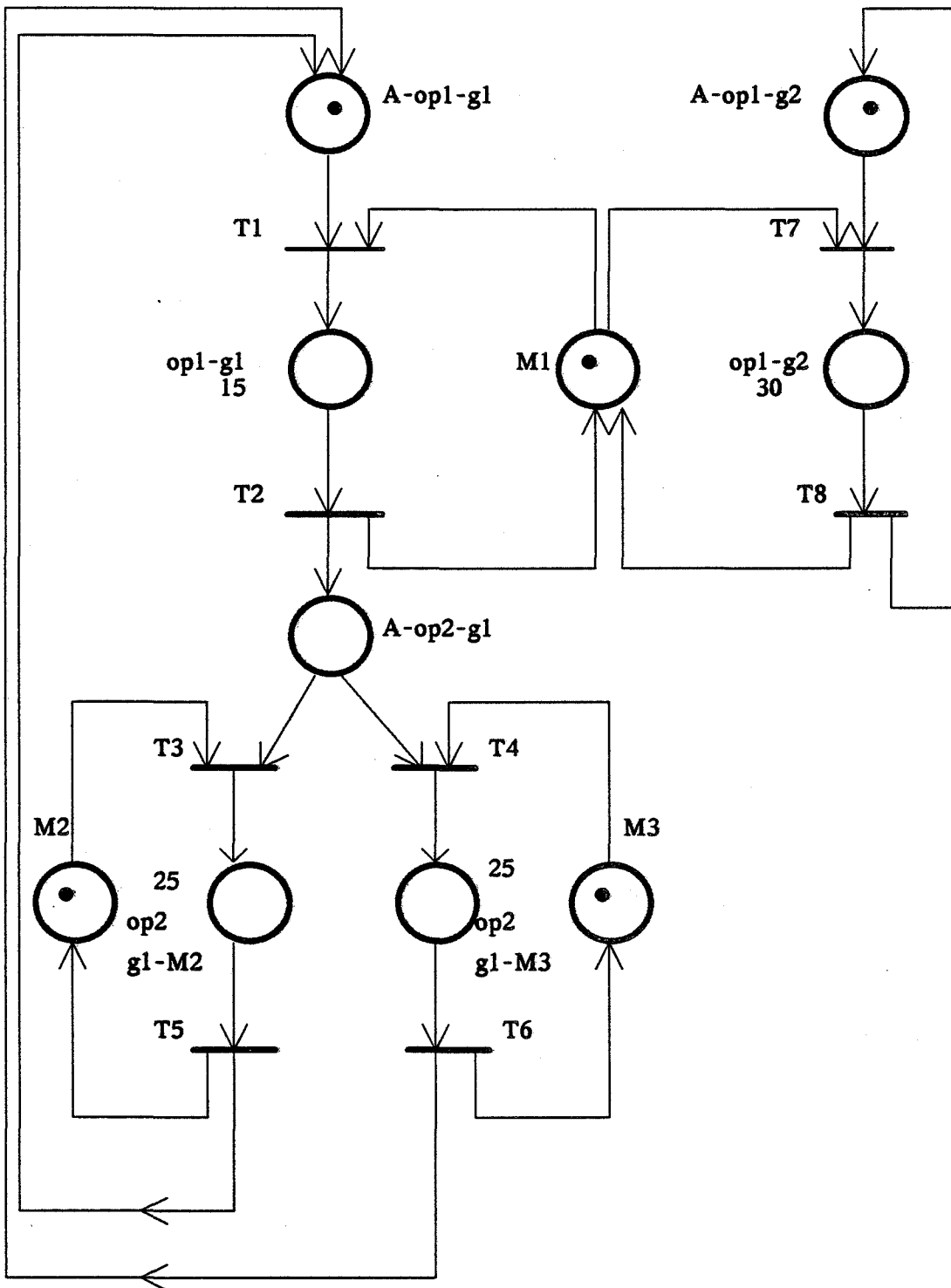


figure II-8: Approche gamme de la modélisation RdPC

II - 3 - 1 - b L'approche atelier

L'approche atelier est illustrée par le RdP de la figure II - 9. Les places représentent l'état des machines qui sont soit libres, soit occupées par P1, soit occupées par P2. Deux places sont introduites pour représenter les stocks de pièces P1 et P2 à usiner.

S'il existe un jeton dans la place S-P1 et un jeton dans la place S-P2, de façon simultanée, et que la machine M1 est disponible, il y a alors un conflit entre les transitions T1 et T6 pour traduire le conflit entre deux pièces candidates sur une même ressource. La transition T2 ne pourra être tirée que lorsqu'il y aura un jeton dans la place M2L, la transition T3 lorsqu'il y aura un jeton dans la place M3L. Si les deux existent simultanément, et que S-A-M1 est marquée, il y aura un conflit entre les transitions T2 et T3 pour traduire le choix d'affectation d'une pièce sur deux machines candidates.

Nous faisons remarquer que lorsque la machine M1 vient de terminer l'opération opl du produit P1, on peut avoir une situation de blocage dans laquelle la machine M1 ne peut être déchargée tant que l'une des machines M2 ou M3 n'est pas disponible. Les spécifications initiales du modèle font donc apparaître une configuration de blocage de l'une des ressources du système.

Les temporisations sont associées aux places Machines Occupées pour traduire la durée de chacune des opérations.

Cet exemple illustre la représentation de l'état et du partage des différentes ressources. Le choix final entre les deux transitions n'est pas exprimé par le Réseau de Petri, il fait appel à un niveau de décision supérieur, choix d'ordonnancement du produit prioritaire entre T1 et T6, choix de routage par rapport à l'utilisation des ressources entre T2 et T3.

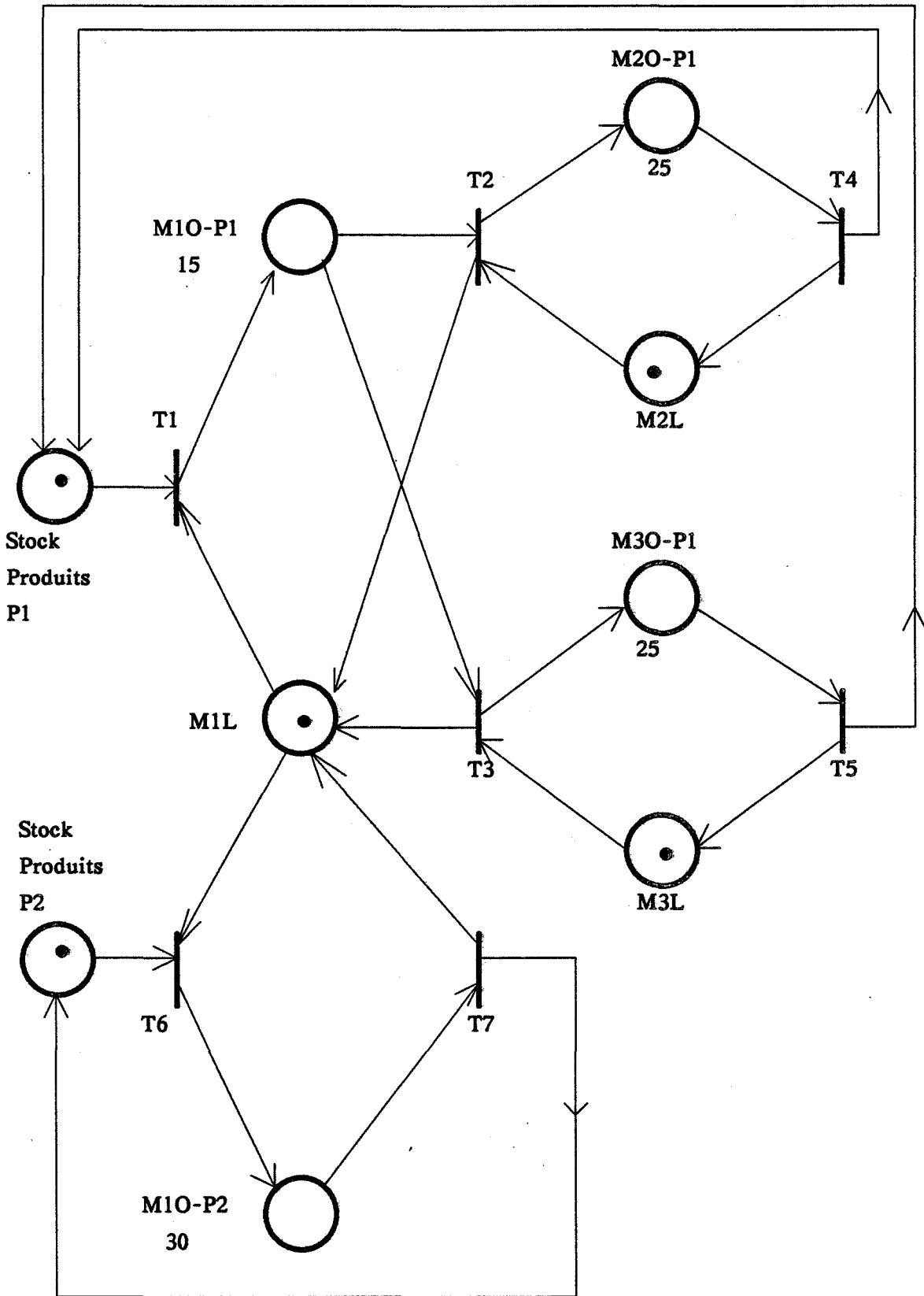


figure II-9: Approche atelier de la modélisation RdPC

II - 3 - 1 - c Comparaison des deux approches

L'approche atelier fait apparaître une insuffisance dans les spécifications du système qui peut conduire à une situation de blocage de la ressource M1. Une solution pourrait être d'introduire un stock de capacité infinie en aval de la machine M1, qui permettrait de stocker le produit p1 en fin de l'opération op1 de sa gamme, dans l'attente de la libération de l'une des machines M2 ou M3. Le réseau correspondant comprendrait alors une place Stock Aval M1 et une transition supplémentaire pour représenter le déchargement de M1.

A ce niveau de coloration des différents modèles, la structure des deux réseaux est semblable (voire identique dans le cas où l'on introduit le stock aval de la machine M1), seule l'interprétation qui est donnée aux places et aux transitions diffère.

Selon l'utilisation qui sera faite du modèle, on préférera l'une ou l'autre des approches.

- Une utilisation pour l'ordonnancement, qui nécessite une représentation globale de l'organisation de l'atelier justifie l'approche gamme. De plus la lecture même du réseau, est conforme au vocabulaire employé ; on parle d'opérations et non plus d'état des ressources.
- Une utilisation pour la simulation en vue d'une aide à la conduite nécessite plutôt une approche atelier. Il est alors nécessaire de faire clairement apparaître chacune des ressources, comme le cas du stock aval de la machine M1 dans l'exemple suivant pour connaître leur état à chaque instant. Cette approche est plus appropriée dans un objectif d'aide à la conception car elle permet le dimensionnement des différentes ressources (exemple ; dimensionnement du Stock Aval de la machine M1). Cette dernière approche permet par ailleurs une description plus détaillée du fonctionnement de l'atelier avec la modélisation des moyens de stockage et des ressources de transport, ainsi que l'introduction d'événements tels que les changements d'outil, les pannes machines et la saturation du système de transport, ...

Les deux approches conduisent à des réseaux de structure différente dès lors que l'on introduit une coloration du modèle RdP utilisé. Le réseau obtenu pour l'approche gamme peut être coloré, si l'on fait porter une couleur $\langle gi \rangle$ relative au numéro de la gamme aux jetons des places A-OP et OP.

Mais la coloration par rapport au numéro du produit $\langle pi \rangle$ et au numéro de la machine $\langle mk \rangle$ pour les machines 2 et 3, est beaucoup plus intéressante pour l'approche atelier, puisqu'elle fait apparaître seulement deux places pour décrire respectivement l'état de la machine 1, des machines 2 et 3 et une place pour représenter le stock de produits P1, P2. Le réseau obtenu, décrit dans la figure II - 10, offre alors une lecture du fonctionnement de l'atelier proche de son implantation physique.

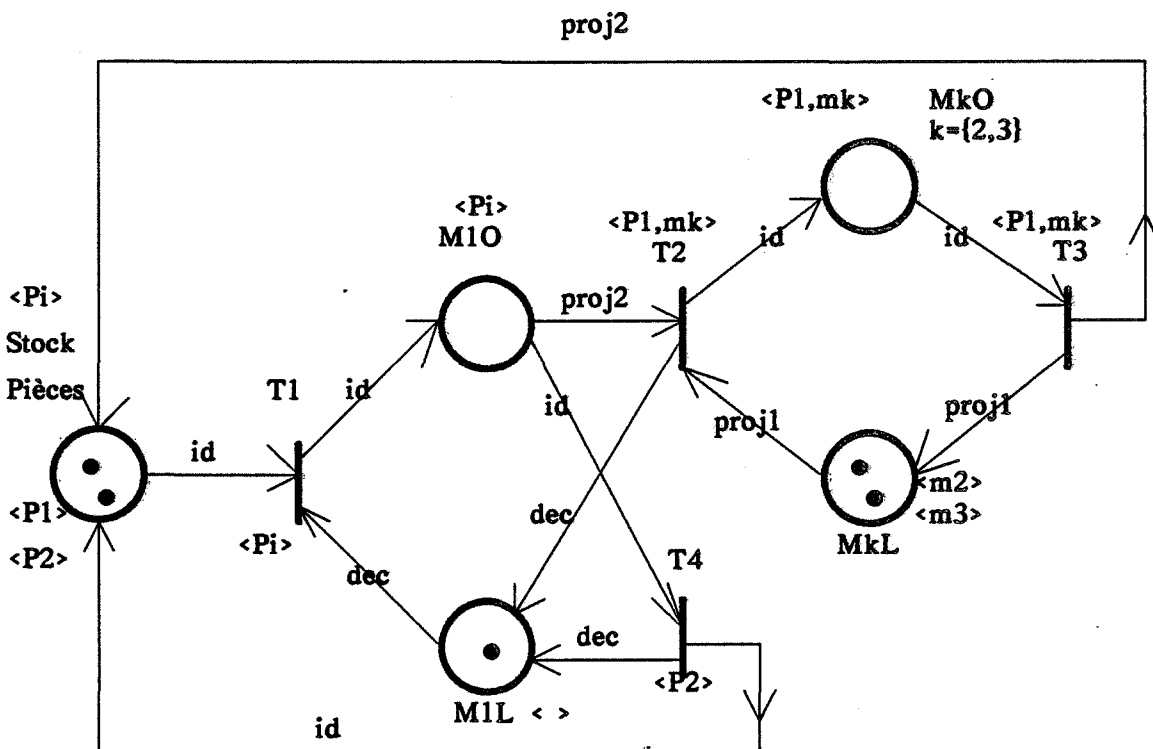


figure II-10: Coloration du modèle de l'approche atelier

Les fonctions utilisées sont très simples ; id, dec, proj, les informations portées par les jetons étant réduites au numéro du produit $\langle pi \rangle$ pour la place Stock et Machine 1 Occupée et au doublet $\langle p1, mk \rangle$ pour les Machines 2 et 3 Occupées.

La transition T1 est tirable pour les couleurs <p1> et <p2>, les transitions T2 et T3 pour la couleur <p1,mk>, tandis que la transition T4 est tirable pour la couleur <p2>. Les transitions T2 et T4 ne seront jamais en conflit, puisqu'elle concernent deux produits distincts. La gamme est donc traduite par l'ensemble des couleurs de tir des transitions.

Notons que le choix d'ordonnancement apparaît ici au niveau de la transition T1 sous la forme d'un conflit jeton lorsque la machine1 est disponible et que <p1> et <p2> sont présents simultanément dans la place Stock Pièces. Le choix de la machine défini par les gammes multiples apparait comme un conflit jeton <m2> ou <m3> à la transition T2. Ces choix ne sont pas résolus par le modèle, ils font appel à un niveau supérieur de décision.

II - 3 - 2 Jusqu'ou colorer?

Le réseau de la figure II - 10 peut à nouveau être coloré par rapport à l'information du numéro de la machine. Mais on se pose alors la question de savoir jusqu'ou colorer, sans perdre la lisibilité du réseau et la correspondance physique des états ?

II - 3 - 2 - a Simplicité des fonctions

Colorer un réseau de Petri revient à faire porter plus d'informations sur les transitions, par la définition des ensembles de tir, et sur les fonctions qui deviennent alors plus complexes. A l'extrême, on peut réduire un réseau à une place et une transition, mais la complexité des fonctions devient très importante et le graphe perd alors tout sens physique.

Dans l'exemple précédent, on peut imaginer un réseau qui regroupe les trois machines en deux places traduisant leur état libre ou occupé. Mais la limite de coloration semble alors être posée par la traduction de la gamme opératoire. La gamme peut être décrite par des fonctions complexes associant à chaque opération, le numéro de la machine concernée par l'opération suivante. Pour chaque produit, à chacune des opérations de sa gamme, on aura une fonction particulière. Le problème qui se pose alors lors de la validation d'un tel modèle est celui du calcul des invariants, la matrice d'incidence ne pouvant être décomposée aisément en matrices diagonales.

Pour privilégier la simplicité des fonctions, nous avons choisi d'introduire une place supplémentaire pour décrire la liste des opérations que peut réaliser une machine. Le marquage initial de cette place comprend pour chaque machine, les numéros de produits et les opérations dans lesquels cette machine intervient.

Les couleurs sont alors définies par le triplet $\langle p_i, op_j, mk \rangle$ où p_i est le numéro de produit candidat à l'opération op_j pouvant être réalisée sur la machine mk . $i \in \{1,2\}$ nombre de produits, $j \in \{1,2\}$ nombre d'opérations de la gamme, $k \in \{1,2,3\}$ nombre de machines.

L'avantage de cette représentation est qu'elle permet de décrire aisément la notion de gamme multiple par la présence des jetons $\langle p_1, op_2, m_2 \rangle$ et $\langle p_1, op_2, m_3 \rangle$ dans la place liste des opérations, sans alourdir le réseau.

La matrice d'incidence obtenue comprend une place supplémentaire par rapport à la matrice qui aurait été construite pour le modèle de la figure II - 10, et contient des fonctions simples. Les fonctions utilisées sont des injections ; à chaque jeton on fait correspondre par la fonction une et une seule couleur image. L'information de la gamme étant alors contenue dans le marquage initial de la place liste des opérations. Il semble donc que cette approche soit intéressante pour la validation formelle du réseau.

La figure II - 11 illustre le réseau obtenu :

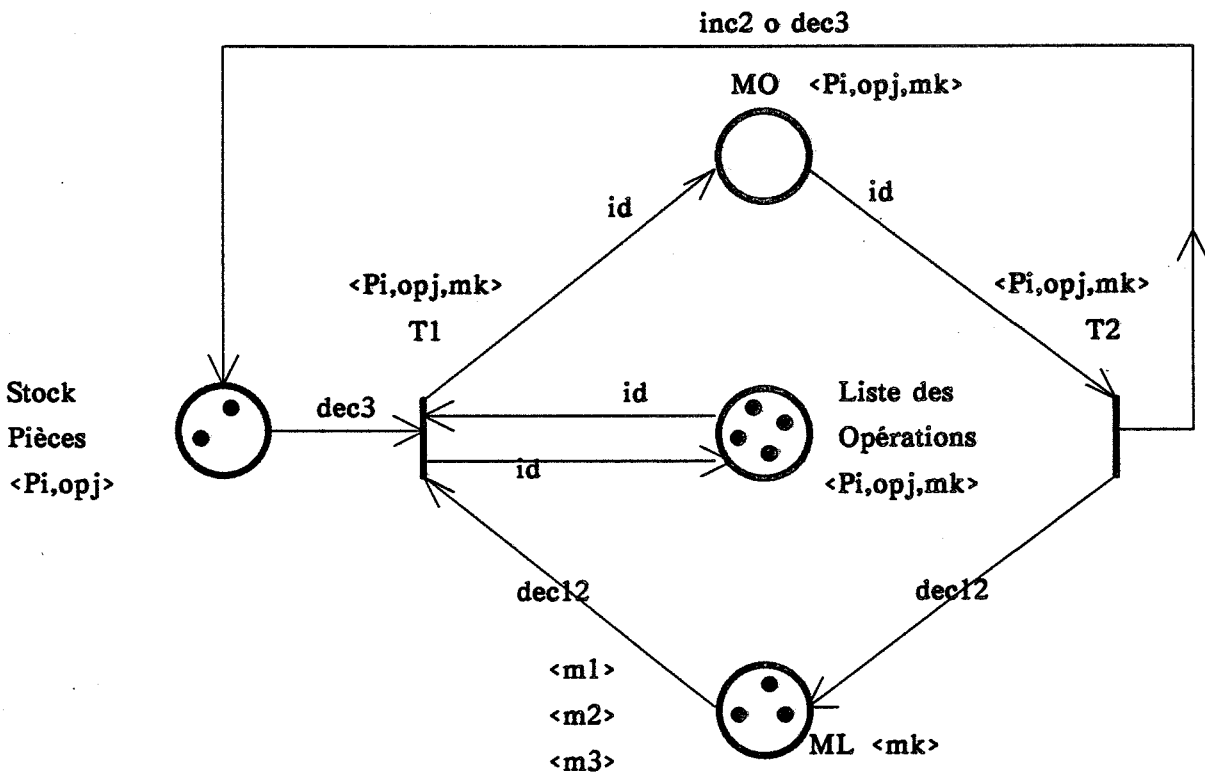


figure II-11: Coloration complète du modèle

La place Stock Pièces de couleur $\langle P_i, op_j \rangle$ comprend les pièces à lancer, mais également les pièces P_i en cours de fabrication lors de leur déchargement de la machine M_1 , ce qui permet de résoudre la configuration de blocage du réseau que nous avons évoqué précédemment.

Les places MO et ML décrivent l'état des machines. Le jeton $\langle mk \rangle$ dans la place ML indique que la machine k est disponible. Le jeton $\langle P_i, op_j, mk \rangle$ dans la place MO indique que le produit P_i à l'opération j de sa gamme est actuellement usiné sur la machine k . La temporisation tétal est une fonction de P_i et de mk pour traduire la durée de l'opération en cours.

La place Liste des Opérations permet de définir l'ensemble des opérations affectée à chaque machine. Au cours de la simulation, son marquage reste identique au marquage initial. Ici, Le marquage initial est défini par les quatre jetons $\langle P_1, op_1, m_1 \rangle$, $\langle P_2, op_1, m_1 \rangle$, $\langle P_1, op_2, m_2 \rangle$ et $\langle P_1, op_2, m_3 \rangle$.

La transition T_1 : chargement d'une machine est tirable par rapport à $\langle P_i, op_j, mk \rangle$.
Si la place Stock pièces comprend un jeton $dec_3 (\langle P_i, op_j, mk \rangle) = \langle P_i, op_j \rangle$, c'est à dire qu'elle comprend un produit P_i à l'opération j quelconque de sa gamme. Si la place ML comprend un jeton $dec_{12} (\langle P_i, op_j, mk \rangle) = \langle mk \rangle$ ($dec_{12} = dec_2$ o dec_1 , composition de fonctions), c'est à dire qu'une machine est disponible. Et que la place Liste des opérations comprend un jeton $id(\langle P_i, op_j, mk \rangle) = \langle P_i, op_j, mk \rangle$, alors l'opération op_j du produit P_i peut se faire sur cette machine mk qui est disponible et la transition T_1 est tirable.
On dépose alors un jeton $id (\langle P_i, op_j, mk \rangle) = \langle P_i, op_j, mk \rangle$ dans la place MO pour traduire le chargement de la machine, et un jeton $id (\langle P_i, op_j, mk \rangle) = \langle P_i, op_j, mk \rangle$ dans la place Liste des Opérations puisque les caractéristiques des machines restent inchangées.

La transition T2 : déchargement d'une machine est tirable par rapport à $\langle P_i, op_j, mk \rangle$.

Si la place MO comprend un jeton id $\langle P_i, op_j, mk \rangle = \langle P_i, op_j, mk \rangle$ disponible, c'est que l'opération op_j du produit P_i sur la machine mk vient de s'achever, la transition T2 est donc tirable.

On dépose alors un jeton $inc2$ o $dec3$ $\langle P_i, op_j, mk \rangle = inc2 \langle P_i, op_j \rangle = \langle P_i, op_{j+1} \rangle$ dans la place Stock pièces et un jeton $dec12$ $\langle P_i, op_j, mk \rangle = \langle mk \rangle$ dans la place ML pour libérer la machine mk .

La politique de lancement utilisée jusqu'à présent dans cet exemple génère automatiquement le lancement d'un nouveau jeton $\langle P_1, op_1 \rangle$ dès l'apparition d'un jeton $\langle P_1, op_3 \rangle$ dans la place Stock pièces, pièce ayant terminé son traitement. De même, l'apparition d'un jeton $\langle P_2, op_2 \rangle$ dans la place Stock pièces, pièce ayant terminé son traitement génère automatiquement le lancement d'un nouveau jeton $\langle P_2, op_1 \rangle$.

II - 3 - 2 - b Présentation du conflit

Cette représentation fait apparaître les conflits décisionnels ; choix de la pièce à lancer sur M1 et choix de la machine pour l'opération 2 du produit 1, sous la forme d'un conflit jeton au niveau de la transition T1. La coloration a donc eu pour effet de transformer le conflit transition en un conflit jeton. Le choix de la forme de présentation du conflit décisionnel peut donc être déterminant pour le niveau de coloration.

Le traitement d'un conflit, lors de sa résolution, fait apparaître plusieurs familles de décision qui répondent aux questions :

- Quel produit charger sur la machine disponible ?
- Quel produit lancer ?
- Quelle orientation donner au système de transport ?
- Sur quelle machine réaliser la tâche ?

Les deux premières questions reviennent à donner un ordre de priorité sur les produits présents dans le stock d'entrée de l'atelier au lancement ou dans les stocks d'en-cours devant les machines. Leur résolution fait appel à des techniques d'ordonnancement, nous les représenterons par des conflits jetons.

La troisième question revient à donner un ordre de priorité sur des activités "géographiques" du système ; évolution du système de transport ou libération d'une ressource machine ou stock. Il semble alors important de pouvoir visualiser graphiquement ces différentes possibilités qui traduisent le routage des produits dans l'atelier. Aussi, nous privilégierons la représentation de cette décision sous la forme d'un conflit transition.

La quatrième question intervient dans le cas de gammes multiples ou dans le cas d'une panne contre laquelle on va réagir en déroutant le produit vers une machine secondaire. Si cette décision de routage ne fait pas intervenir de système de transport dans le modèle, nous la représenterons par un conflit jeton comme dans l'exemple précédent. Si la décision de routage fait intervenir une ressource de transport, représentée dans le réseau par ses états de disponibilité, nous la traduirons par un conflit transition qui présente l'intérêt de représenter graphiquement les différentes possibilités de routage.

La coloration du réseau est donc limitée d'une part par la lisibilité du graphe, la correspondance physique des états et la nécessité de conserver des fonctions simples sur les arcs et d'autre part par le choix de mise en forme du conflit décisionnel.

II - 3 - 3 Conclusion

Pour pallier la grande diversité des réseaux obtenus pour un même atelier, suivant l'approche retenue ou le niveau de coloration choisi, MARTIN a proposé dans sa thèse une méthodologie de simulation [MARTIN - 87]. Cette méthodologie, basée sur l'approche atelier, consiste en la décomposition du système de production en sous modèles de base ou motifs. Les ressources machines, stocks à capacité infinie, stocks à capacité limitée, système de transport sont représentés par autant de sous-graphes définis dans une bibliothèque de modèles. Un convoyeur linéaire uni-sens peut, par exemple, être représenté par le sous-réseau du stock géré en FIFO que nous avons vu précédemment.

Puis, l'intégration des différentes entités et la définition des relations de synchronisation qui les relient permet la construction du réseau de Petri de l'atelier. La figure II - 12 illustre une relation entrée/sortie et la synchronisation entre deux ressources à capacité limitée par exemple une machine et un convoyeur de n positions. La partie sombre du schéma met l'accent sur la relation de synchronisation proprement dite avec sa description arc/transition.

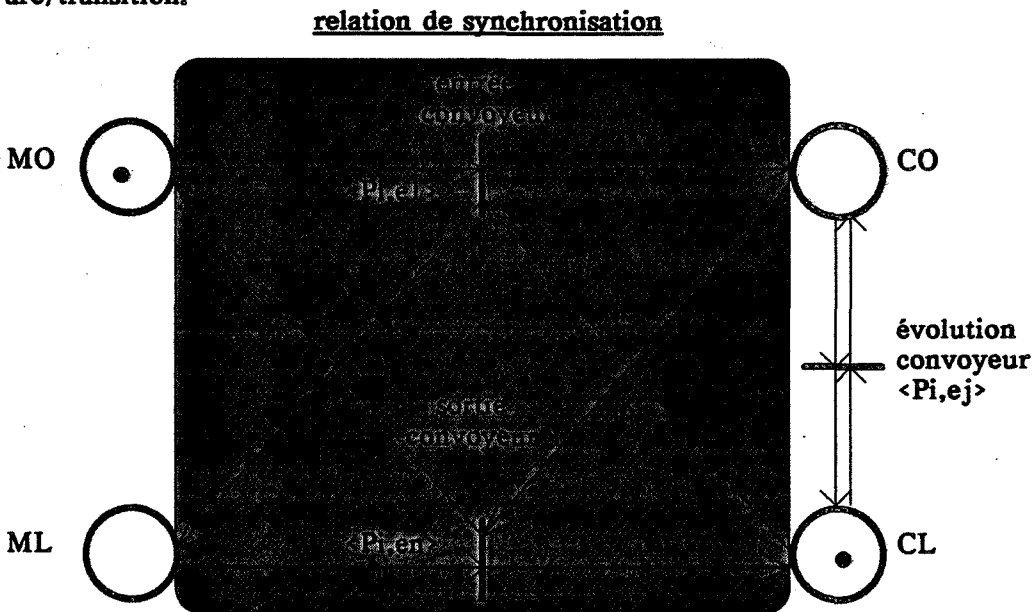


figure II-12: relation de synchronisation entre blocs modèles

Cette méthodologie a permis de mettre en évidence une bibliothèque de fonctions de base rencontrées dans les modèles d'atelier. Ces fonctions peuvent intervenir soit sous leur forme simple, soit sous leur forme composée. Nous en avons déjà rencontré un grand nombre dans les modèles RdPC que nous avons vu précédemment. Soit C un ensemble de n couleurs: $C = \{ \langle c_1 \rangle, \langle c_2 \rangle, \dots, \langle c_i \rangle, \dots, \langle c_n \rangle \}$ et $C_m = C \times \dots \times C$ le produit cartésien à l'ordre m de C .

ensemble	ensemble	fonction	Définition	Commentaire
départ	arrivée			
C	C	Id	$\text{Id}(\langle c_i \rangle) = \langle c_i \rangle$	Identité
C	$\langle . \rangle$	Dec	$\text{Dec}(\langle c_i \rangle) = \langle . \rangle$	Décoloration
C	C	Inc+	$\text{Inc}+(\langle c_i \rangle) = \langle c_{i+1} \rangle . [n]$	Incrément Modulo n
C	C	Inc-	$\text{Inc}-(\langle c_i \rangle) = \langle c_{i-1} \rangle . [n]$	Décrément Modulo n
C_m	C_m	Inc+j	$\text{Inc}+j(\langle c_{i1}, c_{i2}, \dots, c_{ij}, \dots, c_{im} \rangle)$ $= \langle c_{i1}, c_{i2}, \dots, c_{(i+1)j}, \dots, c_{im} \rangle . [n]$	Incrément Modulo n pour un m -uplet
C_m	C_m	Inc-j	$\text{Inc}-j(\langle c_{i1}, c_{i2}, \dots, c_{ij}, \dots, c_{im} \rangle)$ $= \langle c_{i1}, c_{i2}, \dots, c_{(i-1)j}, \dots, c_{im} \rangle . [n]$	Décrément Modulo n pour un m -uplet
C_m	C_{m-1}	Projk	$\text{Projk}(\langle c_{i1}, c_{i2}, \dots, c_{ij}, \dots, c_{im} \rangle)$ $= \langle c_{i1}, c_{i2}, \dots, c_{(i-1)j}, c_{(i+1)j}, \dots, c_{im} \rangle$	Projection sur la jème composante

La description des couleurs des jetons décrit alors l'ensemble des informations sur les produits et les ressources. Soit le quadruplet $\langle c_1, c_2, c_3, c_4 \rangle$ la signification des indices c_i caractérise par exemple:

- $i = 1$ numéro du produit
- $i = 2$ numéro de l'opération en-cours
- $i = 3$ numéro de la machine
- $i = 4$ numéro de l'outil

Les besoins de l'application pourront nécessiter l'emploi d'un grand nombre d'indices qui risquent de pénaliser la rapidité d'exécution de l'interpréteur. Dans la pratique industrielle, il semble que dix indices de couleurs soient suffisants et offrent un juste compromis entre la richesse de l'information et l'efficacité.

Nous décrivons dans le dernier chapitre trois exemples illustratifs de l'application de cette méthodologie.

II - 4 L'INTERPRETEUR DE RdPCT

L'implantation informatique permettant de simuler le fonctionnement de l'atelier modélisé par RdPCT et qui participera au système d'aide au pilotage et à l'ordonnancement avec mise en évidence des points de décision du système, des choix à résoudre s'appuie sur un joueur ou interpréteur de RdPCT. Par ailleurs, la nécessité de pouvoir représenter l'ensemble des mécanismes et ressources rencontrés dans les ateliers flexibles nous a conduit à construire un outil général, plutôt que dédié à un type d'application. Nous sommes donc partis des travaux de F. MARTIN, en conservant l'approche du simulateur général décrit dans sa thèse.

L'interpréteur répond à l'approche par événements des outils de simulation. La progression dans le temps se fait par saut d'une date d'événement à la date de l'événement suivant, grâce à la notion de temporisation. Lorsqu'un jeton temporisé devient indisponible vis à vis des transitions avals de la place qu'il occupe, il est placé dans un échéancier qui contient alors la liste ordonnée dans le temps des événements pouvant faire évoluer le système. A la fin de la temporisation, le jeton est replacé dans la place qu'il occupait, il est dès lors disponible pour toute évolution du système. Le marquage d'une place est alors caractérisé par l'ensemble des jetons disponibles dans cette place et des jetons indisponibles concernant cette place, mais placés dans l'échéancier. Cette approche présente l'intérêt de minimiser le temps de simulation en ne représentant que les états significatifs du comportement du système.

II - 4 - 1 La notion de cycle de simulation

La notion de cycle de simulation de l'interpréteur correspond à une séquence de tirs de transitions qui fait passer le système d'un état stable à un autre.

Le traitement en temps simulé de l'interpréteur permet de considérer que le franchissement d'une transition par rapport à une couleur "validante" est instantané. A un instant donné, caractéristique d'un état stable du système, l'évolution du temps est figée, et on cherche alors à franchir toutes les transitions franchissables. L'évolution du marquage qui en résulte peut autoriser d'autres transitions. On atteint un nouvel état stable lorsqu'il n'existe plus de transitions franchissables. L'évolution du temps se fait alors en affectant à la valeur de l'instant courant, la valeur du prochain événement de l'échéancier (date de fin d'indisponibilité d'un jeton).

La notion de cycle traduit donc ici le franchissement de toutes les couleurs possibles sur l'ensemble des transitions tirables sans évolution du temps.

La difficulté rencontrée est alors de conserver les indéterminismes du modèle afin de mettre en évidence les différents points de décision en vue de leur traitement par un niveau décisionnel supérieur.

II - 4 - 2 La notion de transition à décision

Au cours d'un cycle, le test de toutes les transitions franchissables met en évidence l'ensemble des conflits qui apparaissent tant au niveau des couleurs de tir d'une transition (conflit jeton) qu'au niveau des transitions qui partagent une même ressource (conflit transition).

Certains conflits du réseau sont parfois résolus au niveau de l'atelier par des contraintes technologiques d'implantation, ils ne traduisent plus alors une décision mais une contrainte. Le conflit est résolu au niveau de l'interpréteur et non pas par le niveau décisionnel. Les conflits non résolus par l'implantation seront appelés conflits décisionnels et devront être distingués dès la construction du réseau. Aussi nous avons introduit une notion supplémentaire dans la définition des transitions pour distinguer les conflits décisionnels et ordonner les conflits entre eux.

La représentation des RdPCT à Décision est la suivante:

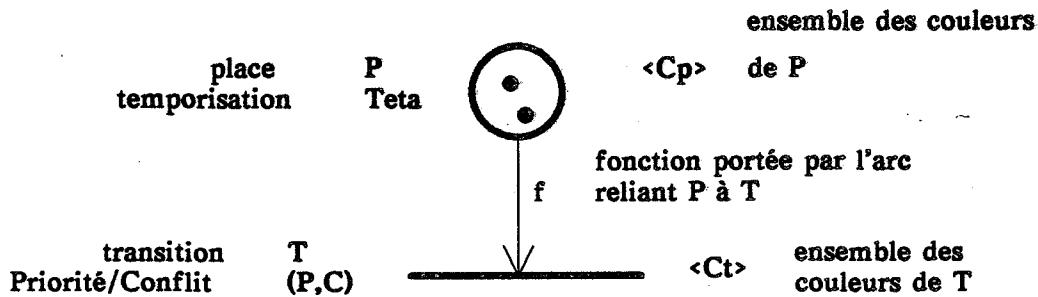


Figure II-13: Représentation des RdPCT à Décision

Chaque transition du réseau est définie par un couple (priorité, conflit) afin d'ordonner les transitions en fonction de leur valeur de priorité et de définir leur niveau de décision. Le conflit décisionnel entre transitions ne devient effectif que lorsque les transitions en cause appartiennent toutes au même niveau de priorité et que l'indicateur conflit a pour valeur 1.

Deux transitions ayant des valeurs de priorité différentes ne seront jamais en conflit, la transition correspondant à la valeur de priorité la plus faible étant par définition prioritaire sur l'autre. La notion de priorité permet donc de constituer des classes de transitions à l'intérieur des quelles on a le même niveau de décision et qui nécessite de ce fait un traitement commun.

Lorsque l'indicateur conflit a pour valeur 0, le conflit est résolu une fois pour toutes soit par l'ordre de déclaration des transitions à l'intérieur de la famille de transitions ayant même valeur (priorité, conflit=0) dans le cas d'un conflit transition, soit par l'ordre de déclaration des jetons dans la place amont dans le cas d'un conflit jeton. Cette valeur permet de résoudre à priori les conflits lorsque la décision qu'ils sous-tendent ne relève pas du niveau décisionnel mais plutôt d'un choix préétabli.

Lorsque l'indicateur conflit a pour valeur 1, et que plusieurs transitions ayant même valeur de priorité sont en conflit, ou que plusieurs couleurs de l'ensemble de tir d'une transition rendent celle-ci tirable, l'interpréteur met en forme le conflit en le caractérisant par son type (conflit transition, conflit jeton) et par l'ensemble des choix possibles.

La présence d'un conflit décisionnel non résolu arrête l'évolution de l'interpréteur, le franchissement des transitions est bloqué tant que le niveau décisionnel n'a pas tranché parmi l'ensemble des choix possibles. Le cycle en-cours ne reprend que lorsque le niveau décisionnel a renvoyé la valeur du choix retenu en fonction de règles de gestion et de décisions qui prennent en compte l'état courant de l'atelier, les historiques de la simulation et les critères de gestion à respecter.

Cette formulation est le garant de la traduction de la décision au niveau du réseau. C'est elle qui définiera l'interface entre le niveau commande proprement dit de l'interpréteur et le niveau décisionnel que nous détaillerons dans le chapitre suivant.

II - 4 - 3 Les temporisations aléatoires

Les pannes sont un phénomène important qu'il convient de prendre en compte dans la simulation. Evénements aléatoires par définition, elles seront traduites dans notre approche par des temporisations dites aléatoires. La temporisation n'est plus alors une valeur réelle constante, mais le résultat d'une fonction probabiliste.

Les fonctions de probabilité implantées caractérisent un grand nombre de phénomènes aléatoires: avec `généal()` fonction de génération de nombre aléatoire

Exponentielle $t = f(a) = - 1/a \cdot \log(1 - \text{généal}())$ avec a valeur moyenne

Uniforme $t = f(a,b) = a + (b - a) * \text{généal}()$ avec a et b paramètres

Normale $t = f(a, b) = a + b \cdot X$ avec a valeur moyenne, b l'écart type
et X loi normale centrée réduite, définie par construction comme somme de variables uniformes sur [0,1].

Nous illustrons l'application de ces temporisations par l'utilisation d'une place Evénement Panne dans le réseau qui génère aléatoirement un jeton panne autour de la valeur du Temps Moyen Entre deux Pannes (MTBF), caractéristique connue dans l'atelier pour chaque ressource. La durée de la panne peut elle aussi être définie aléatoirement autour de la valeur moyenne de durée de la panne, par une temporisation aléatoire sur la place Machine en Panne.

Dans ce modèle, les pannes machines ne sont prises en compte qu'à la fin de l'opération sur la machine, conformément à la majorité des situations industrielles où l'on ne peut interrompre une opération en cours sur une machine. Suivant la nature de la panne, on devra mettre au rebut le produit en-cours, l'évacuer vers une machine secondaire de reprise ou encore continuer la gamme si la panne n'a pas d'incidence sur la qualité du produit.

Dans le cas d'une panne survenant sur une Machine Occupée, le RdPCT obtenu est décrit par la figure II - 14 avec $Teta1 = Gauss$ (MTBF, écart type)

et $Teta2 = Gauss$ (durée panne, écart type).

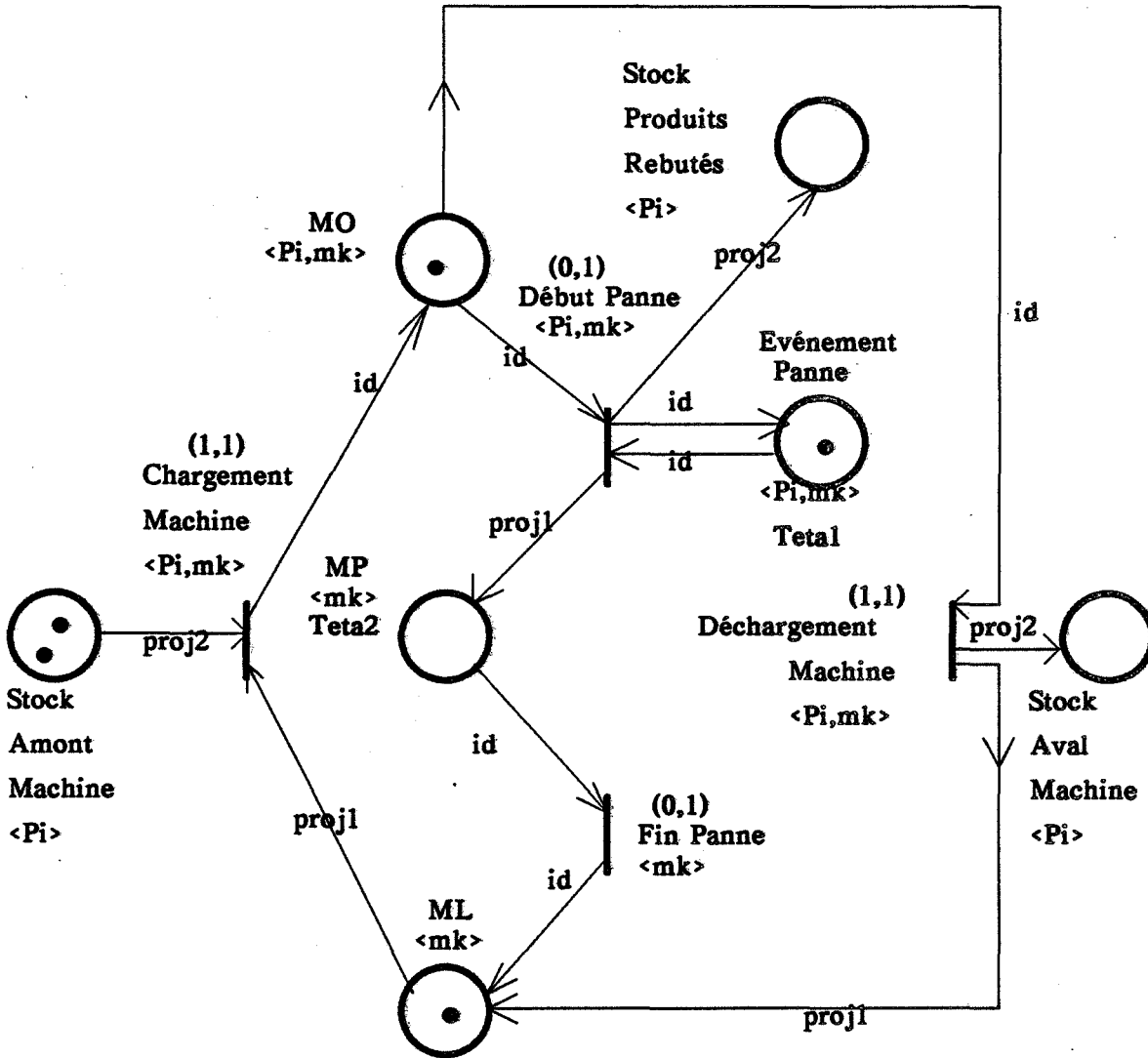


figure II-14 : Modèle RdPCT d'une panne sur machine occupée

Les transitions décisionnelles permettent ici de donner priorité à la transition début de panne par rapport à la transition évacuation de la machine par une valeur de l'indice de priorité plus faible. La valeur de l'indice conflit à 1 peut permettre de choisir entre plusieurs politiques de gestion du conflit, mise au rebut du produit, évacuation vers une machine secondaire, retraitement,...

II - 4 - 4 Le forçage d'un marquage

Une autre solution permettant de prendre en compte une panne consiste à forcer le marquage d'une place. L'utilisateur choisit alors l'instant où il valide cette action pour traduire l'événement panne.

Le danger de cette solution est lié au contrôle de cohérence des actions engendrées. Nous avons pour cela choisi de ne prendre en compte la demande de forçage de marque qu'en fin de cycle, lorsque l'état est stable.

Par ailleurs, ajouter un jeton dans une place Machine en Panne par exemple nécessite de tester l'état antérieur de la ressource machine pour le désactiver en retirant le ou les jetons correspondants. Aussi, nous préférons limiter la possibilité de forçage d'une marque aux seules places traduisant des ressources à capacité infinie (Stock d'Entrée d'un atelier) ou des places traduisant seulement la présence d'un événement (Événement Panne). Le contrôle de cohérence est alors en effet limité à la signification des différents indices de couleurs du jeton introduit.

Le forçage d'une marque nous permettra de représenter par exemple, l'arrivée d'une commande urgente dans le stock d'entrée qui n'était pas prévue dans le plan initial de lancement et qui engendrera ainsi des conflits décisionnels de réordonnement.

II - 4 - 5 L'algorithme de l'interpréteur

La simulation est constituée d'une succession de cycles au cours desquels le marquage est modifié. Elle fait apparaître les conflits décisionnels sans les résoudre au niveau de l'interpréteur.

L'interpréteur a été développé en langage C, sur une plateforme Compaq 386. L'algorithme de base de l'interpréteur traduit ces différents éléments, dans lequel nous noterons (P,C) le couple (Priorité, Conflit) caractérisant les transitions.

lecture du modèle

Répéter

Répéter

mise à jour de la valeur de l'instant

initialise pas-franchissement

initialise (P,C) = (0,0)

Répéter

transition = 1ère transition (P,C)

si C = 0 alors

Répéter

couleur = 1ère couleur de la transition

place amont = 1ère place amont de la transition

Répéter

Répéter

calcul de la couleur d'arrivée des fonctions de

l'arc à partir de la couleur de la transition

si la couleur est dans la place amont alors

validation

place amont = place amont suivante

sinon pas-validation

jusqu'à plus de place amont ou pas validation

Si validation alors

franchissement

calcul des jetons à soustraire

retrait des jetons de la place amont

calcul des jetons à déposer

dépôt de jetons dans les places aval

couleur = couleur suivante de la transition

jusqu'à plus de couleur

transition (P,C) = transition (P,C) suivante

jusqu'à plus de transition pour ce couple (P,C)

```
si C = 1 alors
  Répéter
    initialise pas-conflit
    Répéter
      couleur = lère couleur de la transition
      place-amont = lère place amont de la transition
      Répéter
        Répéter
          calcul de la couleur d'arrivée des
            fonctions de l'arc à partir de la couleur
              de la transition
            si la couleur est dans la place amont
              alors
                validation
                place amont = place amont suivante
              sinon pas-validation
            jusqu'à plus de place amont ou pas validation
            si validation alors
              construction du tableau des
                transitions / couleurs tirables
              couleur = couleur suivante de la transition
            jusqu'à plus de couleurs
            transition (P,C) = transition (P,C) suivante
          jusqu'à plus de transition pour ce couple (P,C)
          lecture tableau des transitions/couleurs tirables
          si il existe des jetons partagés alors
            conflit
            construction tableau conflit
            attente choix niveau décisionnel
            franchissement choix
            calcul des jetons à soustraire
            retrait des jetons des places amont
            calcul des jetons à déposer
            dépôt des jetons dans les places aval
          sinon
            franchissement
            calcul des jetons à soustraire
            retrait des jetons de la place amont
            calcul des jetons à déposer
            dépôt de jetons dans les places aval
        jusqu'à plus de transition (P,C) dans le tableau
          des transition/couleur tirables
      si pas-franchissement alors
        (P,C) = (P,C) suivant
      sinon
        (P,C) = (0,0)
    jusqu'à pas de (P,C) suivant
  jusqu'à pas-franchissement
  recherche dans échéancier prochain jeton disponible
  demande de forçage de marque
jusqu'à fin de simulation
```

Lorsqu'un conflit est résolu par le niveau décisionnel à l'intérieur d'un bloc de transition (P,1), l'interpréteur franchit la transition, puis recalcule l'ensemble des transitions tirables. Le tir de cette transition fait évoluer le système, il peut valider de nouvelles transitions qui elles mêmes peuvent être sources de conflits décisionnels dans ce bloc de transitions. Ce n'est que lorsqu'il n'y a plus de transitions conflictuelles dans le tableau des transitions tirables que les transitions non conflictuelles sont franchies.

Lorsqu'il n'y a plus de transitions tirables à l'intérieur d'un bloc (P,C) et s'il n'y a pas eu de franchissement, on passe au bloc de transitions suivant pour une nouvelle valeur du couple (P,C).

Lorsqu'il n'y a plus de transitions tirables à l'intérieur d'un bloc (P,C) et s'il y a pas eu franchissement, on réinitialise la valeur de (P,C) et on repart du premier bloc de transition pour évaluer l'ensemble des transitions tirables.

L'évolution du temps se fait seulement lorsqu'il n'y a plus de transitions tirables quel que soit le bloc de transition (P,C) testé.

II - 4 - 6 Résultats de la simulation

L'interpréteur fournit en sortie un certain nombre de fichiers qui permettent d'évaluer les résultats de la simulation.

fichier Trace qui contient la description de l'évolution du système dans le temps en termes de marquage des différentes places.

fichier Historique qui contient la description de l'évolution du système dans le temps en termes agrégés d'occupation des ressources machines, stocks et systèmes de transport, et du degré d'avancement des produits dans leur gamme.

fichier Gantt qui permet la construction du diagramme de Gantt d'occupation des différentes ressources en faisant apparaître les différents états possibles : Libre, Occupé, Panne, Bloqué.

fichier Résultat qui contient les résultats de la simulation en fin de période.

fichier Conflit qui contient un historique des conflits rencontrés au cours de la simulation et de leur résolution par le niveau décisionnel.

CHAPITRE III

LES METHODES D'ORDONNANCEMENT

APPORT DE L 'INTELLIGENCE ARTIFICIELLE

III - 1 LES METHODES D'ORDONNANCEMENT

"Ordonnancer, c'est programmer dans le temps l'exécution d'une réalisation décomposable en tâches, en attribuant des ressources à des tâches, et en fixant en particulier leur date de début d'exécution tout en respectant des contraintes données" [CARLIER - 82].

Dans le cadre de l'ordonnancement d'un atelier flexible de type Job-shop, cela revient à rechercher la succession des opérations à réaliser sur chaque machine de façon à satisfaire les critères liés à l'occupation de l'atelier ; occupation des machines, niveau des en-cours, et à la satisfaction des délais de livraison. Nous avons précisé le sens de ces critères dans le chapitre I.

On distingue deux types de contraintes :

- les contraintes potentielles ou contraintes de succession entre les opérations. Elles sont définies :
 - par les gammes caractérisant ainsi le mode opératoire permettant de réaliser le produit,
 - par les dates de début au plus tôt caractérisant ainsi la disponibilité des ressources matières premières,
 - par les dates de fin au plus tard caractérisant ainsi l'obligation de satisfaire les délais de livraison,
- les contraintes de ressource qui définissent la disponibilité des ressources machines nécessaires à la réalisation d'une tâche ou d'une opération. L'utilisation simultanée par plusieurs tâches des différentes ressources induit des contraintes disjonctives et/ou des contraintes cumulatives. Lorsqu'une ressource est unique et non partageable, on parle de contrainte disjonctive. Lorsqu'une ressource n'est pas unique ou est partageable, on parle de contrainte cumulative.

Comme nous avons pu le voir dans le chapitre précédent, chacune de ces contraintes apparaît sous la forme d'un conflit lors de la modélisation de l'atelier en Réseaux de Petri. Le point de décision étant ainsi détecté et mis en évidence, la question est alors de savoir quelles méthodes, quels outils sont à notre disposition pour répondre à ce conflit.

Le grand nombre de contraintes à caractère disjonctif fait du problème d'ordonnement un problème fortement combinatoire. Dans la majorité des cas, le problème est NP complet.

Parallèlement à la complexité du problème posé, il est important de préciser que les problèmes d'ordonnement se divisent en deux types ; problèmes statiques et problèmes dynamiques.

Un problème d'ordonnement est statique si, lorsque l'on recherche à le résoudre, on connaît parfaitement toutes ses caractéristiques ; la période est clairement définie et toutes les caractéristiques des tâches et des ressources sont connues.

Un problème d'ordonnement est dynamique si, les caractéristiques des tâches et les contraintes qui les lient sont connues de manière déterministe ou probabiliste au fur et à mesure de la progression du temps.

Face à ces deux types de problèmes on a vu se développer deux types de méthodes d'ordonnement:

- La recherche d' une solution unique pour les problèmes statiques qui sont supposés avoir un nombre fini de solutions. Selon les cas étudiés, la solution obtenue sera :
 - optimale pour un ou plusieurs critères,
 - réputée bonne si dans un cas complexe on se contente d'une heuristique,
 - admissible, dans les cas où les contraintes sont trop fortes.
- La recherche d' une famille d'ordonnements admissibles pour les problèmes dynamiques : cette approche s'accompagne alors d'un système d'aide à la décision pour aider l'opérateur à choisir parmi ces solutions.

Dans un premier temps, nous allons présenter les différents types de méthodes appliquées aux problèmes statiques qui ont fait l'objet de nombreux travaux de recherche dans le passé.

III - 1 - 1 Les méthodes de résolution du problème statique

Nous distinguerons deux types de méthodes ; une famille de méthodes exactes visant à fournir une solution optimale vis à vis d'un critère donné et une famille de méthodes dites heuristiques qui offrent une solution "bonne", sans assurer qu'elle est la meilleure.

III - 1 - 1 - a Les méthodes exactes

Il s'agit des méthodes classiques d'optimisation basées sur une approche analytique telles que la programmation linéaire en nombres entiers, la programmation dynamique, le "Branch and Bound" ou la théorie des graphes... Cette liste est non exhaustive, mais elle présente les principales méthodes.

La programmation en nombres entiers consiste à formaliser le problème d'ordonnement à l'aide de variables booléennes (la variable booléenne X_{ij} sera égale à 1 si la tâche i se fait sur la machine j et à 0 sinon) puis à rechercher la solution exacte optimale du problème ainsi posé.

La programmation dynamique est une méthode de résolution des problèmes d'optimisation basée sur une énumération implicite des solutions.

La technique du "Branch and Bound" ou séparation et évaluation est basée sur une énumération des solutions potentielles qui sont séparées en sous ensembles puis évaluées afin d'éliminer des sous ensembles à l'aide de théorèmes. La programmation des séparations peut être effectuée en utilisant des procédures récursives ou un algorithme itératif.

Seuls quelques cas particuliers ont été résolus dans la bibliographie selon cette approche, les problèmes à une machine [BAKER - 74], à plusieurs machines en parallèle ou à deux machines en flowshop...

On connaît la séquence optimale qui minimise les critères dans le cas du problème à une machine :

- temps moyen de présence des pièces,
- retard relatif et retard vrai maximal,
- variance du retard dans le cas où les dates de livraison sont les mêmes pour toutes les tâches,
- nombre de tâches en retard,
- somme pondérée des durées d'achèvement,
- somme pondérée des retards vrais,
- coût total des retards,
- temps de réglage ou coûts de réglage.

Le problème à m machines en parallèle a été étudié pour le seul critère de durée entre la date de début de la première tâche et la date de fin de la dernière tâche.

Le problème de n tâches sur un flowshop a été résolu par l'algorithme de Johnson dans le cas où le nombre de machines est inférieur à trois [JOHNSON - 54]. Schématiquement, cela revient à privilégier les tâches commençant par des opérations de courte durée et à terminer par des opérations de longue durée.

L'utilisation de ces méthodes est très vite limitée par :

- la difficulté de représenter, sous forme d'équations, les différentes contraintes et critères, dès lors que le problème d'ordonnement devient plus complexe,
- les temps de calcul nécessaires à la résolution proprement dite du problème d'optimisation.

De telles méthodes ne sauraient donc être utilisées dans le cas de problèmes réels. Dans le cas où il n'est pas possible de trouver une solution optimale par le calcul exact, on peut préférer trouver rapidement une solution satisfaisante vis à vis d'un ensemble de critères, avec les méthodes dites heuristiques.

III - 1 - 1 - b Les méthodes heuristiques

On distingue alors deux familles principales d'heuristiques :

- celles obtenues par simplification de la méthode du "Branch and Bound",
- celles construites sur des règles de priorité locales.

Une première simplification a été faite dans le cas du flow-shop, lorsque le nombre de machines est faible, en appliquant l'algorithme de Johnson sur des groupements de machines successives.

Les heuristiques à base de règles de priorité classent les tâches selon un ordre de priorité. Il existe une grande variété d'indices de priorité, nous classerons les heuristiques associées en trois grands types :

- celles qui généralisent la méthode de Johnson. Nous citerons les méthodes qui introduisent deux machines fictives issues du regroupement des m machines en deux paquets afin de pouvoir appliquer l'algorithme de Johnson [CAMPBELL - 70] [DANNENBRING - 77].
- celles qui consistent à calculer a priori une fonction sur les séries, puis à les classer dans l'ordre croissant ou décroissant de cette fonction. Nous citerons les travaux de Palmer et de Gupta. [PALMER - 65] [GUPTA - 78].
- celles qui visent à minimiser l'inactivité des machines pour minimiser globalement le temps de fabrication. Cette méthode conduit à chaque pas de l'ordonnancement, au calcul de l'attente machine qu'induirait l'ordonnancement de chacune des séries possibles, afin de choisir celle qui induit le moins d'attente. Nous citerons les travaux de Germain. [GERMAIN - 85].

Les règles de priorité présentent l'avantage d'une mise en oeuvre relativement simple. Par ailleurs, elles constituent un pas dans l'approche dynamique du problème de l'ordonnancement.

Malgré les simplifications du modèle statique, ni l'utilisation des techniques analytiques, ni l'utilisation des règles algorithmiques ou des heuristiques n'ont réussi à trouver une solution optimale ou proche de l'optimum dans le cas où le nombre de machines est supérieur à trois.

Le modèle statique a beaucoup aidé la compréhension du problème de l'ordonnancement. Pourtant, les concepts et les termes de cette formalisation ne sont pas réalistes ; toutes les caractéristiques du problème ne peuvent être considérées comme stables. Dans bien des situations réelles, la date d'arrivée des tâches n'est pas connue en début de période.

Le problème de l'ordonnancement est dynamique, l'étude d'un ordonnancement doit donc s'effectuer sur l'échelle du temps, pour tenir compte des propriétés statistiques des différents facteurs d'entrée du système, et des conditions stochastiques du fonctionnement du processus.

III - 1 - 2 Les méthodes de résolution du problème dynamique

Nous nous plaçons maintenant dans le cas de la gestion temps réel d'un atelier où les données sont parfois incertaines et où l'arrivée des produits est aléatoire. Le problème de l'ordonnancement peut alors se définir de la façon suivante : on a un ensemble de machines devant lesquelles des opérations sont en attente : il s'agit de gérer ces files d'attente le "mieux possible". Il paraît en effet vain de vouloir optimiser un tel système, tout juste peut-on définir une famille d'ordonnements admissibles vis à vis de certains critères.

Nous distinguerons deux types de méthodes :

- les méthodes issues de la théorie des graphes,
- les méthodes basées sur la simulation.

III - 1 - 2 - a Les méthodes de la théorie des graphes

Une première famille d'outils basés sur les méthodes graphiques de type PERT (Program Evaluation and Review Technic) permet de résoudre les problèmes à contraintes potentielles dans le cas où l'on veut minimiser la durée globale d'exécution. Ces méthodes présentent toutefois l'inconvénient de ne proposer une solution que vis à vis d'un seul critère.

De nombreux travaux ont été réalisés pour définir les conditions nécessaires et suffisantes pour obtenir un ensemble d'ordonnements admissibles. Les outils utilisés sont les graphes potentiel-tâches conjonctifs ou non conjonctifs [ROY - 70], et l'analyse structurale.

Le LAAS à Toulouse développe depuis 1976 une méthode basée sur l'analyse sous contraintes [ERSCHLER - 76]. La méthode est basée sur une décomposition hiérarchique dans laquelle les critères et objectifs de gestion sont transformés en contraintes externes pour le niveau inférieur. Par la suite, l'analyse sous contraintes permet d'établir une relation entre une décision locale et la satisfaction des contraintes externes.

Cette analyse permet de caractériser un ensemble d'ordonnements admissibles compatibles obéissant aux contraintes. Si l'ensemble ainsi obtenu est vide, on remet en cause les décisions prises aux niveaux supérieurs de la décomposition hiérarchique. Dans le cas contraire, on construit des sous-ensembles de solutions avec des variantes locales afin de permettre à l'utilisateur de réagir aux aléas, dans les limites imposées par les contraintes [THOMAS - 80]. L'ensemble des ordonnements admissibles étant fourni à l'atelier, le choix entre les différents sous-groupes (permutation des opérations à l'intérieur de chaque groupe) est réalisé en temps réel, en fonction des perturbations locales.

Différents travaux ont permis de construire des algorithmes pour la recherche des séquences de groupes admissibles. Ils sont basés sur les principes de scission, de transfert et de regroupement du graphe potentiel-tâches ainsi que sur les techniques de recherche de chemin critique (méthode de Monte-Carlo) [ROUBELLAT - 87].

L'approche par la théorie des graphes vise à chercher les solutions d'ordonnancement admissibles plutôt qu'optimales. Ces solutions sont des séquences de tâches ayant un grand nombre de degrés de liberté de façon à s'adapter aux perturbations de l'atelier.

III - 1 - 2 - b Les méthodes simulatoires

Cette deuxième famille d'outils est basée sur la simulation du fonctionnement de l'atelier par la gestion des files d'attente à l'aide de règles de priorité que l'on cherche à valider.

L'application de ces outils pour tester les performances de politiques d'ordonnement s'est d'abord appliquée à l'étude de systèmes fictifs pour évaluer les propriétés statistiques des règles de priorité [BAKER - 60] [GERE - 66] [CONWAY - 67] [HOLLOWAY - 74].

Les règles de priorité sont utilisées pour choisir parmi les tâches en attente celle qui sera effectuée. Le choix se fait généralement soit à partir d'une information concernant une tâche spécifique telle que sa date de fin désirée, sa durée opératoire, le nombre d'opérations restantes,... soit à partir de l'état des ressources tel que la longueur de la file d'attente.

Les règles de priorité ou règles locales d'ordonnement sont classées en trois familles:

- les règles statiques,
- les règles dynamiques,
- les règles pondérées ou combinées.

Les règles statiques

On désigne par règles statiques les règles de décision dont les paramètres sont indépendants du temps.

Nous donnons ci-dessous quelques exemples de ce type de règles :

- **RANDOM** : sélectionne la tâche de façon aléatoire.
- **SPT (Shortest Processing Time)** : sélectionne la tâche dont la durée opératoire est la plus faible.
- **LPT (Largest Processing Time)** : sélectionne la tâche dont la durée opératoire est la plus grande.
- **SRWPT (Shortest Remaining Processing Time)** : sélectionne la tâche dont le temps opératoire restant à réaliser est le plus faible.
- **LRWPT (Largest Remaining Processing Time)** : sélectionne la tâche dont le temps opératoire restant à réaliser est le plus grand.
- **FOPNR (Fewest Operation Remaining)** : sélectionne la tâche dont le nombre d'opérations résiduelles est le plus faible.
- **MOPNR (Most Operation Remaining)** : sélectionne la tâche dont le nombre d'opérations résiduelles est le plus grand.
- **EDD (Earliest Due Date)** : sélectionne la tâche dont la date de fin désirée est la plus proche.
- **FIFO (First In First Out)** : sélectionne la première tâche arrivée dans la file d'attente.
- **LIFO (Last In Last Out)** : sélectionne la dernière tâche arrivée dans la file d'attente.
- **SPT/WKR** : sélectionne la tâche dont le rapport de la prochaine durée opératoire sur le temps opératoire résiduel est le plus faible.
- **SPT/TWORK** : sélectionne la tâche dont le rapport de la prochaine durée opératoire sur le temps opératoire total est le plus faible.

Les règles dynamiques

On désigne par règles dynamiques les règles de décision pour lesquelles l'instant t où doit se faire le choix intervient directement ou indirectement dans le calcul.

Nous donnons ci-dessous quelques exemples de ce type de règles :

- **SLACK (Slack Time)** : sélectionne la tâche dont la somme des temps disponibles, pour les opérations restantes jusqu'à la date de fin désirée est minimale.
- **S/OPN** : sélectionne la tâche dont le rapport du Slack au nombre d'opérations résiduelles est le plus faible.
- **RSPT1** : sélectionne la tâche pour laquelle le rapport du slack au temps opératoire résiduel est le plus faible.
- **WINQ (Least Work In Next Queue)** : sélectionne la tâche qui, pour sa prochaine opération, doit aller sur la machine qui a le moins de travail.
- **XWINQ (Least Expected Work In Next Queue)** : sélectionne la tâche qui, pour sa prochaine opération, doit aller sur la machine qui a le moins de travail potentiel ou attendu.

Les règles pondérées

On désigne par règles pondérées ou combinées des règles de décision qui essaient de profiter des effets spécifiques de certaines règles simples pour les composer en une seule règle. Dans la pratique, cela revient à éclater une file d'attente en deux ou plusieurs groupes auxquels on applique des règles différentes suivant les circonstances.

Nous donnons ci-dessous quelques exemples de ce type de règles :

- **FIFO/SPT** : si toutes les tâches en attente sont présentes dans cette file depuis une durée inférieure à un seuil clairement établi, on applique la règle SPT. Sinon, pour les tâches qui ont déjà attendu depuis une durée supérieure à la valeur du seuil, on applique la règle FIFO.
- **FIFO(u)/SPT** : sélectionne en fonction de la règle FIFO si le nombre de tâches en attente est inférieur à une certaine valeur u , sinon sélectionne par la règle SPT.
- **WINQ/SPT** : sélectionne la tâche dont la prochaine opération se fait sur la file d'attente la moins chargée selon la règle WINQ. Si toutes les files d'attente destinations sont très chargées, la sélection se fera suivant la règle SPT.
- **WINQ/EDD** : sélectionne selon la règle EDD parmi les tâches arrivées à leur dernière opération et qui vont donc quitter l'atelier ou les tâches qui doivent aller vers une machine présentant une faible file d'attente.
- **LPT(u)/SPT(u)** : sélectionne selon la règle SPT si le nombre de tâches en attente est inférieur à une certaine valeur, sinon, sélectionne selon la règle LPT.
- **FIFO(t1)/SPT(t2)** : sélectionne de façon cyclique, suivant SPT durant un premier intervalle de temps, et suivant la règle FIFO dans le second intervalle de temps.
- **(u)*SPT + (1-u)*WINQ** : sélectionne la tâche pour laquelle la somme de la valeur pondérée de l'opération à réaliser et de la valeur pondérée de la somme de travail dans la prochaine file d'attente est la plus faible.
- **(u)*SPT + (1-u)*S/OPN** : sélectionne la tâche présentant la plus petite somme pondérée du prochain temps opératoire et de la marge par opération restante.

- **MSR** : sélectionne la tâche pour laquelle la date de fin désirée plus la valeur pondérée de la prochaine opération moins la valeur pondérée du travail restant est minimale.

Les résultats de recherche obtenus par simulation d'ateliers fictifs ont permis de valider ces règles. Les plans d'expérience élaborés comportent généralement les étapes suivantes :

[GERMAIN - 86]

- définition de l'ensemble des règles de priorité locales à tester,
- définition des variables d'observation du système vis à vis de ces performances,
- spécification des paramètres du système (nombre de machines, fréquence d'arrivée des commandes, nombre d'opérations par gamme, temps opératoire, règles d'affectation des dates de fin désirées),
- définition des modes de variation des différents paramètres du système, au cours de l'expérimentation,
- simulations de ces différentes configurations du système, en testant par simulation, pour chaque configuration, les effets des différentes règles,
- comparaison des résultats à l'aide des variables d'observation.

Résumer les résultats de ces travaux revient à parler de deux règles qui semblent nettement supérieures aux autres : SPT et S/OPN.

La règle SPT présente l'avantage de réguler le flot moyen des produits dans l'atelier. Tous les produits ayant la même probabilité de présenter à un instant donné une tâche disponible de plus petite durée opératoire, cette règle ne favorise pas les produits par rapport à leur nombre de tâches, elle les mélange et les fait progresser en même temps dans l'atelier. Elle donne de bonnes performances vis à vis de la minimisation de la durée d'achèvement. Toutefois, si elle donne de bons résultats sur le retard moyen, elle conduit à des retards très importants pour certains produits.

La règle S/OPN prend en compte la marge de sécurité restant avant la date de fin désirée et le nombre d'opérations restant à réaliser pour apprécier le niveau d'"urgence" de la tâche. De ce fait, elle donne de meilleurs résultats vis à vis du critère du retard, en particulier sur le nombre de tâches en retard et sur la variance du retard. Toutefois, il semble que ses résultats soient perturbés dès lors que le taux d'engagement de l'atelier est élevé.

Les performances de la règle FIFO apparaissent très médiocres et semblent voisines de celles de la règle RANDOM. De même, des règles comme WINQ et XWINQ ne présentent pas de performances très encourageantes.

Il semble donc difficile d'établir des résultats généraux, indépendants des choix faits lors de la spécification des paramètres du système et de définir les effets des différentes règles vis à vis des objectifs et critères de gestion.

III - 1 - 3 Conclusion

Les outils classiques d'ordonnement ne répondent pas correctement au problème de la commande ou de la conduite des ateliers flexibles réels. Les hypothèses des méthodes de résolution du problème statique ne sont pas réalistes, car trop simplistes vis à vis de la complexité du problème réel. Les résultats des méthodes de résolution du problème dynamique sont difficilement généralisables mais semblent toutefois les plus adaptées au caractère dynamique et temporel du problème de l'ordonnement.

Dans le contexte réel de la commande et de la conduite d'un atelier flexible, l'arrivée fréquente d'aléas comme les pannes ou l'apparition d'une nouvelle demande de fabrication rend irréalizable l'ordonnement calculé. Il faut alors soit recalculer un plan complet d'ordonnement, ce qui nécessite d'importants temps de calcul, incompatibles avec les contraintes de production de l'atelier, soit se limiter à des actualisations périodiques de façon à tenir compte du cumul des aléas passés.

Dans le cas d'un atelier réel, les spécifications étant définies par les contraintes de production, l'approche par simulation, avec utilisation des règles de priorité nous a paru intéressante pour l'étude de l'ordonnement. Elle permet de définir des stratégies "temps réel" de lancement de production ainsi que des règles locales de gestion des machines. Par ailleurs, l'utilisation de la simulation comme outil d'aide à la prise de décision temps réel dans le cas d'un environnement perturbé peut répondre au problème de conduite de l'atelier.

L'objectif n'est plus alors de définir un ordonnancement optimal, mais plutôt de fournir au décideur un outil informatique qui lui permettra de tester les conséquences de chaque décision qu'il envisage de prendre.

La difficulté réside cependant dans la pertinence des règles de décision à évaluer et dans l'évaluation de leur adéquation vis à vis des différents critères de gestion. Cette approche présente l'avantage d'exploiter la connaissance et l'expérience des hommes de l'entreprise :

- le chef de production qui connaît le comportement de son outil de production,
- le responsable de l'ordonnancement qui connaît les contraintes de la production d'une part et l'ensemble des règles potentielles de résolution d'autre part,
- le chef d'entreprise qui connaît les priorités qu'il veut donner et les critères de gestion qu'il souhaite respecter.

Nous développerons cette approche au cours des travaux présentés par la suite, avec une méthodologie de formalisation et d'exploitation de cette connaissance.

III - 2 APPOINT DE L'INTELLIGENCE ARTIFICIELLE

III - 2 - 1 Différence entre approche procédurale et approche intelligence artificielle

Jusqu'à une date récente, la plupart des langages de programmation utilisés pour mettre en oeuvre des méthodes d'ordonnancement étaient de type procédural, ce qui nécessite une parfaite connaissance des différentes étapes de la méthode et des directions que le processus algorithmique peut suivre. Les algorithmes, pour lesquels tous les cas ont été envisagés, sont alors déterminés une fois pour toutes.

La mise à jour de tels programmes réclame beaucoup de temps et rend inenvisageable une amélioration ultérieure qui alourdirait considérablement son fonctionnement.

L'originalité des techniques de programmation de l'Intelligence Artificielle consiste en :

- La définition de l'ensemble des connaissances, livrées en vrac, qui représentent l'expertise du domaine.
- La définition d'un ensemble de procédures chargées d'utiliser ces connaissances pour résoudre le problème posé par l'utilisateur.

La séparation très nette de ces deux ensembles permet d'isoler la structure de contrôle de la partie déclaration des connaissances. Il est ainsi possible d'augmenter ou de modifier le système, de l'extérieur, à tout instant, sans risque de répercussions indésirables que l'on rencontrerait inmanquablement dans les programmes classiques, de part leur caractère séquentiel. D'où le caractère très modulaire de tels programmes.

Une autre originalité tient dans le caractère déclaratif de la programmation "intelligente". Les connaissances sont entrées sous forme de représentations : règles, réseaux sémantiques, et sont délivrées en vrac par l'expert. Les unités de connaissances sont indépendantes et ne communiquent entre elles que par la seule base des faits. La notion d'ordre disparaît permettant des remises en cause de certains faits et des retours en arrière.

Nous développerons par la suite l'approche par Systèmes Experts qui est la principale voie des applications actuelles de l'intelligence artificielle pour les problèmes d'ordonnement.

III - 2 - 2 Les systèmes experts

Les Systèmes Experts sont une famille particulière des langages d'intelligence" artificielle particulièrement dédiés à un domaine, où la connaissance a été clairement définie par l'expert humain. Le système se comporte alors comme l'expert, en se basant sur les connaissances spécifiques de celui-ci, plutôt que sur des principes généraux logiques ou algorithmiques.

Les systèmes experts sont basés sur l'interaction d'une base de connaissances, qui contient l'ensemble des connaissances de l'expert, et d'un moteur d'inférence chargé de reproduire les mécanismes du raisonnement.

Le schéma de la figure III-1 illustre les différents éléments constitutifs d'un système expert, que nous détaillons par la suite.

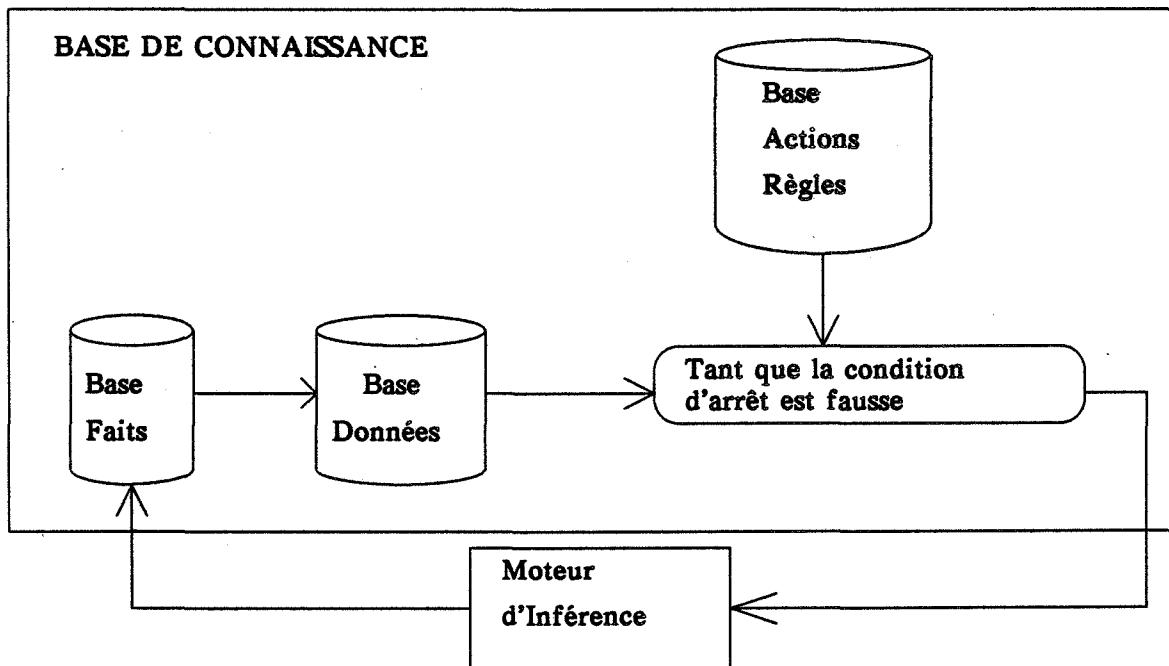


figure III-1: Structure d'un Système Expert

III - 2 - 2 - a La base de connaissances

La base de connaissances contient l'ensemble des informations nécessaires à l'expert lors de sa prise de décision.

Elle est composée, suivant une conceptualisation abstraite [FARRENY - 85], par :

- Une base de données qui comporte les déclarations et les spécifications des objets, des concepts, des buts, des états, des hypothèses et des faits qui décrivent le domaine d'application du système. Dans le cas d'un système expert de conduite, le but peut être de donner la prochaine tâche à réaliser sur la ressource conflictuelle. Les faits peuvent décrire l'état de l'atelier au moment de la prise de décision.

- Un ensemble d'actions ou de règles qui manipulent les objets, les concepts, ou les états de la base de données. Les règles comportent une partie SI qui définit la situation, les conditions ou prémisses d'application de la règle, et une partie ALORS qui définit les actions à entreprendre. Ces éléments peuvent être utilisés lors d'un cycle de contrôle ou inférence, pour produire une modification de la base de connaissance elle-même. Cela peut consister en la génération de nouveaux faits à partir de faits déjà existants ; créer de nouveaux objets, en supprimer, en modifier, transformer des hypothèses vérifiées en faits, définir de nouveaux buts, etc... Dans le cas d'un système expert de conduite, ces règles caractérisent l'ensemble des décisions que l'expert peut prendre, sous la forme des règles que nous avons vues précédemment : FIFO, SPT, EDD, WINQ,...

La représentation de la connaissance de l'expert par des règles confère au système expert toute sa modularité, une règle individuelle de la base de connaissances peut être modifiée effacée, ou ajoutée indépendamment des autres. On parle parfois d'îlot de connaissance pour illustrer le caractère autonome et indépendant de la description par règles.

Cette représentation est particulièrement adaptée aux domaines où la connaissance est "diffuse" et "éparpillée", et où les actions à entreprendre sont indépendantes, ce qui est le cas des problèmes d'ordonnancement et de pilotage.

- Une base de faits qui représente la description dynamique du système au cours de l'inférence. Elle comprend des variables globales (valeurs calculées, buts, données d'entrée,...), un agenda décrivant la liste des actions devant être exécutées par le système, un historique de l'inférence décrivant le chemin lors du processus de raisonnement.

- Une condition d'arrêt qui permet au mécanisme d'inférence de savoir si la résolution du problème est ou non achevée.

III - 2 - 2 - b Le moteur d'inférence

Le moteur d'inférence est chargé de reproduire les mécanismes du raisonnement ou plus précisément, la façon dont l'expert utilise sa connaissance. Il appelle les règles validées et accumule les nouveaux faits déduits dans un tampon.

On a l'habitude de faire apparaître trois phases pour décrire l'utilisation des connaissances :

- L'acquisition de connaissances supplémentaires qui pose le problème sous-jacent du contrôle de cohérence.
- L'extraction de la base de connaissances de l'ensemble des faits et règles validées par l'état actuel du système.
- Le raisonnement qui permet de déduire et de vérifier les nouveaux faits à partir de ceux qui le sont déjà.

Nous distinguerons alors deux types de raisonnement :

- le raisonnement en chaînage avant, (Forward Reasoning) ou dirigés par les données qui consiste en l'enchaînement des données avérées vraies à partir de l'état initial du système pour atteindre une conclusion ou un but qui soient solution du problème.
- Le raisonnement en chaînage arrière, (Backward Reasoning) ou dirigé par les buts qui consiste à l'inverse, à partir d'une conclusion ou d'un but que l'on cherche à tester, et à remonter vers les faits ou les données qui pourraient la confirmer ou l'infirmer.

La plupart des systèmes experts combinent ces deux approches en un chaînage mixte pour répondre au mieux aux problèmes réels où il est fréquent d'avoir à la fois des informations diverses sur le problème à résoudre, et des intérêts particuliers ou présomptions pour certaines conclusions.

Devant la complexité des structures à mettre en place pour réaliser et contrôler un moteur d'inférence, on a vu se développer un ensemble d'outils appelés "générateurs de systèmes experts" qui fournissent un moteur d'inférence indépendant. Ceci permet à l'utilisateur de remplir sa base de connaissance en fonction de son domaine d'application, sans se soucier de la réalisation informatique du moteur d'inférence.

III - 2 - 3 Les outils d'aide à la décision en ordonnancement

Comme nous l'avons vu précédemment, l'approche par système expert semble pouvoir s'appliquer aux problèmes d'ordonnancement, pour lesquels la connaissance est "diffuse" et "éparpillée" dans les mains des différents experts de l'entreprise.

Cette approche a donné lieu à de nombreux travaux de recherche et à des études toujours en cours.

Le système expert ORDF [ROCHE - 84], construit à partir du générateur de systèmes experts EAQUE, est dédié aux problèmes de partage de ressources en nombre limité pour un problème d'ordonnancement statique. Il résout les conflits rencontrés, lors de la construction du graphe de conception de l'ensemble des tâches à réaliser, à partir de règles heuristiques de priorité.

Le système expert ISIS [FOX - 84] est le premier système à avoir été utilisé dans l'industrie, pour un atelier de production de composants de turbines de la firme Westinghouse. Les techniques d'ordonnancement ne sont ni des modèles d'optimisation, ni des techniques issues de l'approche par règles de priorité, puisqu'il ne simule pas le fonctionnement de l'atelier. Il simule plutôt la démarche de l'expert en ordonnancement travaillant sur un diagramme de Gantt. Le graphique illustre les contraintes ; délais possibles de livraison, niveau des en-cours, ressources disponibles, intervalles libres des machines,... Il tient également compte des contraintes opérationnelles de fabrication ; gammes multiples, relations de précédence,... Pour planifier un ordre, le système effectue une recherche heuristique dans l'espace des plans possibles en tenant compte des gammes et des contraintes, l'objectif étant de placer l'ordre dans les intervalles libres du diagramme de Gantt.

Le système expert SOJA [LEPAPE - 85] est un système d'ordonnancement journalier d'atelier qui étudie la capacité d'un atelier à partir des coefficients de productivité des machines, des pannes connues et des arrêts provisoires prévus, de la disponibilité du personnel également. A partir de la capacité de l'atelier, des délais à respecter et des différents critères de gestion à respecter, il sélectionne les opérations à exécuter suivant des règles heuristiques de priorité, avant de les ordonnancer par la construction d'un treillis de sélection suivant les techniques des graphes potentiel-tâches.

Le système expert OPAL [BEL - 87] [BEL - 88] intègre et fait coopérer des méthodes classiques d'ordonnancement telles que l'analyse sous contraintes de délai et de ressources [ERSCHLER - 76], la recherche ordonnée de solutions, les règles de priorité et des règles de séquençement, connaissances expertes issues de l'expérience du chef d'atelier. Il établit un ordonnancement prévisionnel qui ne tient pas compte des pannes.

Le système expert MASCOT [ESQUIROL - 87] concerne plus particulièrement le problème de la modélisation des effets des interactions entre les différentes tâches, dues au partage des ressources et à la limitation des intervalles de temps alloués. La limitation des capacités des ressources est à l'origine de conflits entre les différentes tâches utilisatrices. La résolution de ces conflits se traduit par des conditions logiques nécessaires de séquençement entre les tâches concurrentes qui peuvent être exprimées de façon analogue aux contraintes de séquençement des gammes. Il caractérise alors l'ensemble des ordonnancements admissibles à travers des faits séquentiels et temporels auxquels il applique les techniques de l'analyse sous contraintes.

Ces différents systèmes apportent une aide à la décision pour l'ordonnancement d'atelier. Toutefois, nous devons signaler que dans le cas d'aléas survenant lors de la fabrication, le plan d'ordonnancement doit être recalculé de façon globale pour tenir compte des nouvelles données du problème. Aucun de ces systèmes ne prend en compte l'évolution dynamique de l'état de l'atelier, au cours de son inférence.

Nous citerons quelques travaux de recherche visant à construire non plus des systèmes d'aide à l'ordonnancement seulement, mais des systèmes d'aide à la conduite qui se caractérisent alors par un suivi de l'état de l'atelier.

Une première approche développée dans le projet SAGEP [VOYATZIS - 87] est basée sur l'utilisation de la mémoire artificielle. Ceci consiste à stocker intelligemment dans la mémoire d'un ordinateur des états du système de production et des règles de gestion possibles avec les conséquences qu'elles engendrent pour chacun de ces états. Lorsque l'atelier atteint l'un de ces états, on cherche alors à rattacher cette situation à une situation connue, en recherchant en mémoire, pour chaque règle répertoriée, les conséquences probables de son application. La réalisation et la construction de la mémoire artificielle reposent sur des techniques de l'analyse de données (classification automatique, analyse discriminante, classification hiérarchique et analyse en composantes principales) plus que sur des techniques d'intelligence artificielle.

La deuxième approche est basée sur l'utilisation d'une base de données directement reliée au système de commande de l'atelier. Nous citerons le système expert SYGAM [ARNOUX - 86] qui permet de prendre en compte les aléas de production et leur traitement dans le cadre d'une gestion de production assistée par ordinateur. Le système se présente comme une interface intelligente entre l'atelier et la GPAO ; il récupère l'ensemble des données caractérisant l'état de l'atelier afin de diagnostiquer la cause des aléas, de faire un historique des aléas pour enrichir la base de connaissances, et de déterminer l'ensemble des informations pertinentes pour la GPAO. L'expertise recueillie auprès des spécialistes permet de décider si la GPAO doit procéder à un nouvel ordonnancement des tâches tenant compte de la dégradation observée, ou si une modification de l'ordre de fabrication initial est suffisant.

La dernière approche est basée sur l'utilisation de la simulation pour tester les effets des différentes décisions. Nous citerons le projet PROMIP qui contient un module de simulation intelligente PETREL [ATABAKHCHE - 87]. La simulation est utilisée pour suivre le fonctionnement de l'atelier. En cas d'incident, le moteur d'inférence est appelé pour faire un diagnostic puis choisir un traitement d'exception. Si l'accident est de faible conséquence, le jeu de règles permet d'adapter le plan de production aux nouvelles contraintes. Sinon, le système OPAL cité précédemment est appelé pour définir une nouvelle planification.

L'approche par simulation présente l'intérêt de pouvoir prévoir le comportement de l'atelier face à des aléas, mais aussi face aux différentes règles de décision et de pilotage testées. Elle permet de mettre en évidence les différents points de décision au niveau local en tenant compte de l'état réel de l'atelier. Le problème de l'ordonnancement se ramène alors à un problème de décisions successives prises en temps réel. L'apport d'une approche par système expert est d'aider le décideur dans le choix d'une action compatible avec les contraintes du problème, l'état de l'atelier, qui satisfasse au mieux les critères de gestion.

III - 3 APPROCHE RETENUE

Le simulateur SAGASSE est un système d'aide à la conception et à la conduite des ateliers flexibles qui intègre les aspects fonctionnement de l'atelier et gestion de production réduite au routage et à l'ordonnancement des pièces.

Il s'agit de réagir à tout moment, selon un processus temps réel, à toute perturbation survenue au sein de l'atelier ou au niveau du carnet de commandes. Dans le premier cas, nous considérons les modifications de l'état des ressources, pannes, stocks saturés, maintenance préventive. Dans le deuxième cas, nous prenons en compte tout changement du plan de fabrication décidé à la suite d'une commande urgente qui bouleverse les priorités et exige une modification de la politique de lancement. Le système proposé effectue ce choix en calculant, pour chaque événement si nécessaire, un nouvel ordonnancement tenant compte de l'état atteint par l'atelier et de la nature de l'événement. Le nouvel ordonnancement est alors immédiatement pris en compte, pour une évaluation dynamique par la simulation.

L'approche retenue pour le système SAGASSE repose sur l'interaction dynamique entre l'interpréteur RdPCT présenté précédemment, et un système expert garant du pilotage et de la gestion de l'atelier.

Lorsque l'interpréteur rencontre un conflit (décision non résolue au niveau du modèle) ou lorsqu'il détecte un mode de fonctionnement critique (arrivée d'une commande urgente, machine en panne, moyen de transport saturé, stock saturé, état prédéfini,...) il arrête la simulation, et "passe la main" au système expert chargé de prendre la décision. L'interpréteur met en évidence le point de décision, mais ne le traite pas, il déclenche la consultation des différents experts concernés et attend leur réponse pour prendre en compte l'apparition des aléas au niveau de l'atelier. Voir figure III-2.

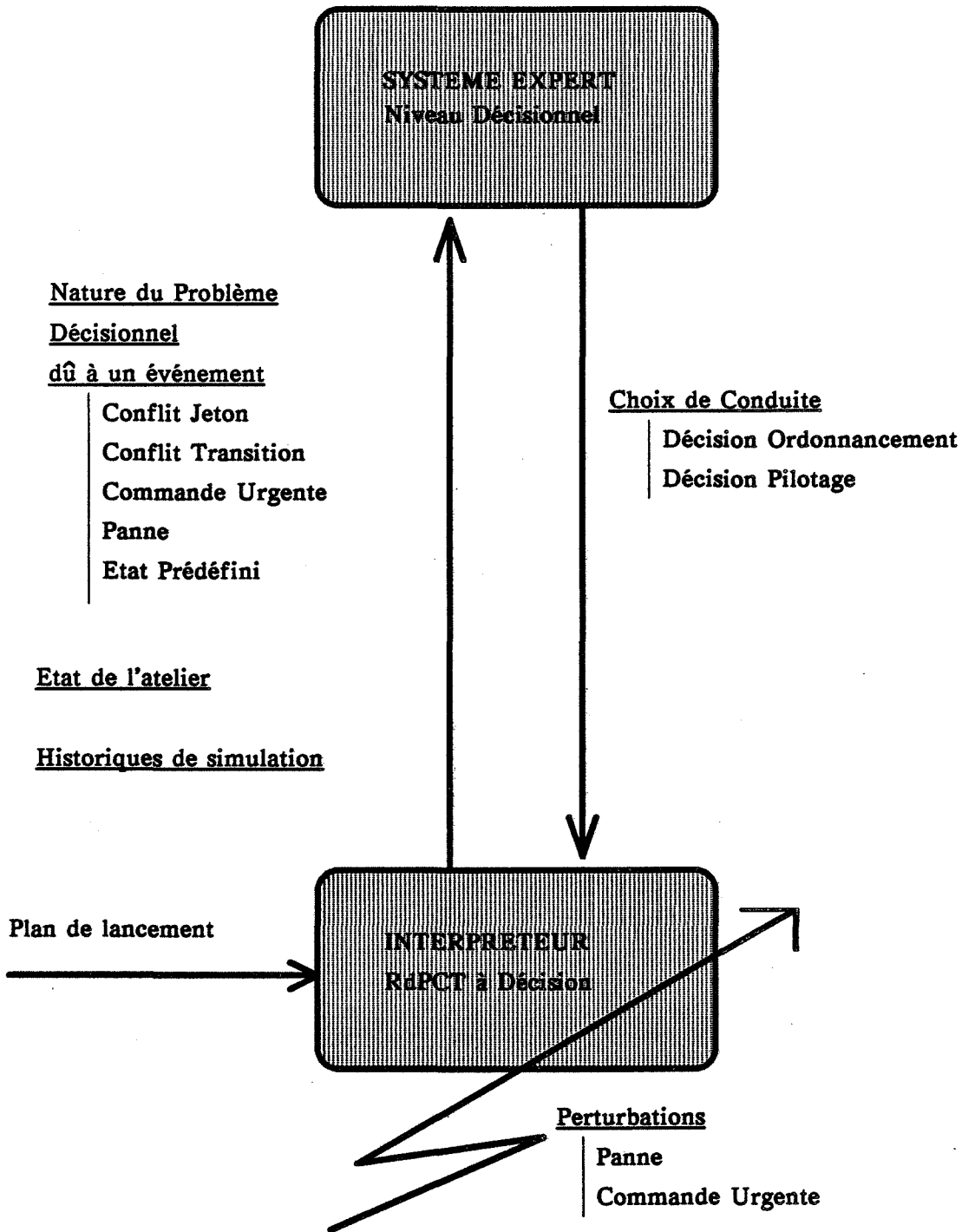


figure III-2: Le simulateur SAGASSE

Comme nous avons pu le voir, la problématique du système expert en ordonnancement ou en pilotage réside en trois points :

- Le caractère fortement temporel du problème à résoudre. Un ordonnancement peut devenir totalement irréalisable par l'apparition d'une panne machine, l'arrivée d'un lot urgent. Pour conserver son réalisme, l'ordonnancement doit donc être fait localement, à l'instant où se pose le choix, par des règles de priorité heuristiques, en relation avec l'outil de suivi de l'état de l'atelier, ici la simulation.
- La diversité de nature du savoir des différents experts. Le chef de production connaît le comportement spécifique de son outil de production. Il gère les contraintes technologiques, pannes machines,...., les contraintes humaines, compétence du personnel, pour garantir la qualité du produit. Le responsable de l'ordonnancement connaît les contraintes de la production d'une part et l'ensemble des règles potentielles de résolution d'autre part. Il travaille sur un atelier au fonctionnement idéal, où les aléas n'existent pas. Le chef d'entreprise connaît les priorités qu'il veut donner et les critères de gestion qu'il souhaite respecter. Il suit les différents indicateurs de gestion pour évaluer la productivité et la flexibilité de son entreprise.
- L'imprécision du savoir due au caractère empirique de la connaissance. Comme nous l'avons vu précédemment, il n'existe pas de solution optimale satisfaisant toutes les conditions. Seule l'expérience et des connaissances heuristiques à base de règles de décisions permettent d'avoir des solutions admissibles qu'il est très difficile de généraliser. La difficulté est alors d'évaluer la pertinence des règles de décision vis à vis d'une situation et vis à vis des critères de gestion.

Ces différents éléments nous ont conduit à élaborer un environnement de programmation qui puisse intégrer les différents savoirs, en aidant les experts à formuler, puis à mettre en commun leurs connaissances. Cet environnement doit être suffisamment souple pour permettre de modifier et de corriger les différents paramètres, mais doit aussi être suffisamment simple pour permettre une expression de la connaissance sous forme linguistique. Nous avons pour cela utilisé une approche par langage objet que nous détaillons dans le paragraphe suivant.

III - 3 - 1 Les langages objets

La modélisation du système d'information par l'approche orientée objet permet de donner une représentation de la connaissance sous forme déclarative et procédurale [FARRENY - 87].

Le concept essentiel est celui d'objet qui permet de représenter, d'associer, en une seule entité, les données et leurs procédures d'exploitation.

Un objet est défini comme un atome auquel est rattaché un ensemble de propriétés. Une propriété est une caractéristique d'un objet désigné par un nom, pouvant correspondre à un attribut qui peut prendre une valeur, une fonction, ou un sous-objet composant.

Une classe est un objet possédant une liste d'héritage.

Une instance est un objet appartenant à une classe et dont les attributs sont valorisés.

Un objet est une instance de la classe à laquelle il appartient.

Une classe d'objets spécifie la structure et le comportement commun à un ensemble d'objets. Les objets d'une même classe possèdent le même ensemble de propriétés et ne diffèrent que par les valeurs de celle-ci. La spécification progressive des classes d'objets composant une base de données objets permet de modéliser le comportement commun des objets de manière modulaire et extensible.

Le modèle orienté objet permet non seulement de modéliser des objets, mais aussi de modéliser leur comportement, en spécifiant les procédures aussi appelées méthodes qui manipulent les objets. Une méthode est une opération associée à une classe qui manipule ou retourne l'état d'un objet ou d'une partie d'objet de la classe.

Exemple :

Un atelier d'usinage comprend 5 stocks intermédiaires en amont des machines de traitement, des stocks de matières premières, des stocks d'outils, un stock de produits finis.

Un stock intermédiaire particulier peut être décrit par un objet ayant :

- un nom : "Stock 2"
- une classe : "Stock intermédiaire"
- des propriétés :
 - "amont-de" qui désigne la machine qu'il dessert.
 - "capacité- max" qui désigne la capacité du stock.
 - "valeur-actuelle" qui contient le nombre de pièces actuellement en stock.
 - ...
- des méthodes :
 - une méthode de "déchargement" permettant au stock d'enregistrer et de satisfaire une demande de chargement de la machine. Cette méthode a le nom de la machine comme argument.
 - une méthode de "temps de stockage" permettant au stock de calculer le temps depuis lequel une pièce donnée est en stock. Cette méthode a le nom de la pièce émettrice comme argument.
 - ...

Les objets communiquent entre eux par des envois de messages comportant le nom de la méthode, l'objet destinataire et les paramètres du message. Un objet réagit à un message en exécutant la méthode correspondante et en retournant les paramètres retour de la méthode, en général sous forme d'un message envoyé à la méthode appelante.

Afin d'éviter la répétition de toutes les propriétés, les attributs et les méthodes pour chaque classe d'objets, il est possible d'utiliser les propriétés d'héritage empruntées aux modèles sémantiques de données. L'héritage permet la transmission des propriétés d'une classe vers une sous-classe (généralisation/spécialisation). Un objet hérite des propriétés de la classe à laquelle il appartient (classification/instanciation), comme un sous-objet hérite de l'objet dont il est élément (aggrégation/décomposition).

Exemple :

"Stock2" est une instance de la classe "Stock-intermédiaire" qui est elle-même une sous classe de la classe "Stock", tandis que les n positions "position-i" sont des sous-objets de "stock2".

Le schéma de la figure III-3 à la page suivante, illustre un exemple d'héritage entre ces différentes entités.

Les concepts fondamentaux de modélisation des données, instanciation et généralisation, sont construits d'après le paradigme orienté objet. Le modèle des données est donc la combinaison d'objets indépendants et d'un réseau sémantique défini intrinséquement. A un niveau, les noeuds du réseau représentent les objets abstraits ou classes, tandis qu'au niveau du détail, ces noeuds contiennent des données physiques.

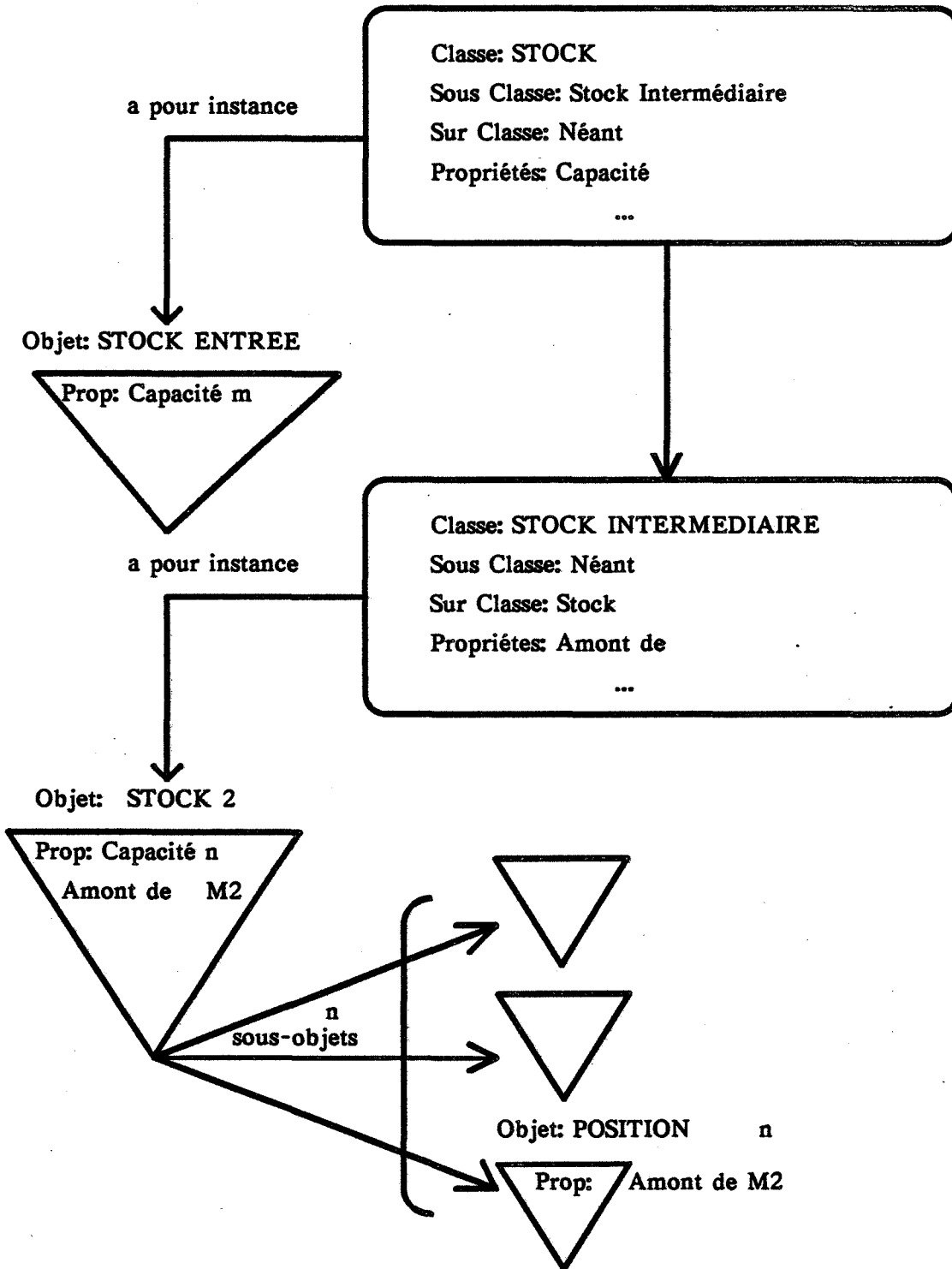


figure III-3: Illustration des propriétés d'héritage

III - 3 - 2 Organisation de la connaissance

La difficulté dans l'utilisation des règles locales de décision pour résoudre le problème de l'ordonnancement et du pilotage réside dans :

- le choix de la règle,
- la pertinence des règles de décision à évaluer par simulation,
- et l'évaluation de leur adéquation vis à vis des différents critères de gestion.

Les résultats des différents travaux de recherche n'étant pas généralisables, nous avons cherché à travailler non plus avec un atelier standard imaginaire, mais avec un atelier réel, pour lequel il semble plus facile de définir des règles spécifiques et donc de tirer des conclusions.

Par ailleurs, l'expérience de la simulation nous a montré que certaines règles de décision étaient satisfaisantes, de par leurs effets correcteurs, dans certaines situations bien précises. Nous rappelons, par exemple, les résultats de recherche en ordonnancement qui ont montré que la règle S/OPN a de bons résultats vis à vis des coûts tant que le taux d'engagement de l'atelier est inférieur à 80%.

Cette constatation nous a conduit à envisager des règles de conduite de l'atelier qui ne soient plus fixes au cours de toute la simulation (FIFO,SPT,...) mais variables en fonction de l'état de l'atelier. Cette hypothèse semble être tout à fait nouvelle dans les systèmes d'aide à la conduite. De plus, elle est compatible avec l'évolution des technologies de commande des stocks de type carrousel qui permettent le choix d'une pièce selon un critère défini par le processus central de conduite de l'atelier.

Nous proposons alors une méthode d'approche qui, s'appuyant sur le système expert, devra permettre de formaliser les conditions et limites d'application des différentes règles, pour un atelier donné, à travers un grand nombre de simulations.

L'idée est de fournir aux différents experts de l'entreprise une grille commune de formalisation de la connaissance spécifique qu'ils ont de leur atelier, afin de choisir au mieux la règle locale à appliquer à l'instant où se pose la prise de décision.

Cette grille d'analyse met en évidence les différents niveaux de connaissance nécessaires à la prise de décision en ordonnancement et fait appel à trois types d'expertise :

- La reconnaissance du type de décision :

décision d'ordonnancement dans le cas;

- d'un conflit jeton détecté par l'interpréteur,
- ou de l'arrivée d'une commande urgente,

décision de pilotage ou de routage dans le cas;

- d'un conflit transition détecté par l'interpréteur,
- de l'apparition d'une panne,
- ou encore la reconnaissance d'un état critique prédéfini par les experts.

Elle permet de caractériser la décision à prendre au niveau local.

L'expert en réseau de Petri traduit alors les données du conflit jeton/transition ou de l'état critique, ainsi que l'ensemble des choix possibles, en termes compréhensibles par le niveau décisionnel.

Pour les problèmes d'ordonnancement : Quel produit charger sur la machine disponible?

Pour l'arrivée d'une commande urgente : Quel produit lancer ?

Pour les problèmes de gamme multiple : Sur quelle machine réaliser la tâche ?

Pour les problèmes de pannes : Quelle machine utiliser ?

Pour les problèmes de routage : Quelle orientation prendre ?

- La description de la situation dans laquelle la règle s'applique, permet d'introduire des informations relatives à l'état de l'atelier n'entrant pas directement dans la description du conflit, mais qui peuvent intervenir dans le choix d'une stratégie.

Ces informations peuvent être des indicateurs historiques sur la production déjà réalisée, mis à jour à chaque instant par l'interpréteur (taux d'occupation des machines, des stocks, temps de passage des produits déjà réalisés, taux de blocage de pannes, des machines,...), ou encore des données instantanées sur l'état courant de l'atelier (niveau d'un stock, état d'une machine,...).

Comparées à des seuils établis de façon empirique ; soit par expérience, soit par simulation, elles permettent de sélectionner un ensemble de règles plus particulièrement adaptées à cette situation.

Cette étape permet de prendre en compte l'expérience de terrain du chef de production qui a une connaissance intuitive du comportement de l'atelier et des solutions à apporter.

- L'estimation du degré de pertinence d'une règle par rapport à un critère de gestion, relève de la connaissance de l'expert en ordonnancement qui au niveau du bureau des méthodes doit répondre aux objectifs du chef d'entreprise.

Les règles de priorité sont évaluées en fonction de leurs effets sur :

- le respect des délais,
- l'optimisation du taux d'occupation des machines,
- la minimisation des en-cours.

Pour cela, nous utilisons la représentation pyramidale, présentée dans le chapitre I, qui permet de situer graphiquement l'état courant de l'atelier au moment de la prise de décision, vis à vis des coûts dus aux différents critères.

Cette construction apporte à chaque instant, une visualisation des effets des différentes règles testées (cf figure I - 4). Au cours de différentes simulations, la comparaison des pyramides obtenues aide l'expert à préciser l'efficacité des actions correctrices qu'il envisage.

Cette dernière étape permet le choix de la règle parmi l'ensemble des règles sélectionnées par l'étape précédente. A ce niveau, la règle s'exprime de façon très simple comme un algorithme de tri parmi l'ensemble des choix possibles (FIFO,SPT,EDD,...).

Illustration de l'application de la grille à travers deux exemples simplifiés de règles :

Si conflit-ordonnement dans le stock-2 amont machine-2

taux-occupation.machine-2 <= 80%

niveau.stock-2 <= 80% capacité.stock-2

critère = optimisation de l'occupation machine

Alors règle S/OPN

tri parmi les choix possibles de la pièce du stock-2 dont le rapport du Slack au nombre d'opérations résiduelles est le plus faible.

Si conflit-ordonnement dans le stock-2 amont machine-2

taux-occupation.machine-2 > 80%

niveau.stock-amont-opération-suivante <= 80% capacité.stock

critère = minimisation de l'en-cours

Alors règle

tri parmi les choix possibles de la pièce du stock-2 devant aller, pour l'opération suivante de sa gamme, dans le stock amont de la machine comprenant le moins de pièces, lorsque ce stock est à moins de 80% de sa capacité, au moment de la prise de décision.

Cette grille permet par ailleurs de structurer l'ensemble des règles en faisant apparaître un ensemble de règles sur les règles, ou métarègles, qui accélèrent la résolution proprement dite.

La figure III - 4 illustre l'utilisation de métarègles.

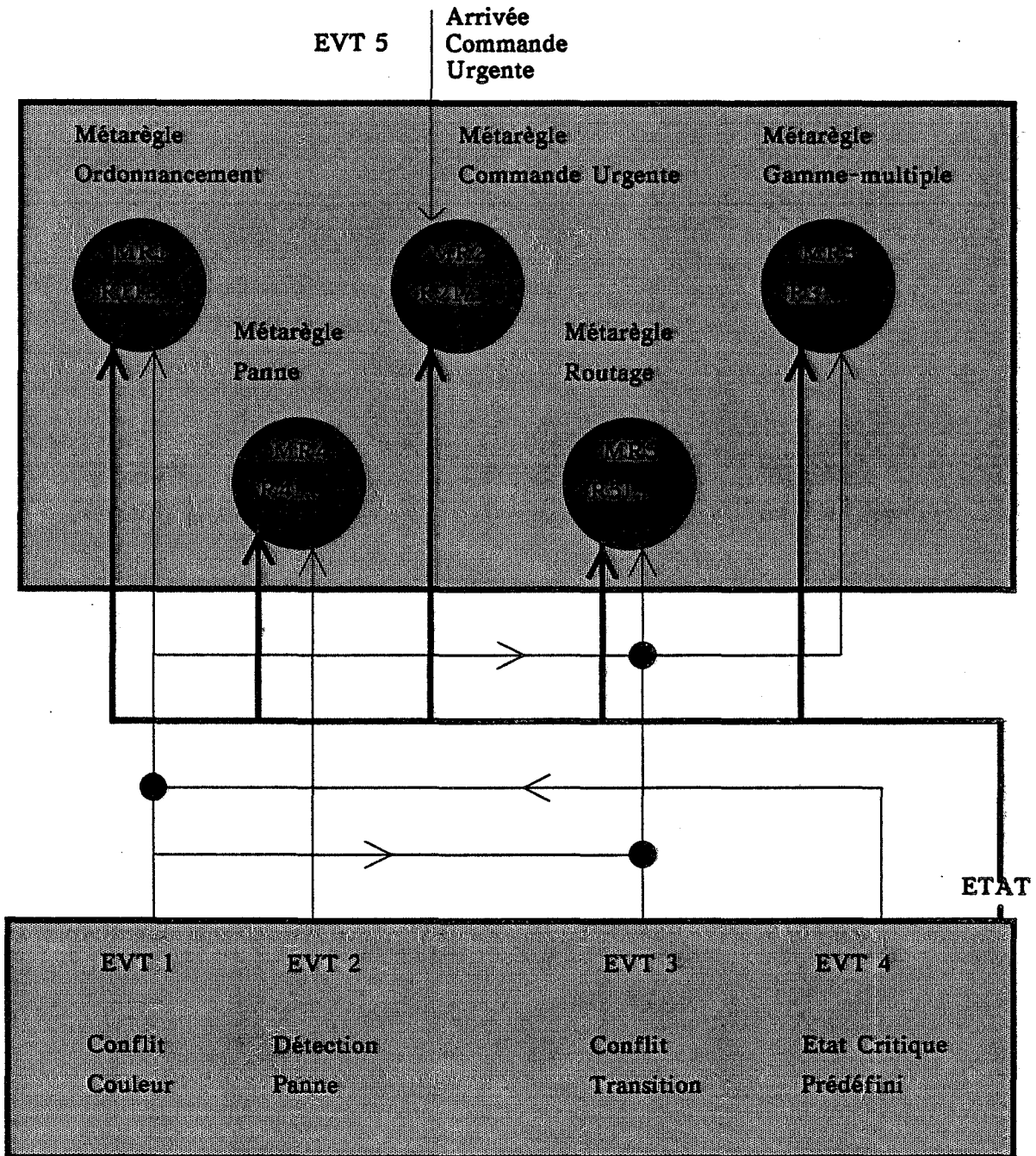


figure III-4: Illustration de l'utilisation des Métarègles

On peut remarquer que la grille marque les étapes de la décomposition hiérarchique de la prise de décision dans les problèmes de gestion de production.

Partant du niveau de description le plus bas, celui de l'atelier, elle remonte jusqu'au niveau de l'ordonnancement court terme puis du plan directeur, par le biais des critères de gestion à satisfaire.

Partant de la nature de la décision à prendre au niveau local dans l'atelier, elle prend en compte l'état instantané des ressources et les indicateurs historiques de la production déjà réalisée pour définir si la situation actuelle relève de règles de conduite connues. Le plan directeur intervient dans la construction de la pyramide, lorsque l'on compare les performances simulées (avec prise en compte des aléas) aux performances attendues (zéro retard, taux charge attendu, niveau des en-cours attendu) pour définir sur quel critère il faut entreprendre une action correctrice.

L'originalité de notre approche tient essentiellement dans son caractère ascendant qui permet une meilleure prise en compte de l'état réel de l'atelier et nous semble plus proche de ce qui se passe réellement dans l'industrie.

Une dernière difficulté réside toutefois, dans la détermination par les experts des différents seuils d'observation et des conditions d'application qui sont très souvent approximatifs et entachés d'incertitude. Nous avons alors utilisé des techniques de traitement de l'incertain.

III - 3 - 3 Traitement de l'incertain

L'implémentation informatique du module expert du prototype du système SAGASSE a été réalisée sur le générateur de systèmes experts NEXPERT-OBJECT développé par NEURON-DATA [NEURON-DATA - 88]. Ce générateur propose un certain nombre d'utilitaires permettant la représentation de l'incertain dans les systèmes à base de connaissances.

On distingue plusieurs niveaux d'incertain dans l'expression et la mise en oeuvre de la connaissance :

- La représentation de connaissances imprécises qui traduit des faits non nécessairement bien définis.
- La représentation de déductions incertaines.
- La représentation et la manipulation d'informations conflictuelles qui sont parfois contradictoires, inconsistantes, ou non-pertinentes.

Ces différents niveaux illustrent en fait, soit l'imprécision de l'information, soit l'incertitude de la connaissance. Deux méthodologies sont alors proposées :

- Les coefficients de certitude sur les faits : ils permettent de représenter des notions vagues comme l'usure d'un outil en fonction du nombre d'utilisations. Chaque donnée est associée à un facteur de certitude CF dont la valeur est comprise dans l'intervalle $[-1,1]$.

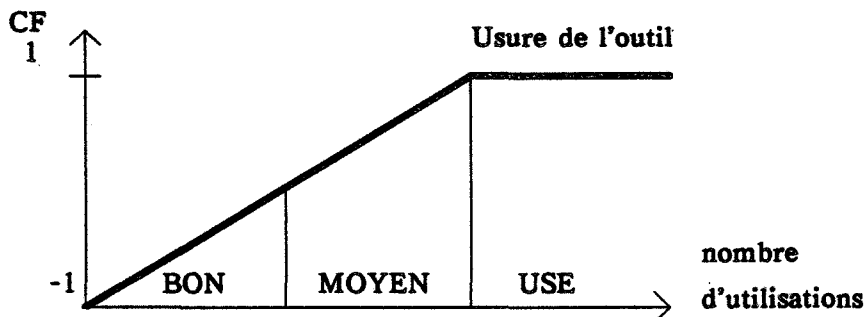


figure III-5: Coefficients de certitude sur les faits

- Les coefficients de certitude sur les règles : ils permettent de représenter le degré de fiabilité de la règle. Chaque règle est associée à un facteur de certitude CR dont la valeur est comprise dans l'intervalle [-1,1]. Deux règles ayant le même coefficient de certitude traduisent une indétermination de l'expert quant au choix de la règle la plus appropriée soit à la situation, soit au critère de gestion. Le choix entre ces deux règles se fera par l'intermédiaire du moteur d'inférence par la propagation des facteurs de certitude.

La propagation des facteurs de certitude à travers l'inférence se fait selon la formule algébrique suivante : (avec CF conclusion initialisé à 0)

$$\text{CF conclusion} = \text{CF conclusion} + [(1 - \text{CF conclusion}) * (\text{CF règle} * \text{CF conditions})]$$

Le processus d'inférence procède alors à un vote de la conclusion dont le facteur de certitude calculé de la conclusion est le plus proche de 1.

III - 3 - 4 Conclusions et Perspectives

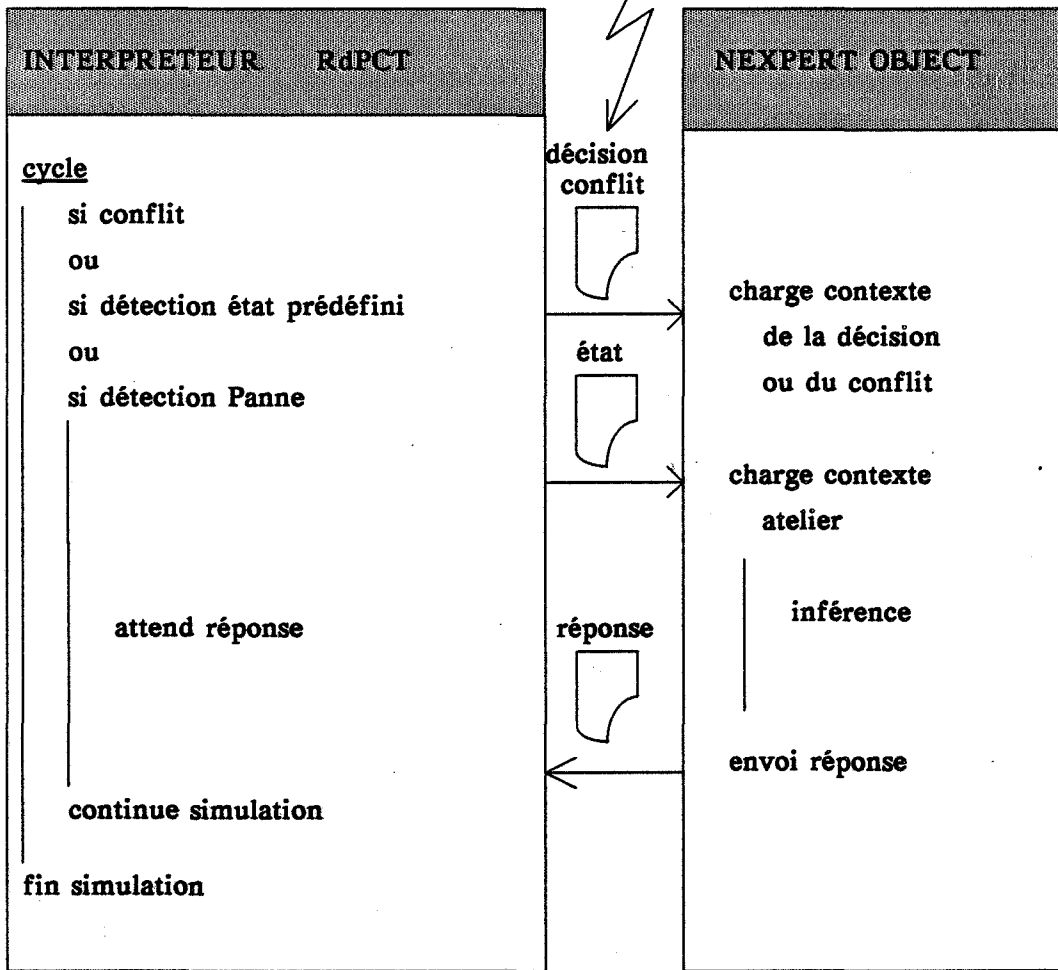
Tout en restant dans les deux niveaux inférieurs de la décomposition hiérarchique de la décision en gestion de production; l'ordonnancement court terme et le suivi de production (cf figure I -1), l'originalité de notre approche tient dans l'interaction entre les outils propres à chacun de ces niveaux. L'utilisation de l'interpréteur RdPCT répond aux problèmes liés au fonctionnement séquentiel des ateliers flexibles, tandis que les règles de gestion et de décision du système expert apportent une aide à la conduite, au niveau ordonnancement et pilotage.

L'interface entre ces deux éléments est réalisée pour une décision donnée, fonction de l'état atteint par l'interpréteur, auquel le système expert doit répondre par un choix de pilotage ou une décision d'ordonnancement.

Ensemble ils forment un système interactif qui permet au concepteur de tester, non plus le seul fonctionnement de l'atelier projeté pour une politique de gestion fixe tout au long de la simulation, mais simultanément le fonctionnement de l'atelier et les différentes politiques de gestion possibles. L'objectif n'est pas de trouver la meilleure règle dans l'absolu, mais plutôt de choisir celle qui paraît la mieux appropriée à l'état de l'atelier et aux objectifs de l'instant, définissant ainsi une sorte de "conduite adaptative" par période.

L'utilisation du système SAGASSE comme outil de simulation peut apporter une aide tant au niveau conception, pour le dimensionnement des différentes ressources, qu'au niveau conduite pour localiser les "bons" paramètres pour une situation donnée (état atelier, pertinence règle, coefficients certitudes) et tester les effets des différentes règles de décision vis à vis des différents critères de gestion.

si commande Urgente



- trace simulation
- historique simulation
- historique Résolution conflits/décisions
- résultats simulation
- diagramme Gantt

figure III-6: Structure interactive de SAGASSE

La structure même de SAGASSE; modélisation de l'atelier d'une part, et description, dans la base de connaissance, de règles spécifiques à l'entreprise d'autre part, en font un outil d'aide à la conduite par l'ordonnancement et le pilotage dédié à un atelier particulier. La base de connaissance initiale contient la structure des objets et un nombre minimal de règles. Ce sont les experts de l'entreprise, qui, au cours de campagnes de simulation, vont affiner leurs connaissances, en créant leur propre base de règles.

L'aspect convivial de l'approche objet dans la description de la connaissance permet à l'utilisateur de paramétrer et de modifier aisément les objets mais aussi les règles du système expert.

Les expériences de validation industrielle du prototype de SAGASSE, développées dans le chapitre IV, nous ont permis de nous rendre compte qu'il pouvait être un outil intéressant, du point de vue humain, pour la formalisation et la mise en commun des différentes connaissances des experts, trop souvent cloisonnées par la structure hiérarchique de l'entreprise.

Il semble toutefois, que les techniques d'apprentissage automatique, de plus en plus utilisées dans les langages d'intelligence artificielle, pourraient apporter une grande aide, au niveau de l'ajustement des différents paramètres ou seuils intervenants dans l'écriture des règles, ou dans l'ajustement des coefficients de certitude des faits et règles utilisées. Les techniques d'apprentissage par expérience et les techniques d'apprentissage par exemples apparaissent comme une perspective de développements ultérieurs du prototype déjà réalisé.

CHAPITRE IV

EXEMPLES D'APPLICATIONS

VALIDATION INDUSTRIELLE

Dans ce dernier chapitre, nous présentons trois exemples d'application industrielle du simulateur SAGASSE qui nous ont permis de valider notre approche en construisant des bases de connaissances adaptées à chaque type d'atelier. Chaque type d'atelier est lui même caractéristique d'une classe de problèmes rencontrés lors de la conduite d'un atelier flexible.

Pour des raisons de confidentialité, nous avons simplifié chacun des ateliers pour ne faire apparaître que les points spécifiques caractérisant un type d'utilisation du simulateur SAGASSE.

Le premier atelier est un exemple d'utilisation du simulateur pour des problèmes d'ordonnancement dans le cas de gammes multiples. Les temps de transfert entre machines sont pris en compte, mais on ne représente pas la disponibilité des ressources de transport engagées qui sont donc supposées infinies.

Le deuxième atelier illustre l'utilisation du simulateur pour des problèmes de pilotage proprement dit. Les ressources de transport représentent une ressource critique du système, elles devront être prises en compte pour déterminer des stratégies de routage pertinentes.

Le troisième atelier illustre l'utilisation du simulateur pour des problèmes plus généraux de conduite incluant l'ordonnancement et le pilotage dans le cas d'un atelier de fabrication en continu où les pannes machines ne doivent en aucun cas bloquer la production.

IV - 1 EXEMPLE 1 : PROBLEME DES GAMMES MULTIPLES

L'atelier présenté est une cellule flexible composée de trois centres d'usinage M1, M2, M3, d'un stock intermédiaire SE de type carrousel, de capacité limitée à dix, contenant les produits à usiner et les produits en cours de fabrication, et d'un stock de produits finis SF supposé à capacité infinie. La figure IV-1 schématise l'organisation de l'atelier :

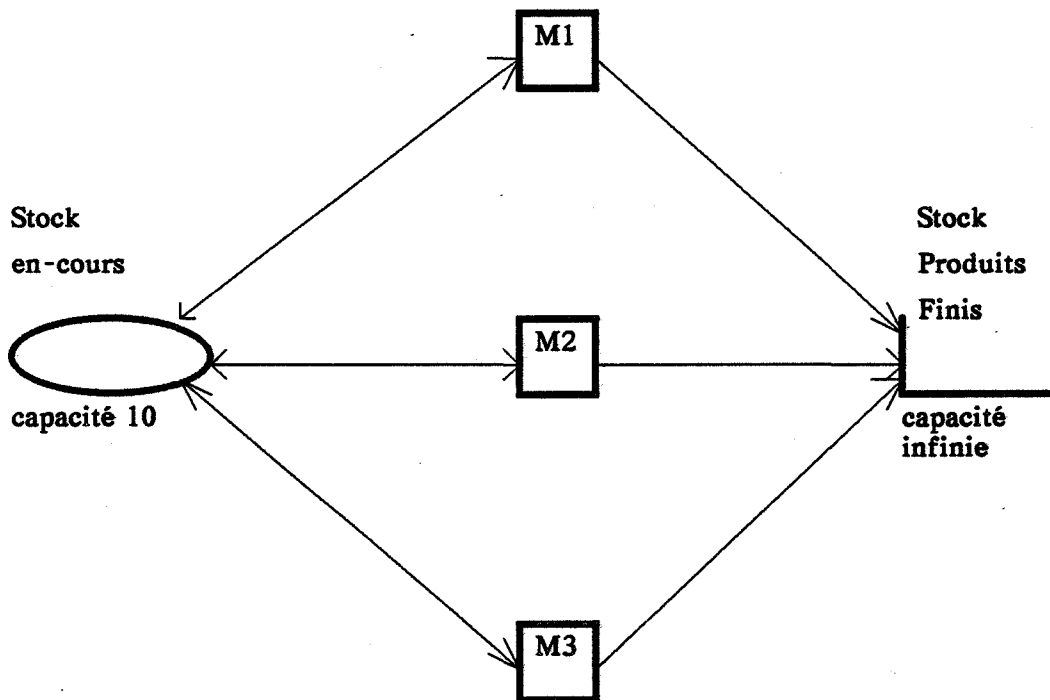


figure IV-1: Cellule 3 machines

Le système de transport entre les machines et les stocks est un robot, mais nous ne l'avons pas représenté en temps que ressource puisque nous voulons ici mettre en avant les problèmes d'ordonnancement liés aux gammes multiples. Seuls les temps de transfert, constants quelques soient le produit et la machine, sont pris en compte.

Les machines M1 et M2 sont semblables, elles peuvent réaliser certaines opérations de façon identique mais ne donnent pas les mêmes temps opératoires, M2 étant plus ancienne que M1. Toutefois, les magasins d'outils, non pris en compte dans cet exemple simplifié, expliquent que certaines opérations doivent obligatoirement être réalisées soit sur M1, soit sur M2.

Les produits P1, P2, P3 sont usinés suivant les gammes définies dans le tableau suivant:

opérations produits	op1	op2	op3
P1	M1 ou M2 10 30	M2 15	M3 8
P2	M3 13	M2 ou M1 22 32	M1 16
P3	M2 18	M3 7	M1 ou M2 15 35

Les produits sont palettisés dès leur entrée dans le stock SE. Chaque palette est caractérisée par son numéro et par le nombre de pièces qui y sont rivées et ne comprend que des pièces de même type ayant le même degré d'avancement dans la gamme. Toute palette commencée sur une machine doit être usinée complètement, elle ne retournera dans le stock SE que si toutes les pièces rivées ont fini leur usinage.

Lorsque la palette a terminé la dernière opération de sa gamme, elle est évacuée vers le stock de produits finis, et elle peut provoquer l'entrée d'une nouvelle palette de pièces à usiner dans le stock SE.

L'objectif est alors de définir une politique d'approvisionnement du stock SE ainsi que des règles d'ordonnancement appropriées au d'une pièce dans le stock d'une part, et à l'affectation d'une opération à une machine dans le cas des gammes multiples d'autre part.

La première étape est l'élaboration du modèle RdPCT associé avec la mise en évidence des différents points de décision du système. La figure IV - 2 illustre le réseau obtenu :

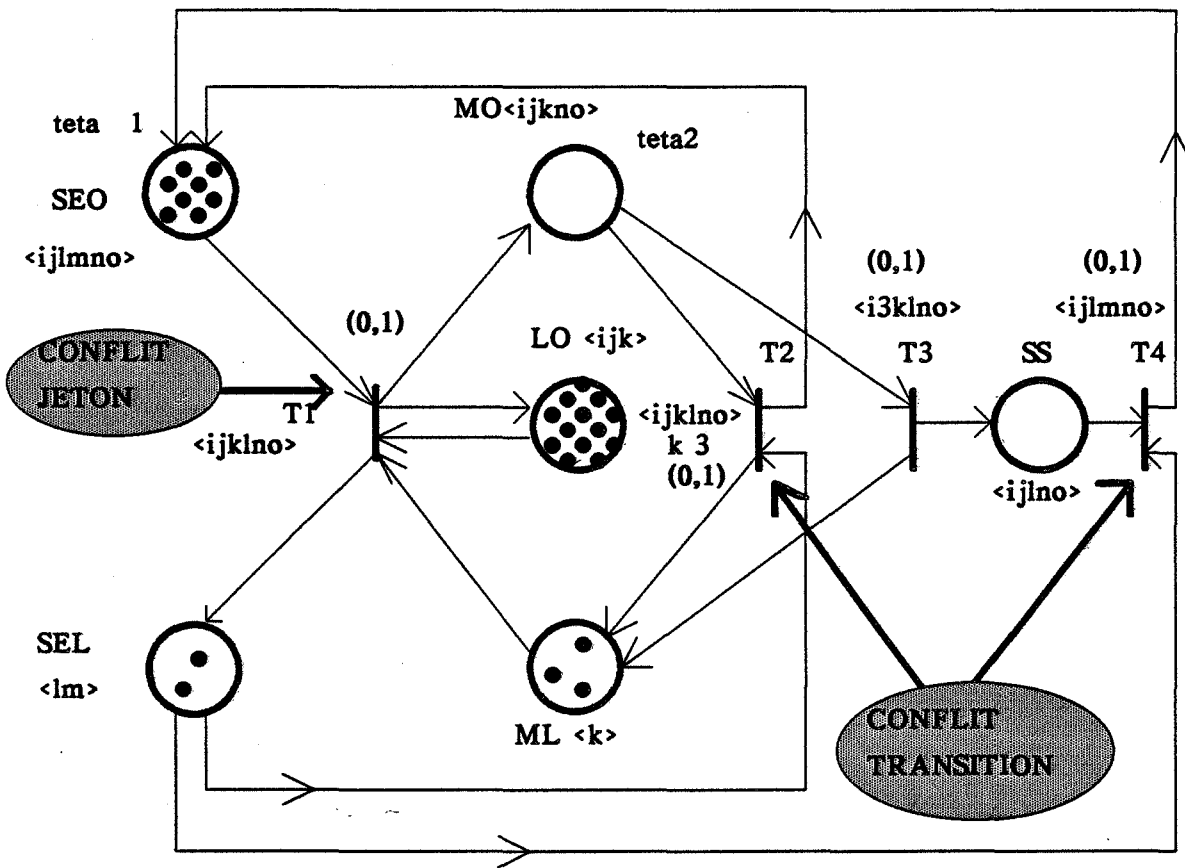


figure IV-2; modèle RdPC de la cellule 3 machines

Soit $\langle ijklmno \rangle$ le 7-uplet des indices de couleur des jetons avec :

i numéro produit $i=\{1,2,3\}$,

j numéro opération $j=\{1,2,3\}$,

k numéro machine $k=\{1,2,3\}$,

l numéro stock $l=\{1,2\}$,

m position dans le stock $m=\{1,2,\dots,10\}$,

n numéro palette $n \in \mathbb{N}$,

o nombre de pièces par palettes $o < 15$.

Les places SEO et SEL représentent l'état des 10 emplacements de SE. Teta1 est la temporisation représentant le délai de transfert du robot.

Les places MO et ML représentent l'état des machines M1 M2 M3. Teta2 est une fonction de $\langle i,j,k,o \rangle$ pour représenter la durée de l'opération d'usinage fonction de la tâche en cours et du nombre de pièces sur la palette.

La place LO contient la liste des opérations pour chaque machine, c'est elle qui permet la prise en compte des gammes multiples.

La place SS représente le stock de produits finis.

La transition T1 représente le chargement d'une machine.

La transition T2 représente l'évacuation d'une machine vers le stock SE lorsque le produit en cours n'est pas à la dernière opération de sa gamme.

La transition T3 représente l'évacuation d'une machine vers le stock SS dès lors que le produit a subi la dernière opération de sa gamme.

La transition T4 représente l'introduction d'une nouvelle palette de pièces.

Les fonctions n'ont pas été introduites sur le graphe car elles pénaliseraient la compréhension du modèle. Elles relèvent des fonctions élémentaires décrites dans la méthodologie d'approche de la modélisation.

Nous porterons plus particulièrement notre attention sur les notions de conflits mis en évidence sur le graphe et sur les règles de décision alors mises en place. On rencontre en effet dans cet exemple les deux types de conflit : conflit jeton et conflit transition.

IV - 1 - 1 Conflit jeton

Le conflit jeton sur la transition T1 est un conflit d'ordonnancement. Il traduit le choix d'une pièce dans un stock lorsqu'aucune politique de gestion n'a été fixée au préalable.

La comparaison des n couleurs validant la transition fait apparaître des valeurs d'indice communes et des valeurs d'indice différentes qui caractérisent les différents choix possibles. Les 6 indices de couleur du jeton font ainsi apparaître 6^2 possibilités de combinaison des valeurs d'indice pour caractériser les différentes possibilités de tir de la transition et donnent 36 définitions de conflits. En fait, seuls 5 types de conflits sont caractéristiques de situations décisionnelles.

Le conflit machine correspond à la situation où une palette n du produit i , à l'opération j de sa gamme, peut passer sur x machines distinctes.

Ce conflit relève de l'hypothèse des gammes multiples. En fonction du critère de gestion à respecter, respect des délais, niveau des en-cours, occupation des machines, et en fonction de l'état de l'atelier et de la production, on choisira par exemple, la machine donnant la plus petite durée opératoire (SPT). L'expérimentation nous a permis de constater en effet que cette règle donnait les meilleurs résultats si la répartition des lancements ne privilégiait pas le produit P2, du fait de l'équilibrage naturel des charges machines M1 et M2 dû aux opérations op1 pour P3 et op2 pour P1.

Le conflit palette correspond à la situation où x palettes du produit i , à l'opération j de sa gamme, peuvent passer sur la machine k .

Ce conflit relève d'une politique d'ordonnancement qui peut, par exemple, privilégier la palette la plus chargée tant que le taux d'occupation des machines est équilibré.

Le conflit produit-palette correspond à la situation où x palettes de produits différents y , à l'opération j de leur gamme, peuvent passer sur la machine k .

Ce conflit relève d'une politique d'ordonnancement qui peut, par exemple, privilégier le produit ayant la date de fin désirée la plus proche de l'instant courant.

Le conflit palette-gamme correspond à la situation où x palettes du produit i , à diverses opérations y de la gamme, peuvent passer sur la machine k .

Ce conflit relève d'une politique d'ordonnancement qui peut par exemple, privilégier la palette la moins chargée et la plus avancée dans la gamme.

Le conflit produit-palette-gamme correspond à la situation où x palettes de produits différents y à diverses opérations y de leur gamme peuvent passer sur la machine k .

Ce conflit relève d'une politique d'ordonnancement qui peut par exemple, privilégier la palette la moins chargée et la plus avancée dans la gamme, du produit dont la date de fin désirée est la plus proche.

Ces exemples de règles illustrent le grand nombre de possibilités de résolution du problème d'ordonnancement. Le domaine d'application d'une règle est fonction du critère de gestion à respecter, respect des délais, niveau des en-cours, occupation des machines, et des seuils empiriques liés à l'état de l'atelier et établis par simulation.

IV - 1 - 2 Conflit transition

Les transitions T2 et T3 ne sont pas en conflit effectif puisqu'elles ne concernent pas la même opération ; T3 est validée uniquement pour la dernière opération de la gamme du produit i, tandis que T2 exclut cette possibilité de tir.

Les transitions T2 et T4 sont en conflit effectif dès lors qu'il existe une position libre dans le stock SE validant simultanément soit le déchargement d'une machine, soit l'introduction d'une nouvelle palette.

Ce conflit relève d'une politique de lancement dont dépendra toute l'efficacité du système de production.

Si l'on associe à T2 et T4 les couples (1,1) et respectivement (0,1), la définition du couple (Priorité, Conflit) donne systématiquement priorité à la transition lancement d'une nouvelle palette. Cette configuration conduit à des situations critiques où le stock SE saturé ne permet plus le déchargement des machines et bloque la production.

Si l'on associe à T2 et T4 les couples (0,1) et respectivement (1,1), la définition du couple (Priorité, Conflit) donne systématiquement priorité à la transition déchargement machine, ce qui équivaut à terme, à vider le stock d'en-cours avant de lancer de nouveaux produits. Politique de sécurité, qui semble toutefois devoir être affinée pour améliorer les performances de production.

Si l'on associe à T2 et T4 les couples (0,1) et respectivement (0,1), la définition du couple (Priorité, Conflit) fait apparaître le conflit au niveau décisionnel et déclenche la consultation des experts. Un exemple de règle pourra consister à donner la priorité au lancement à partir d'un seuil d'occupation donné du stock SE. Le choix du produit à lancer ainsi que le nombre de pièces sur la palette à introduire est alors le résultat d'une heuristique de lancement, fonction de l'état de l'atelier et de la production déjà réalisée.

IV - 2 EXEMPLE 2 : PROBLEME DE PILOTAGE

L'atelier présenté est un atelier flexible composé de trois centres d'usinage M1, M2, M3, d'un stock matière première SE supposé de capacité infinie, de deux stocks d'en-cours S2 et S3 placés en aval et en amont de M3, respectivement de capacité 5 et 2, et d'un stock SS de produits finis supposé à capacité infinie. Les différentes ressources sont organisées autour d'un convoyeur central bouclé, pouvant contenir 24 palettes, la longueur du convoyeur étant discrétisée selon le nombre de palettes qu'il peut contenir. La figure IV-3 schématise l'organisation de l'atelier :

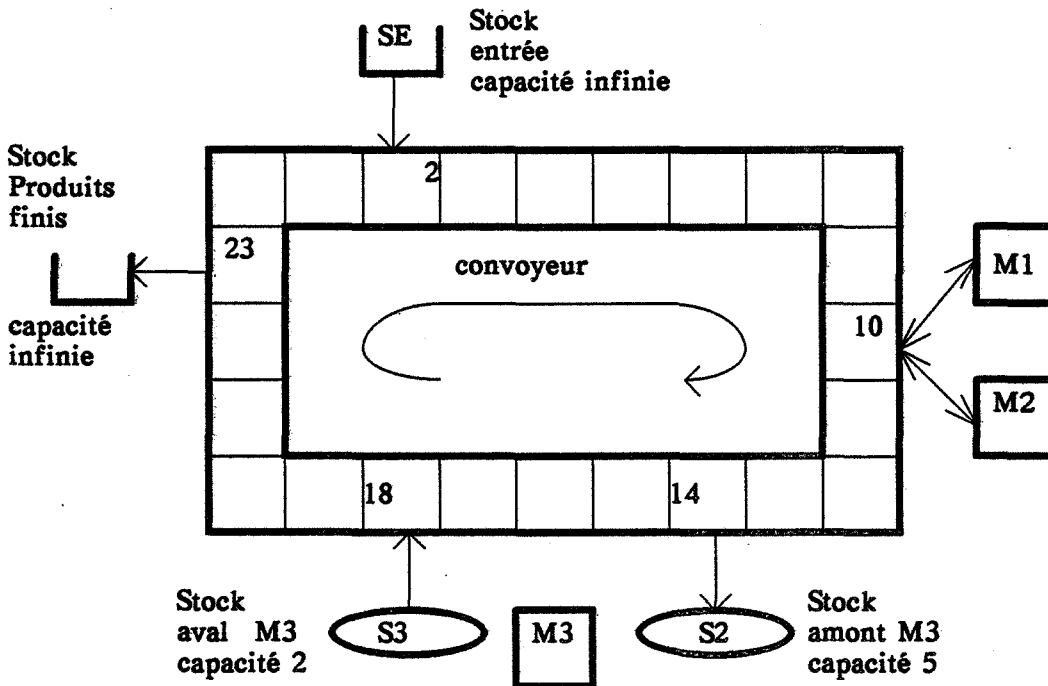


figure IV-3: Atelier avec convoyeur

La vitesse du convoyeur est constante, représentée par le temps mis par une palette pour passer d'une position à la position suivante du convoyeur. Les produits sont palettisés dès leur entrée dans le stock SE. Chaque palette est caractérisée par son numéro, mais elle ne peut contenir qu'une seule pièce à la fois.

Les produits P1, P2, P3, P4 sont usinés suivant les gammes définies dans le tableau suivant :

opérations produits	op1	op2	op3
P1	M1 10	M3 20	
P2	M2 12	M3 10	M1 ou M2 5 5
P3	M3 15	M2 10	
P4	M1 30	M3 10	

L'opération op3 du produit P2 peut être réalisée indifféremment soit sur la machine M1, soit sur M2 qui partagent la même position d'entrée sortie avec le convoyeur.

Lorsque la palette a terminé la dernière opération de sa gamme, elle est évacuée vers le stock de produits finis. Les lancements se font de façon périodique toute les 15 unités de temps dans le stock SE, en plaçant successivement un produit P1, puis P2, puis P3, puis P4.

L'objectif est alors de définir une politique de pilotage du convoyeur en gérant les différents niveaux de priorité des différentes ressources entre elles.

La première étape est l'élaboration du modèle RdPCT associé avec la mise en évidence des différents points de décision du système. La figure IV - 4 illustre le réseau obtenu :

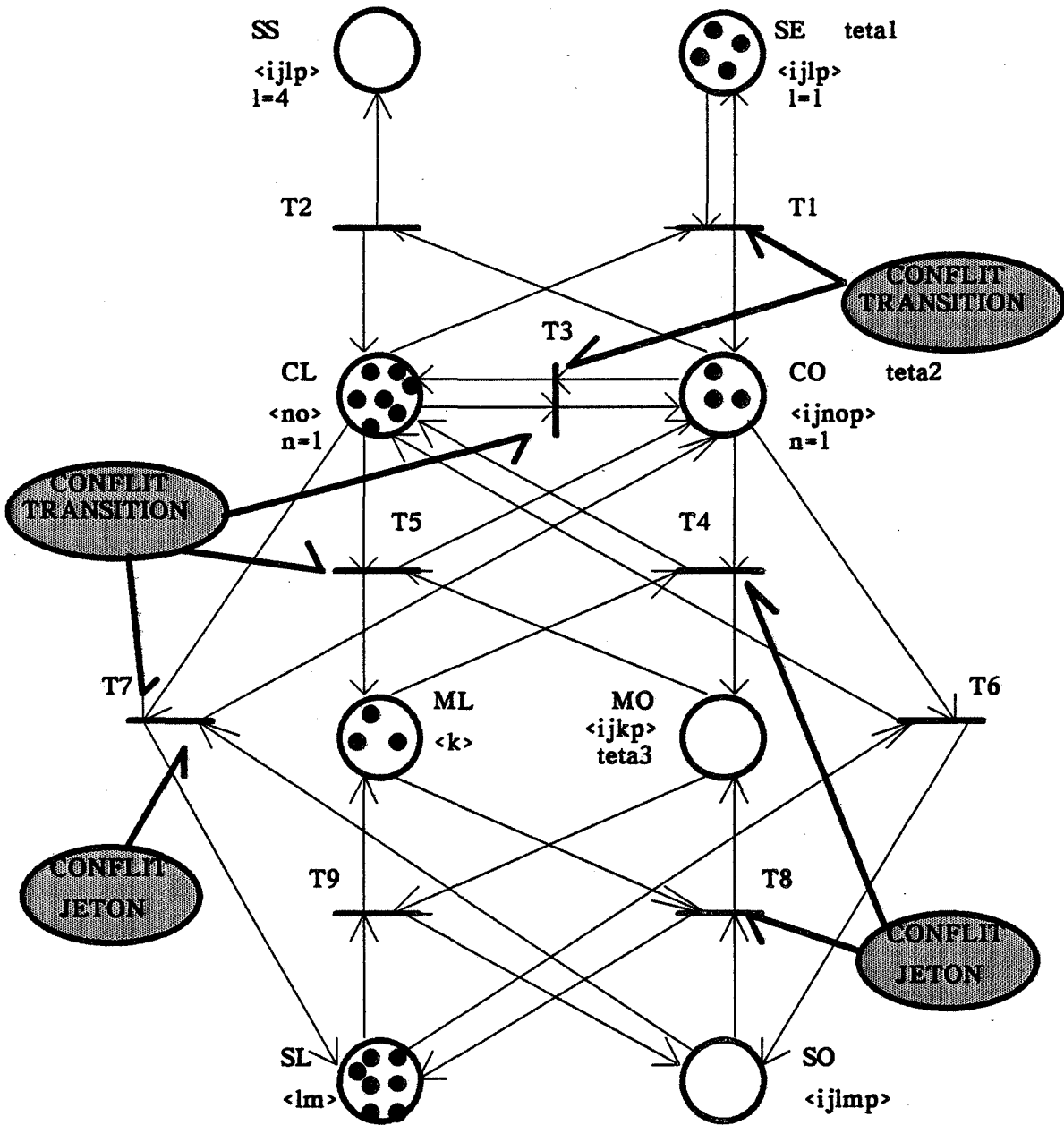


Figure IV-4: modèle RdPC de l'atelier avec convoyeur

Soit $\langle ijklmnop \rangle$ le 8-uplet des indices de couleur des jetons avec :

i numéro produit $i=\{1,2,3,4\}$

j numéro opération $j=\{1,2,3\}$

k numéro machine $k=\{1,2,3\}$

l numéro stock $l=\{1,2,3,4\}$

m position dans le stock $l=2 \quad m=\{1,2,3,4,5\} \quad l=3 \quad m=\{1,2\}$

n numéro convoyeur $n=1$

o position convoyeur $o=\{0, \dots, 23\}$

p numéro palette $p \in \mathbb{N}$

La place SE représente l'état du stock d'entrée, $\langle i,j,l=1,p \rangle$ le 4-uplet des indices de couleurs des jetons qu'elle contient. Teta1 est la temporisation représentant la période de lancement d'une nouvelle palette.

La place SS représente l'état du stock de produits finis, $\langle i,j,l=4,p \rangle$ le 4-uplet des indices de couleurs des jetons qu'elle contient.

Les places CO $\langle i,j,n=1,o,p \rangle$ et CL $\langle n=1,o \rangle$ représentent l'état du convoyeur. Teta2, la temporisation sur la place CO, représente la vitesse du convoyeur.

Les places MO $\langle i,j,k,p \rangle$ et ML $\langle k \rangle$ représentent l'état des machines M_k avec $k=\{1,2,3\}$. Teta3, la temporisation sur la place MO, est une fonction de $\langle i,j,k \rangle$, elle représente la durée de la tâche en cours.

Les places SO $\langle i,j,l,m,p \rangle$ et SL $\langle l,m \rangle$ représentent l'état des stocks S2 et S3 avec $l=\{2,3\}$.

La transition T1 représente l'entrée d'une palette du stock SE vers la position 2 du convoyeur, elle est tirable par rapport à $\langle i,j,l=1,n=1,o=2,p \rangle$.

La transition T2 représente la sortie d'une palette de la position 23 du convoyeur vers le stock SS, elle est tirable par rapport à $\langle i,j,l=4,n=1,o=23,p \rangle$.

La transition T3 représente l'évolution du convoyeur avec le passage d'une palette de la position en-cours à la position suivante, elle est tirable par rapport à $\langle i,j,n=1,o,p \rangle$.

La transition T4 représente le chargement de la machine M1 ou M2, elle est tirable par rapport à $\langle i,j,k=1,n=1,o=10,p \rangle$ et $\langle i,j,k=2,n=1,o=10,p \rangle$.

La transition T5 représente le déchargement des machines M1 ou M2, elle est tirable par rapport à $\langle i,j,k=1,n=1,o=10,p \rangle$ et $\langle i,j,k=2,n=1,o=10,p \rangle$.

La transition T6 représente le chargement du stock S2 amont de M3, elle est tirable par rapport à $\langle i,j,l=2,m,n=1,o=14,p \rangle$ avec $m=\{1,2,3,4,5\}$.

La transition T7 représente le déchargement du stock S3 aval de M3, elle est tirable par rapport à $\langle i,j,l=3,m,n=1,o=18,p \rangle$ avec $m=\{1,2\}$.

La transition T8 représente le chargement de la machine M3, elle est tirable par rapport à $\langle i,j,k=3,l=2,m,p \rangle$ avec $m=\{1,2,3,4,5\}$.

La transition T9 représente le déchargement de la machine M3, elle est tirable par rapport à $\langle i,j,k=3,l=3,m,p \rangle$ avec $m=\{1,2\}$.

Les fonctions n'ont pas été introduites sur le graphe car elles pénaliseraient la lisibilité du graphe. Elles relèvent des fonctions élémentaires décrites dans la méthodologie d'approche de la modélisation.

Les conflits mis en évidence sur le graphe nous permettrons présenter les règles de décision alors mises en place.

IV - 2 - 1 Conflit jeton

Le conflit jeton sur la transition T1 est résolu par le choix de la politique de lancement. Les fonctions assurent le lancement cyclique des produits P1, P2, P3, P4 et le conflit n'apparaît pas comme un conflit décisionnel.

Le conflit jeton sur la transition T4 est du au niveau de coloration du modèle, il traduit la possibilité pour l'opération op3 du produit P2 d'être réalisée soit sur M1 soit sur M2. Les deux machines étant physiquement reliées à la même position du convoyeur, il paraissait en effet opportun de transformer par coloration ce conflit transition en conflit jeton.

Ce conflit machine pourra par exemple être résolu en choisissant d'équilibrer les charges des deux machines puisque les durées opératoires sont égales.

Le conflit jeton sur la transition T7 est un conflit de pilotage, il traduit le choix de la palette du stock S3 à renvoyer sur le convoyeur.

Suivant la définition des critères de gestion ; respect des délais, niveau des en-cours, occupation des machines, et suivant l'état de l'atelier, on pourra par exemple choisir le produit le plus avancé dans sa gamme et qui, de part ce fait, sortira le plus vite de l'atelier, allégeant ainsi la charge du convoyeur.

Le conflit jeton sur la transition T8 est un conflit d'ordonnancement, il traduit le choix de la palette du stock S2 à charger sur M3.

Ce conflit pourra être résolu de façon analogue aux conflits d'ordonnements présentés dans l'exemple de la cellule 3 machines présenté précédemment. Nous citerons pour exemple une règle qui choisit la palette du produit dont l'opération sur la machine M3 n'est pas la dernière opération pour équilibrer les charges machines et faire circuler les produits sur le convoyeur.

Il serait fastidieux de présenter toutes les règles qui ont été testées et implémentées au niveau du système expert, l'idée est de donner ici un aperçu des possibilités de combinaison des différents paramètres en fonction de la nature du conflit, des choix de gestion, et de l'état de l'atelier.

IV - 2 - 2 Conflit transition

Les transitions T1 et T3 sont en conflit effectif dès lors que la position 2 du convoyeur est libre et que la position 1 est occupée d'une part, qu'une palette du stock SE est candidate pour entrer sur le convoyeur d'autre part. Le conflit illustre le choix entre l'évolution du convoyeur et l'entrée d'une nouvelle palette.

Les transitions T3 et T5 sont en conflit effectif dès lors que la position 10 du convoyeur est libre et qu'elle valide simultanément soit le déchargement des machines M1 et/ou M2 soit l'évolution du convoyeur.

Les transitions T3 et T7 sont en conflit effectif dès lors que la position 18 du convoyeur est libre et qu'elle valide simultanément soit le déchargement du stock S3 soit l'évolution du convoyeur.

Ces conflits relèvent d'une politique de pilotage du convoyeur dont dépendra toute l'efficacité du système de production.

Si on associe à T1, T3 respectivement les couples (0,1) et (1,1), la définition du couple (Priorité, Conflit) donne systématiquement priorité à la transition lancement d'une nouvelle palette. Cette configuration conduit à des situations critiques où le convoyeur saturé ne permet plus le déchargement des machines. Les produits non achevés ne pouvant être évacués vers le stock SS tournent alors sur le convoyeur en attente de la libération des machines concernées. Très vite, on atteint une situation de blocage pour laquelle aucun produit ne sort de l'atelier et pour laquelle toutes les machines sont bloquées

Si on associe à T3, T5 respectivement les couples (0,1) et (1,1), le couple (Priorité, Conflit) donne systématiquement priorité à l'évolution du convoyeur, ce qui équivaut, à terme, à ne jamais décharger les machines M1 et M2 lorsqu'elles ont terminé leur opération. Cette configuration conduit à terme à des situations critiques où les machines M1 et M2 sont bloquées et où les produits tournent sans cesse sur le convoyeur en attendant leur disponibilité sans pouvoir être évacués vers le stock SS. Très vite cette situation bloque la production.

Si on associe à T3, T7 respectivement les couples (0,1) et (1,1), le couple (Priorité, Conflit) donne systématiquement priorité à l'évolution du convoyeur, ce qui équivaut à terme, à ne jamais décharger le stock S3. Cette configuration conduit à des situations critiques pour lesquelles le stock S3 saturé ne permet pas le déchargement de la machine M3 qui elle même provoque la saturation du stock S2. Les produits tournent alors sans cesse sur le convoyeur en attendant la disponibilité de positions dans les stocks S2 ou S3 sans pouvoir être évacués vers le stock SS.

La pertinence du choix des stratégies de priorité du convoyeur vis à vis des autres ressources apparaît vite comme un élément déterminant pour l'efficacité de l'atelier. Le choix d'affectation des couples (priorité, conflit) est alors très important pour sélectionner les différents types de priorité et mettre en évidence les points de décision du système qui doivent être résolus par le niveau décisionnel pour adapter la stratégie de priorité à l'état de l'atelier.

L'expérimentation nous a permis d'affecter les valeurs de (P,C) avec le plus de pertinence possible :

les transitions les plus prioritaires seront celles qui déchargent le convoyeur :

T2 (0,0), T6 (0,0), transitions non décisionnelles

T4 (0,1), transition décisionnelle

les transitions prioritaires seront celles qui font évoluer le système à l'extérieur du convoyeur:

T9 (1,0), transition non décisionnelle

T8 (1,1), transition décisionnelle

les transitions les moins prioritaires seront celles qui chargent le convoyeur :

T1 (2,1), T5 (2,1), T7 (2,1), T3 (2,1), transitions décisionnelles

Les transitions décisionnelles provoquent la consultation du système expert pour choisir une stratégie locale de décision fonction de l'état de l'atelier et des critères de gestion à respecter. Nous avons pu mettre en évidence des seuils limites d'occupation du convoyeur au delà desquels il était nécessaire de modifier les stratégies de priorité des ressources vis à vis du convoyeur pour atteindre les objectifs de production calculés.

L'observation de ces seuils critiques nous a conduit à vouloir anticiper la prise de décision en développant un ensemble de règles de conduite plus particulièrement adaptées à ces situations. Dès que l'atelier atteint un état prédéfini, voisin des valeurs seuils, l'interpréteur déclenche la consultation du système expert pour prendre une décision d'exception comme le forçage d'une marque, par exemple, pour introduire ou enlever une palette.

IV - 3 EXEMPLE 3 : PROBLEME DES PANNES

L'atelier présenté est un atelier flexible composé de cinq machines M1, M2, M3, M4, M5, d'un stock matière première SE supposé de capacité infinie, de trois stocks d'en-cours S2, S3, S4 placés en aval des machines M2, M3 et M4 (respectivement de capacité 12, 6, 4) et d'un stock SS de produits finis supposé à capacité infinie. La figure IV-5 schématise l'organisation de l'atelier:

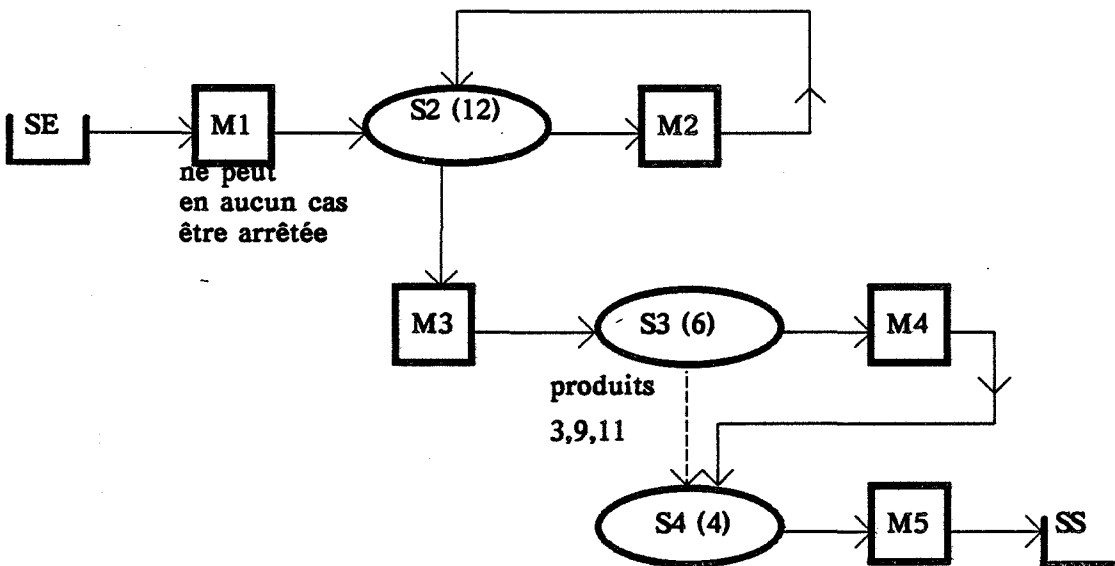


figure IV-5: Atelier de fabrication en continu

La production est de type continu, les produits passent d'abord sur M1 puis sur M2, M3, M4, M5. Seuls certains produits ne nécessitent pas le traitement de la machine M4, auquel cas ils sont directement orientés du stock S3 vers le stock S4.

Le système de transport entre les machines et les stocks n'est pas représenté en temps que ressource puisque nous voulons ici mettre en avant les problèmes liés aux pannes machines. Seuls les temps de transfert, fonctions de la distance entre le stock et la machine, sont pris en compte.

On suppose que tous les produits à traiter lors de la période de simulation sont initialement présents dans le stock SE. Les produits sont introduits dans le stock SE sur des "bobineaux" caractérisés par leur numéro et la longueur du produit qu'ils supportent. Deux produits identiques sur des "bobineaux" différents peuvent avoir des longueurs distinctes.

Les produits P1, P2,...., P11 ont des durées opératoires différentes sur chaque machine indépendamment de la longueur du "bobineau". Les durées sont définies dans le tableau suivant :

machine opération	M1	M2	M3	M4	M5
P1	41	39	30	37	43
P2	41	39	30	37	43
P3	41	39	30	—	46
P4	45	40	28	40	51
P5	45	40	29	40	51
P6	52.5	49	51	51	68
P7	52.5	49	49	51	68
P8	49	47	37	42	54
P9	49	47	37	—	54
P10	50	47	29	47	47
P11	50	47	29	—	47

La machine M1 ne peut en aucun cas être arrêtée, le traitement en cours ne peut être interrompu et la machine ne doit jamais tourner à vide. Les produits ne doivent pas séjourner plus longtemps que la durée opératoire dans la machine, la machine ne doit donc jamais être bloquée par un manque de place dans le stock aval.

On suppose que la politique de maintenance préventive de l'entreprise permet d'assurer que les machines ne tombent jamais en panne. Par contre, les machines M3 et M4 qui réalisent des opérations délicates occasionnent des "casses" du produit qui immobilisent la machine.

Nous assimilerons ces événements à des pannes machines ne pouvant intervenir que sur des machines occupées. Une fois la "casse" réparée le "bobineau" continue son traitement dans l'atelier, seule sa longueur a été modifiée.

La durée moyenne de la "casse" sur M3 est de 45 UT, le temps moyen de bon fonctionnement MTBF de la machine M3 suit une répartition Gaussienne autour d'une valeur moyenne de 480 UT pour les produits 1,2,3,4,5,8 et de 460 UT pour les produits 6,7,9,10,11 (écart type 2).

La durée moyenne de la "casse" sur M4 est de 52 UT, le temps moyen de bon fonctionnement MTBF de la machine M4 suit une répartition Gaussienne autour d'une valeur moyenne de 420 UT pour les produits 1,2,4,6,10 et de 410 UT pour les produits 5,7,8 (écart type 2).

L'objectif est alors de définir une politique de lancement de la production qui puisse compenser les effets des "casses" sur machine sans saturer les stocks, et surtout, sans arrêter la production sur la machine M1.

La première étape est l'élaboration du modèle RdPCT associé avec la mise en évidence des différents points de décision du système.

La figure IV - 6 illustre le réseau obtenu:

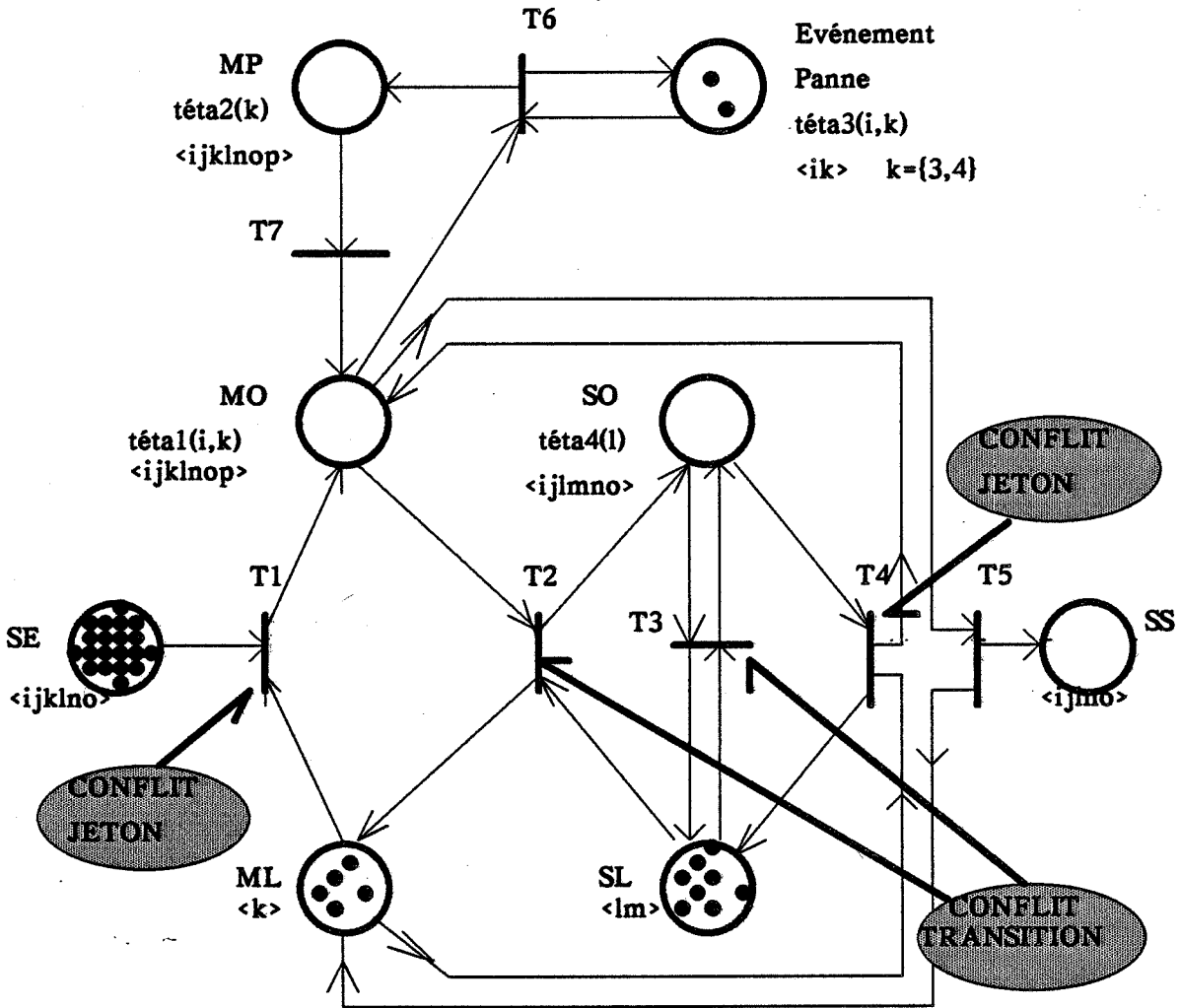


figure IV-6: modèle RdPC de l'atelier en continu

Soit $\langle ijklmnop \rangle$ le 8-uplet des indices de couleur des jetons avec :

i numéro produit $i=\{1,2,\dots,11\}$

j numéro opération $j<6$

k numéro machine $k=\{1,2,3,4,5\}$

l numéro stock $l=\{1,2,3,4,5\}$

m position dans le stock $l=2 \ m=\{1,2,\dots,12\} \ l=3 \ m=\{1,2,\dots,6\} \ l=4 \ m=\{1,2,\dots,4\}$

n numéro "bobineau" $n \in \mathbb{N}$

o longueur du "bobineau" $o \in \mathbb{N}$

p flag panne $p=\{0,1\}$

La place SE représente l'état du stock d'entrée, $\langle i,j,k=1,l=1,n,o \rangle$ le 6-uplet des indices de couleurs des jetons qu'elle contient.

La place SS représente l'état du stock de produits finis, $\langle i,j,l=5,n,o \rangle$ le 5-uplet des indices de couleurs des jetons qu'elle contient.

Les places MO $\langle i,j,k,n,o,p \rangle$ MP $\langle i,j,k,n,o,p \rangle$ et ML $\langle k \rangle$ représentent l'état des machines Mk. L'indice $p=1$ indique si la machine occupée vient de subir une panne. Téta1 la temporisation sur la place MO est une fonction de $\langle i,j,k,p \rangle$; si $p=0$ elle représente la durée de la tâche en cours, si $p=1$ elle est de valeur nulle pour permettre l'évacuation instantanée du produit ayant subi une "casse" vers le stock aval. Téta2 est une fonction de $\langle k \rangle$, elle représente la durée de la panne. Téta3 est une fonction de $\langle i,k \rangle$, elle suit une répartition Gaussienne autour de la valeur moyenne de durée entre deux pannes.

Les places SO $\langle i,j,l,m,n,o \rangle$ et SL $\langle l,m \rangle$ représentent l'état des stocks S2 et S3 avec $l=\{2,3\}$. Téta4 est une fonction de $\langle l \rangle$, elle traduit le temps de transfert entre les machines et les stocks.

La transition T1 représente l'entrée d'un "bobineau" sur la machine M1, elle est tirable par rapport à $\langle i,j=1,k=1,l=1,n,o \rangle$.

La transition T2 représente le déchargement d'une machine et l'entrée du "bobineau" dans le stock aval, elle est tirable par rapport à $\langle i,j,k,l,m,n,o \rangle$ pour k différent de 5.

La transition T3 représente l'évacuation directe du stock S3 vers S4 pour les produits 3,9,11, elle est tirable par rapport à $\langle i=3,j,l=3,m,n,o \rangle$, $\langle i=9,j,l=3,m,n,o \rangle$, $\langle i=11,j,l=3,m,n,o \rangle$.

La transition T4 représente le chargement des machines M2, M3, M4 ou M5, elle est tirable par rapport à $\langle i,j,k,l,m,n,o \rangle$ pour k différent de 1.

La transition T5 représente le déchargement de la machine M5 vers le stock SS, elle est tirable par rapport à $\langle i,j,k=5,l=5,n,o \rangle$.

La transition T6 représente le début d'une panne sur les machines M3 ou M4 lorsqu'elles sont occupées, elle est tirable par rapport à $\langle i,j,k=3,n,o,p=0 \rangle$ et $\langle i,j,k=4,n,o,p=0 \rangle$

La transition T7 représente la fin de la panne sur les machines M3 ou M4, elle est tirable par rapport à $\langle i,j,k=3,n,o,p=1 \rangle$ et $\langle i,j,k=4,n,o,p=1 \rangle$

Les fonctions utilisées sont des fonctions élémentaires décrites dans la méthodologie d'approche de la modélisation, elles n'ont pas été introduites sur le graphe pour en améliorer la lisibilité.

Nous caractériserons tout particulièrement les notions de conflits relevées sur le graphe et les règles de décision testées. On rencontre les deux types de conflit : conflit jeton et conflit transition.

IV - 3 - 1 Conflit jeton

Le conflit jeton sur la transition T1 relève du choix de la politique de lancement. Ce conflit pourra être résolu en jouant, par exemple sur le lancement des produits 3,9,11 qui n'utilisent pas la machine M4, pour assurer la continuité de la production pendant ou avant une "casse" sur la machine M4.

Le conflit jeton sur la transition T4 est un conflit d'ordonnancement dû au choix d'un produit dans le stock aval des machines. Toute l'efficacité du système de production dépend de ce choix pour assurer le non engorgement des stocks et par la suite, le non blocage de la machine M1.

Nous donnerons simplement un aperçu des possibilités de combinaison des différents paramètres (fonction de la nature du conflit, des choix de gestion, de l'état de l'atelier), lors de la construction de la base de règles du système expert. Dans cet exemple, le suivi de l'état de l'atelier est primordial pour assurer le non blocage de la machine M1, malgré les perturbations importantes causées par les pannes. L'observation de l'état de l'atelier permet l'utilisation de seuils basés sur le niveau des différents stocks pour déclencher tel ou tel type de règle.

IV - 3 - 2 Conflit transition

La survenue d'une "casse" sur les machines M3 et M4 et son traitement est prioritaire sur toute l'évolution du système. Les transitions T6 et T7 ont des valeurs du couple (Priorité conflit) = (0,0) assurant ainsi leur priorité sur les autres transitions pour lesquelles (P,C) = (1,1) sans que cela apparaisse comme un conflit décisionnel.

Les transitions T2 et T3 sont en conflit effectif dès lors qu' une position du stock S4 est libre permettant simultanément soit le déchargement de la machine M4, soit le transfert d'un produit 3,9, ou 11 du stock S3 vers le stock S4. Ce conflit relève d'une politique de pilotage du convoyeur dont dépendra toute l'efficacité du système de production.

Si les couples associés aux transitions T2 et T3 sont respectivement (1,1) et (2,1), le couple (Priorité, Conflit) donne systématiquement priorité à la transition déchargement de la machine M4. Cette configuration conduit à des situations critiques pour lesquelles le stock S3 se trouve très vite saturé par des produits de types 3,9,11 ne pouvant continuer leur évolution dans l'atelier. La saturation de S3 provoque le blocage de la machine M3 qui elle-même provoque en cascade la saturation du stock S2 puis le blocage des machines M2 et M1. On atteint une situation de blocage : aucun produit ne sort de l'atelier, toutes les machines sont bloquées, y compris la machine M1.

Si les couples associés aux transitions T2 et T3 sont respectivement (2,1) et (1,1), le couple (Priorité, Conflit) donne systématiquement priorité à l'évacuation de stock à stock entre S3 et S4. Cette configuration conduit à bloquer la machine M4 qui ne peut être déchargée, puis à saturer le stock S3 de produits en attente de passer sur M4. A terme, la saturation en cascade des différentes ressources conduit au blocage de la machine M1.

Si les couples associés aux transitions T2 et T3 sont respectivement (1,1) et (1,1), la définition du couple (Priorité, Conflit) fait apparaître le conflit au niveau décisionnel et déclenche la consultation des experts. Un exemple de règle pourra être de donner la priorité soit au déchargement de la machine tant que le stock S3 ne contient pas un certain seuil de produits de type 3,9,11 soit à l'évacuation stock à stock au-dessus de ce seuil.

Les transitions décisionnelles provoquent la consultation du système expert pour choisir une stratégie locale de décision fonction de l'état de l'atelier et des critères de gestion à respecter. Nous avons pu mettre en évidence des seuils limites d'occupation des différents stocks au delà desquels il était nécessaire de modifier les stratégies de pilotage d'une part et d'ordonnancement d'autre part pour atteindre les objectifs de production calculés malgré les pannes très fréquentes.

IV - 4 CONCLUSIONS

Pour chacun des exemples que nous avons présentés dans ce chapitre, nous avons été amenés à construire une base de règles différente, caractéristique de cette application.

Au cours de l'expérimentation par simulation, la convivialité de l'approche par système expert, avec la possibilité de modifier facilement les règles par simple ajout ou retrait dans la base de règle, nous a permis d'affiner notre connaissance des ateliers étudiés et de leur comportement vis à vis des différentes politiques de conduite testées.

Chaque base de règles comprend des règles d'ordonnancement bien connues, comme celles présentées dans le chapitre III, mais aussi des règles empiriques très simples, issues de l'expérience des hommes de l'entreprise.

Cependant, l'ajustement des paramètres des différentes règles définissant les conditions d'application fonction de l'état de l'atelier, le degré de pertinence vis à vis des critères de gestion, les coefficients de certitude sur les faits et sur les règles, sont des opérations très longues.

Cette limite tient essentiellement aux problèmes de communication des différents experts entre eux et à la difficulté de formaliser ces différentes connaissances. Il est apparu qu'il était important que cette étape soit coordonnée par un informaticien qui joue le rôle de centralisateur des informations au cours de la construction de la base de connaissance.

Les techniques de l'intelligence artificielle, avec l'apprentissage automatique devraient pouvoir permettre d'automatiser cette étape de construction de la base de règles, sous certaines conditions d'expérimentation.

Le principal intérêt d'une telle approche a été non pas de définir des règles de conduite optimale, mais de montrer que les règles de conduite pouvaient être réévaluées pour chaque décision en fonction de l'état de l'atelier.

L'approche dynamique par simulation, permet d'adapter les lois de conduite, tant au niveau de l'ordonnancement qu'au niveau commande, en fonction de l'état courant de l'atelier et de l'écart entre les résultats escomptés et les résultats de la simulation à l'instant de la décision.

Nous avons cherché par ce biais à trouver des solutions locales aux problèmes décisionnels du niveau de l'atelier tout en tenant compte des contraintes de gestion, par une approche "par le bas", venant de l'atelier pour remonter jusqu'au niveau gestion de production.

Cette approche présente l'intérêt d'être plus proche de la formalisation du problème telle qu'on la rencontre dans les entreprises; elle répond aux questions du chef d'atelier qui doit localement prendre un certain nombre de décisions et dont vont dépendre en partie les performances du système de production.

CONCLUSION

Face à la complexité croissante des systèmes de production, la simulation apporte une aide aux spécialistes de l'entreprise dans la résolution des problèmes de conduite de production.

Mais un système d'aide à la conception puis à la conduite, basé sur la simulation ne peut réellement apporter enseignement et support de réflexion que s'il intègre, outre le fonctionnement de l'atelier, une part importante de la gestion du procédé. Il doit permettre la manipulation des règles de décisions souvent complexes, toujours présentes dans un système flexible.

Nous avons présenté dans ce mémoire nos travaux de recherche sur ce problème : ils ont abouti à la réalisation d'un simulateur, SAGASSE, basé sur l'interaction dynamique entre un interpréteur de Réseaux de Petri Colorés Temporisés, garant du suivi de l'état de l'atelier, et un système expert, chargé de prendre les décisions de conduite.

Nous n'avons pas cherché à construire de nouveaux algorithmes pour la résolution des problèmes d'ordonnancement, l'objectif de nos travaux n'étant pas de trouver une méthode standard optimale, mais plutôt de donner un outil de simulation qui permet une recherche de solutions adaptées à chaque atelier.

Les validations industrielles que nous avons réalisées nous ont permis de constater que la construction de la base de connaissance propre à un atelier modélisé, permet aux différents experts de l'entreprise d'approfondir leur formalisation du problème et de mettre en commun leur savoir. Elle nous a, par la même, conduit à relever les difficultés propres à une telle démarche que l'on rencontre par ailleurs, dans toutes les expériences de mise en place d'un système expert.

Pour autant, le cadre de réflexion et le support de simulation fournis par SAGASSE, permettent une meilleure concertation et une responsabilisation des cadres de l'entreprise autour de leur programme productique, ce qui n'est pas, à notre sens, le moindre des résultats de cette approche.

L'ADEPA a financé une grande partie de nos travaux de recherche et souhaite utiliser SAGASSE comme un outil didactique pour une approche des problèmes de conduite d'atelier flexible.

BIBLIOGRAPHIE

- [ALLA - 84] H. ALLA, P. LADET, J. MARTINEZ, M. SILVA : "Modelling and validation of complex systems by coloured Petri Nets ; Application to a flexible manufacturing system". Advances in Petri Nets, Edited by G. Rosenberg, 1984.
- [ALLA - 87] H. ALLA : "Réseaux de Petri Colorés et Réseaux de Petri Continus, Application à l'étude des systèmes à événements discrets". Thèse d'Etat de l'Institut National Polytechnique de Grenoble, 1987.
- [ARNOUX - 86] M. ARNOUX, G. BECKER, M.C THOMAS : "Un système expert de gestion des aléas en production". In 2ème Colloque International d'Intelligence Artificielle, Marseille, pp. 15-30, 1986.
- [ATABAKHCHE - 87] H. ATABAKHCHE FALLAHI : "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 de Toulouse, 1987.
- [BAKER - 60] C.T BAKER, B.D DZIELINSKI : "Simulation of a simplified job-shop", Mgmt Science, 1960.
- [BAKER - 74] K.R BAKER : "Introduction to sequencing and scheduling". Wiley, 1974.
- [BEHRTUL - 85] C. BEHRTUL : "Pilotage d'ateliers flexibles en présence de perturbations". Thèse de Docteur Ingénieur, Université Paris Dauphine, 1985.

- [BEL - 85] G. BEL, D. DUBOIS : "Modélisation et simulation de systèmes automatisés de production". Revue des modèles de systèmes automatisés de production, APII, Volume 19, N° 1, pp. 3-43, 1985.
- [BEL - 88] G. BEL, E. BENSANA, D. DUBOIS : "OPAL : Un système à base de connaissance pour l'ordonnancement prévisionnel d'atelier". In International Conference CAD/CAM/CIM and Robotics : Contribution of Artificial Intelligence, Bordeaux, 15-17 Mars, pp. 37-42, 1988.
- [BESOMBES - 88] B. BESOMBES, B. JULLIEN, P. LADET : "SAGASSE - Système expert pour l'aide à la gestion d'atelier flexible". Colloque International Productique et Robotique sur "Les apports de l'Intelligence Artificielle", AFCET, 15-17 Mars, Bordeaux, 1988.
- [BESOMBES - 89] B. BESOMBES, P. LADET : "SAGASSE - An expert system based simulator for flexible manufacturing systems". In International Workshop on Decisional Structures in Automated Manufacturing, IFAC - CIRP - IFIP - IFORS, 18-21 Septembre, Genova (Italie), 1989.
- [BESOMBES - 89] B. BESOMBES : "Un simulateur à événements discrets pour l'aide à la conduite d'une ligne de production papier". Conférence Invitée in 42^{ème} Congrès de l'Association Technique Industrie Papetière, ATIP - CTP, 21-23 Novembre, Grenoble, 1989.

- [BESOMBES - 90] B. BESOMBES, P. LADET : "La formalisation de la connaissance dans un système d'aide au pilotage d'atelier flexible". Publication proposée, in 3ème Congrès International sur le Génie Industriel, facteur d'intégration et d'efficacité des entreprises, ADEPA - AFCET - GGI, 20-22 Mars, Tours, 1991.
- [BRAMS - 83] G. W BRAMS : "Réseaux de Petri - Théorie et pratique". Tomes 1 & 2 , Ed. Masson, 1983.
- [CAMPBELL - 70] H.G CAMPBELL, D.A DUDEX, M.L SMITH : "An heuristic algorithm for the n-job m-machine sequencing problem". Management Science, Volume 16, N° 10, pp. 630-631, 1970.
- [CANALS - 86] D. CANALS : "Ordonnancement d'atelier par simulation. Etude des règles de priorité et aide au lancement". Thèse de Doctorat 3ème cycle ENSAE, 1986.
- [CAVILLE - 82] J.B CAVAILLE, D. DUBOIS : "Intérêt de la théorie des réseaux de file d'attente pour l'évaluation des performances d'un atelier flexible". In Analysis and Optimisation of Systems. Lect. notes in Control and Information Sciences, Volume 44, Springer-Verlag, Berlin, pp. 320-331, 1982.
- [CAVILLE - 88] J.B CAVAILLE, J.M PROTH : "Pratique de la simulation en productique discontinue". SIPRODIS, Collection Novotique, Ed. EC2, 1988.

- [CHRETIENNE - 83] P. CHRETIENNE : "Les réseaux de Petri Temporisés". Thèse d'Etat, Université Pierre et Marie Curie, Paris, 1983.
- [COHEN - 83] G. COHEN, D. DUBOIS, J.P QUADRAT, M.VIOT : "Analyse du comportement périodique de systèmes de production par la théorie des dioïdes". Rapport de Recherche INRIA, N° 191, 1983.
- [COURTOIS - 89] A. COURTOIS, M. PILLET, C. MARTIN : "Gestion de Production". Les Editions d'Organisation, 1989.
- [CONWAY - 67] R.W CONWAY, W.L MAXWELL, L.W MILLER : "Theory of scheduling". Addison-Weley, 1967.
- [DANNENBRING - 77] P.G DANNENBRING : " An evaluation of flow-shop sequencing heuristics". Management Science, Volume 23, N° 11, pp. 1173-1183, 1977.
- [DOUMEINGTS - 79] G. DOUMEINGTS, F. ROUBELLAT : "Système de conduite de production et aide à la décision". RAIRO Automatique, Volume 13, N° 1, pp. 77-92, 1979.
- [ERSCHLER - 76] J. ERSCHLER : "Analyse sous contraintes et aide à la décision pour certains problèmes d'ordonnancement". Thèse d'Etat, Université Paul Sabatier de Toulouse, Toulouse, 1976.

- [ERSCHLER - 86] J. ERSCHLER, P. ESQUIROL : "Decision aid job-shop scheduling : a knowledge based approach." IEEE International Conference on Robotics and Applications ; San Francisco, April 1986.
- [ESQUIROL - 87] P. ESQUIROL : "Règles et processus d'inférence pour l'aide à l'ordonnancement de tâches en présence de contraintes." Thèse de Doctorat 3ème cycle, Université Paul Sabatier de Toulouse, 1987.
- [FARRENY - 85] H. FARRENY : "Les systèmes experts ; principes et exemples". Ed. CEPADUES, Collection Techniques Avancées de l'informatique, 1985.
- [FARRENY - 87] H. FARRENY, M. GHALLAB : "Eléments d'intelligence artificielle", Ed. HERMES, Traité des Nouvelles Technologies, Série Intelligence Artificielle, 1987.
- [FOX - 84] M. FOX, S. SMITH : "ISIS, a knowledge based system for factory scheduling". Expert System, Volume 1, N°1, pp. 25-49, 1984.
- [GELENBE - 82] E. GELENBE, G. PUJOLLE : "Introduction aux Réseaux de files d'attente". Collection Technique et Scientifique des Télécommunications. Ed. Eyrolles, 1982.
- [GENRICH - 86] H. J GENRICH : "Predicate/ Transition Nets", Advanced Course on Petri nets, Bad Honnef, 8-19 sept, 1986.

- [GERE - 66] GERE : "Heuristics in job-shop scheduling". Mgmt Science, Volume 13, N° 3, 1966.
- [GERMAIN - 85] R. GERMAIN, C. SRISKANDARAJAH, B. PICARD D'ESTELAIN : "Heuristics for general flow-shop and job-shop scheduling". APMS COMP CONTROL'85, 1985.
- [GINTY - 81] J. Mc GINTY : "Flexible Manufacturing Systems". First Boston Research Industry Report. 1981.
- [GONDRAN - 87] M. GONDRAN, M. MINOUX : "Dioid theory and its applications". In Seminaire CNRS/INRIA, Algèbres exotiques et systèmes à événements discrets, 3-4 Juin, Issy-les-Moulineaux, 1987.
- [GUPTA - 78] J.N.D GUPTA, R. REDDI : "Improved dominance conditions for three machine Flow-Shop scheduling problem". Operational Research Quaterly, Volume 26, N° 1, pp. 200-208, 1978.
- [HERITIER - 90] C. HERITIER : "Conception d'un système de production : Une méthode d'aide à la décision basée sur la simulation et l'analyse des données". Thèse à soutenir Université Lyon I Claude Bernard (INSA Lyon).
- [HOLLOWAY - 74] C.A HOLLOWAY, R.T NELSON : "Job-shop scheduling with due dates and variables processing times". Mgmt Science, Volume 20, N°9, pp. 1264-1275, 1974.

- [JENSEN - 80] K. JENSEN : "Coloured Petri Nets and the invariant method", DAIMI, PB- 104, August, 1980.
- [JENSEN - 82] K. JENSEN : "High level Petri Nets". Daimi PB 151, Aarhus University, September 1982.
- [JOHNSON - 54] S.M JOHNSON - : "Optional two and three stage production scheduling with set-up time included". Nav. Res. Logist. Quart, Volumel, pp. 61-68, 1954.
- [LEARY - 83] D. O'LEARY : "Les ateliers flexibles". Ingersoll Engineers France. Ed. de l'Usine Nouvelle, 1983.
- [LEOPOULOS - 85] V.I LEOPOULOS : "LORIC, un simulateur réseau de Petri écrit en Maclisp". Rapport de Recherche N° 371, INRIA, Rocquencourt, Mars 1985.
- [LEPAPE - 85] C. LEPAPE : "SOJA ; un système d'ordonnancement journalier d'atelier". In 5ème Journées Internationales Intelligence Artificielle, Les systèmes experts et leurs applications, Avignon 13 - 15 Mai, Volume 2, pp. 849-867, 1985.
- [MARTIN - 87] F. MARTIN : "Méthodologie de modélisation et simulation de systèmes complexes décrits par Réseaux de Petri Colorés". Thèse de Doctorat de l'Institut National Polytechnique de Grenoble, 1987.
- [MELESE - 72] J. MELESE : "L'analyse modulaire des systèmes de gestion". Ed. Hommes et Techniques, 1972.

- [MERCIER - 88] R. MERCIER des ROCHETTES : "Sur l'utilisation des Réseaux de Petri Colorés pour la commande des systèmes de production ; Mise en oeuvre sur un atelier flexible". Thèse de Doctorat 3ème cycle du Laboratoire d'Automatique de Grenoble, 1988.
- [NEURON-DATA - 88] NEURON-DATA : "NEXPERT-OBJECT Version 1.0 Manuels de documentation". High St Palo Alto, California, USA, 1988.
- [PALMER - 65] D.S PALMER : "Sequencing jobs through a multistage processing the minimum total time. A quick method of obtaining a near optimum". Operational Research Quaterly, Volume16, N°1, pp. 101-107, 1965.
- [PROTH - 86] J.M PROTH, H. QUENTIN DE GROMARD : "Systèmes Flexibles de Production - Conception préliminaire", Ed. MASSON, 1986.
- [REISIG - 86] W. REISIG : "Place/Transition systems", Advanced course on Petri nets, Bad Honnef, 1986.
- [ROCHE - 84] C. ROCHE : "EAQUE-LRO ; Génération de systèmes experts- Application à des problèmes d'ordonnancement." Thèse de Doctorat 3ème cycle de l'Institut National Polytechnique de Grenoble, 1984.
- [ROUBELLAT - 87] F. ROUBELLAT, V. THOMAS : "Une méthode et un logiciel pour l'ordonnancement en temps réel d'ateliers". 2ème Conférence Internationale Systèmes de Production, Paris, 6-10 Avril, pp. 87-101, 1987.

- [ROY - 70] B. ROY : "Algèbre moderne et théorie des graphes". Tome II, Paris, Ed. Dunod, 1970.
- [RUSSEL - 87] R.S RUSSEL, Z.M DAREL : "A comparative analyse of covert job sequencing rule using various shop performances measures". Intelligent journal process result, Volume 25, N°10, 1987.
- [SIBERTIN - 85] C. SIBERTIN-BLANC : "High Level Petri Nets with Data Structures", 6th European Workshop on Application and Theory of Petri Nets, Finland, 1985.
- [SIBERTIN - 88] C. SIBERTIN-BLANC : "Le prototypage des applications interactives à l'aide de Réseaux de Petri", Séminaire d'informatique de l'IIE, Evry, 26-27 Octobre, pp. 130-149, 1988.
- [THOMAS - 80] V. THOMAS : "Aide à la décision pour l'ordonnancement d'atelier en temps réel". Thèse de Doctorat 3ème Cycle, Université Paul Sabatier de Toulouse, 1980.
- [VALETTE - 85] R. VALETTE, V. THOMAS, S. BACHMANN : "SEDRIC, un simulateur à événements discrets basé sur les réseaux de Petri". APII Systèmes de production, N°19, pp. 423-436, 1985.
- [VOYATZIS - 87] K. VOYATZIS : "Utilisation de la mémoire artificielle pour les problèmes d'ordonnancement". Thèse de Doctorat 3ème cycle Paris Dauphine, 1987.

NOM : BESOMBES (avec précision du nom de jeune fille, le cas échéant) Prénoms : Béatrix, Angèle, Rose, Marie		DATE de SOUTENANCE 8 Novembre 1983
TITRE : Un Système d'Aide à la Conduite d'Ateliers Flexibles basé Réseaux de Petri Colorés.		
NATURE :		Numéro d'ordre : 2193
DIPLOME DE DOCT. <input checked="" type="checkbox"/>	DOCTEUR-INGENIEUR <input type="checkbox"/>	DOCTORAT D'ETAT <input type="checkbox"/>
		DOCTORAT DE 3e CYCLE <input type="checkbox"/>
		Spécialité : Automatique
Cote B.I.U. - Lyon : T 50/210/19 / et bis		CLASSE :
RESUME : Les outils classiques de simulation ne permettent pas de prendre facilement en compte dans leur modèle, le niveau décisionnel lié à l'ordonnancement des tâches. La politique de décision est figée tout au cours d'une simulation. Par ailleurs, les outils classiques de gestion de production ne permettent pas de contrôler correctement le flux de production, parcequ'ils ne prennent pas en compte l'état réel de l'atelier (temps morts devant les machines, pannes,...). Ils induisent des surdimensionnements de stocks ou de mauvais taux d'utilisation des ressources. Nous proposons dans ce mémoire un outil d'aide à la conduite d'atelier flexible intégrant les aspects fonctionnement de l'atelier et gestion de production réduite au routage et à l'ordonnancement des tâches. Le simulateur proposé est basé sur l'interaction dynamique entre un interpréteur de Réseaux de Petri Colorés Temporisés, garant de l'état courant de l'atelier, et un niveau décisionnel rempli par un système expert. La présence d'une décision de routage, d'ordonnancement, ou la détection d'un état prédéfini se traduit au niveau de l'interpréteur par un conflit dans le modèle RdPCT. La construction d'un ensemble de règles de décisions, par les experts humains de l'entreprise permet au système expert de résoudre le conflit. Avec l'introduction des techniques de l'Intelligence Artificielle, le système SAGASSE se caractérise par une approche ascendante du problème de conduite d'atelier flexible, du système physique vers le système décisionnel. Des exemples industriels sont cités dans ce mémoire pour valider l'approche retenue.		
MOTS-CLES : Atelier flexible, Simulation, Réseaux de Petri Colorés Temporisés, Ordonnancement, Aide à la conduite, Système Expert, Système base de règles		
Laboratoire(s) de recherches :		Département Stratégie du Développement Ecole des Mines de St-Etienne
Directeur de recherches :		Mr P. LADET
Président du jury :		
Composition du jury : M. DUBRETT, M. PÉPIN, M. BÉZ, M. LADET, M. LADET, M. HATHON		