

50376
1986
27

50376
1986
27

N° d'ordre : 34

THÈSE

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE FLANDRES ARTOIS

pour obtenir le titre de

DOCTEUR

Spécialité : Automatique

par

DJEGHABA Messaoud

Ingénieur E.N.P.A.

PROBLEMES DE DECISION DANS UNE CELLULE DE PRODUCTION UTILISANT LA COOPERATION ENTRE ROBOTS



Thèse soutenue le 20 Mars 1986 devant la Commission d'Examen

Membres du Jury :

P.	VIDAL	Président et Rapporteur
R.	SOENEN	Rapporteur
M.	STAROSWIECKI	Directeur de recherche
J.G.	POSTAIRE	Examineur
P.	MONTCUQUET	Invité
D.	JOLLY	Invité

SOMMAIRE

INTRODUCTION GENERALE	2
I - SYSTEMES DE PRODUCTION FLEXIBLES	
1 - INTRODUCTION	7
2 - ATELIERS FLEXIBLES	8
3 - CONFIGURATIONS D'UN ATELIER FLEXIBLE	9
3.1 ATELIER DE PRODUCTION UNITAIRE	9
3.2 ATELIER DE PRODUCTION PETITE ET MOYENNE SERIES	10
3.3 ATELIER DE PRODUCTION GRANDE SERIE	12
4 - PRESENTATION DE LA CELLULE DE PRODUCTION FLEXIBLE ENVISAGEE	13
4.1 PROBLEMES SOULEVES	13
4.1.1 Synchronisation et communication entre les robots	14
4.1.2 Evitement d'obstacles	15
4.1.3 Contrôle et diagnostic	16
4.1.4 Allocation dynamique des tâches et des ressources	17
4.2 CARACTERISTIQUES PRINCIPALES	17
4.2.1 Moyens matériels	18
a) Moyens de production	18
b) Moyens de transport	19
c) Architecture matérielle	20
4.2.2 Moyens logiciels	21
a) Architecture logicielle	21
b) Algorithmes de décision	21
c) Communication avec l'extérieur	21
5 - PRESENTATION DE QUELQUES CELLULES DE PRODUCTION UTILISANT LA COOPERATION ENTRE ROBOTS	22
6 - PRESENTATION DU SITE EXPERIMENTAL	23
6.1 INTRODUCTION	23
6.2 DESCRIPTION DE LA CELLULE	24
7 - CONCLUSION	28
II - SYSTEMES DE CONDUITE D'ATELIERS	
1 - INTRODUCTION	30
2 - ORDONNANCEMENT	31
2.1 PROBLEME GENERAL	31
2.2 DESCRIPTION DU PROBLEME	32

2.2.1	Tâches	32
2.2.2	Processeurs	33
2.2.3	Ressources	33
2.2.4	Contraintes	34
2.2.5	Fonctions objectifs	35
3 -	MODELES DE CONDUITE DES SYSTEMES DE PRODUCTION	36
3.1	ORDONNANCEMENT STATIQUE	38
3.1.1	Méthodes de programmation linéaire à variables mixtes	38
3.1.2	Méthodes par Séparation et Evaluation Progressive ou Séquentielle (SEP ou SES)	39
3.1.3	Méthodes d'analyse combinatoire	40
3.1.4	Méthodes heuristiques	40
3.1.5	Simulation	41
3.2	ORDONNANCEMENT DYNAMIQUE	43
4 -	APPROCHE RETENUE	44
4.1	STRUCTURE DECISIONNELLE	44
4.2	POSITION DU PROBLEME	45
5 -	CONCLUSION	47

III - ETUDE DE LA CELLULE DE FABRICATION FLEXIBLE

1 -	INTRODUCTION	50
2 -	MODELISATION	50
2.1	ETAT DU PRODUIT EN COURS DE REALISATION	51
2.1.1	Décomposition de la réalisation du produit	51
a)	Définition d'une opération	52
b)	Caractéristiques d'une opération	52
c)	Exemple	53
2.1.2	Etats d'une opération	55
2.1.3	Evolution de l'état du produit	61
2.2	ETAT DE L'ENVIRONNEMENT	62
2.3	ETAT DES ROBOTS	64
2.4	ENSEMBLE DES COMMANDES ADMISSIBLES	66
3 -	STRUCTURE DU SYSTEME DE CONDUITE	66
3.1	INTRODUCTION	66
3.2	INITIALISATION	68
3.2.1	Description de la gamme de fabrication	68
3.2.2	Description de l'environnement	69
3.2.3	Description des robots	69
3.3	SYSTEME D'INFORMATION	70
3.3.1	Identification	70
3.3.2	Contrôle	70
3.3.3	Diagnostic	70
3.3.4	Mise à jour	71
a)	Etat de l'environnement	71

b) Etat des robots	71
c) Etat du produit	72
3.4 SYSTEME DE DECISION	72
3.4.1 Affectation	72
3.4.2 Lancement	73
3.5 COMMUNICATIONS EXTERNES	73
3.5.1 Communication Homme-Machine	73
3.5.2 Communication Machine-Machine	73
4 - CONCLUSIONS	73

IV - MODULE DE DECISION

1 - INTRODUCTION	76
2 - PROBLEME DE LA DECISION OPTIMALE	76
3 - APPROCHE STOCHASTIQUE	80
3.1 RAPPELS SUR LES PROCESSUS DE MARKOV	80
3.1.1 Définition	80
3.1.2 Programmation Markovienne	80
3.2 CONCLUSION	82
4 - APPROCHE MULTICRITERES	83
4.1 INTRODUCTION	83
4.2 POSITION DU PROBLEME	86
4.3 CHOIX DES CRITERES	87
4.4 METHODE DE RESOLUTION FONDEE SUR LA NOTION DE SURCLASSEMENT	93
4.4.1 Introduction	93
4.4.2 Description de la méthode ELECTRE	94
4.5 APPLICATION	95
4.5.1 PREMIERE APPROCHE	96
4.5.2 DEUXIEME APPROCHE	97
4.5.3 SEUILS DE DISCORDANCE	99
4.5.4 PONDERATION DES CRITERES	102
4.5.5 EXEMPLES	103
5 - CONCLUSION	107
CONCLUSION GENERALE	109
ANNEXE	112
BIBLIOGRAPHIE	122

INTRODUCTION GENERALE

Coopérer : Agir conjointement avec quelqu'un

Coopération : Action de coopérer. Méthode d'action économique par laquelle des personnes ayant des intérêts communs constituent une entreprise où les droits de chacun à la gestion sont égaux et où le profit est réparti entre les seuls associés au prorata de leur activité.

"Grand Larousse"

Jusqu'à ces dernières années, les robots industriels ont été conçus et utilisés pour remplacer l'homme dans la réalisation des tâches pénibles et répétitives. Ils ont permis une nette amélioration des conditions de travail, ainsi que des gains importants en productivité.

Le développement de la technologie (calculateurs, capteurs, ...) a ouvert de nouveaux horizons à la robotique et a conduit naturellement l'homme à rechercher une utilisation des robots pour d'autres fonctions plus "intelligentes" (i.e adaptation à l'environnement par exemple).

L'accroissement de la compétition économique dans l'industrie, dans l'espace pousse à l'automatisation de tâches de plus en plus complexes, nécessitant parfois la collaboration de plusieurs individus en temps normal.

Ainsi, une nouvelle voie était ouverte en robotique : la coopération entre robots.

Deux domaines nous semblent être son terrain de prédilection :

- les milieux hostiles ;
- les systèmes de production.

Dans les milieux hostiles, donc inaccessibles directement à l'homme, ou dangereux, comme les fonds marins, les zones polluées, le cœur des centrales nucléaires, l'intérieur des canalisations, l'espace, etc..., la nécessité de procéder à des expérimentations, à des travaux de réparation, à des recherches, ..., est évidente.

Le développement d'engins capables de remplacer l'homme dans les différentes opérations citées, est un problème réel, et d'actualité.

Aujourd'hui, la plupart des engins utilisés sont télémanipulés dans leur grande majorité. Un exemple récent est celui du bras manipulateur associé à la navette spatiale américaine, qui a permis la récupération d'un satellite en panne dans l'espace, afin de le réparer à bord.

Les perspectives de développement de cette branche conduisent non pas au remplacement de la téléopération, par des systèmes robotisés autonomes, ce qui est difficile à envisager à l'heure actuelle, mais par l'intégration de "modules" ou "cellules" autonomes, pour la réalisation d'une classe de problèmes donnée, dans un système de téléopération.

Ce système, où il y a une coopération entre hommes et robots est une étape intermédiaire (pour quelques années encore) nécessaire pour arriver à la mise en œuvre de systèmes robotisés totalement autonomes.(HAB.81)

Dans les systèmes de production industriels, tous les moyens sont utilisés afin de satisfaire un objectif commun (maximisation de la productivité, de la production, amélioration de la qualité, etc...). Il y a donc coopération. Mais celle-ci peut être appréciée différemment suivant le niveau d'interaction entre les divers éléments évoluant dans les systèmes de production.

Dans les ateliers dit "classiques", tels les chaînes transfert par exemple, les moyens de production sont spécialisés. Ils obéissent à un programme de travail figé, qui ne peut être modifié qu'avec une remise en cause de l'ensemble de l'organisation matérielle et de la commande de la chaîne.

Or dans les milieux industriels, de nouvelles données ont bouleversé le marché, surtout dans la production petite et moyenne série. En effet l'apparition continue sur le marché d'une multitude de produits nouveaux, engendre une supériorité de l'offre sur la demande. Ainsi, la durée de vie de certains produits devient pratiquement imprévisible. On comprend alors le besoin et l'intérêt de l'adaptabilité et la flexibilité des systèmes de production. Ils doivent être capables, d'une part, de répondre à de fortes demandes, en augmentant la cadence de production, d'autre part de changer de production en cas d'une très faible demande, sans que cela ne perturbe trop la structure initiale du système et suffisamment rapidement pour garder au produit sa compétitivité.

On voit aujourd'hui de plus en plus d'études consacrées à ces systèmes et plusieurs réalisations d'ateliers flexibles ont déjà vu le jour, un peu partout dans le monde (JAPON, U.S.A., FRANCE, R.F.A., ...).

Ces ateliers se caractérisent par :

- la réalisation de plusieurs produits simultanément ;
- une décision en temps réel, tenant compte de l'état courant de l'atelier.

Dans la plupart des ateliers flexibles opérationnels aujourd'hui, les divers moyens de production obéissent chacun à un programme de travail établi par un niveau central. Mais ils ont la possibilité de prise de décision locale pour certaines fonctions (changement d'outils, détection de défauts, ...).

En cas d'événements imprévus ou de changement d'objectif, c'est le niveau de décision hiérarchiquement supérieur qui décide de modifier le programme de travail ou d'en télécharger un autre.

Il n'y a pas d'interaction directe entre les moyens de production.

Actuellement, plusieurs travaux sont consacrés à l'étude de systèmes de production utilisant une plus forte interaction entre les moyens de production.

L'équipe du L.A.A.S., développe dans le cadre du projet A.R.A., une cellule d'assemblage flexible expérimentale, utilisant des robots. [ALA 83].

Une cellule d'assemblage de composants électroniques est décrite dans [SAN 83], il existe d'autres exemples sur cet aspect de la flexibilité, dans la littérature [BOU. 84] [OKH. 84].

C'est justement dans ce cadre précis que nous avons situé notre travail, puisqu'il s'agit pour nous d'étudier les problèmes liés à la conduite d'une cellule de production flexible, où un certain nombre de robots collaborent à la réalisation d'un produit commun.

Notre travail se limite à la recherche d'un modèle de conduite de cette cellule, où les robots évoluent dans un contexte perturbé.

Dans le premier chapitre, nous donnons un aperçu sur les systèmes de production flexibles existants. Nous définissons ensuite les caractéristiques principales de la cellule de production envisagée. Enfin nous décrivons le site expérimental.

Dans le second chapitre, est défini le problème de la conduite de la cellule, c'est-à-dire le problème d'ordonnement-affectation. Les différentes méthodes de résolution, que l'on peut trouver dans la littérature, sont exposées. Enfin, nous développons l'approche que nous avons retenue, pour

résoudre ce problème dans le cadre de la cellule envisagée, la structure décisionnelle y est décrite, ainsi que la position du problème et les hypothèses de travail émises.

Le troisième chapitre est composé de deux parties. Dans la première, nous utilisons une modélisation de la cellule, par des graphes d'états et de transitions [BEL. 85]. Nous décrivons ensuite, dans la seconde partie la structure du système de conduite proposée pour le pilotage de la cellule. Ses différents modules y sont présentés. Une simulation d'une partie de ce système a été réalisée en MULISP sous CP/M 80.

Nous exposons enfin dans le quatrième chapitre, deux approches pour aborder le problème décisionnel.

La première est stochastique. Elle associe aux transitions entre états des probabilités et considère le système comme Markovien. Nous montrons qu'elle est difficilement envisageable pour un cas réel.

La seconde est déterministe. Elle ne tient compte dans son processus de décision que de l'état courant de la cellule, sans anticiper sur son évolution. Nous proposons dans ce cas une méthode basée sur une approche multicritères.

Son application à notre cellule de production est étudiée et des exemples sont traités.

Enfin quelques remarques générales, sur l'étude réalisée et ses perspectives de développement conclueront ce travail.

Chapitre I

SYSTEMES DE PRODUCTION FLEXIBLES

1 - INTRODUCTION

L'atelier flexible ne définit pas à l'heure actuelle une structure industrielle bien précise. On trouve sous ce vocable plusieurs réalisations profondément différentes d'un pays à un autre et d'une firme à une autre. Cependant on peut dire que d'une manière générale, ils correspondent à l'automatisation des chaînes de production des petites et moyennes séries, même si actuellement on envisage leur extension aux productions en grande série.

L'utilisation de la flexibilité se justifie par le fait que l'automatisation des chaînes transfert classiques, pour la production en petites et moyennes séries, est une opération très coûteuse. En effet leur adaptation aux changements de gammes de produits nécessite une totale refonte de leur structure physique et de leur structure d'exploitation.

Lorsqu'on sait qu'une très grosse proportion (3/4) de la production industrielle concerne la fabrication en petite et moyenne séries, on mesure le besoin et l'intérêt de la conception de structures flexibles.

Leur principal atout est justement de pouvoir fabriquer simultanément plusieurs produits et donc de s'adapter sans trop de frais à de nouvelles gammes de production.

La flexibilité d'un atelier ne peut être possible que par celle de l'ensemble de ses composants :

- matériels (moyens de production, moyens de transport, ...)
- informatiques.

Le rôle et les caractéristiques de tous ces éléments seront détaillés dans les paragraphes suivants.

D'après une étude faite dans [DUP 82], sur les systèmes de production flexibles réalisés dans plusieurs pays, il ressort trois grands axes de développement :

- modules ou cellules flexibles ;
- lignes transfert flexibles (type Flow-shop ou Job-shop) ;

- systèmes flexibles où l'emplacement des moyens de production n'obéit pas à une structure linéaire (type Open-shop).

Dans le cadre des projets, celui des Japonais, M.U.M. (Methodology for Unmanned Manufacturing System), mérite d'être cité. Il s'agit de la mise en œuvre d'un complexe de production flexible qui regroupe plusieurs cellules spécialisées dans le formage, l'usinage et l'assemblage.

C'est d'ailleurs dans cette structure d'atelier, composée de différentes cellules de production que s'inscrit notre démarche, puisqu'il s'agit de l'étude des problèmes de conduite d'une cellule de production flexible utilisant la coopération entre robots.

Le but de ce chapitre est alors de caractériser cette cellule.

2 - ATELIERS FLEXIBLES [DUP. 82] [BON. 85]

Une définition rigoureuse d'un atelier flexible n'existe pas. On peut néanmoins dire que selon la littérature consultée, on s'accorde sur les éléments de définition suivants :

"Un atelier flexible est un système de production automatisé capable de réaliser simultanément plusieurs types de produits et en quantité variable.

L'opérateur humain n'intervient pas dans le processus de fabrication.

Enfin, l'atelier est géré en temps réel par un ordinateur".

On peut ajouter que c'est un système de production adaptatif.

On a vu que la flexibilité de l'atelier est en fait liée à celle de ses composants (moyens de production, de transport, et informatiques).

Les moyens de production doivent être capables de traiter plusieurs types de pièces, donc de s'adapter à de nouvelles gammes de produits d'une manière automatique et sans que cela ne perturbe la production globale (limitation des retards, ...).

Les moyens de transport sont flexibles dans le sens d'un cheminement non figé dans l'atelier, d'une possibilité de manutention de produits divers, de stockage, de destockage, etc...

L'approche informatique dans la conduite de l'atelier doit être modulaire afin de permettre une intégration facile d'autres éléments, sans remettre en cause la structure déjà implantée.

La spécification de l'ensemble de ces éléments, organisationnels matériels et informatiques est arrêtée dans les différentes phases d'étude de l'atelier.

Il existe en fait deux phases. La première est une phase de conception et est relative aux structures et moyens matériels. La seconde est une phase de conduite, et correspond au monde de gestion de l'atelier (planification, ordonnancement, gestion des stocks, diagnostic de pannes, ...).

3 - CONFIGURATIONS D'UN ATELIER FLEXIBLE

Lorsque l'on dit qu'une chaîne flexible est capable de réaliser plusieurs produits différents, il faut préciser que cela n'est concevable que si ces produits appartiennent à une même famille. On ne peut pas sur le même site produire des carters, des bielles, des pistons, etc.

De ce point de vue, il est alors logique de penser organiser l'atelier en groupes de machines, où chacun est spécialisé dans une classe d'opérations donnée.

C'est un premier niveau d'organisation de l'atelier. Un second niveau correspond à la structure du groupe lui-même.

Cette structure dépend de la famille de produits à réaliser, du matériel utilisé, et du mode de production.

Les ateliers flexibles sont généralement classés en trois types, relativement aux trois modes de production suivants :

- la production unitaire ;
- la production petite et moyenne série ;
- la production grande série, qui est nouvellement envisagée.

3.1 ATELIER DE PRODUCTION UNITAIRE

Ce sont des modules ou cellules flexibles composés pour la plupart d'un centre d'usinage et parfois seulement d'un centre de tournage. Il n'en existe pas à notre connaissance, pour d'autres opérations comme l'assemblage par exemple, où seules des chaînes très spécialisées sont utilisées à l'heure actuelle et dans la plupart des cas, elles sont réservées à la production en séries.

Ces modules sont caractérisés par l'automatisation des opérations de chargement et de déchargement à l'aide de manipulateurs industriels et par l'existence d'un stock leur assurant une certaine autonomie dans le cas d'une perturbation dans l'alimentation en pièces.

Celles-ci, de forme quelconque, mais appartenant à une même famille de taille, peuvent subir toutes les opérations réalisables dans ces cellules.

3.2 ATELIER DE PRODUCTION PETITE ET MOYENNE SERIES

Il existe deux modes d'organisation de ce type d'atelier. Le premier est très proche des lignes transfert classiques, c'est ce qu'on appelle les lignes transfert-flexibles. Le second est une structure non linéaire, où les moyens de production sont "éparpillés" dans l'atelier. Il correspond aux systèmes flexibles "non-alignés" [DUP. 83].

3.2.1 Ligne transfert flexible

C'est une association en ligne, de plusieurs moyens de production (machines à commande numérique, ou machines spécialisées).

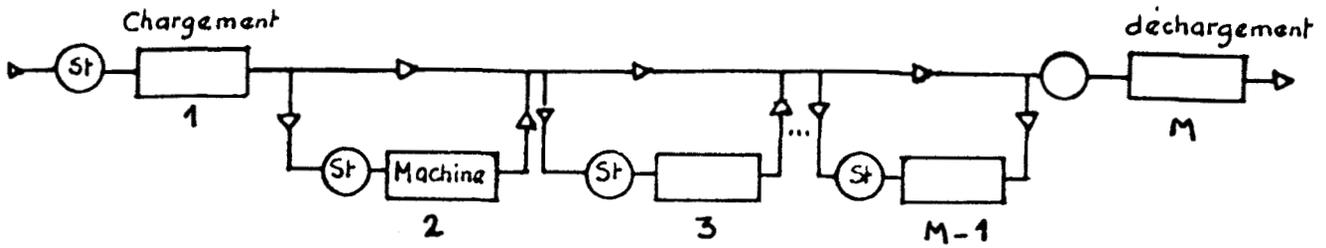
Ils sont alimentés par un système de transport, qui est généralement un tapis à rouleaux.

La flexibilité se traduit à travers les aptitudes différentes de chacune des machines de la ligne, et ainsi par la possibilité de fabriquer plusieurs types de pièces simultanément (de 4 à 100) [LEV. 82]. Ceci peut exiger un nombre d'outils considérable.

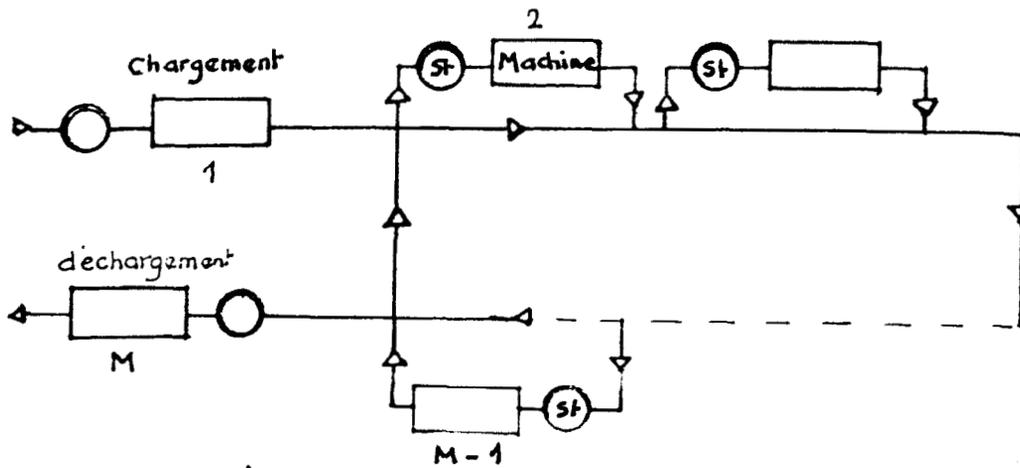
Selon la spécification du mouvement de la pièce dans l'atelier, on aboutit à deux configurations différentes de celui-ci.

Dans le cas où l'ordre de passage des pièces sur les machines est imposé et est le même pour chaque pièce, on dit que l'atelier est de type "FLOW-SHOP" (Fig. 1). Les pièces ne passent pas forcément sur chaque machine.

Si l'ordre de passage reste imposé, mais peut différer d'une pièce à une autre, on dit alors que l'atelier est de type "JOB-SHOP" (Fig. 2). Cela suppose l'existence de moyens de transport entre les différents postes de travail.



- fig 1 -



- fig 2 -

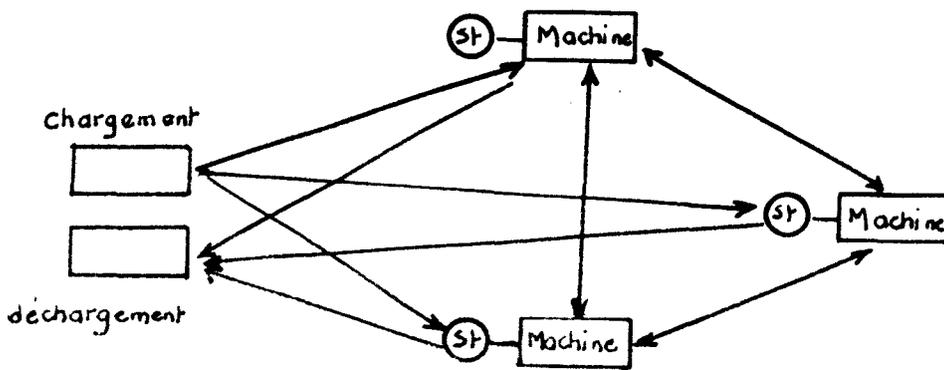
3.3.2 Système de production flexible "non-aligné"

La disposition des moyens de production n'obéit pas à une structure linéaire. C'est une organisation plus souple, qui nécessite des moyens de transport adaptés.

L'organisation physique de l'atelier est donné en Fig. 3.

Lorsqu'aucun chemin n'est fixé à priori pour la pièce dans l'atelier, on dit de celui-ci qu'il est du type "OPEN-SHOP".

Les ateliers flexibles de CITROEN à Meudon, ou d'Allis-Chalmer (U.S.A.) en sont des exemples. Dans ce dernier, les chemins de manutention sont implantés de façon à permettre une livraison aléatoire aux postes de travail.



. Fig 3 .

3.3 ATELIER DE PRODUCTION GRANDE SERIE

Si pour la production petite et moyenne séries, la nécessité de la flexibilité est reconnue, pour les grandes séries, cette approche est nouvelle.

On explique habituellement ce manque d'intérêt par les raisons suivantes :

La longévité d'une chaîne de production d'un produit donné, compense dans une large mesure le temps consacré à sa mise en œuvre. Elle est certes, rigide mais elle optimise la production.

Si la demande sur le marché de ce type de produit ne s'atténue pas et si la production sur la chaîne se maintient, le problème de la flexibilité ne se pose pas.

Or, il n'en est pas toujours ainsi, et ceci pour différentes raisons. La principale est relative aux éventuelles et nécessaires retouches que l'on est contraint d'apporter au produit de par l'évolution technologique, une nouvelle législation, etc...

Par exemple, certaines chaînes (tubulures d'admission et d'échappement des moteurs) ont dû être refaites trois à quatre fois à quelques années d'intervalle. Une conception flexible aurait permis de réduire considérablement les coûts relatifs à ces changements.

C'est une approche qui n'est pas encore adoptée mais elle est sérieusement prise en compte par plusieurs grands groupes, notamment par Général Motors. Ce dernier avait envisagé de fabriquer tous ses produits

dans des centres d'usinage. Nous n'avons pas d'informations concernant l'aboutissement de ce projet.

Après la présentation des différentes configurations d'ateliers flexibles, il convient maintenant de préciser le système de production dans lequel la coopération entre robots est étudiée ainsi que ses caractéristiques principales.

4 - PRESENTATION DE LA CELLULE DE PRODUCTION FLEXIBLE ENVISAGÉE

L'étude de la coopération entre robots ne peut être envisagée dans le cadre d'un atelier, mais dans un système de dimensions limitées, c'est-à-dire une cellule de production flexible.

Le but que l'on assigne à cette dernière est d'être un cadre où des robots peuvent collaborer les uns avec les autres afin de réaliser un produit commun.

Pour ce faire la structure de la cellule tant matérielle qu'informatique et donc de l'ensemble des moyens qui le composent doit être adaptée, c'est-à-dire permettre l'étude des problèmes posés et l'expérimentation des solutions proposées.

Dans la première partie de ce paragraphe, sont décrits les principaux problèmes liés à l'étude de la conduite de cette cellule. La seconde partie concerne la spécification des caractéristiques principales des divers composants, matériels et informatiques qui interviennent dans la cellule.

4.1 PROBLEMES SOULEVES

La coopération entre des robots interagissant entre eux afin de réaliser un produit commun peut se traduire de plusieurs manières :

- 1 - Par une coordination des mouvements de plusieurs robots œuvrant à la réalisation d'une même opération.

Exemple : Opération à deux bras.

A partir de cet exemple, il ressort au moins deux problèmes généraux :

- synchronisation et communication entre les robots ;
- évitement d'obstacles, ou anti-collision.

- 2 - Par la préparation par un robot des ressources nécessaires à l'exécution d'une opération par un autre robot.

Exemple : Mise en stock de pièces.

Le problème posé par cet exemple est relatif à :

- l'allocation dynamique des ressources (espace de stockage).

- 3 - Par le remplacement d'un robot indisponible, par un autre robot, pour la réalisation d'une opération.

Exemple : Robot défaillant à remplacer.

Les problèmes que pose cet exemple, concernent :

- l'allocation dynamique des tâches ;
- le contrôle et le diagnostic.

D'autres problèmes peuvent encore être cités, mais nous nous limitons à l'exposé de ceux qui nous semblent être les principaux.

4.1.1 Synchronisation et communication entre les robots

La réalisation d'une tâche nécessite obligatoirement un contrôle, soit continu, soit périodique en cours d'exécution pour constater la conformité ou non de la tâche par rapport à sa définition et réagir en conséquence.

Exemples :

- serrage avec contrôle de l'effort ;
- suivi de joint de soudure en temps réel.

D'autres tâches ont besoin en plus de synchroniser leurs exécutions respectives, ce qui se traduit par une coordination des mouvements des différents robots qui les réalisent.

Exemples :

- assemblage de deux pièces par deux bras manipulateurs.

La connaissance de leurs états respectifs en temps réel est nécessaire pour la conduite à bien de la synchronisation de leurs mouvements. Cela suppose donc des moyens de communication et de contrôle adéquats (protocole de transmission, vitesse de transfert de l'information, performance des capteurs utilisés, etc...) et une organisation matérielle et logicielle qui répondent à la situation.

La structure matérielle correspond généralement à un réseau local (en anneau, ou en étoile) [NIC. 79].

La structure logicielle est relative à l'organisation du contrôle, centralisée ou répartie.

Dans cette dernière, le robot a une totale autonomie dans la décision. Il gère l'échange d'informations, le partage des ressources communes, avec l'ensemble des robots afin de réaliser son objectif.

C'est une approche qui peut sembler intéressante a priori quoique difficile à mettre en œuvre. En effet, la mise en communication de plusieurs processus nécessite un débit important d'informations échangées. De ce fait, leur gestion devient complexe ; une analyse exhaustive ne peut être envisagée sans pénalité, l'élimination totale du risque d'interblocage n'est pas toujours garantie.

Dans l'approche centralisée, un organe central ou superviseur a connaissance de l'état global de la cellule et en fonction de celui-ci, décide de l'évolution de chaque élément de la cellule, c'est-à-dire en allouant les tâches aux robots, en mettant à jour la base de données, en reconfigurant la cellule, etc...

C'est une approche sûre, mais très contraignante par sa rigidité et sa lourdeur.

Entre ces deux approches extrêmes, il existe une approche mixte. Elle consiste à donner aux robots une certaine autonomie de décision pour des opérations spécifiques (choix d'une trajectoire par exemple) et laisser à un superviseur la gestion des conflits éventuels entre les robots, et/ou d'autres fonctions nécessaires à la conduite de la cellule.

4.1.2 Evitement d'obstacles

La coopération entre les divers moyens de production admet l'évolution de plusieurs d'entre eux dans un espace commun. Il y a donc un risque certain de collision qui doit être pris en considération.

L'évitement d'obstacles dynamiques est à lui seul un sujet très complexe. Il est approché de deux manières. L'une est analytique, l'autre est basée sur les méthodes de l'intelligence artificielle, et plus particulièrement la génération de plans.

Un exemple remarquable sur la coordination des mouvements de deux robots évoluant dans un espace commun est décrit dans [PARK. 84].

L'approche utilisée est basée sur la représentation de l'espace d'état des deux robots. Néanmoins, elle est difficilement envisageable d'une part, par la difficulté de mise en œuvre et d'autre part, par les moyens de calcul qu'elle exige.

La génération de trajectoires dans un milieu dynamique utilisant les techniques de l'intelligence artificielle, n'a pas d'exemple d'application concret, à notre connaissance. C'est un sujet de recherche actuel et les seuls exemples sont ceux des laboratoires.

4.1.3 Contrôle et diagnostic

La coopération entre robots dans une cellule de production exige un contrôle strict de l'état des divers éléments évoluant dans cette cellule, afin de permettre le bon déroulement des opérations.

Suivant le résultat du contrôle, un diagnostic doit pouvoir être établi afin qu'une décision soit prise le plus rapidement possible et ainsi éviter que d'autres décisions soient fondées sur une base de données incohérentes, parce qu'elle n'a pas été remise à jour à temps.

Le diagnostic est un sujet complexe. Selon la finesse avec laquelle on veut localiser la défaillance, il peut être approché de diverses manières, par exemple par un système totalement automatique. Celui-ci aura à détecter soit précisément l'élément défaillant, soit globalement la région défaillante, pour permettre sa prise en compte rapide par le système de décision. Le diagnostic fin sera laissé dans ce cas, à l'opérateur ou à un système hors ligne d'aide à la décision. Ce système pourra être conçu autour d'un système expert qui permet interactivement avec un opérateur, de déterminer précisément la défaillance.

Celle-ci peut être :

- une mauvaise exécution d'une opération de fabrication ou de stockage, suite à une panne de robot, à une ressource non conforme, ... ;
- une perte d'aptitudes d'un robot, à cause de l'usure ou de la casse d'un outil, d'un défaut sur un capteur (vision, effort, ...) ;
- une mise hors service complète d'un robot, pour la maintenance, ...

Le passage en fonctionnement en mode dégradé doit être effectué automatiquement par le système, sur la base du diagnostic.

4.1.4 Allocation dynamique des tâches et des ressources

L'affectation des tâches aux robots, des ressources aux tâches (pièces à assembler par exemple), ou aux robots (outils partageables par les robots), est un problème essentiel dans la conduite d'une cellule de production. C'est de la manière dont il est résolu que dépendent les performances globales de la cellule.

Il s'agit le plus souvent d'un problème d'optimisation d'un coût, d'un délai de réalisation ou d'un autre critère. Ceci conduit dans la plupart des cas à la recherche d'une maximisation du parallélisme dans la réalisation des tâches, tout en respectant leurs contraintes de synchronisation, de succession, de disjonction, ...

La prise en compte de la dynamique de la cellule de production, qui peut se traduire par des événements imprévus tels que des pannes (partielles ou totales) sur les robots, une rupture dans un stock de ressources, des perturbations dans l'alimentation en pièces, etc..., conduit nécessairement à l'utilisation de méthodes d'allocation dynamiques.

L'affectation d'une tâche à un robot, à un instant donné, en vue d'optimiser un certain critère, dépend de plusieurs facteurs :

- de l'état courant des robots (en panne, actifs, libres, liste de leurs aptitudes) ;
- de l'état du produit en cours de réalisation ;
- de l'état du stock intermédiaire (s'il en existe un).

Les différentes approches et méthodes utilisées pour ce problème situé dans un cadre général feront l'objet du chapitre suivant.

La distribution des tâches est à notre sens le cœur du problème de la coopération entre robots.

L'étude de ces différents problèmes ne peut se faire que dans un cadre spécifique. C'est pourquoi, nous précisons dans ce qui suit, les principales caractéristiques de la cellule de production flexible envisagée.

4.2 CARACTERISTIQUES PRINCIPALES

L'étude et la conception d'une cellule de production flexible utilisant la coopération entre robots nécessitent un choix adéquat des moyens et structures matériels et informatiques. Ceci afin qu'elles répondent

aux exigences de la flexibilité et de la conduite en temps réel d'un processus industriel.

Les critères essentiels qui guident ce choix sont :

Flexibilité : La cellule doit pouvoir réaliser plusieurs produits différents et se reconfigurer en fonction des demandes.

Extensibilité : La cellule doit s'adapter à des configurations réduites ou étendues, selon les besoins de l'utilisateur.

Fiabilité : La cellule doit prendre en charge des pannes qui surviennent sur le matériel et garder toujours la possibilité de reprise en main de la commande.
Une défaillance simple ne doit jamais bloquer le système de conduite. Le fonctionnement doit être sûr.

Pour satisfaire ces critères, les composants de la cellule, matériels et informatiques, doivent présenter certaines caractéristiques que nous exposons dans ce qui suit.

4.2.1 Moyens matériels

a) Moyens de production

Les moyens de production utilisés dans la cellule de fabrication sont des robots.

Un des aspects de la coopération entre ces moyens est leur interchangeabilité quant à la réalisation de certaines opérations en cas d'indisponibilité de l'un d'entre eux. Ils doivent être capables d'exécuter plusieurs opérations différentes et donc être à aptitudes multiples. Ils permettent ainsi à la cellule, de se reconfigurer au cas où cela est nécessaire.

Ces aptitudes dépendent en fait, d'une part, des possibilités intrinsèques du moyen de production, comme pour un robot par exemple, la vitesse d'exécution, la force développée en bout de l'organe terminal, le nombre de degrés de liberté, d'autre part des outils disponibles sur le site et qui s'adaptent sur son porte-outil.

Un deuxième aspect de la coopération, relatif aux robots est lié à la coordination de leurs mouvements, lorsqu'ils réalisent des opérations synchronisées.

Ces robots doivent donc être munis de moyens de commande et de mesure conséquents (effecteurs, capteurs).

Parmi les fonctions que doivent remplir ces derniers, nous pouvons citer :

- la vision
- la mesure d'une distance, d'une force, ...
- l'appréciation par les robots de leurs états respectifs (position de l'outil terminal, vitesse, ...).

Une des qualités que l'on exige de ces robots qui interviennent dans un système très automatisé et flexible, est la fiabilité. Ceci afin d'éviter de fréquentes interventions d'opérateurs et de maintenir les mêmes performances, le plus longtemps possible pour garantir une même qualité de la production.

b) Moyens de transport

La principale fonction des moyens de transport est la distribution des produits à mettre en œuvre dans la cellule. Ces produits peuvent être des pièces ou des outils, ...

Cette distribution est accomplie conjointement par les moyens de transport propres à la cellule dont l'objectif est la circulation des produits d'un poste de travail à un autre, et par ceux de l'atelier dont le rôle est l'alimentation de la cellule en produits, l'évacuation de ceux-ci après exécution, ou utilisation, etc...

Le transport est organisé de manière à satisfaire certains critères, dont les principaux sont :

- rapidité et fiabilité dans l'acheminement des produits aux postes de travail, afin de maximiser l'engagement de ces derniers, et d'être sûr pour le personnel et le matériel.

Les systèmes de transport généralement utilisés dans une cellule, se limitent à notre connaissance au seul tapis roulant (à rubans, ou à rouleaux).

Par contre, pour alimenter la cellule, on rencontre le plus souvent :

- des chariots guidés (par fil enterré ou par bande de peinture)
- des chariots tractés à l'aide de chaînes enfouies dans le sol.

c) Architecture matérielle

Les moyens matériels informatiques représentent les différents automates et calculateurs qui doivent dans le cadre d'une architecture donnée et en fonction de certaines règles, commander et contrôler, l'ensemble des processus physiques qui constituent la cellule.

L'architecture matérielle ou le support de communication entre les divers matériels informatiques peut être organisé de deux manières :

- liaisons point-à-point ;
- liaisons multipoints ou réseau.

C'est cette dernière qui intéresse de plus en plus les milieux industriels et qui est parfaitement envisageable pour la cellule de production.

Elle doit répondre à certaines caractéristiques :

- Souplesse, fiabilité :

Le support physique de transmission de l'information doit permettre des vitesses de transfert élevées afin de prendre en compte rapidement l'état courant de la cellule.

Il doit avoir une haute immunité aux parasites, afin d'éviter des incidents éventuels sur le produit à réaliser ou sur le matériel.

Il existe deux moyens de transmission :

- par fil électrique (câble coaxial, bifilaire)
- par fibre optique.

Ce dernier n'est pas encore très utilisé en raison des difficultés de mise au point dans le milieu industriel.

- Modularité :

L'organisation du système de communication doit permettre l'intégration (ou l'élimination) de matériels dans (de) la cellule sans perturbation de la structure globale. Ainsi, la reconfiguration du système de production en vue de satisfaire un nouvel objectif est assurée sans trop de frais.

- Adaptabilité :

L'utilisation possible de matériels d'origines diverses dans la cellule, oblige à faciliter leur intégration par l'élaboration, par exemple, d'un protocole d'échange d'informations unique, qui s'interface sur chacun des matériels.

4.2.2 Moyens logiciels

a) Architecture logicielle

Les caractéristiques de l'architecture matérielle, souplesse, fiabilité, modularité, doivent aussi être demandées à l'architecture logicielle. La plus importante est à notre sens, la modularité.

Elle permet la modification d'un programme (correspondant à un module), le rajout d'un autre, suite à l'intégration d'un nouvel élément dans la cellule, sans aucun changement dans la structure globale. Elle facilite par conséquent l'extension, ou la reconfiguration de la cellule.

L'architecture informatique doit permettre un niveau élevé de dialogue entre les éléments de la cellule, et entre celle-ci et l'extérieur.

Un autre aspect important qu'elle doit autoriser, est relatif à la répartition de la décision au niveau des divers éléments de la cellule.

Entre les approches extrêmes, l'une totalement répartie et l'autre totalement centralisée, il nous semble plus réaliste d'envisager une approche mixte (& 4.1.1). Cette dernière est utilisée dans beaucoup de systèmes de production automatisés [BUZ. 82].

b) Algorithmes de décision

La caractéristique principale de la cellule, est la possibilité de prise en compte de son état courant, et donc des éventuels imprévus, dans la décision. Il est donc évident que les algorithmes qui seront choisis doivent tenir compte de cette dynamique et s'adapter à toutes les situations.

c) Communication avec l'extérieur

Les liaisons de la cellule avec l'extérieur doivent satisfaire plusieurs fonctions relatives :

- aux phases de mise au point des programmes ;
- à la reconfiguration de la cellule par l'opérateur ;

- aux phases de CAO ou de simulation ;
- au changement de la gamme de fabrication (ou téléchargement par un niveau supérieur de G.P.A.O.) ;
- au rattrapage d'une erreur ou d'un incident ;
- au changement d'objectif ;
- à l'affichage de l'état courant de la cellule.

5 - PRESENTATION DE QUELQUES CELLULES DE PRODUCTION UTILISANT LA COOPERATION ENTRE ROBOTS

Dans la bibliographie que nous avons consultée sur les systèmes de production flexibles, nous avons pu constater le nombre très limité de travaux concernant des cellules flexibles utilisant la coopération entre robots.

Toutes ces études ont été réalisées dans le cadre de cellules d'assemblage. Parmi celles-ci, deux méritent d'être présentées.

1 - Cellule d'assemblage flexible développée au LAAS (ALA, 83) (LAN, 83)

Dans le cadre du projet ABA, l'équipe du LAAS a développé un système de programmation flexible, NNS, d'un système robotisé d'assemblage. Ce dernier est composé de deux robots (un SCEMI 6 axes, et un TH8), d'un tapis roulant sur lequel arrivent des pièces d'une manière aléatoire, et d'un système de vision.

Le premier robot prend la pièce du tapis et la positionne sur un étai, le deuxième robot la prend et l'insère dans un support disposé dans un magasin. La caméra placée à l'entrée du tapis permet d'identifier les pièces.

L'organisation du système NNS, est hiérarchique et modulaire. Les différents sous-systèmes (robots, vision, ...) sont associés à des processus informatiques évoluant dans un environnement multitâches. La coordination est assurée par un processus maître.

Un des intérêts d'une telle démarche est de fournir à l'opérateur la possibilité de programmer sa manipulation de façon aisée, en écrivant le module maître.

2 - Station flexible d'assemblage à l'Université de Carnegie Mellan [SAN. 83]

La station d'assemblage développée à l'Université de Carnegie Mellan est composée de quatre manipulateurs (deux PUMAS, un SEIKO, une table X-Y-O), qui opèrent dans un espace de travail commun, d'un système de vision, ...

Le problème de la coordination de ces divers éléments est résolu au moyen d'une architecture "maître-esclave". Cette dernière a l'avantage de permettre la programmation de l'application sur le "processeur maître", et sur chacun des "processeurs esclaves", est implantée un sous programme relatif à ce qu'il doit faire.

Des expérimentations ont été effectuées sur cette station, dans le domaine de l'assemblage des composants électroniques afin de tester la faisabilité d'une telle cellule.

Remarques :

Nous constatons que dans les deux cas, la structure informatique choisie est centralisée et que les problèmes traités sont plus orientés informatiques, c'est-à-dire ayant trait aux problèmes de synchronisation et de communication entre les divers éléments évoluant dans la cellule. C'est le problème exposé en §4.1.1.

La particularité essentielle est qu'elles offrent à l'utilisateur la possibilité de programmer facilement son application.

6 - PRESENTATION DU SITE EXPERIMENTAL

6.1 INTRODUCTION

Dans le cadre du pôle productique régional du Nord-Pas-de-Calais, le laboratoire d'Automatique de l'U.S.T. de Lille, développe un atelier flexible expérimental dans le domaine de la confection.

La particularité de cet atelier par rapport à ceux conçus dans les industries mécaniques est la manipulation d'objets non rigides et déformables, que sont les pièces de tissus, de cuir, etc...

Cet atelier est composé de plusieurs postes de travail interconnectés de manière à prendre en charge tout le processus de confection, c'est-à-dire :

- alimentation et inspection en continu de la matière première ;
- découpe en continu selon un placement dynamique, tenant compte de l'inspection ;
- préhension et stockage des pièces ;
- assemblage automatique à l'aide de robots travaillant en coopération ;
- couture et évacuation.

Chaque poste de travail soulève un certain nombre de problèmes théoriques et pratiques qui occupent actuellement plusieurs chercheurs du centre d'Automatique, notamment l'étude d'un système de préhension de pièces déformables, le développement d'algorithmes de placement avant la découpe, la coopération entre robots en vue de l'assemblage des pièces et l'étude d'un système de supervision qui permet par le biais d'un réseau local de communication, de coordonner les différentes fonctions dévolues à chacun de ces postes de travail.

Notre cellule de fabrication intervient dans cet atelier flexible, au niveau des opérations d'assemblage.

Dans un premier temps et afin de nous affranchir des problèmes spécifiques à la préhension des pièces de tissus, nous avons préféré utiliser des pièces rigides et de forme simple qui peuvent être manipulées aisément par les robots.

Ceci, n'influence nullement le problème de la conduite de la cellule, qui est l'objet principal de cette étude.

6.2 DESCRIPTION DE LA CELLULE

La cellule de fabrication qui a été implantée sur le site expérimental est composée d'un certain nombre d'éléments robotiques, péri-robotiques, et de calcul.

a) Deux robots SCEMI 4C06

Ce sont des manipulateurs de type bras articulé rigide, à quatre degrés de liberté, correspondant à deux rotations principales du bras et à deux mouvements relatifs à l'organe terminal : une rotation de celui-ci et une translation verticale le long de son axe.

Ils sont munis au niveau de l'organe terminal d'une ventouse qui joue le rôle de préhenseur. Elle s'adapte parfaitement aux objets manipulés (cubes).

L'espace de travail de chacun des robots est asymétrique, ce qui a impliqué un aménagement particulier de leur implantation afin d'obtenir une zone de travail commune la plus grande possible.

Leur disposition est telle qu'elle offre trois plans de travail possibles, dont un leur est commun.

Leur programmation s'effectue à l'aide d'un langage de robotique de haut niveau, LM (langage de manipulation) [MAZ. 81] [MIR. 84] développée à l'IMAG.

b) Un système de vision VISIOMAT

C'est un système développé par MATRA, qui utilise un langage de programmation de haut niveau, LV (Langage de vision).

Il comprend un numériseur d'images, un extracteur de contours, un processeur de commande, construit autour d'un microprocesseur 16 bits.

c) Un tapis roulant

Il est linéaire et accessible aux deux robots. Des cellules photoélectriques sont installées sur son support afin de signaler la présence d'un objet devant les robots ou sous la caméra.

d) Calculateur : HP 1000

L'ensemble des éléments de la cellule est géré par l'intermédiaire d'un calculateur HP 1000 de la série A 600, construit autour du microprocesseur 2801.

Il travaille sous le système d'exploitation multi-tâches temps réel RTE. A.

L'organisation physique de la cellule de fabrication doit être telle que les différents problèmes cités et qui sont inhérents à la coopération entre les robots, puissent apparaître et donc être étudiés.

La structure que nous avons choisie est celle représentée par la figure 4.

Cette organisation offre de multiples possibilités d'expérimentations, tout en autorisant une augmentation croissante de la difficulté.

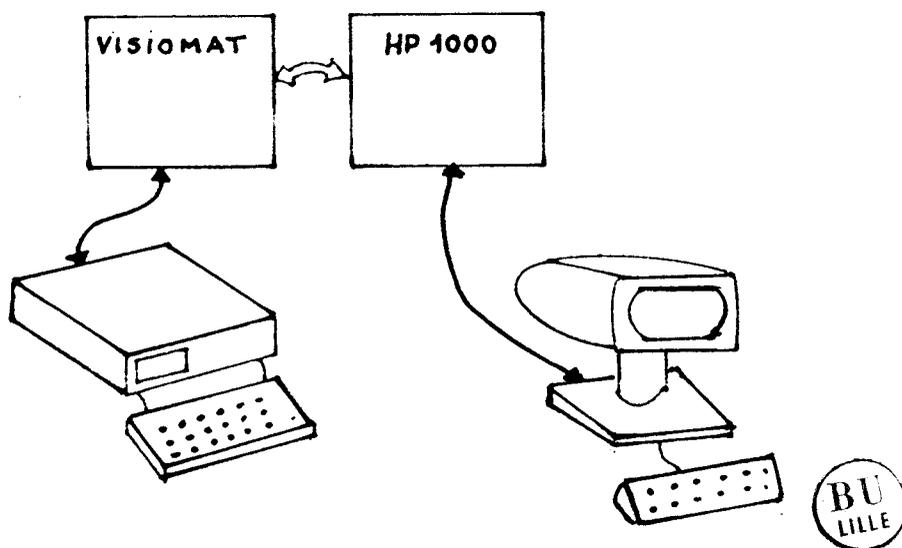
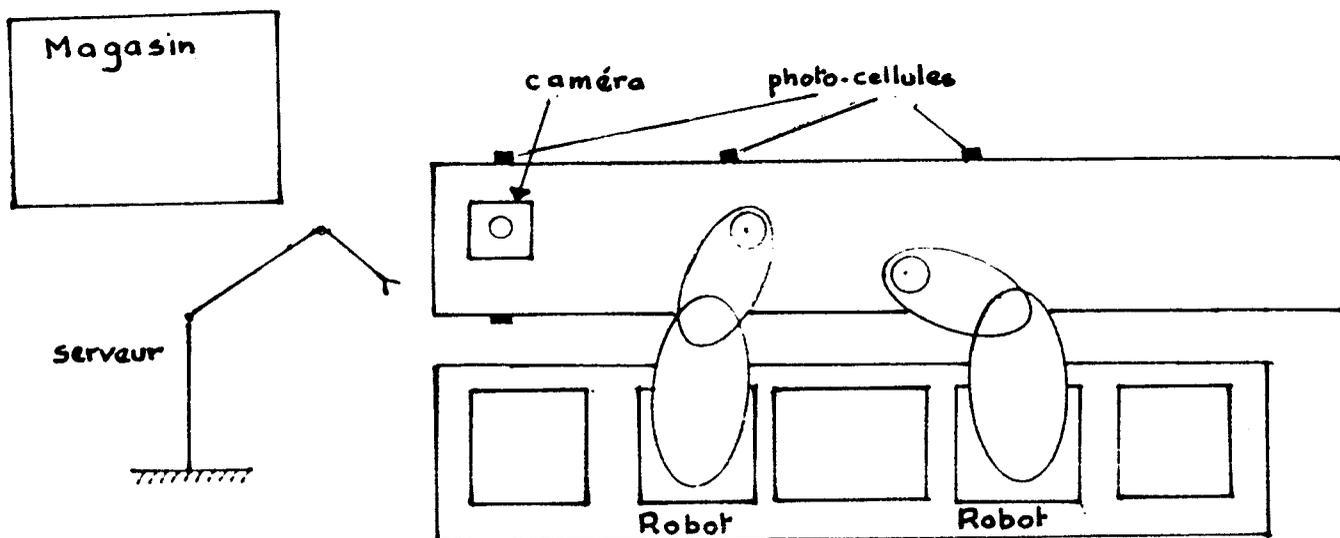
En dehors de l'atelier expérimental dans lequel elle est située, la cellule peut s'intégrer facilement dans d'autres structures plus importantes, comme par exemple une chaîne de production modulaire.

Elle illustre donc parfaitement une réalité industrielle dont on observe la naissance actuellement.

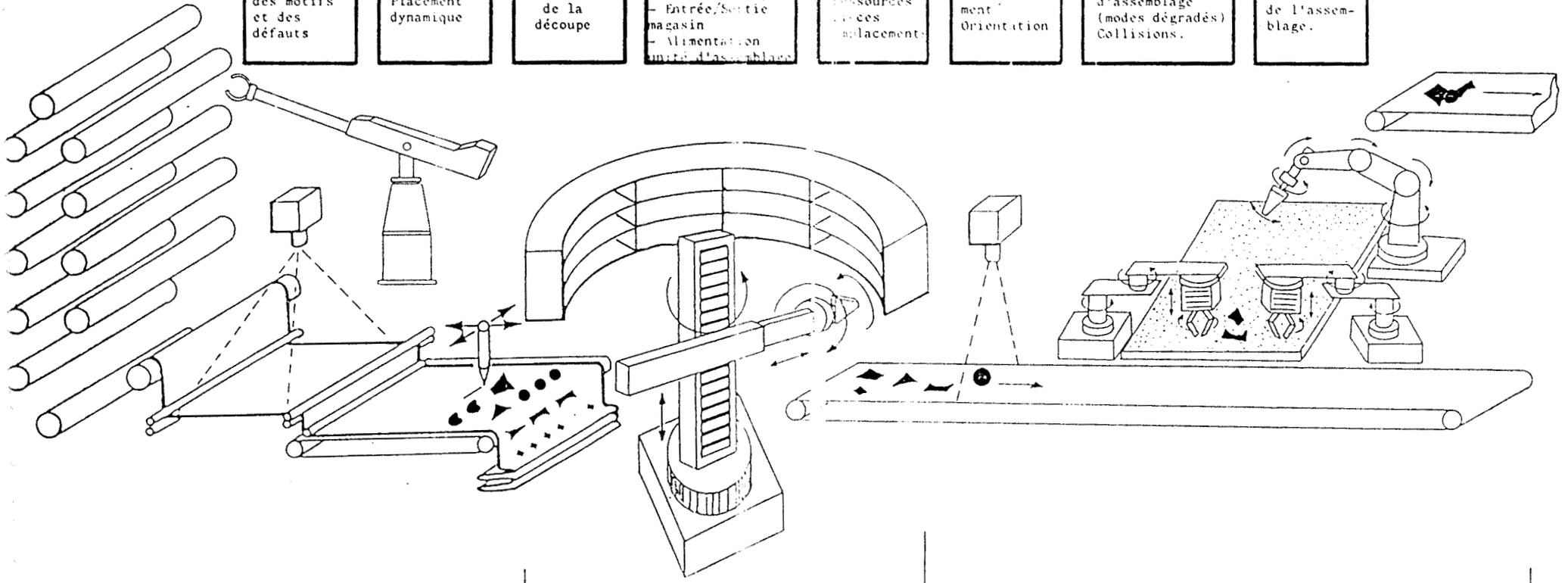
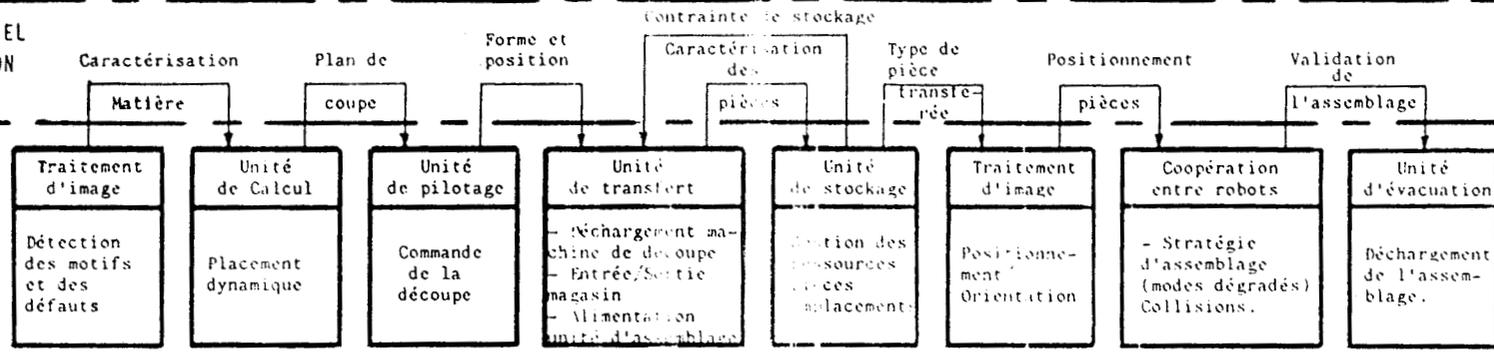
Le but de cette cellule est de permettre la fabrication d'un ou de plusieurs produits simultanément. Ceux-ci nécessitent généralement différentes opérations (assemblage, collage, ...) qui doivent obligatoirement être compatibles sur un même site.

Certaines de ces opérations ont besoin de pièces ou de matière première pour leur exécution. Celles-ci sont supposées produites en amont de la cellule et disponibles dans un magasin.

Ces ressources arrivent aux robots par l'intermédiaire du tapis roulant. Leur transfert du magasin au tapis est supposé accompli par un robot serveur.



RÉSEAU INDUSTRIEL
DE COMMUNICATION
ENTRE MACHINES



UNITE DE PREPARATION

MAGASINAGE

UNITE D'ASSEMBLAGE



7 - CONCLUSION

L'étude de la conduite d'une cellule de production flexible utilisant la coopération entre robots engendre un certain nombre de problèmes dont nous avons exposé les principaux.

Parmi ces derniers, nous nous sommes limités dans le cadre de ce travail à l'étude de celui qui nous semble être le cœur du problème de la conduite de la cellule, c'est-à-dire l'allocation des tâches aux robots.

La définition du problème de conduite en général, ainsi que les différentes approches et méthodes utilisées, font l'objet du chapitre suivant.

Chapitre II

SYSTEMES DE CONDUITE D'ATELIERS

1 - INTRODUCTION

L'objet de la conduite d'un système de production est de satisfaire les objectifs de production assignés, tout en utilisant au mieux les ressources disponibles.

La dimension et la complexité de ces systèmes sont telles qu'il est devenu classique de les décomposer en une structure hiérarchisée, ou pyramidale, afin d'essayer de prendre en compte toutes les composantes de cette complexité.

Cette approche hiérarchique semble actuellement la plus apte au développement de modèles de structures complexes d'organisation.

Cette décomposition est réalisée suivant deux axes. Le premier horizontal, correspond à la décomposition du processus physique (en machines uniques, en îlot de fabrication, ...). Le second vertical, correspond à celle du système de décision en plusieurs niveaux. A chacun d'eux, il est associé un horizon relatif à la validité de la décision prise. Ils sont composés généralement de plusieurs centres de décision.

Les liaisons qui existent dans cette structure sont de deux types :

- inter-niveaux, ou coordination ;
- intra-niveaux, ou coopération [KAL. 85].

Dans notre travail, nous nous situons au niveau bas de cette hiérarchie, puisque notre étude concerne la conduite d'une cellule de production, c'est-à-dire les problèmes d'ordonnancement-affectation et lancement, des opérations au niveau des robots compte-tenu de l'état courant de la cellule et de l'objectif à satisfaire.

Pour situer notre problème et l'approche que nous avons retenue pour le résoudre, notamment en ce qui concerne la structure décisionnelle et la méthode de résolution proposée, nous exposons les différents modèles de conduite étudiés dans la littérature.

Si pour les ateliers dits "classiques" où l'ordonnancement est statique, des méthodes analytiques peuvent être appliquées, il en est autrement des ateliers flexibles, où l'ordonnancement ne peut être que dynami-

que. En effet, le plus souvent, seules des méthodes heuristiques, ou fondées sur la simulation sont utilisées dans ce cas. De plus, les algorithmes appliqués dans les quelques ateliers flexibles opérationnels, sont jalousement tenus secrets pour des raisons "commerciales" sans doute.

Le but de ce chapitre est d'abord de définir le problème de l'ordonnancement, de présenter ensuite les méthodes de résolution dans les cas statique et dynamique et enfin de préciser la structure de conduite envisagée pour la cellule et les hypothèses sous lesquelles nous travaillons.

2 - ORDONNANCEMENT

2.1 PROBLEME GENERAL [ROY. 64] [DOU. 83] [CAR. 82]

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

Le problème d'ordonnancement est en fait, un problème d'optimisation où il s'agit de réaliser une action, en optimisant un ou des critères donnés, sous le respect de certaines contraintes.

Il se compose des éléments suivants :

- # Action ou tâches à réaliser
- # Critères ou fonction objectif à optimiser
- # Contraintes ou caractéristiques des processeurs, des ressources et des tâches.

La grande diversité des domaines que recouvrent les problèmes d'ordonnancement va de l'industrie (gestion de production, d'ateliers, ...) jusqu'à l'informatique (gestion de fichier, de mémoire, allocation de ressources,...) en passant par la construction (suivi de projet, ...), l'administration (emploi du temps, ...), etc...

La disparité entre ces domaines très variés et les problèmes spécifiques qu'elle engendre, le manque de définitions précises du vocabulaire utilisé (tâches, processus, processeur, ressources, contraintes, etc...) rend difficile une formalisation rigoureuse et unique du problème

général de l'ordonnancement. Chaque problème est un cas particulier avec ses contraintes propres, ses notations, son formalisme, etc...

Néanmoins, on peut dire qu'il existe deux grandes classes d'ordonnancement :

- Dans la première, l'ensemble des tâches à réaliser est parfaitement identifié (gamme d'usinage, conduite d'un chantier, ...). L'ordonnancement est fait hors ligne et n'est plus remis en cause.

- Dans la seconde, l'arrivée des tâches est aléatoire, et par conséquent, l'ordonnancement ne peut être que dynamique. Le système d'exploitation d'un ordinateur multipostes en est un exemple. En effet, celui-ci est accessible à plusieurs personnes qui l'utiliseront d'une manière totalement imprévisible.

Il existe différentes méthodes de résolution de ces divers problèmes, elles seront exposées dans (6.3.).

Poser un problème d'ordonnancement, c'est spécifier les caractéristiques des éléments qui le définissent, et que nous détaillons dans ce qui suit.

2.2 DESCRIPTION DU PROBLEME

2.2.1 Tâches

Une tâche est une opération spécifique au problème posé. Certaines données générales sont communes à toutes les tâches.

a) Données associées à une tâche

- Durée de l'exécution ;
- Localisation temporelle de l'exécution avec :
 - une date d'exécution au plus tôt ;
 - une date d'exécution au plus tard.
- Pénalité en cas de réalisation de la tâche, au delà de la date au plus tard ;
- Liste des processeurs capables de réaliser la tâche ;
- Liste des autres ressources associées à la tâche :

- . périphériques pour une tâche informatique ;
- . pièce pour une tâche d'assemblage, ...

b) Relations entre les tâches

Il y a deux possibilités. Dans la première, toutes les tâches sont indépendantes, c'est-à-dire qu'elles peuvent être réalisées en parallèle. Dans la seconde, les tâches sont liées par des relations de succession ou de synchronisation.

} autre cas
deux

c) Préemption

Dans plusieurs problèmes, la possibilité de réaliser une tâche en plusieurs parties existe et facilite le plus souvent la résolution du problème. Cela signifie qu'une tâche lancée peut être interrompue par une autre tâche plus prioritaire et est reprise à un autre moment. On dit que la tâche est préemptive.

Dans le cas contraire où l'interruption d'une tâche en cours est interdite, on dit qu'elle est non préemptive.

2.2.2 Processeurs

C'est l'élément qui représente l'outil de réalisation de la tâche. Il peut correspondre à un homme pour une tâche manuelle, à une machine-outil pour une tâche de fraisage, de tournage ou autre, à une unité de calcul pour une tâche informatique, de calcul par exemple. Le mot processeur est donc utilisé au sens large et regroupe tous les outils de réalisation (transformation, manutention, transport, etc...)

Ces processeurs peuvent être à aptitudes multiples, c'est-à-dire capables de réaliser plusieurs tâches différentes, ou spécialisés dans une seule opération.

2.2.3 Ressources

Des ressources autres que l'outil de réalisation (processeur), sont souvent nécessaires à l'exécution d'une tâche. Elles peuvent représenter une pièce pour une opération d'assemblage, des fichiers de données pour une tâche informatique, ou de l'argent pour une tâche d'investissement par exemple.

Leur disponibilité en fonction du temps est un paramètre très important dans l'établissement de l'ordonnancement. En effet, elles peuvent être en quantité suffisante pour le travail à réaliser, auquel cas, il n'y aura

pas de contrainte sur ces ressources ou en quantité insuffisante et selon une disponibilité en fonction du temps, déterministe ou aléatoire.

2.2.4 Contraintes

Les principaux types de contraintes sont donnés dans [ROY. 64] [CAR. 82]. Ce sont les contraintes de type potentiel, de type disjonctif, et enfin de type cumulatif.

a) Contraintes potentielles

Elles sont communes à la presque totalité des problèmes d'ordonnement. En fait, elles sont de deux types. Le premier est relatif à la localisation temporelle de la tâche et le second concerne les relations de succession entre les tâches.

Soit i une tâche qui ne peut être réalisée avant une date S_i . Son temps de début d'exécution t_i doit donc obéir à l'inéquation :

$$t_i \geq S_i \text{ (localisation temporelle)}$$

Cette tâche i , ne peut être réalisée que si les tâches j et k le sont. Si d_j représente la durée d'exécution de la tâche j , on aura :

$$t_i \geq t_j + d_j$$

$$t_i \geq t_k + d_k$$

Ces inéquations traduisent les contraintes de succession.

b) Contraintes disjonctives

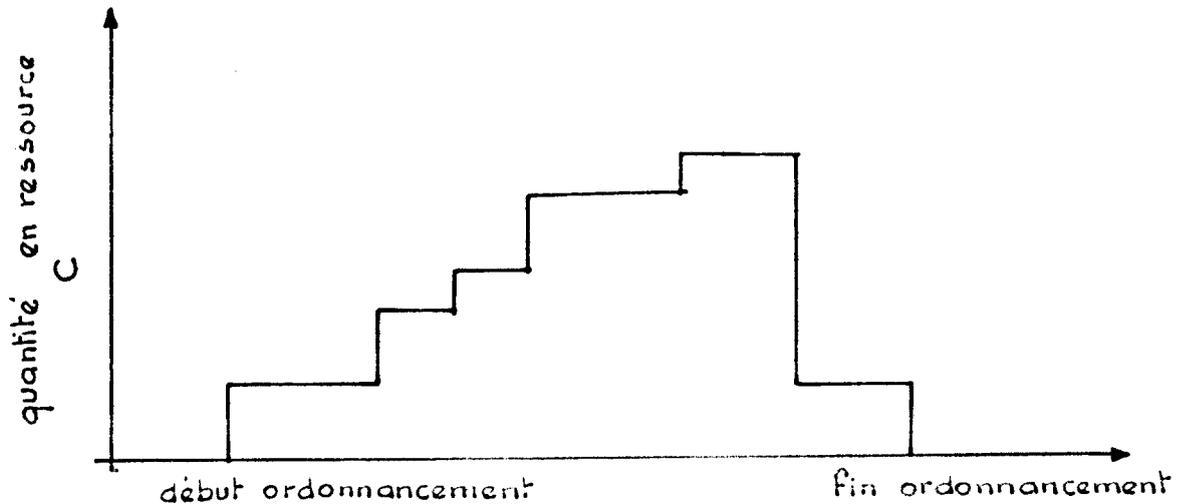
Ces contraintes traduisent l'interdiction à des tâches de s'exécuter en même temps sur un même processeur par exemple, lorsque celui-ci ne peut en réaliser qu'une seule à la fois.

On imagine que si, entre toutes les tâches, l'ordre de succession est fixé, les contraintes disjonctives se trouveront automatiquement satisfaites. Mais généralement dans des problèmes complexes comme ceux que l'on rencontre dans la réalité, il est souvent difficile de prévoir l'ordre de succession de l'ensemble des tâches à lancer, parce qu'elles dépendent de paramètres exogènes par exemple. Il est alors impossible d'éliminer ces contraintes disjonctives.

c) Contraintes cumulatives

Les besoins d'une tâche en ressource d'une certaine catégorie C, dépendent parfois du temps. Si on fait le cumul des besoins pour cette ressource, sur l'ensemble des tâches, on obtient ce qu'on appelle une courbe C_c relative à la ressource C, qui traduit en fonction du temps, le besoin global en ressource C, dans l'ordonnancement.

Exemple :



Les contraintes cumulatives sont celles qui pèsent sur la courbe de charge C_c .

Cela signifie, que l'on ne peut pas à tout moment satisfaire l'ensemble des tâches à lancer par l'octroi de cette ressource C, qui existe en quantité limitée.

L'idéal serait d'avoir des courbes de charge qui traduisent un besoin constant en main d'œuvre par exemple, mais cela est difficile à tenir dans la pratique.

2.2.5 Fonction objectif

La fonction objectif (F), ou économique dépend du problème posé.

Les variables intervenant dans cette fonction peuvent être associées aux tâches, aux processeurs ou aux ressources.

a) Tâches

La fonction F peut être :

- $\min (C_{\max})$, si C_{\max} est la date d'achèvement de la dernière tâche de l'ordonnancement
- $\max \sum_i p_i(1-U_i)$, avec $U_i = 1$ lorsque la tâche i est en retard, et $U_i = 0$ sinon ; p_i est le profit réalisé lorsque la tâche i est réalisée sans retard.

L'objectif est donc de maximiser la somme des profits sur l'ensemble des tâches.

Dans certains cas, il existe des pénalités sur les retards dans l'exécution des tâches.

b) Processeurs

L'objectif dans ce cas peut être la minimisation du temps d'inactivité des processeurs, ou l'équilibre entre leurs temps de travail.

c) Ressources

Plusieurs cas peuvent être considérés :

- minimisation des stocks en cours ;
- optimisation du placement des ressources en stock, ...

Si des variables intervenant dans F, sont aléatoires, la fonction économique représente généralement l'espérance mathématique du coût engendré par cette (ou ces) variable (s).

Après la définition du problème général de la conduite d'un système de production, c'est-à-dire l'ordonnancement-affectation, nous présentons dans le paragraphe suivant, les divers modèles de conduite ou méthodes utilisés dans la littérature pour le résoudre.

3 - MODELES DE CONDUITE DES SYSTEMES DE PRODUCTION

Le problème d'ordonnancement consiste généralement en la réalisation d'un certain nombre n de pièces par les m machines de l'atelier. Chacune de ces pièces nécessite au plus m opérations dans cet atelier.

Il s'agit alors de trouver le meilleur ordonnancement relativement aux critères choisis, qui se réduisent le plus souvent aux délais de réalisation.

De nombreuses approches ont été étudiées dans la littérature pour résoudre ces problèmes :

Approche déterministe :

- méthode combinatoire analytique ;
- programmation linéaire à nombres mixtes ;
- heuristiques ;
- simulation, ...

Approche probabiliste :

- par la théorie des files d'attente ;
- par les chaînes de Markov, ...

Ces approches se basent sur certaines hypothèses, comme :

- l'ordre de passage des pièces devant les machines ;
- la fréquence d'arrivée des pièces ;
- la possibilité de stockage intermédiaire (fini ou infini) ;
- les temps opératoires, connus ou aléatoires ;
- la définition du modèle : continu ou discret ;
- la prise en compte ou non de l'état courant de l'atelier dans l'ordonnancement, etc ...

C'est par rapport à cette dernière hypothèse que l'on peut distinguer les deux grandes classes des problèmes d'ordonnancement (§2.1) : [DEG. 85] [SOU. 81].

Les problèmes statiques, où les tâches à ordonnancer sont connues.

Aucun autre ensemble de tâches ne sera pris en compte tant que les premières ne sont pas réalisées.

Les problèmes dynamiques, où l'ordonnancement est remis en cause chaque fois qu'un événement imprévu apparaît (arrivée aléatoire des tâches par exemple).

La plupart des techniques citées sont utilisées dans le premier problème, notamment les méthodes analytiques qui donnent des résultats optimaux dans certaines configurations d'ateliers. Leur adaptation aux problèmes dynamiques n'est pas facile, à cause du nombre important de

variables mises en jeu. C'est pourquoi, l'on se tourne le plus souvent vers la recherche d'heuristiques plus ou moins performantes, mais qui ont l'avantage d'être rapides. Elles satisfont ainsi l'exigence temps réel des ordonnancements dynamiques.

Parmi les méthodes citées, nous n'exposons que celles qui sont les plus utilisées, dans un contexte déterministe, en mettant en relief leurs conditions et limites d'utilisation, surtout vis-à-vis du problème dynamique.

3.1 ORDONNANCEMENT STATIQUE

Le principe de toutes les méthodes appliquées dans ce cas est de déterminer avec précision le séquençement des différentes opérations à réaliser, avant le début d'exécution des tâches.

Lorsque seules les contraintes potentielles sont considérées, l'ordonnement des tâches, de manière à minimiser le délai total d'exécution est facilement solutionné par les méthodes CPM/PERT ou MPM [ROY. 64] [CAR. 82]. Celles-ci sont basées sur la recherche d'un chemin de valeur maximale dans un graphe valué.

Seulement, dans la plupart des problèmes réels, on tient compte d'autres contraintes liées aux ressources (machine, main d'œuvre, matière première, ...) dont la quantité ou la disponibilité est limitée. La complexité des problèmes s'accroît considérablement, et l'on doit faire appel à d'autres techniques pour tenter de leur trouver des solutions.

3.1.1 Méthodes de programmation linéaire à variables mixtes

C'est l'une des formulations les plus couramment utilisées. L'ordonnement est décrit à l'aide d'un programme linéaire utilisant deux types de variables :

- les variables, temps de début d'exécution (d_i), et temps d'exécution de l'opération (t_i). Elles expriment les contraintes de succession.

Exemple :

Si, i a pour successeur j , alors :

$$t_j \geq t_i + d_i \quad \text{ye diais} \quad d_j \geq t_i + d_i$$

- les variables booléennes, qui expriment les contraintes disjonctives sur les moyens (une machine ne peut réaliser qu'une seule opération à la fois).

La fonction objectif peut correspondre à un ou plusieurs parmi les critères que nous avons déjà cités (§ 2.4).

La prise en compte de variables booléennes augmente considérablement la dimension du problème.

C'est donc une méthode qui permet d'obtenir la solution optimale, mais son coût en temps de calcul devient prohibitif si la taille de l'atelier est relativement grande.

Exemple :

Pour réaliser 20 tâches, sur 7 machines dans un atelier "Flow-shop", on a :

- 140 inconnues (s_j) ($m.n$)
- 1330 variables booléennes devant vérifier
2800 contraintes ($m.n^2$) [DEG. 85]

Il semble assez maladroit de résoudre les problèmes d'ordonnement sous contraintes sur les moyens, à l'aide de cette méthode [SOU. 81]

3.1.2 Méthodes par Séparation et Evaluation Progressive ou Séquentielle (SEP ou SES)

Les méthodes qui se fondent sur cette approche sont arborescentes. Elles consistent à parcourir l'arbre des solutions possibles et à en choisir la meilleur compte tenu des critères retenus [ROY. 69].

La SEP (ou S.E.S.) est une méthode d'énumération implicite plus connue sous le nom de "Branch and Bound". Elle repose sur les principes suivants :

a) Principe de séparation, qui consiste à partitionner l'ensemble des solutions en deux sous-ensembles complémentaires.

b) Principe d'évaluation, qui à partir d'une fonction définie sur tout sous-ensemble résultant d'une séparation, permet d'apprécier par excès ou par défaut la valeur des solutions (s'il y en a), des sous-ensembles considérés.

L'évaluation des solutions peut être faite par rapport à divers critères (temps de passage des pièces, durée de l'ordonnancement, ...).

L'efficacité de la méthode dépend surtout de celle de la fonction d'évaluation, c'est-à-dire du calcul du minorant (ou du majorant), du critère retenu.

C'est une méthode qui peut aboutir à des résultats optimaux, mais les coûts en temps de calcul restent assez importants. Mais si l'on se contente de résultats sous-optimaux, la méthode donne rapidement des solutions. elle reste quand même limitée aux problèmes de faible taille.

3.1.3 Méthodes d'analyse combinatoire

Le principe de base de ces méthodes est d'analyser l'ensemble des solutions possibles, pour en dégager des sous-ensembles remarquables et de réduire ainsi la dimension du problème à résoudre.

La propriété qui permet de déterminer ces sous-ensembles est la dominance [ERS. 82] [THO. 80] [FON. 80] [COU. 79] [ERS. 85]. Elle repose sur deux conditions :

- la dominance par rapport à l'admissibilité (contraintes temporelles, et sur les moyens) ;
- la dominance par rapport aux critères.

Les conditions nécessaires et suffisantes d'admissibilité sont établies dans [ERS. 82]. Dans [COU. 9] et [FON. 80] sont démontrées les conditions nécessaires de dominance à travers l'étude des structures pyramidales des diagrammes de GANTT pour le premier, et la recherche de circuit dans le graphe pour le second.

Ces méthodes sont intéressantes dans un cadre d'aide à la décision puisqu'elles peuvent proposer plus d'une solution satisfaisante et permettent donc au décideur d'orienter son choix.

Ainsi, leur application à des ordonnancements temps réel est possible. Une telle approche est développée dans [THO. 80]. Néanmoins, les résultats obtenus par ces méthodes concernent surtout des problèmes de faible dimension.

3.1.4 Méthodes heuristiques

L'inconvénient majeur des méthodes présentées est leur incapacité à résoudre des problèmes de dimensions réalistes à cause d'une part, des besoins en place en mémoire qui risquent d'être importants, et d'autre part, du temps mis pour obtenir la solution optimale.

Les méthodes heuristiques pallient à cet inconvénient mais malheureusement sans garantir un résultat optimal.

• "Une heuristique est une règle de choix permettant d'agir en l'absence d'un résultat théorique sûr" [SIM. 84].

Il est d'ailleurs souvent difficile de juger de leur efficacité. Néanmoins, elles permettent d'obtenir des solutions très rapidement, à des problèmes complexes. La plupart des algorithmes heuristiques sont à une passe [SOU. 82].

Ils consistent généralement en le classement des opérations "ordonnançables", c'est-à-dire celles dont les antécédents sont déjà ordonnancés, selon une priorité donnée. Celle-ci peut être une règle PAPS (premier arrivé, premier servi), ou selon les temps d'exécution croissants, ou leurs dates au plus tard croissantes, etc...

La recherche de solutions sous-optimales, n'obéit donc pas à un besoin abstrait de traiter des problèmes complexes mais à un besoin réel de trouver des solutions aux problèmes qui se posent, dans l'industrie par exemple, et qui doivent être résolus. C'est tout l'intérêt de ces méthodes.

3.1.5 Simulation

Les limites des modèles analytiques orientent les concepteurs vers l'utilisation de la simulation.

Celle-ci ne résout pas les problèmes posés dans le système, mais permet la description de son évolution dans le temps, à chaque occurrence d'un événement. Cette description est aussi fine que cela est nécessaire pour résoudre le problème posé.

Dans les modèles à événements discrets pour la simulation, l'approche la plus connue et la plus utilisée, est celle par événements. D'autres approches sont décrites dans [BEL. 85]. C'est la seule que nous présentons.

Avant d'aboutir au programme de simulation, il faut spécifier deux choses. La première consiste à répertorier tous les événements pouvant apparaître dans le système, ce qui nécessite une analyse détaillée. La seconde concerne la modélisation de la logique de changements d'états, relative à ces événements.

Ceci étant fait, le programme n'aura qu'à exécuter la décision associée à l'événement, à la date où celui-ci apparaîtra.

L'évolution dans le temps s'effectue de deux manières :

- soit à l'aide d'une horloge, où à chaque unité de temps, on teste s'il y a occurrence d'un événement pour le traiter ;
- soit par l'apparition de l'événement lui-même en passant directement de la date de l'événement courant à celle du plus proche dans le temps. Les dates d'événements sont donc ordonnées dans une liste. Cet ordre n'est pas figé, il est remis en cause chaque fois que le traitement d'un événement l'exige.

De nombreux langages spécifiques utilisent cette approche : GPSS-FORTRAN, SIMSCRIPT, GASP IV, ... [BEL. 85] [LEV. 82].

Il semble toutefois que pour des modifications de structures des systèmes de production, la plupart de ces langages ont leurs limites. Dans le but d'éviter ces contraintes, l'institut de robotique de Carnegie-Mellon a élaboré un outil d'aide à la simulation KBS (Knowledge Based Simulation), qui est une extension du langage SRL [ARA. 84].

Plusieurs exemples concrets d'utilisation de modèles de simulation de ce type, méritent d'être présentés.

a) Laboratoire d'Automatique de Grenoble [DAL. 82]

Un programme de simulation à événements discrets a été développé à partir d'une modélisation du système de production à l'aide du GRAFCET.

Cette simulation a été appliquée à l'atelier flexible de CITROEN à Meudon. Il a permis l'étude des stratégies de lancement et de gestion des pièces dans l'atelier.

b) D.E.R.A. du C.E.R.T. [ARA. 84]

Le D.E.R.A. a utilisé la simulation pour étudier :

- une cellule flexible d'usinage d'aubes de compresseur avec la SNECMA ;
- un atelier de fabrication de vitrages automobiles avec Saint Gobain.

c) S.O.D.E.T.E.G.-T.A.I.

Un système de simulation à événements discrets a été appliqué à l'atelier de fabrication de pales d'hélicoptères de l'usine de l'Aérospatiale qu'il a permis de reconfigurer pour une productivité double.

Beaucoup d'autres exemples peuvent encore être cités. Ceci montre l'attrait et donc l'intérêt de la simulation dans l'étude des systèmes de production en général.

Remarques générales :

Par rapport à l'optimalité de la solution, nous classons les méthodes en deux groupes :

- méthodes optimales (P.L.M., S.E.P., S.E.S.) ;
- méthodes sous-optimales (M.A.C., heuristiques).

Nous avons relevé les conditions et les limites d'utilisation de chacune d'elles. Nous constatons que leur application, exception faite pour les heuristiques, est difficile à envisager pour des problèmes réels. En effet, dans ces derniers, il est nécessaire de prendre en compte le contexte perturbé d'un atelier. L'ordonnancement recherché devient dans ce cas, dynamique. C'est un problème plus complexe que dans le cas statique et qui exige des techniques spécifiques pour sa résolution.

3.2 ORDONNANCEMENT DYNAMIQUE

L'ordonnancement dynamique est un problème qui se pose dans les systèmes de production flexibles.

La particularité de ces derniers est leur possibilité de fabrication simultanée de différents produits, et de réagir en ligne aux éventuelles perturbations qui apparaîtraient dans l'atelier.

Les modèles qui seront proposés pour conduire de tels systèmes doivent absolument tenir compte de cette contrainte et utiliser des algorithmes d'ordonnancement temps réels.

Certains auteurs ont essayé d'étendre des résultats trouvés dans le cas statique, pour des lignes transfert "classiques", aux lignes transfert flexibles. Ceci a eu pour conséquence l'application de règles simples d'ordonnancement, telle P.A.P.S. par exemple.

Plusieurs approches ont été étudiées pour développer des modèles et méthodes de conduite de ces systèmes :

- la simulation est très utilisée [DAL. 82] [ARA. 84] ;
- les heuristiques ;
- les méthodes analytiques dans lesquelles nous regroupons les méthodes déterministes combinatoires [THO. 80] [SOE. 77], les

méthodes fondées sur la théorie des files d'attente, etc... [AMM. 85] [LEV... 82] [DUP. 83] [KUS. 85].

Nous venons de relever (§ 3.1.), la difficulté qu'ont ces techniques à résoudre les problèmes statiques. Leur adaptation au problème dynamique donc plus complexe semble irréalisable à cause du nombre de variables à considérer, et du coût très élevé en temps de calcul.

Le problème que l'on étudie dans ce travail concerne la conduite d'une cellule de production flexible, qui nécessite obligatoirement un ordonnancement ou une affectation dynamique.

Dans l'approche que nous avons retenue, nous définissons d'une part la structure décisionnelle, et d'autre part, la position du problème avec toutes les hypothèses de travail émises.

4 - APPROCHE RETENUE

4.1 STRUCTURE DECISIONNELLE

Le problème posé est l'étude de la conduite (ordonnancement-affectation) de la cellule de production flexible que nous avons présentée au chapitre 1.

Cette cellule représente un sous-système parmi plusieurs autres dans l'atelier flexible.

La structure décisionnelle de celui-ci peut être décomposée en une structure hiérarchisée en plusieurs niveaux. Chacun d'eux est associé à un horizon de prise de décision, au delà duquel, cette dernière est remise en cause [KAL. 85].

Pour notre part, nous nous situons au niveau bas de cette hiérarchie, puisque l'on se limite à l'étude de la conduite d'une seule cellule.

L'état courant de la cellule doit obligatoirement être pris en compte instantanément dans la décision.

Dans les logiciels de gestion de production, la conduite d'atelier est généralement décomposée en deux fonctions : ordonnancement et lancement [KAL. 85].

Nous considérons que cette décomposition reste aussi valable dans le cas d'une cellule de production. La conduite de celle-ci peut être alors

représentée par une structure décisionnelle à deux niveaux. Le premier relatif à l'ordonnancement, et le second au lancement. Ce dernier est décomposé horizontalement en plusieurs centres de décision, relatifs aux machines intervenant dans la cellule.

Deux approches principales peuvent être considérées dans ce cas :

- 1 - Approche centralisée, où le niveau ordonnancement prend en charge tous les problèmes de la conduite, et dans ce cas, les centres de lancement (machines) ne seront que des exécutants.
- 2 - Approche décentralisée, où le niveau ordonnancement spécifie l'ensemble des opérations à réaliser pour chaque centre de lancement, et c'est ce dernier qui aura à déterminer la séquence de réalisation des opérations en interaction, forcément, avec les autres centres de lancement. C'est une approche très complexe à mettre en œuvre.

L'inconvénient principal de l'approche centralisée, pourrait être le temps de reconfiguration de l'ordonnancement en cas de perturbations imprévisibles. Il serait incompatible avec l'exigence d'une décision temps réel. Cela serait valable dans le cas d'un système complexe de production, mais au niveau d'une cellule de production composée d'un faible nombre de moyens de fabrication, nous pensons que le problème est différent et qu'une structure décisionnelle centralisée peut être adoptée.

C'est l'approche que nous avons retenue dans ce travail.

4.2 POSITION DU PROBLEME

Le problème général de la conduite d'un système de production (ordonnancement-affectation) est un problème d'optimisation (§ 2), où il s'agit d'optimiser une fonction objectif, sous un ensemble de contraintes.

Nous nous sommes limités dans le cadre de ce travail au seul objectif :

"minimisation des délais d'exécution des produits à réaliser".

Les hypothèses que nous avons retenues sont les suivantes :

- 1 - Il existe un seul plan de travail dont une partie est en zone commune, et deux autres parties propres à chacun des robots.
- 2 - Le magasin est alimenté par des systèmes divers. Les aléas provoquent des perturbations dans la distribution des

ressources. Nous considérons alors, leur arrivée dans le magasin comme aléatoire.

Ces ressources propres, sont à la disposition de plusieurs cellules, qui peuvent les "consommer" à tout instant. Ainsi, pour une cellule donnée, ces ressources apparaissent de disponibilité imprévisible.

Ces deux hypothèses ont été prises dans plusieurs études sur les systèmes de production flexibles.

L'entreprise SODETEG. TAI a étudié pour la SNIAS, l'automatisation d'un atelier de traitement de surface des pièces d'avion. Cette étude devait prendre en compte l'arrivée aléatoire des pièces dans l'atelier [E. Ind. 83 Fev.].

Dans l'étude d'une cellule d'assemblage flexible, la prise en compte de l'arrivée aléatoire des pièces, a été envisagée dans [ALA. 83], [LOZ. 83].

3 - La capacité du magasin est finie.

4 - Il existe un stock tampon, de capacité limitée, situé en zone commune et donc accessible aux deux robots.

Ce stock se justifie par le fait de la disponibilité aléatoire des ressources dans le magasin. Ainsi, une cellule peut "stocker" des pièces "rares" afin d'éviter d'être bloquée dans la réalisation du travail demandé

5 - Les robots sont à aptitudes multiples, dont quelques unes leur sont communes. Ceci, afin de permettre à un robot, de prendre en charge des opérations destinées à l'autre, au cas où celui-ci serait dans l'incapacité de les réaliser (panne ou une quelconque indisponibilité).

6 - La zone commune n'est accessible qu'à un seul robot à la fois.

C'est une contrainte qui peut sembler forte, seulement le problème de l'évolution de plusieurs manipulateurs dans un espace commun est très complexe, et il est rare de trouver des études ayant abouti à une réalisation pratique. Un des seuls exemples que l'on peut citer est celui de [PARK. 84]. Il étudie la coordination des mouvements de deux robots évoluant dans un espace commun.

L'approche utilisée est basée sur la représentation de l'espace d'état des deux manipulateurs.

A partir de cet espace d'état, la méthode consiste à générer une carte des régions de collision (collision-map), en fonction des variables d'état des robots. Cette carte est ensuite compressée et enfin mémorisée. Il suffit alors, lors de la commande des robots de comparer les valeurs des variables

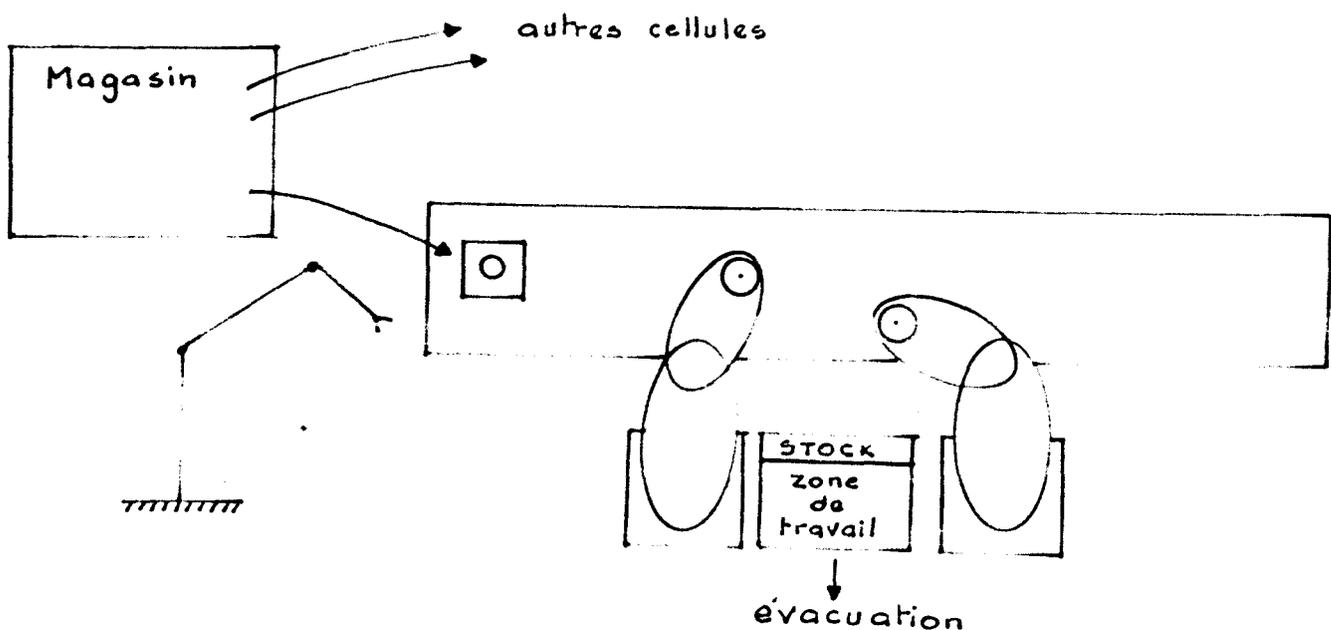
d'état courantes avec celles, mémorisées, et de générer la suite des valeurs des variables d'état associées à une trajectoire sans collision. La génération de la trajectoire se fait en ligne.

La détermination de la carte de collision, en supposant quatre variables d'état pour , a nécessité huit heures de temps de calcul sur un VAX 750.

Elle reste une approche difficile à envisager.

- 7 - Le temps de transport de la ressource propre du magasin vers les robots n'est pas pris en compte.

La configuration physique de la cellule est la suivante :



Le problème à résoudre consiste alors, en fonction de l'état courant de la cellule (état des robots, état du magasin, état du stock) et de la fonction objectif retenue, à choisir la tâche ou la séquence de tâches à lancer, parmi celles qui sont exécutables.

C'est un problème d'ordonnancement-affectation dynamique dont la présentation détaillée fait l'objet du prochain chapitre.

5 - CONCLUSION

Après la définition du problème d'ordonnancement, nous avons présenté différentes approches ou méthodes utilisées pour sa résolution.

Nous avons constaté la disproportion qui existe dans la littérature, entre le nombre de travaux concernant les ordonnancements statique et dynamique. Si pour le premier, diverses méthodes existent, il n'en est pas de même pour le second, où seuls des travaux ponctuels lui sont consacrés. C'est un problème complexe qui exige des techniques adaptées.

Après avoir spécifié le problème que nous étudions, nous présentons dans le chapitre suivant l'approche que nous avons retenue pour décrire le comportement de la cellule de production flexible, et la structure informatique du système de conduite qui en a découlé.

La méthode de résolution proposée pour résoudre ce problème et qui correspond au module de décision du système de conduite, sera détaillée dans le chapitre IV.

Chapitre III

ETUDE DE LA CELLULE DE FABRICATION FLEXIBLE

1 - INTRODUCTION

Le degré d'automatisation d'un système de production flexible comme la cellule que nous étudions, rend sa conduite difficile à appréhender.

Le besoin d'outils performants pour décrire le comportement de teils systèmes est évident. C'est pourquoi, depuis des années, on trouve dans la littérature des propositions de modèles les concernant. Ils sont basés pour la plupart, soit sur des approches analytiques, soit sur des représentations graphiques (Réseau de Pétri, Grafset) [OH. 82], soit enfin sur des modèles de simulation [BEL. 85] [HO. 83].

Les systèmes de production automatisée sont généralement décrits par des modèles à événements discrets, qui signifient qu'à chaque occurrence d'un événement dans le système, les changements d'états conséquents sont effectués.

Dans le cadre de ce travail, nous utilisons une modélisation par des graphes d'états et de transitions, qui décrivent globalement l'état de l'ensemble des éléments intervenant dans la cellule.

La dynamique du modèle, c'est-à-dire les changements d'états, est dictée, soit par des contraintes technologiques (libération de l'espace de travail, relations de précedence dans la gamme de fabrication, ...), soit par des règles de conduite du système (ou algorithmes de décision).

Le but de ce chapitre est, d'une part, de présenter ce modèle et d'autre part, de proposer une structure du système de conduite de la cellule de production flexible.

2 - MODELISATION [DJE. 85] [STA. 85] [STA. 85 bis] [FIN. 85]

La cellule de production flexible est décrite aux moyens de graphes d'état associés au produit à réaliser, à l'environnement et enfin aux robots.

Elle est modélisée par le quadruplet :

$\langle X_t, E_t, R_t, U_t \rangle$

où,

X_t représente l'état du produit en cours

E_t représente l'état de l'environnement

R_t représente l'état des robots

U_t représente l'ensemble des commandes admissibles.

2.1 ETAT DU PRODUIT EN COURS DE REALISATION

2.1.1 Décomposition de la réalisation du produit

L'objectif à satisfaire étant la réalisation du produit, il est nécessaire que celle-ci soit bien définie pour aboutir au résultat souhaité.

Cette définition passe par la décomposition de la réalisation en un ensemble d'opérations directement exécutables par un robot ou plusieurs synchronisés.

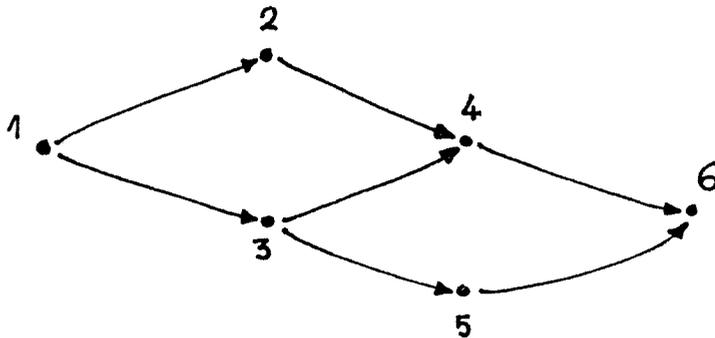
Ces opérations ne sont liées que par des relations de succession.

Elles forment alors ce que l'on appelle la gamme de fabrication. Celle-ci, qui traduit donc l'enchaînement de toutes les opérations à réaliser est décrite à l'aide d'un graphe orienté acyclique $G(N,A)$, où N est l'ensemble des nœuds et A celui des arcs. [BERG. 83]

Ce graphe fait apparaître :

- toutes les opérations à réaliser, correspondant aux nœuds de N ;
- les relations de succession entre ces opérations représentées par les arcs de A .

Exemple :



* Une gamme de fabrication composée de 6 opérations.

L'obtention de la gamme de fabrication peut être manuelle au sein d'un bureau des méthodes, ou bien automatique à l'aide d'un générateur de gammes par exemple.

Ce dernier qui représente une approche récente, se base surtout sur les méthodes de l'intelligence artificielle, notamment celles de la génération de plans [DES 81].

Dans le cadre de notre étude, nous supposons ce problème de la décomposition, résolu.

a) Definition d'une opération

Une opération de la gamme de fabrication est un ensemble d'actions élémentaires synchronisées, de robots.

b) Caracteristiques d'une opération

- Une opération est non préemptive
- Une opération nécessite pour sa réalisation :
 - . des ressources propres (pièces, matières premières) ; non renouvelables
 - . des ressources partageables (robots, espace de travail, ...). renouvelables
- Toutes les ressources nécessaires doivent être disponibles au début de l'exécution de l'opération. Ceci, afin d'éviter tout

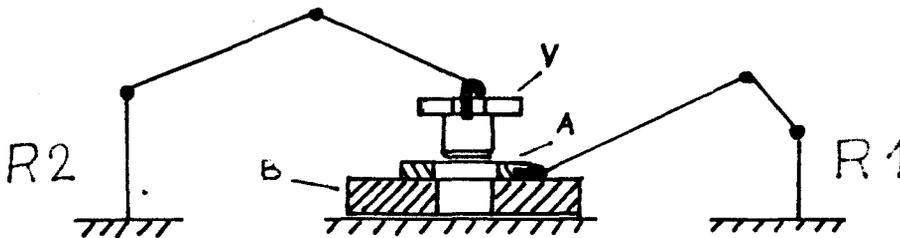
blocage dû à une indisponibilité prolongée de l'une d'entre-elles.

- Dès qu'une ressource partageable n'est plus utile à la poursuite de l'exécution d'une opération, elle est libérée.
- L'exécution d'une opération de la gamme exige une synchronisation entre les actions élémentaires du ou des robots qui la réalisent.

La synchronisation de ces actions peut être représentée à l'aide d'un réseau de Petri, ou d'un Grafset, par exemple.

c) Exemple

Soit une opération d'assemblage d'une pièce A et d'une pièce B, par l'intermédiaire d'une vis V ; B étant positionnée par un système externe.



Cette opération est représentée par un nœud du graphe associé à la gamme de fabrication.

Elle nécessite pour sa réalisation, la disponibilité de :

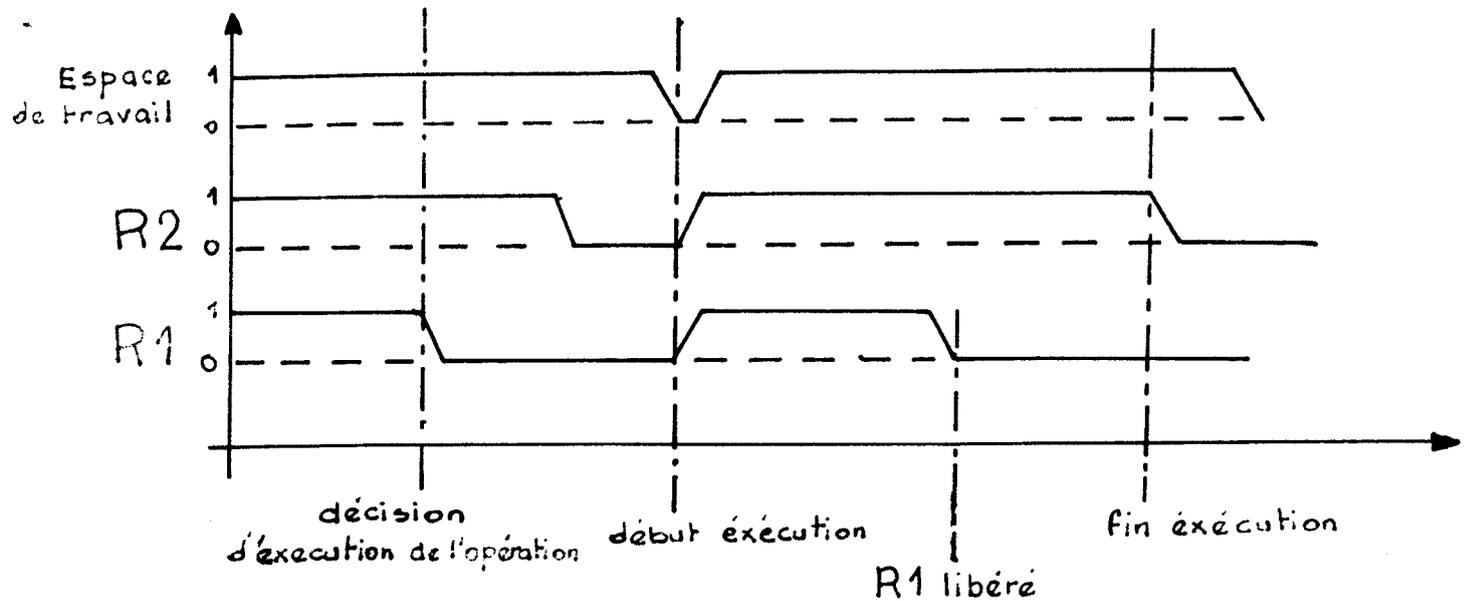
- pièces A et B, et la vis V ;
- deux robots, ayant les aptitudes nécessaires.

Lorsque les ressources propres (A, B et V) sont disponibles, les ressources partageables (robots, espace de travail, ...) ne seront pas forcément libres.

La décision de réaliser l'opération implique donc la prise en compte du temps d'indisponibilité de ces ressources.

On associe ainsi à l'opération deux instants relatifs :

- à la décision d'exécution ;
- au début effectif de l'exécution.

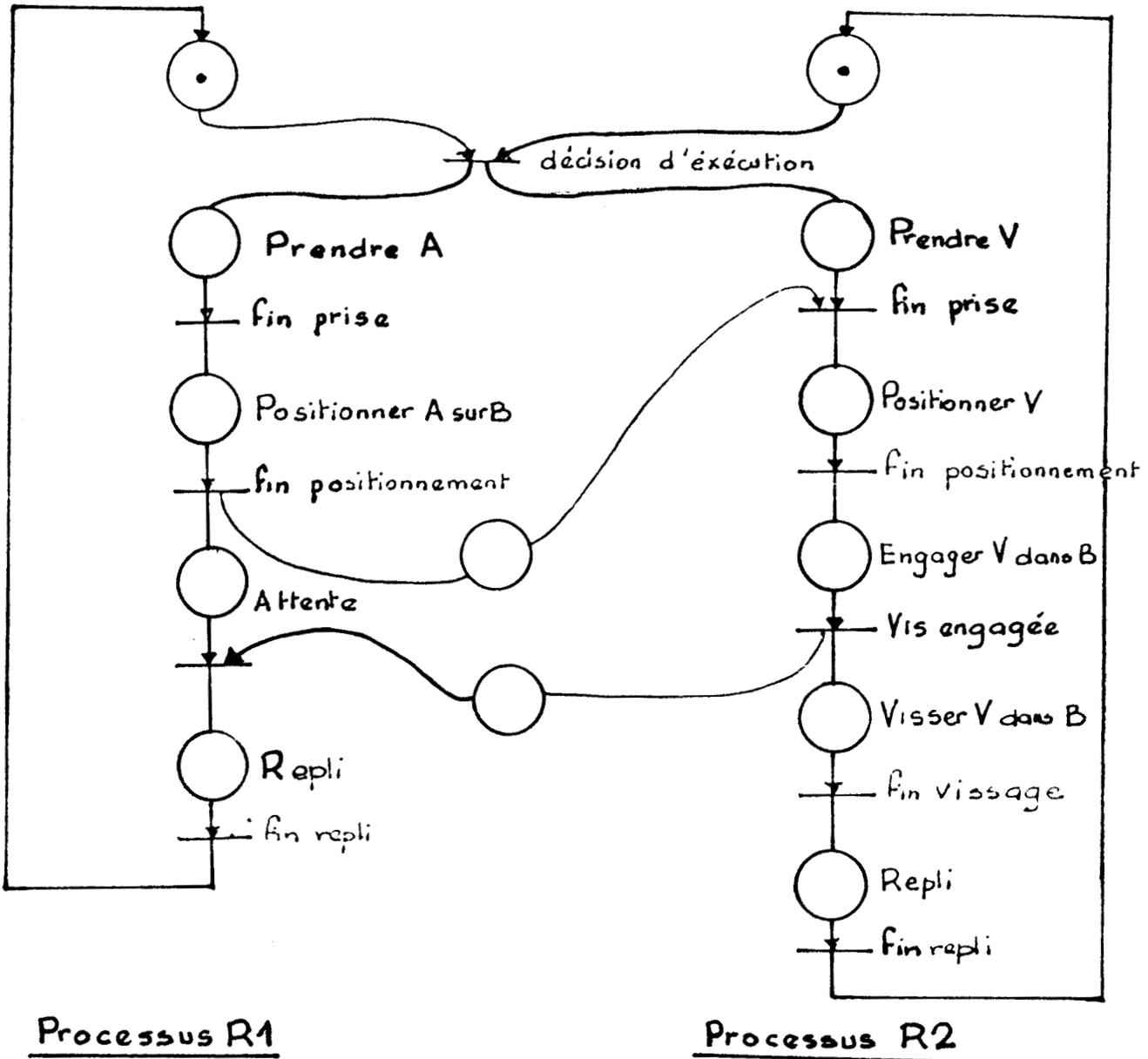


On peut faire les remarques suivantes sur cette approche :

- la décision de "requisitionner" les ressources partageables disponibles peut conduire à des périodes importantes d'inactivité pour les robots et/ou de blocage pour l'espace de travail. Cependant, il faut remarquer que la procédure de décision intègre ce temps d'indisponibilité au niveau des critères pour fournir un résultat globalement optimal.
- dès qu'un robot n'est plus utile pour la poursuite de l'opération, il est libéré (R1 dans l'exemple).

La spécification de la synchronisation est un problème complexe qui nécessite un outil de représentation adéquat. Indépendamment de toute structuration informatique, les réseaux de Pétri sont un outil très utilisé

Notre exemple peut être représenté par les deux processus suivants :



2.1.2 Etat d'une opération

L'état du produit est représenté à chaque instant par l'état des opérations qui le composent.

Une opération de la gamme de fabrication est en fait un processus informatique associé à un ou plusieurs robots. Or, les états d'une tâche informatique sont bien définis. Nous sommes alors tentés de reprendre les mêmes définitions.

La configuration de la cellule flexible est caractérisée par les points suivants :

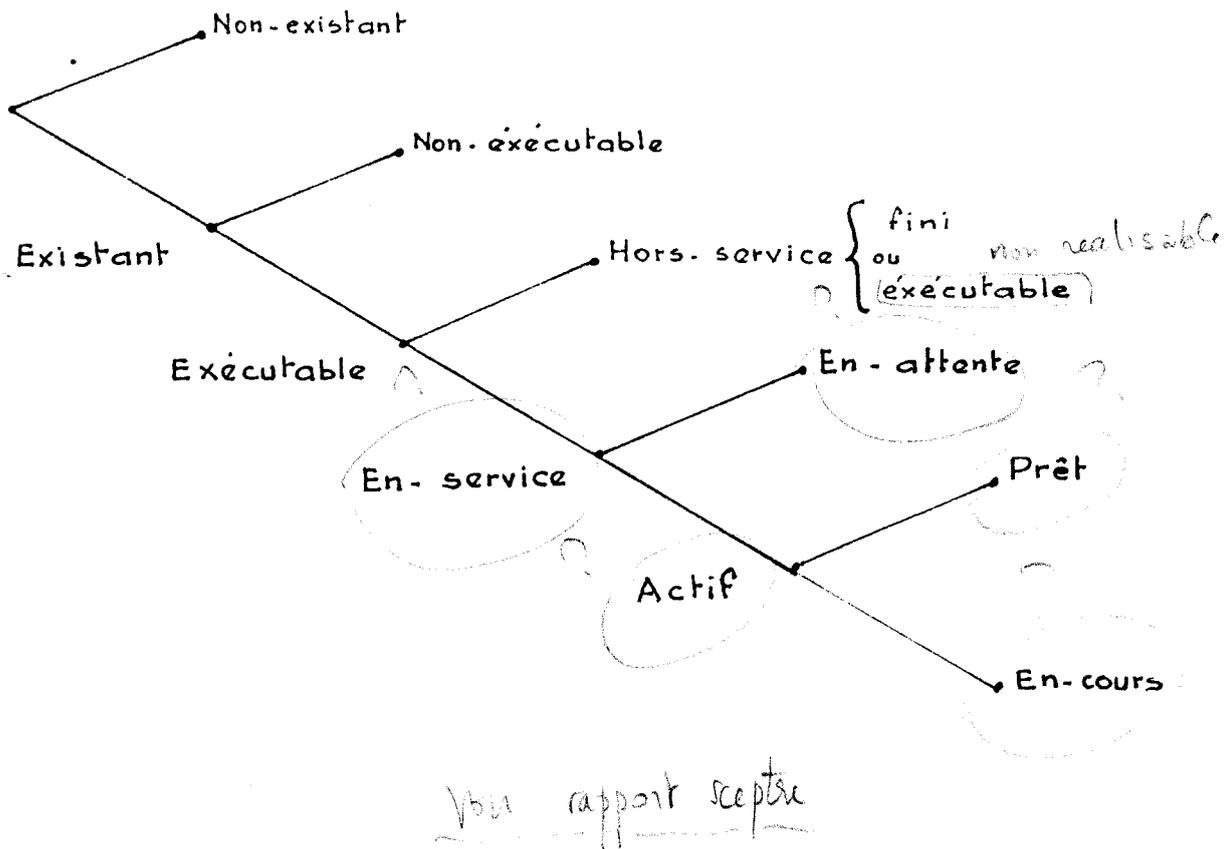
- les opérations ne sont pas préemptives ;
- l'ensemble des opérations à réaliser est parfaitement connu à l'avance, ainsi que les contraintes potentielles qui structurent cet ensemble ;
- les ressources propres nécessaires à l'exécution de ces opérations sont de disponibilité aléatoire. ?

Dans ce sens, nous avons choisi d'utiliser les mêmes vocables que ceux utilisés pour les tâches informatiques, pour les états semblables, et nous définissons les autres en utilisant un vocabulaire propre.

Le vocabulaire utilisé dans la littérature pour désigner les états d'une tâche informatique varie quelque peu d'un auteur à un autre.

Nous avons choisi comme référence, le rapport SCEPTRE, qui est une proposition de standard de noyau pour les exécutifs temps réel [SCEPTRE].

Dans ce dernier, les états d'une tâche sont représentés par l'arbre suivant :



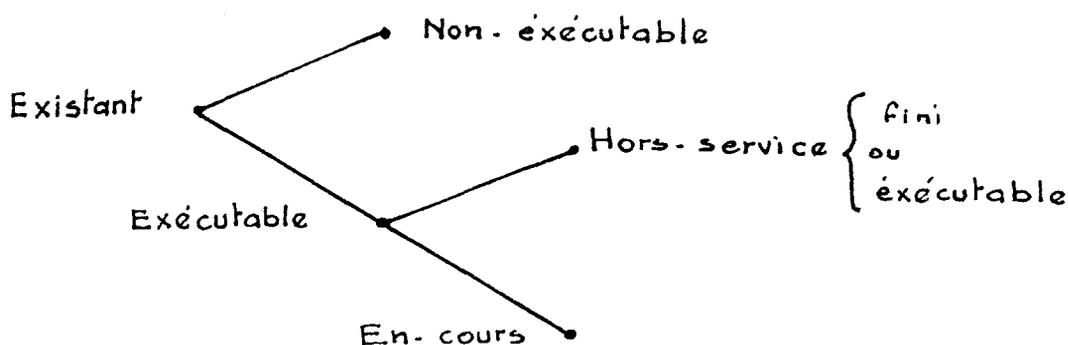
Parmi tous ces états, ceux qui sont relatifs à la préemption d'une tâche (ou d'un processeur), ne peuvent pas concerner une opération de fabrication.

Ces états sont notamment :

- En service \Rightarrow en attente
- Actif \Rightarrow prêt

Nous notons que toutes les opérations sont déclarées en même temps, elles sont donc toutes existantes.

Nous obtenons, ainsi, l'arbre réduit suivant.

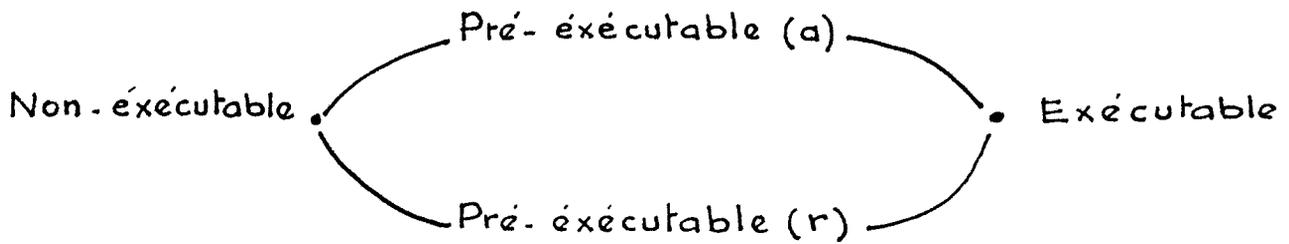


Il représente partiellement les états d'une opération de fabrication. Dans le cadre de nos hypothèses de travail, il est nécessaire de prendre en compte des états supplémentaires qui permettent de mieux cerner l'évolution de la cellule et d'améliorer ainsi la prise de décision.

Nous les définissons ci-dessous.

Soient a et r , deux variables logiques correspondant respectivement à : antécédents réalisés et ressources propres disponibles.

Le passage de l'état non-exécutable à exécutable s'effectue lorsque la fonction logique $a.r$ est satisfaite. Mais dans notre cas, les états intermédiaires correspondant aux fonctions $\bar{a}.r$ et $a.\bar{r}$ sont pris en compte séparément par le module de décision. Ce qui donne le schéma partiel suivant :



Par ailleurs, à partir de l'état en cours, une opération de fabrication peut :

- bien se terminer ;
- mal se terminer, mais ne bloque pas la suite des opérations ;
- mal se terminer, et rend la gamme non-réalisable.

Les deux derniers états ne sont pas représentés dans l'arbre décrit plus haut, mais sont pris en compte dans l'établissement d'une décision.

Nous obtenons ainsi, pour une opération de production, les états suivants :

Non exécutable : Tous les antécédents de l'opération ne sont pas réalisés, et toutes ses ressources propres ne sont pas disponibles.
Il correspond à la fonction logique : $\bar{a} \cdot \bar{r}$

Pré-exécutable : Cet état intermédiaire correspond en fait à deux situations :

- antécédents réalisés
- ou
- ressources propres disponibles.

C'est-à-dire, $\bar{a}r + a\bar{r}$
On les note respectivement par :
Pré-exécutable (a)
Pré-exécutable (r)

Exécutable : Les deux conditions suivantes sont vérifiées :

- Antécédents réalisés ;
- et
- Ressources propres disponibles.

C'est-à-dire $a.r$

En cours : L'opération est exécutable et les conditions suivantes satisfaites :

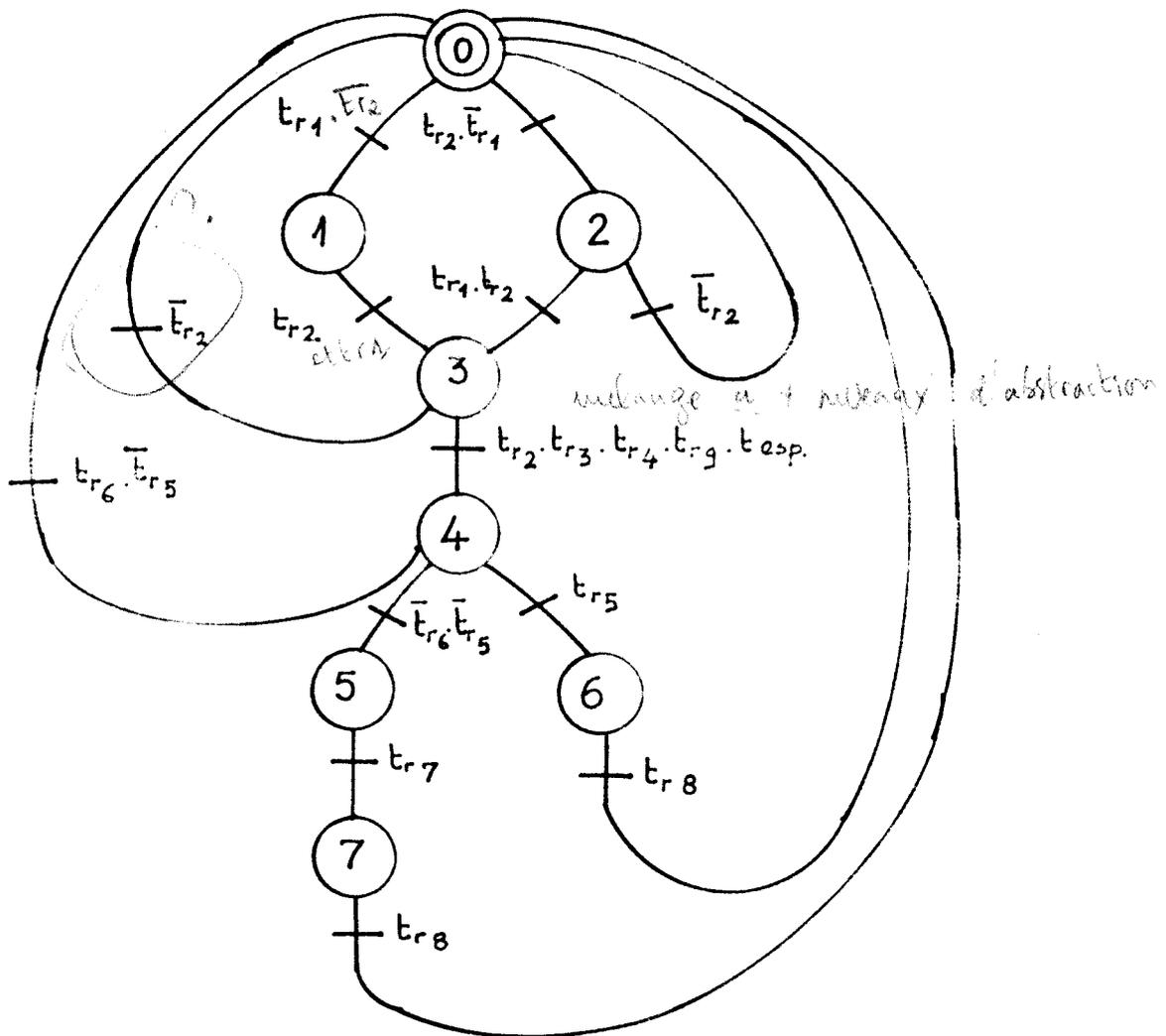
- Toutes les ressources sont disponibles et
- La décision de réaliser l'opération est prise.

Fini : L'opération est correctement réalisée.

Non-réalisable : L'opération ne peut plus être poursuivie.

Attente : L'opération attend une décision de réinitialisation.

L'évolution d'une opération entre ces différents états peut être décrite à l'aide du graphe d'état suivant :



- | | |
|------------------------|--------------------|
| 0 : Non-exécutable | 4 : En cours |
| 1 : Pré-exécutable (a) | 5 : Non réalisable |
| 2 : Pré-exécutable (r) | 6 : Fini |
| 3 : Exécutable | 7 : Attente |

Transitions :

- tr₁ : Les antécédents de l'opération sont finis.
- tr₂ : Les ressources propres de l'opération sont disponibles (pièce ou matière première).
- tr₃ : Un robot capable de réaliser l'opération est libre.
- tr₄ : La décision d'exécuter l'opération est prise.
- tr₅ : L'opération est correctement réalisée.
- tr₆ : Après une mauvaise exécution, l'opération peut être réexécutée.
- tr₇ : Appel d'urgence
- tr₈ : Réinitialisation globale, après la fin de toutes les opérations, ou après qu'une opération devienne non-réalisable.
- tr₉ : La ressource propre est dans la zone de prise du robot.
- tesp : L'espace de travail de l'opération est libre.

A chacune de ces opérations est associé un ensemble de paramètres caractéristiques :

- identificateur de l'opération ;
- enchaînement des actions robots → ?
- ressources nécessaires à sa réalisation :
 - spécification du type d'outil ✓
 - matière première ✓
 - pièce, ... ✓
- paramètres d'exécution :
 - position d'approche de l'outil ✓
 - vitesse de déplacement de l'outil, ... ✓
- temps probable d'exécution, etc... ✓

Ces paramètres doivent identifier parfaitement l'opération pour permettre son exécution.

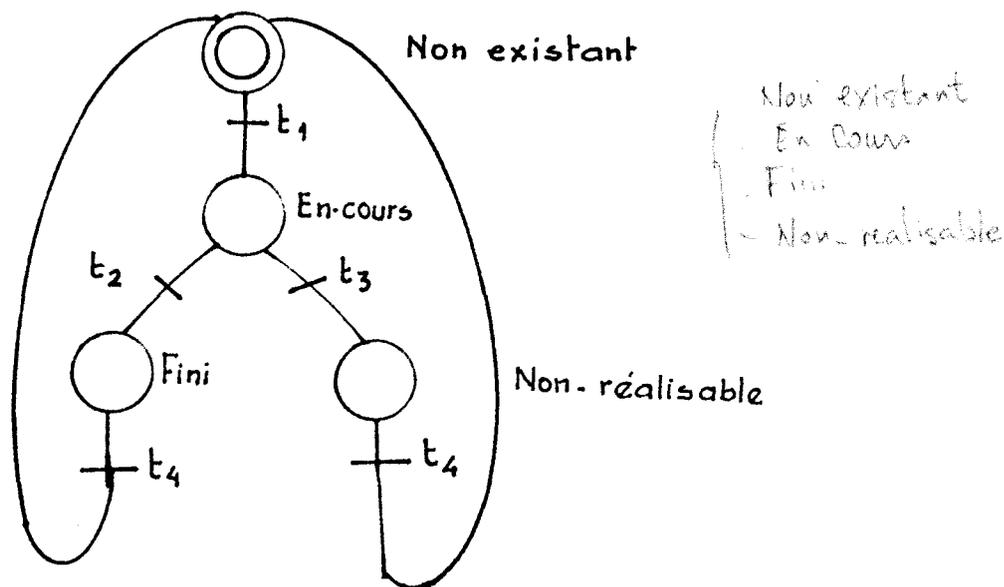
2.1.3 Evolution de l'état du produit

L'évolution de l'état du produit dépend de celle des opérations qui le constituent.

Sa réalisation sera terminée lorsque toutes les opérations seront dans l'état fini.

Si une seule d'entre elles passe dans l'état non-réalisable, le produit le devient automatiquement. Il est fait appel dans ce cas à un système de dégagement du produit (manuel ou automatique).

Les états du produit peuvent être représentés par le graphe suivant :



Avec,

t_1 : décision de commencer la réalisation du produit ;

t_2 : toutes les opérations de la gamme sont finies ;

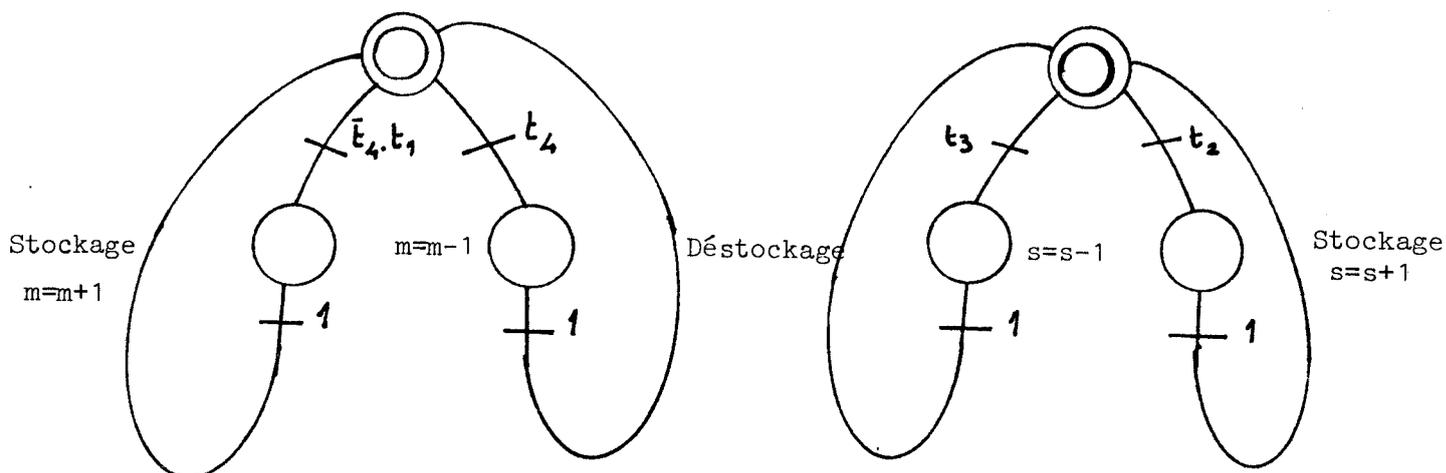
t_3 : une opération de la gamme passe à l'état non-réalisable ;

t_4 : ré-initialisation.

2.2 ETAT DE L'ENVIRONNEMENT

L'environnement correspond aux ressources propres des opérations de la gamme de fabrication qui sont présentes dans la cellule de production, c'est-à-dire dans le magasin ou dans le stock situé en zone commune.

A chacun de ces derniers correspond un graphe d'état décrivant l'évolution des ressources.



avec, m et s , le nombre de ressources présentes respectivement dans le magasin et en stock intermédiaire.

- t_1 : une ressource attend son stockage ;
- t_2 : décision de stocker une ressource en stock intermédiaire ;
- t_3 : décision de consommer une ressource à partir du stock ;
- t_4 : décision de consommer une ressource à partir du magasin
 ($t_4 = \sum_{\text{cellules}} t_2$)

A chacune des ressources est associé un ensemble de paramètres caractéristiques, relatifs à :

- son identification ;
- sa position courante ;
- sa fréquence d'apparition ;

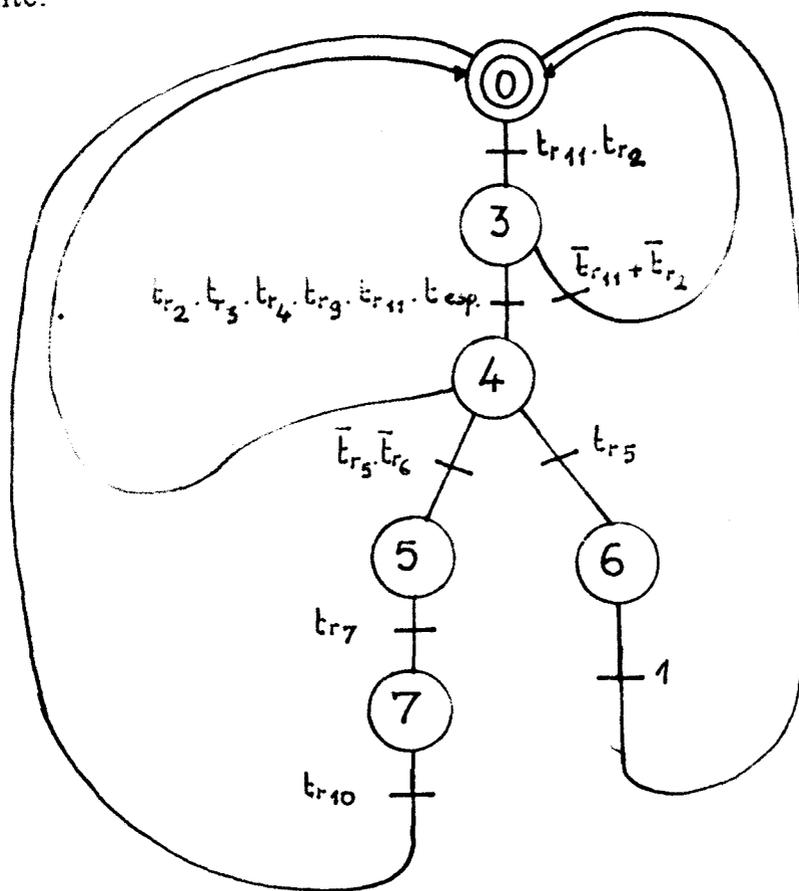
- son risque de disparition (au profit d'une autre cellule, lorsqu'elle est dans le magasin).

L'évolution de l'état de l'environnement dépend d'événements externes comme l'arrivée et le départ des ressources dans le magasin, et aussi des actions robots. Celles-ci sont associées à des opérations de deux types :

- a - Opérations de la gamme de fabrication ;
- b - Opérations de stockage des ressources.

Certains des états de ces opérations ont une influence directe sur l'état de l'environnement.

A chaque robot, nous associons une opération de stockage dont le graphe est donné ci-dessous. Celui d'une opération de fabrication a déjà été présenté.



- | | |
|--------------------|--------------------|
| 0 : Non-exécutable | 5 : Non-réalisable |
| 3 : Exécutable | 6 : Fini |
| 4 : En cours | 7 : Attente |

Les états sont désignés par les mêmes vocables que ceux du graphe d'état d'une opération de fabrication. Les transitions ayant le même

numéro ont aussi le même sens, sauf qu'elles sont relatives à une opération de stockage.

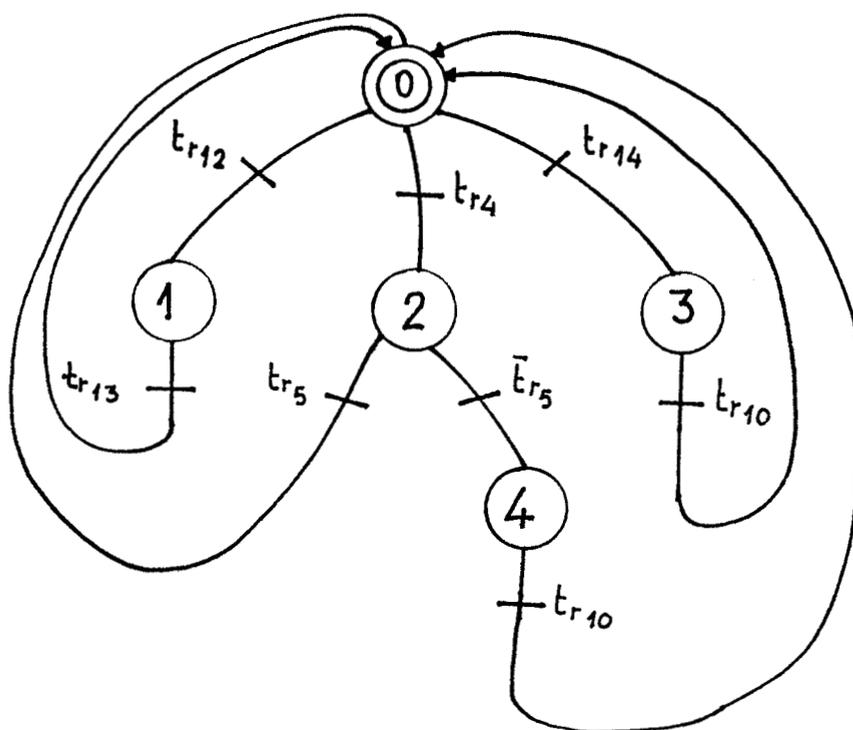
Les seules transitions particulières à ce graphe sont :

tr_{10} : réinitialisation (la réparation, la maintenance ou l'intervention demandée est finie).

tr_{11} : il existe au moins une place pour stocker la ressource propre.

2.3 ETAT DES ROBOTS

A tout instant, l'état d'un robot est donné par le graphe d'état suivant :



Définition des états :

(0) - Libre : Le robot n'est lié par aucune décision. Il est à la disposition du module de décision.

(1) - Attente : Un robot est mis en attente dans les situations suivantes :

a - il n'y a pas d'opérations exécutables qu'il sache réaliser à cet instant.

b - il est affecté à une opération qui n'est pas exécutable, mais va le devenir dans un avenir proche. C'est une affectation prévisionnelle.

Celle-ci n'est pas définitive. Elle peut être remise en cause si l'évolution de l'état de la cellule diffère de celle qui a été prévue.

- (2) -Actif : Le robot est en cours d'exécution d'une opération.
- (3) -Maintenance : Le robot est à la disposition du service d'entretien.
- (4) -Diagnostic : Le robot est à la disposition du système de diagnostic.

Signification des transitions :

tr₁₂ : décision, attente d'événements.

Ce peut être :

- libération de ressources partageables (zone de travail)
- nouvelles opérations exécutables.

tr₁₃ : événements arrivés.

tr₁₄ : décision de mise en maintenance

Les autres transitions ont la même signification que celles des graphes d'état des opérations de fabrication et de stockage.

A chacun des robots est associée une liste de paramètres, permettant son identification exacte et ses capacités à tout instant :

- liste des actions élémentaires qu'il sait réaliser

Exemples :

- collage
- assemblage
- découpage

- période de révision

- performances en :
 - vitesse
 - précision
 - répétabilité

L'ensemble des robots évoluant dans la cellule sont représentés par :

$$R_t = (R_t^1, R_t^2, \dots, R_t^m)$$

où R_t^i représente la liste des aptitudes du robot à l'instant t , c'est-à-dire les actions qu'il peut réaliser.

2.4 ENSEMBLE DES COMMANDES ADMISSIBLES

A chaque instant t , l'ensemble des opérations que le robot i est capable de réaliser est connu.

L'ensemble des commandes admissibles par ce robot est alors :

$$U_t^i : (R_t^i \cap EXEC_t) \cup ATT \cup MAINT$$

où :

$EXEC_t$, est l'ensemble des opérations exécutables, dont les ressources partageables associées sont disponibles à l'instant t .

ATT , est une commande particulière, attente d'événements.

$MAINT$, est une commande de mise en maintenance.

3 - STRUCTURE DU SYSTEME DE CONDUITE [STA. 85 ter] [SIL. 82].

3.1 INTRODUCTION

Un système de conduite est composé d'un système de décision et d'un système d'information [BER. 83].

Compte tenu de la modélisation de la cellule, le système de conduite est organisé de façon modulaire.

Les différents modules sont activés par des changements d'états de la cellule.

Le système d'information a pour rôle l'acquisition, le traitement et le stockage des données nécessaires au fonctionnement de la cellule de production. Il comprend les modules identification des ressources (pièces, ...), de contrôle de l'exécution des tâches, de diagnostic en cas d'incident et de mise à jour de l'état ou de la base de données.

Le système de décision a pour rôle le choix de l'opération à lancer, en fonction des informations disponibles et de l'état courant du système de production. Il comprend les modules d'affectation et de lancement.

La partie du système d'information assurant les échanges avec l'extérieur de la cellule est regroupée dans deux modules. Le premier concerne la communication homme-machine et le second, la communication machine-machine.

La structure du système de conduite est présentée par la figure 5.

Une partie de ce système a été simulée sur micro-ordinateur. Nous avons utilisé un des dialectes de LISP, MULISP sous CP/M 80. Notre choix a été orienté par la bonne interactivité du langage et par ce qui fait sa force, le traitement de symbole, qui facilite beaucoup la représentation des structures de données [annexes].

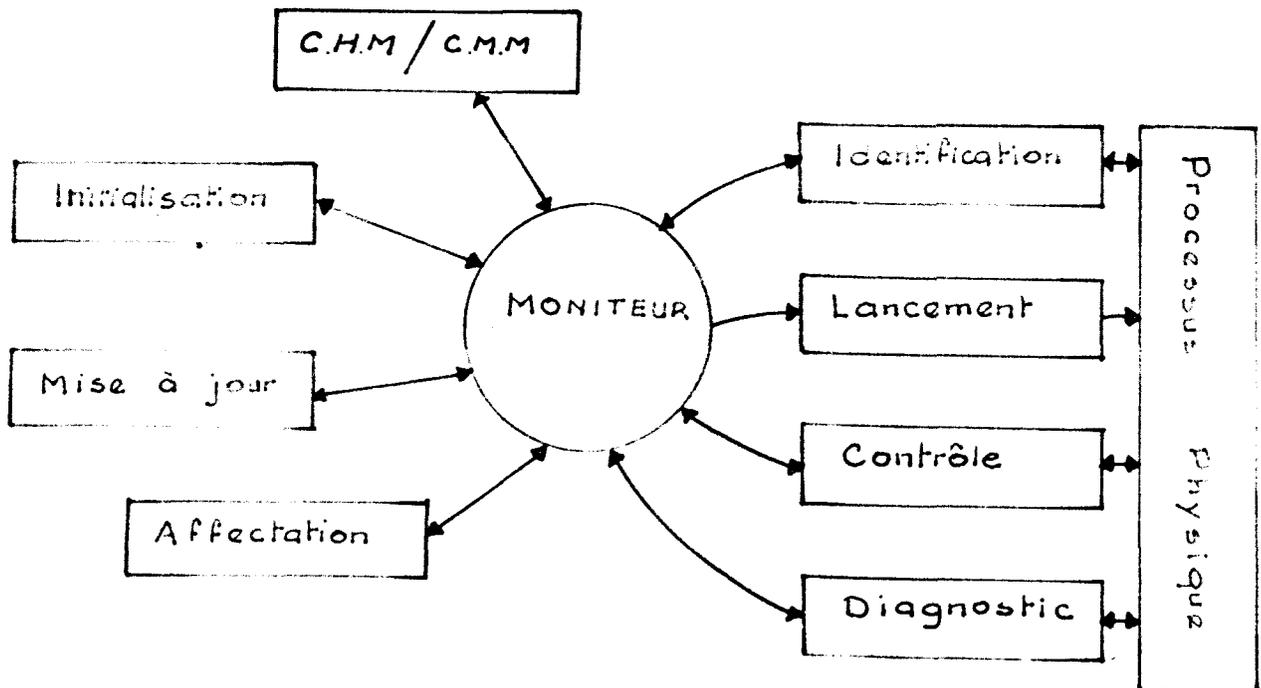


Fig. 5

L'implantation sur le site est en cours actuellement [annexe]

3.2.2 Description de l'environnement

L'environnement est représenté par la liste, initialement vide des pièces présentes dans la cellule. Chacune des pièces est elle même décrite par une liste contenant ses caractéristiques :

(Identificateur de la Pièce (Position courante)
(Paramètres de prise)
(Fréquence d'apparition))

où

Position courante correspond à la position de prise sur le tapis, ou la position dans le magasin ou dans le stock tampon. Cette information peut être utile non seulement pour la prise de la pièce, mais aussi pour une éventuelle gestion du placement dans le stock (surface, ou volume occupé).

Paramètres de prise représentant l'orientation, le volume, le poids et éventuellement la forme de la pièce.

Fréquence d'apparition c'est la fréquence d'arrivée de la pièce dans la cellule.

Cette liste peut être, soit introduite par un opérateur, soit téléchargée par un niveau supérieur dans la base de données de la cellule où les paramètres, position courante et fréquence d'apparition sont mis à jour automatiquement.

3.2.3 Description des robots

Un robot est décrit par la liste de ses aptitudes (actions qu'il sait réaliser) et sa période de maintenance ou date à laquelle le risque de tomber en panne n'est plus négligeable.

Il est décrit par une liste de la forme :

(Identificateur du robot (Période de maintenance)
(Type d'action, Paramètre 1, Paramètre 2 ...)
(Type d'action, " " " "))

où, paramètre i est une spécification pour la réalisation de l'action.

3.3 SYSTEME D'INFORMATION

3.3.1 Identification

Ce module, en liaison avec le système de vision, permet de relever certaines caractéristiques de manipulation des pièces qui arrivent sur le tapis roulant. Elles peuvent être, des coordonnées du centre de gravité, orientation selon un axe choisi de la pièce, etc...

Ces données sont nécessaires pour la commande des robots et l'adaptation éventuelle de l'outil de prise.

3.3.2 Contrôle

Ce module permet, à partir de capteurs, de vérifier la bonne exécution des opérations. Il transmet à l'organe de conduite, une information de fin d'exécution ou de mauvaise exécution.

Les moyens exigés pour un contrôle efficace peuvent être considérables (capteur d'effort, de proximité, de vision, etc...). De même, la spécification des paramètres à contrôler n'est pas une tâche évidente. Elle est d'autant plus complexe que le contrôle demandé est fin.

Ce module est activé chaque fois que la réalisation d'une opération est commencée.

3.3.3 Diagnostic

C'est un module "expert" qui suivant un problème détecté par le module de contrôle, est capable de diagnostiquer la panne et de donner la marche à suivre pour y remédier.

Pour certaines défaillances, il peut être amené à opérer des tests sur le processus physique (robots, système de vision, tapis roulant, ...).

Il est activé par une "exécution incorrecte" d'une opération. Il décide alors, si cette opération redevient pré-exécutable (a) ou si sa mauvaise exécution interdit la poursuite de la fabrication.

Exemples :

Si (test 1 vrai) et (test 2, vrai) et (pince 1 fermée)
alors (Axe i défaillant)

Si (test 2, vrai) et (test 3, vrai)
alors (Robot inapte)
(Appel opérateur)

Si (Axe i défaillant) et (Robot en cours)
alors (Robot inapte partiellement)
(Mise à jour de l'état de l'opération en cours)
(Mise à jour des aptitudes du robot)
(Appel opérateur)

C'est un module qui représente, en lui-même, un sujet de recherche sur lequel nous poursuivons nos réflexions.

3.3.4 Mise à jour

a) Etat de l'environnement

La mise à jour de l'environnement est activée chaque fois qu'une pièce :

- arrive dans le magasin ou en stock tampon ;
- sort du magasin ou du stock tampon.

Toutes ces informations sont générées par les trois modules :

- identification ;
- affectation ;
- contrôle.

Au magasin et au stock tampon sont associées deux listes, composées d'autant de sous-listes que de pièces emmagasinées. Chacune de ces sous-listes englobe toutes les informations relatives à la pièce (§ 3.2.2.).

b) Etat des robots

La mise à jour de l'état des robots est activée à deux niveaux différents :

- à l'issue de chaque prise de décision d'affectation ;
- à l'issue du contrôle d'exécution de l'opération allouée.

Celle des aptitudes, est activée :

- à l'issue d'un diagnostic.

c) Etat du produit à réaliser

L'état du produit à réaliser est décrit par l'état des opérations qui le composent.

Les opérations qui sont dans un même état sont regroupées dans une même liste. La mise à jour de l'état du produit revient donc à la mise à jour des listes représentant les états des opérations.

Il est clair que cette description ne s'applique qu'aux produits en cours de fabrication, c'est-à-dire pour lequel aucune opération n'est dans l'état non-réalisable.

Cette mise à jour est activée par les modules :

- identification ;
- affectation ;
- contrôle ;
- diagnostic.

3.4 SYSTEME DE DECISION

3.4.1 Affectation

C'est le module principal de décision. Il dispose de la liste des opérations exécutables sur lesquelles, il doit opérer un ordonnancement qui satisfait aux critères retenus, en vue de leurs exécutions par les robots.

Cette liste étant dynamique, il doit à chaque fois en fonction de l'état global de la cellule, ré-ordonnancer les opérations qu'elle contient.

L'algorithme de choix ou d'ordonnancement dynamique forme le noyau de ce module. Il peut être fondé sur plusieurs approches différentes.

Dans le cadre de notre travail, nous en étudions deux. La première est une approche stochastique qui prend en compte les probabilités de transition entre les différents états possibles de la cellule. La seconde est une approche multicritère qui utilise la notion de surclassement pour comparer les opérations, relativement à plusieurs critères.

Ces deux approches font l'objet du chapitre suivant. Au niveau de la simulation, seule la seconde approche a été implantée.

Ce module d'affectation est activé dès qu'un robot devient libre.

3.4.2 Lancement

C'est un module en liaison avec le processus physique. Il permet l'activation du robot pour la réalisation de l'opération qui lui a été affectée. Pour ce faire, il a connaissance des paramètres d'exécution de l'opération.

Ce module n'a pas été simulé puisque il représente, en fait, l'interface entre la décision d'affectation et son application réelle sur le robot.

3.5 COMMUNICATIONS EXTERNES

3.5.1 Communication Homme-Machine

C'est un module interface entre l'opérateur et la cellule. Il permet d'une part, l'acquisition des données nécessaires à la réalisation du produit, d'autre part, la possibilité pour l'opérateur de connaître l'état de la cellule à n'importe quel moment.

La communication entre l'opérateur et la cellule est prise en charge par un système conversationnel basé sur un langage simple ne nécessitant aucune connaissance particulière en informatique.

3.5.2 Communication Machine-Machine

Ce module permet à la cellule de communiquer avec d'autres cellules par l'échange de certaines données par exemple, ou bien d'acquérir directement, d'un niveau supérieur, le plan de production (gamme d'assemblage). Ce niveau peut être celui de système de gestion de production assisté par ordinateur (G.P.A.O.).

4 CONCLUSION

Nous avons décrit le modèle d'évolution de la cellule, à l'aide des graphes d'états et de transitions, ainsi que la structure de son système de conduite.

L'implantation d'un tel système peut être envisagée selon deux approches :

Dans la première, on associe à chacun des robots, un calculateur dont le rôle sera d'assurer l'exécution des décisions de réalisation des opérations commandées par le niveau superviseur. Celui-ci incombe à un autre calculateur coordonnateur, dont la fonction est la mise à jour de l'état de la cellule et la prise de décision au niveau global.

Dans la seconde, un seul ordinateur prend en charge le moniteur de pilotage et les procédures d'exécution des opérations par les robots. Il est évident que, dans ce cas, toutes les procédures (ou tâches) doivent pouvoir évoluer en parallèle et de ce fait, sous un système d'exploitation multi-tâches temps réel.

La structure informatique, disponible actuellement dans la cellule, comprend un seul ordinateur (HP 1000), qui par l'intermédiaire du langage de manipulation LM, commande les deux robots sous le système d'exploitation multi-tâches temps réel, RTE. A. Il peut sembler donc a priori, possible d'utiliser une configuration selon la seconde approche citée plus haut. Malheureusement, cette solution ne peut être envisagée dans l'état actuel de développement de LM. En effet, ce langage est monotâche et interdit donc l'exécution simultanée de plus d'une tâche LM, sous RTE. A. L'implantation actuelle est, de ce fait, très dégradée.

Notre intérêt, dans ce travail, porte plus particulièrement sur le problème de la décision qui est indépendant de la structure informatique choisie.

La présentation de ce problème et la méthode proposée pour le résoudre, font l'objet du chapitre suivant.

Chapitre IV
MODULE DE DECISION

1 - INTRODUCTION

Le cœur du système de conduite d'une cellule de production flexible, est le module de décision. Son rôle est primordial puisque c'est de la manière dont les décisions sont prises que dépendent les performances de la cellule.

Le choix de la procédure de prise de décision est donc capital.

Le problème de décision qui se pose est généralement un problème de commande optimale, où il s'agit en fonction de critères retenus, d'optimiser une certaine fonction objectif. Mais la particularité du problème retenu dans notre travail concerne la prise en compte de la dynamique du processus étudié. En effet, c'est un problème d'ordonnancement-affectation dynamique d'un ensemble d'opérations à exécuter sur un ensemble de robots, afin de minimiser sa durée de réalisation.

Nous avons vu au chapitre III, l'inexistence de méthodes rigoureuses pour appréhender ce type de problème, le plus souvent très complexe.

L'approche que nous proposons dans cette étude afin de résoudre ce problème d'une manière satisfaisante sinon optimale, est basée sur la méthode ELECTRE, et donc sur la notion de surclassement entre les opérations.

Elle sera détaillée dans ce qui suit mais après que nous ayons posé le problème de la décision optimale et exposé brièvement une approche stochastique de ce problème, étudiée en collaboration avec l'Institut de contrôle et systèmes de WROCLAW (Pologne).

2 - PROBLEME DE LA DECISION OPTIMALE

Le problème de la commande est pris en compte par un exécutif temps réel, composé de deux parties principales :

Information : Elle est activée à chaque franchissement d'une transition d'un des graphes décrivant l'état de la cellule. Elle remet à jour cet état, afin de fournir

une information correcte et cohérente pour la partie décision.

Décision : Elle a pour fonction de choisir parmi les décisions admissibles, celle qui satisfait le mieux aux critères utilisés.

Une décision est en fait un choix d'une opération à réaliser parmi plusieurs possibles. Or, à cette dernière, est associé un ensemble de ressources propres et de ressources partageables dont la disponibilité est une condition d'admissibilité de la décision.

L'activation de ce module de décision peut donc être envisagée de diverses manières, selon que les ressources partageables associées à l'opération choisie, sont toutes disponibles, en partie seulement, ou non disponibles.

Nous aboutissons alors à une décision d'affectation, qui peut être soit définitive, soit prévisionnelle, ou bien à une décision de mise en attente du robot.

Le tableau suivant récapitule les diverses situations :

Robot	Res. propres	Res. partag.	Affectation
Libre	Disponibles	Disponibles	Définitive
	"	Non Dispo.	Mise en att.
Occupé	"	Disponibles	Prévisionn.
	"	Non Dispo.	Prévisionn.

Dans le cadre de notre travail, nous nous sommes placés dans la première hypothèse, c'est-à-dire l'affectation définitive.

L'activation du module de décision ne concerne donc que les robots libres et les opérations exécutables dont les ressources partageables sont disponibles.

Le choix a été fait dans un souci de résoudre d'abord, le problème de la décision pour le cas qui nous semble le plus abordable, pour ensuite prendre comme base cette résolution, et intégrer au problème, graduelle -

ment, des hypothèses plus contraignantes, notamment, l'affectation prévisionnelle, ou par anticipation.

Nous ne considérons donc, que les décisions appartenant à l'ensemble restreint des commandes admissibles, donné par :

$$U_t^i = R_t^i \cap EXEC_t \quad (i = 1, 2, \dots, m)$$

Le problème posé est la réalisation d'un produit en minimisant le temps final.

Soit J_t , ce temps. Le problème se pose alors comme :

$$\begin{aligned} \min \quad & J(X_t, E_t, R_t) \quad , \quad t = 0, 1, 2, \dots \\ & u_t \in U_t \end{aligned} \quad (1)$$

objectif minimiser le temps final

où,

X_t : état du produit

E_t : état de l'environnement

R_t : état des robots

donner le temps de réalisation

Les instants de décision sont définis comme suit :

Soit t , l'instant de prise de la décisions, u_t et $\mathcal{T}(u_t)$ le temps d'exécution de cette décision.

Soit \mathcal{C}_t , l'ensemble des temps d'exécution restant des opérations en cours de réalisation. La prochaine décision prise par le système le sera après un temps :

$$\Delta(u_t) = \min(\mathcal{T}(u_t) \cup \mathcal{C}_t)$$

min(temps de (u_t) et C_t)

En utilisant le principe d'optimalité de BELLMAN, le temps final peut être exprimé à l'instant t par :

$$J(X_t, E_t, R_t, U_t) = \Delta(u_t) + J^*(X_{t+\Delta(U_t)}, E_{t+\Delta(U_t)}, R_{t+\Delta(U_t)})$$

peut être

où, $J^*()$ est la valeur optimale de J .

Le temps final minimum est donc donné par le programme dynamique suivant :

$$J^*(X_t, E_t, R_t) = \min_{u_t \in U_t} (\Delta(U_t) + J^*(X_{t+\Delta(U_t)}, E_{t+\Delta(U_t)}, R_{t+\Delta(U_t)})) \quad (2)$$

OK

dont l'initialisation se fait à l'instant final t_f , par :

$$J^*(X_{t_f}) = 0, \quad \forall E_{t_f} \text{ et } R_{t_f}$$

On sait en outre (par hypothèses), que l'état de l'environnement $E_{t+\Delta}(U_t)$ dépend des variables aléatoires, arrivée et disponibilité des ressources propres. L'état des robots $R_{t+\Delta}(U_t)$ dépend de la variable aléatoire, panne de robot.

C'est donc un problème stochastique d'optimisation monocritère à espaces d'état et de temps de décision discrets.

On peut le résoudre en utilisant des méthodes basées sur des modèles stochastiques.

Si nous adoptons un modèle de représentation markovien, toutes les transitions dans les différents graphes d'état décrivant la dynamique du système, seront associées à des probabilités. Elles représentent alors les probabilités de transition d'un état du système à un autre.

Les espaces d'état et des temps étant discrets, on utilise pour résoudre ce problème, la programmation dynamique sur un processus Markovien afin d'aboutir à une solution optimale du problème posé.

Nous montrerons d'une manière succincte, que pour cette approche, l'obtention d'un résultat optimal en temps réel, est difficile au vu de la dimension de l'espace d'état du système.

C'est pourquoi, nous avons utilisé une autre méthode fondée sur les remarques suivantes, concernant la dépendance de J^* vis-à-vis de ses arguments.

En effet, sa minimisation conduit à celle des temps d'inactivité des robots. Or, celle-ci est le résultat d'une maximisation du nombre d'opérations exécutables. Il devient donc, dans ce cas important, de savoir choisir parmi ces opérations, celles qui après leur exécution permettront au plus grand nombre d'opérations de devenir exécutables.

Lorsque l'on sait qu'un grand nombre de paramètres sont associés à une opération, le choix devient une tâche difficile.

Ces paramètres sont considérés comme des fonctions objectifs et le problème original mono-critère est remplacé par un problème multi-critères.

Ces deux approches font l'objet des paragraphes suivants.

3 - APPROCHE STOCHASTIQUE

3.1 RAPPELS SUR LES PROCESSUS DE MARKOV

3.1.1 Définition [MART. 78]

"Un processus stochastique est dit Markovien, si la probabilité conditionnelle d'un événement futur, étant donné l'état actuel du système et des événements passés, est indépendante des événements passés et dépend seulement de l'état actuel du système".

3.1.2 Programmation Markovienne

Elle s'applique à des processus à espaces d'états et de temps discrets et est basée sur le principe d'optimalité de BELLMAN.

En programmation dynamique, on résoud la récurrence d'un programme en commençant par l'étape finale, en supposant un horizon fini T . On peut donc considérer qu'à un instant t donné, où le système est dans un des M états possibles, la probabilité de transition d'un état i à l'instant t , vers un état j à l'instant $t + 1$, est donnée par :

$$P(Y_{t+1} = j / Y_t = i) = P_{ij}$$

On suppose que cela engendre un coût C_{ij} de la transition de i vers j .

On définit $V_{T-t}(i)$, comme le coût moyen engendré lorsque i est l'état de départ. Il est égal à la somme du coût de la transition vers l'état j , à l'instant $t + 1$ et de la valeur de l'état à cet instant, c'est-à-dire $V_{T-t-1}(j)$.

Or le passage à l'état j n'est qu'estimé par une probabilité P_{ij} .

A partir donc de la définition de l'espérance mathématique d'une variable aléatoire discrète, on peut écrire que :

$$V_{T-t}(i) = \sum_{j=1}^M P_{ij} (C_{ij} + V_{T-t-1}(j))$$

Lorsqu'en plus, on a des décisions à prendre à chaque instant pour faire évoluer le système d'un état vers un autre, on aura un problème décisionnel Markovien.

Soit U_t , l'ensemble des décisions admissibles à l'instant t . La probabilité de transition P_{ij} , comme le coût C_{ij} dépendent maintenant de la décision u_t de U_t . On aura alors $P_{ij}(u_t)$ et $C_{ij}(u_t)$.

Le problème devient donc la recherche de la décision u_t qui engendre le coût moyen optimum.

D'où

$$V_{T-t}(i) = \underset{u_t \in U_t}{\text{Opt}} \left(\sum_{j=1}^M P_{ij}(u_t) (C_{ij}(u_t) + V_{T-t-1}(j)) \right)$$

ok

Dans notre cas, l'algorithme de décision consiste en la recherche de la décision d'affectation u_t dans U_t .

Le critère choisi est la minimisation du temps de réalisation du travail J. Or celui-ci dépend entre autres de la variable aléatoire, disponibilité des ressources. On ne peut donc que minimiser son espérance mathématique E(J).

Pour garder un aspect général au problème, on remplace J par un coût de la réalisation Q, qui s'écrit :

$$Q = \sum_{t=1}^T C_{ij}(u_t)$$

pourquoi ?

Le coût global moyen est donné par l'espérance mathématique de Q.

$$E(Q) = E \left(\sum_{t=1}^T C_{ij}(u_t) \right)$$

$\forall i \text{ sur } [0, \dots, T] \quad \forall j \text{ sur } [0, \dots, T]$

Le problème est donc la minimisation de ce coût global moyen, par l'algorithme de décision. Le résultat optimal ne peut être atteint que si la probabilité de l'état initial $P(Y_0 = j)$ et les probabilités de transitions sont connues.

La solution est obtenue à l'aide du programme dynamique Markovien suivant :

$$V_1(i) = \min_{u_{T-1} \in U_{T-1}} \left(\sum_{j=1}^M C_{ij}(u_{T-1}) \cdot P_{ij}(u_{T-1}) \right)$$

$$V_{T-t}(i) = \min_{u_t \in U_t} \left(\sum_{j=1}^M P_{ij}(u_t) (C_{ij}(u_t) + V_{T-t-1}(j)) \right)$$

3.2 CONCLUSION

Cette approche ne peut malheureusement pas être appliquée au vu de la dimension du problème pour un système réel.

Le nombre d'états possibles M devient astronomique même si l'on considère une cellule composée de quelques robots ayant à réaliser seulement quelques tâches.

Exemple :

Soit un système composé de :

- 3 robots ($m = 3$), ayant chacun 3 états possibles : libre, occupé, en panne ;
- 6 tâches à réaliser ($n = 6$), ayant chacune 3 états possibles : réalisée, non-réalisée, en cours ;
- 6 ressources ($k = 6$), ayant chacune 2 états possibles : disponible, non-disponible.

Le nombre d'états possibles est alors égal à :

$$M = 3^m \cdot 3^n \cdot 2^k = 3^9 \cdot 2^6 = 1\,259\,712$$

Si l'on ajoute les moyens d'observation de ces états et la connaissance des probabilités de transition, le problème devient inabordable par cette méthode.

Cette approche a été plus particulièrement étudiée par l'Institut de Contrôle et Systèmes de l'Université technique de WROCLAW (Pologne), dans le cadre d'une collaboration avec notre Laboratoire [BUB. 85] [REY. 84] [REY. 84 bis].

Nous avons étudié une cellule d'assemblage à deux robots et décomposé le processus de décision globale en deux processus relatifs chacun à un robot. Cette décomposition a permis de réduire le nombre d'états possibles pour chacun des processus de décision, néanmoins, ceci semble difficilement applicable à un cas réel.

Une autre approche multi-critères nous a permis de proposer une méthode de résolution basée sur la notion de sur-classement qui fera l'objet du paragraphe suivant.

4 - APPROCHE MULTICRITERES [ROY. 80] [LAG. 78] [LAG. 80] [BOU. 75]

4.1 INTRODUCTION

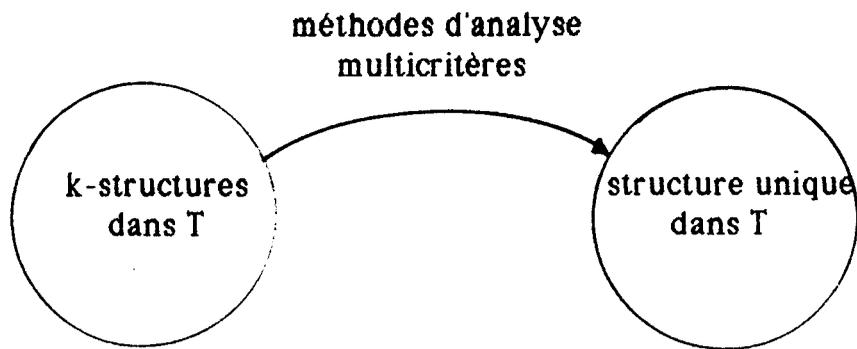
Lorsqu'un organe de décision ou décideur doit choisir une tâche ou une solution parmi plusieurs possibles, il essaie naturellement d'opter pour celle qu'il juge la plus satisfaisante, sinon la meilleure par rapport aux critères qu'il sait fixer. Mais pour ce faire, il faudrait qu'il puisse les comparer.

Dans une perspective de décision monocritère, le problème est simple. Le décideur est censé être capable d'opérer un choix afin de satisfaire son objectif, en maximisant une certaine fonction utilité.

Mais dans le cadre d'une décision à critères multiples, la solution n'est pas aisée. En effet, si l'on adopte pour un critère la définition suivante [BERN. 71] :

" Un critère sur un certain ensemble de tâches T,
est ce qui établit une structure sur cet ensemble".

La structure pouvant être une relation quelconque (d'ordre, de pré-ordre, d'équivalence, ...), nous aurons alors, dans T, autant de structures que de critères considérés. Le problème est alors la recherche d'une structure unique dans T, à partir d'une agrégation des k-structures initiales (k étant le nombre de critères).



Soit $T = (T_1, T_2, T_3, \dots, T_n)$

La question qui se pose alors, est comment agréger ces structures (ordinales, cardinales hétérogènes), en une structure unique ?

Il existe plusieurs approches pour résoudre les problèmes multi-critères, mais elles restent attachées en fait à deux grandes écoles. La première est américaine et s'intéresse à des aspects plus théoriques [MOU.81]. La seconde est française et est beaucoup plus pragmatique. Elle a permis de nombreux traitements de problèmes multicritères.

Selon B. ROY, l'un des chefs de file de l'école française, quatre approches peuvent être retenues [ROY.70] [BOL.82] [LAG.83] :

1 - Hiérarchisation des critères ou méthode lexicographique

Si la hiérarchisation des critères est possible, on résout le problème multicritère de la manière suivante.

On cherche une solution relativement au critère le plus important J_1 . On obtient un ensemble de solutions optimales $T(J_1)$. On considère ensuite le deuxième critère dans l'ordre d'importance J_2 . On résout le problème, mais sur l'ensemble $T(J_1)$, on obtient alors $T(J_2)$, l'ensemble des solutions optimales. On continue ainsi jusqu'au dernier critère J_k , avec $T(J_k)$, l'ensemble des solutions au problème posé.

2 - Agrégation des critères

C'est l'agrégation d'un ensemble de critères en un critère unique qui conduit à une fonction globale d'utilité.

Généralement, le modèle d'agrégation est linéaire. Par exemple, la fonction d'utilité F , pour une tâche T_i , dont le choix est lié à la satisfaction de plusieurs critères (J_1, \dots, J_k) , s'écrit :

$$F(T_i) = \sum_{j=1}^k J_j(T_i) P_j$$

P_j : coefficient de pondération

Mais le plus souvent, la détermination de cette fonction est assez difficile et le décideur est contraint au choix, soit d'un critère unique, soit à des utilités sur chacun des critères, cardinales et exprimées dans la même unité.

3 - Construction des relations de surclassement

La difficulté d'agrégation des critères en un critère unique a poussé à la recherche d'autres moyens de résolution. Les méthodes de surclassement s'inscrivent dans cette perspective.

Soient deux tâches T_i et T_j de T . Une relation de surclassement S est une relation binaire définie sur T , telle que :

- $T_i S T_j$ et non $T_j S T_i$, signifie que T_i surclasse T_j , ou est préférée à T_j
- $T_i S T_j$ et $T_j S T_i$, signifie qu'il y a indifférence
- Il n'existe pas de relations entre T_i et T_j . Dans ce cas, les tâches sont incomparables, mais l'équivalence n'est pas exclue.

4 - Méthodes interactives ou définition progressive et partielle des préférences

Ce sont des méthodes d'aide à la décision. Elles permettent au décideur de changer de contraintes de décisions antérieures, etc..., afin de mieux définir son choix et donc d'atteindre son objectif.

Toutes ces méthodes sont généralement utilisées en économie, ou dans l'industrie, pour résoudre des problèmes complexes, comme le choix des investissements, la planification des projets, les problèmes de transport, les problèmes d'ordonnancement, etc...

Ces problèmes font intervenir beaucoup de critères et le but des méthodes d'analyse présentées, est beaucoup plus d'aider à la décision que de résoudre les problèmes.

Le problème posé dans ce chapitre, concernant l'allocation des tâches à un ensemble de robots à aptitudes multiples, n'appartient pas à la classe des problèmes cités par le fait de sa dynamique. Celle-ci est due entre autre aux différentes perturbations imprévues qui peuvent survenir dans la cellule et qui exigent une décision en ligne. Cet aspect "temps réel", est primordial dans notre étude, et à notre connaissance, il n'a jamais été abordé par les méthodes citées. Ceci est d'ailleurs parfaitement compréhensible par le fait qu'elles n'ont pas été conçues pour cette classe de problèmes.

Nous avons vu dans (§ 2), que le problème original monocritère pouvait être transformé en un problème multicritères, en associant à chaque tâche un ensemble de critères.

Dans une première partie, nous formulons le problème et présentons les critères qui nous semblent caractériser au mieux une tâche ou opération de la gamme de fabrication. Dans une seconde partie, nous exposons l'approche que nous avons adoptée pour résoudre ce problème et la méthode utilisée.

Enfin quelques exemples seront traités pour juger des résultats obtenus.

4.2 POSITION DU PROBLEME

Le temps de réalisation du produit s'exprime de la manière suivante :

$$\begin{aligned} \text{Temps de réalisation} &= \text{Temps d'activité des robots} \\ &+ \\ &\text{Temps d'oisiveté des robots} \end{aligned}$$

Le temps d'oisiveté de chaque robot est nul, si à tout instant t où il se libère, l'ensemble des tâches qu'il sait réaliser est non vide. Pour augmenter la probabilité de cet événement, il convient d'agir en sorte que cet ensemble soit le plus grand possible.

Par ailleurs, le temps d'activité des robots est d'autant plus petit que chaque tâche j est affectée au robot optimal pour son exécution. La probabilité pour que cette affectation soit possible est d'autant plus grande que l'ensemble des tâches exécutables à l'instant où le robot devient libre, est plus grand.

Ce raisonnement heuristique nous conduit à chercher une procédure d'affectation qui tend à maximiser l'ensemble des tâches exécutables. Dans ces conditions, la probabilité pour qu'un robot libre trouve une tâche à réaliser de manière optimale, est élevée.

Le choix de la tâche qui déblocuera après son exécution le plus grand nombre de tâches exécutables n'est pas évident. En effet, elles sont soumises à des contraintes de succession de disjonction, etc... Elles sont caractérisées par leurs positions dans la gamme de fabrication, et par des paramètres spécifiques, tels la disponibilité de leurs ressources, leurs durées de réalisation, ...

Tous ces paramètres sont exposés dans le paragraphe suivant.

4.3 CHOIX DES CRITERES

La détermination des paramètres qui caractérisent une opération exécutable ne relève pas d'une méthode exacte mais beaucoup plus d'une démarche intuitive.

De plus, l'importance de leur influence sur le choix de l'opération n'est pas connue a priori, elle ne peut être qu'estimée par l'expérience ou par la simulation.

Parmi l'ensemble de ces paramètres, nous présentons ceux qui nous semblent les plus significatifs.

1 - Nombre de successeurs d'une opération

Dans une gamme de fabrication, les différentes opérations s'exécutent selon un enchaînement, que nous avons décrit à l'aide d'un graphe orienté acyclique. On associe à une opération une ou plusieurs opérations successeurs immédiats dans la gamme, et sa réalisation contribue à les rendre pré-exécutables (a). Il semble alors judicieux de choisir parmi les opérations exécutables, celles dont le nombre de successeurs immédiats est le plus élevé.

Un autre paramètre que nous jugeons intéressant est celui du nombre de successeurs total (immédiats et autres), dont les ressources propres sont disponibles dans la cellule (magasin ou stock intermédiaire). Ces successeurs ont l'avantage de devenir directement exécutables, lorsque leurs antécédents sont réalisés.

2 - Coefficient de déblocage

La réalisation de chacun des antécédents d'une opération (si elle en a plusieurs), contribue à la rendre pré-exécutable (a). On associe ainsi, à chacun de ces antécédents, un coefficient dit de déblocage. Il dépend de

l'état, réalisé ou non des antécédents de l'opération et prend sa valeur maximale lorsqu'il reste un seul antécédent non-réalisé.

3 - Temps de réalisation de l'opération

Le temps d'exécution d'une opération peut dépendre du robot qui la réalise. Il est donc important de savoir à un moment donné, lorsque le robot le plus performant est occupé, s'il faut l'attendre ou réaliser l'opération avec un autre robot, disponible mais de performances moindres.

Si le temps d'exécution est indépendant des robots, il nous semble que le choix des opérations les plus courtes est intéressant puisqu'il peut contribuer à l'accroissement du nombre d'opérations exécutables.

4 - Fréquence d'apparition des ressources propres dans le magasin

Une ressource qui arrive dans le magasin est susceptible d'être utilisée par plusieurs cellules. Il y a donc, pour chacune de ces dernières, un risque de perte des ressources nécessaires à la réalisation de son objectif au profit des autres cellules.

Celles-ci doivent alors avoir une estimation de la durée de la disponibilité (ou de l'indisponibilité) de la ressource dans le magasin afin d'orienter leur choix en conséquence.

Ainsi, les pièces "rares", c'est-à-dire celles dont la fréquence d'apparition est faible et qui de plus sont convoitées par plusieurs cellules, sont choisies en priorité.

5 - Espace libéré sur le stock

Les ressources propres sont généralement de dimensions, de formes, d'envergures, différentes. Elles occupent donc en stock, des emplacements de tailles variables.

L'utilisation d'une ressource libère un espace sur le stock et augmente ainsi la zone de stockage.

On associe à chaque ressource propre un coût relatif au stockage, proportionnel à l'espace occupé (ou libéré).

6 - Probabilité de bonne résolution

Les caractéristiques d'un robot ou d'une machine évoluent avec le temps. Il est donc important de tenir compte de sa fiabilité à un moment donné, pour l'affecter à la réalisation de certaines opérations.

Cette fiabilité peut être représentée par une appréciation du risque d'une mauvaise exécution de l'opération par le robot, ou par la probabilité de bonne exécution.

Les paramètres présentés ne forment pas un ensemble exhaustif, d'autres pourraient être pris en considération dans un cadre plus général.

Dans le cas où plusieurs produits sont réalisés simultanément, nous pouvons considérer par exemple :

- le niveau de l'opération dans le graphe (la gamme) ;
- la priorité relative entre les produits.

Dans ce qui suit, nous allons présenter plus précisément les paramètres que nous avons choisis dans notre travail.

Associons à chaque décision u_j , c'est-à-dire à chaque opération j , les définitions suivantes :

- $S(j)$, ensemble des successeurs immédiats de j
- $P(j)$, " des prédécesseurs immédiats de j
- $\mathcal{S}(j)$, " de tous les successeurs de j

où :

$$\mathcal{S}(j) = \bigcup_{i=0}^{N_j} S^i(j), \quad (N_j \text{ est tel que, } S^{N_j+1}(j) = \emptyset)$$

avec $S^k(j) = S(S^{k-1}(j))$ et $S^0(j) = S(j)$

- $RD(t)$, ensemble des ressources propres disponibles à l'instant t
- $RN(S(j))$, ensemble des ressources propres nécessaires à l'ensemble des opérations $S(j)$

- $D_t(k/j)$, coefficient de déblocage. j n'étant pas en général le seul antécédent de l'opération k , sa réalisation ne rend pas automatiquement k exécutable. On calcule un coefficient de déblocage $D_t(k/j)$ de la façon suivante :

Soit $P_t(k)$, l'ensemble des antécédents de l'opération k , non encore réalisés à l'instant t .

Le coefficient de déblocage est alors défini par :

$$D_t(k/j) = \frac{1}{|P_t(k)|}$$

Comme une opération j peut avoir plusieurs successeurs, il lui est associé un ensemble de coefficients de déblocage. Soit $D_t(j)$, cet ensemble.

Nous avons :

$$D_t(j) = \{ D_t(k/j), \forall k \in S(j) \}$$

Il existe plusieurs façons de comparer deux opérations par rapport à ces coefficients.

Celle qui permet d'avantager une opération dont au moins un coefficient est de valeur maximale et qui débloque ainsi après son exécution au moins une opération, nous paraît la plus judicieuse.

Parmi l'ensemble des critères présentés, ceux qui nous semblent les plus caractéristiques d'une opération exécutable, et que nous avons choisis d'utiliser dans nos exemples sont les suivants :

- a - $J_1(j) = |S(j)|$, nombre de successeurs immédiats de j
- b - $J_2(j) = |RN(\mathcal{S}(j)) \cap RD_t|$, nombre d'opérations de $S(j)$ dont la ressource propre est disponible à l'instant t
- c - $J_3(j) = f(D_t(k/j))$, coefficient de déblocage

Nous avons choisi de ne prendre en considération que le coefficient de valeur maximale, c'est-à-dire :

$$f (Dt (k/j)) = \max_{k \in S (j)} \{ Dt (k/j) \}$$

d - $J_4(j)$, temps de réalisation de l'opération j.

d-1 Temps de réalisation indépendant de robots

Dans ce cas, on a simplement :

$$J_4(j) = t_j \quad (t_j : \text{temps de réalisation de } j)$$

d-2 Temps dépendant du robot qui réalise l'opération

Soit t_{ij} , le temps d'exécution de l'opération i par le robot j.

La solution la plus satisfaisante est celle qui permet la réalisation de toutes les opérations de la gamme avec leurs temps minimum. Or ceci est difficilement atteignable au vu de la dynamique du système, et donc de l'affectation dynamique qui en résulte. Le robot le plus performant d'une opération n'est pas forcément disponible au moment de l'affectation.

C'est pourquoi, nous associons à toutes les opérations exécutables, non pas leurs temps d'exécution, mais l'écart relatif entre leurs temps d'exécution minimum absolu, et le temps d'exécution minimum réalisé par un des robots disponibles.

Soit e_j , l'écart associé à l'opération j

On a :

$$e_j(t) = \left(\min_{\forall i \in R_t} \{ t_{ji} \} - \min_{\forall i \in R_t} \{ t_{ji} \} \right) / \min_{\forall i \in R_t} \{ t_{ji} \}$$

/ i libre à t

Donc,

$$J_4(j) = e_j(t)$$

Seul le cas d-1 a été considéré dans nos exemples.

e - $J_5(j)$, temps d'attente moyen d'une ressource propre par l'opération j

C'est un paramètre qui tient compte :

- du nombre n_j , d'arrivée de ressources propres de j par unité de temps
- de la probabilité p_j que la ressource propre de j soit prise par une autre cellule.

Par rapport à ce critère, on doit avantager les opérations associées aux pièces ou matières premières "rares" :

Une opération j sera préférée à k lorsqu'en cas de perte de leurs ressources propres, j attend en moyenne plus longtemps que k , l'arrivée d'une autre ressource propre.

Le temps d'attente moyen associé à j , tm_j s'exprime par :

$$tm_j = \frac{1}{n_j} P_j$$

On remarque que lorsque la ressource propre est en stock intermédiaire de la cellule, le temps d'attente est nul ($P_j = 0$).

Les P_j sont connus simplement. Soient,

- \hat{n}_j^i , le nombre d'arrivée de ressources propres de j , de la cellule i
- $n_j = \sum_{i=1}^k n_j^i$ (k , nombre de cellules dans l'atelier)

La probabilité que la cellule i prenne la ressource j , est estimée par :

$$n_j^i / n_j$$

Ainsi la probabilité que la ressource propre de j , soit prise par une autre cellule que la cellule i est donnée par :

$$P_j = 1 - \frac{n_j^i}{n_j}$$

$$\text{et donc, } tm_j = \frac{1}{n_j} \left(1 - \frac{n_j^i}{n_j} \right) = \frac{n_j - n_j^i}{n_j^2}$$

$$\text{et, } J_5(j) = tm_j$$

4.4 METHODE DE RESOLUTION FONDEE SUR LA NOTION DE SURCLASSEMENT

4.4.1 Introduction

De toutes les approches présentées pour la résolution des problèmes multicritères, nous avons retenu l'approche fondée sur la notion de surclassement pour les raisons suivantes :

- prise en compte de critères hétérogènes ;
- mise en œuvre facile.

Le système étant dynamique, l'aspect temps réel de la prise de décision est essentiel. Il doit donc être pris en compte dans l'application de la méthode multicritères.

Parmi celles qui utilisent la notion de dominance, les méthodes développées par B. ROY sous le nom d'ELECTRE (Elimination Et Choix Traduisant la Réalité) sont les plus représentatives et les plus utilisées. Les deux principales sont ELECTRE 1 et 2 [GUI. 77] [MOU. 81] [BOU. 75].

La première est un modèle de décision avec partition de l'ensemble des tâches exécutables T. La seconde est un modèle de décision avec classement dans T.

Comme tous les modèles de décision aboutissent soit à une partition, soit à un classement, ces deux méthodes couvrent tout le champ de la décision, dans l'approche fondée sur le surclassement.

La décision d'affectation d'une tâche à un robot n'intervient que lorsque celui-ci devient libre. Il semble par conséquent plus approprié d'utiliser ELECTRE 1.

Dans le cas où plusieurs robots sont libres en même temps, la méthode sera appliquée autant de fois que cela est nécessaire, afin de satisfaire le plus grand nombre d'entre-eux, sinon la totalité.

4.4.2 Description de la méthode ELECTRE 1

Elle a pour but la bi-partition de l'ensemble des tâches exécutables T , entre la tâche choisie et le reste des tâches éliminées. Le choix est effectué en fonction de multiples critères, en utilisant la notion de surclassement.

On dit que la décision $u_t(i)$ (choix de la tâche T_i) surclasse $u_t(j)$, si des tests de concordance et de non-discordance sont satisfaits. Il y a alors un nombre suffisant de critères en accord avec cette assertion et l'évaluation des critères en désaccord ne dépasse pas un seuil, défini comme seuil de discordance.

Chaque décision u_t , qui peut être prise à l'instant t est évaluée selon un ensemble de critères :

$$J(u_t) = (J_1(u_t), J_2(u_t), \dots, J_k(u_t))$$

Chacun de ces k critères peut prendre ses valeurs (qualitatives ou quantitatives) dans un ensemble ordonné et on lui associe un coefficient de pondération p_i , $i = 1, \dots, k$.

$$J_{ij}^+ = (k/J_k(u_i) > J_k(u_j)) , (u_i = u_t(i))$$

$$J_{ij}^= = (k/J_k(u_i) = J_k(u_j))$$

$$J_{ij}^- = (k/J_k(u_i) < J_k(u_j))$$

(">" signifie meilleur que).

Nous posons aussi que :

$$P_{ij}^+ = \sum P_k , k \in J_{ij}^+$$

$$P_{ij}^= = \sum P_k , k \in J_{ij}^=$$

$$P_{ij}^- = \sum P_k , k \in J_{ij}^-$$

$$(P = \sum P_k)$$

Une décision u_i surclassera une décision u_j si et seulement si deux classes de conditions sont satisfaites :

1 - Conditions concernant la véracité du surclassement

Ces conditions expriment le degré de véracité de l'assertion "u_i surclasse u_j".

Elles sont généralement écrites sous la forme :

$$\mathcal{E} (P_{ij}^+ , P_{ij}^- , P_{ij}^-) \geq c$$

où $\mathcal{E} ()$, est une fonction croissante de P_{ij}^+ et décroissante de P_{ij}^- . Le coefficient c est un paramètre exogène donné qu'on appelle seuil de concordance.

Une forme classique de ces conditions est donnée par :

$$P_{ij}^+ \geq P_{ij}^-$$

$$(P_{ij}^+ + P_{ij}^-) / P \geq c$$

2 - Conditions concernant la non-véracité du surclassement

Elles expriment le degré de non-véracité de l'assertion "u_i surclasse u_j" et sont écrites sous la forme :

$$D (J_k (u_i) , J_k (u_j)) \leq d_k , \forall k \in J_{ij}^-$$

Comme exemple pour des critères quantitatifs, ces conditions peuvent s'écrire :

$$J_k (u_j) - J_k (u_i) \leq d_k$$

où d_k est un paramètre exogène qu'on appelle seuil de discordance. Ce coefficient est défini pour chaque critère.

Ces seuils de concordance et de discordance sont établis par le décideur et dépendent de l'échelle d'évaluation adoptée pour les critères k . Une grande part de subjectivité intervient dans la fixation de leur valeur [LAG. 83].

4.5 APPLICATION

L'application de la méthode ELECTRE offre quatre situations possibles pour un couple d'opérations à comparer (i, j).

$i S j$	$j S i$	Conclusion
0	0	Incomparables
1	0	$i > j$
0	1	$j > i$
1	1	équivalentes

Ces quatre situations sont obtenues en vérifiant les conditions de concordance et de discordance par rapport aux seuils choisis, respectivement c et d_k ($k = 1, 2, \dots$).

Une valeur forte du seuil c ($c = 1$), permet un surclassement très fort, mais le nombre d'opérations incomparables risque d'être très élevé. Si, par contre, on choisit une valeur faible ($c = 0,5$), c'est le nombre d'opérations équivalentes qui deviendrait élevé et la notion de surclassement perdrait tout son sens.

L'application d'ELECTRE conduit ainsi, souvent, à utiliser différents seuils de concordance avant de proposer le surclassement d'une opération par une autre. Ceci exige donc plusieurs itérations qui se répercutent directement sur le temps de réponse du décideur.

La particularité de notre problème est relative à la contrainte de décision temps réel. Il est donc difficile d'accepter une méthode qui ne répondrait pas à cette contrainte.

Notre démarche a pour but d'accélérer le processus de décision en minimisant le nombre d'itérations et donc le temps de calcul. Elle se caractérise par deux approches.

4.5.1 PREMIERE APPROCHE

Elle consiste en l'application d'ELECTRE telle qu'elle est définie dans § 4.3.2, avec la fixation d'un seul seuil de concordance c , et les seuils de discordance relatifs à chaque critère.

La décision doit être prise en une seule itération s'il y a équivalence ou incomparabilité ; seuls des ajustements du seuil de concordance peuvent lever l'ambiguïté, or cela implique plus de temps de calcul.

C'est pourquoi dans ces deux cas, nous considérons que le choix des opérations est indifférent (celle qui demandera moins de tests sera choisie).

i	j	Conclusion	Choix
0	0	Incomparables	Indifférent
1	1	Equivalentes	Indifférent

4.5.2 DEUXIEME APPROCHE

Une autre façon d'accélérer les calculs est de ne prendre en compte la condition de non discordance que dans le cas où les indices de concordances ont des valeurs voisines.

Une manière de calculer ces indices est la suivante :

$$e_{ij} = \frac{P^+ + P^-}{P}$$

$$e_{ji} = \frac{P^- + P^=}{P}$$

($P = P^+ + P^- + P^=$, pondérations obtenues pour l'assertion i S j)

Soit C_r , l'écart au delà duquel, les seuils de discordance interviennent dans la validation du surclassement.

Les différentes situations auxquelles on peut aboutir, sont résumées dans le tableau suivant :

$e_{ij} - e_{ji}$	Discordance	Conclusion	Choix
$\geq Cr$	\emptyset	surclass. accepté	i
$< Cr$	0	surclass. accepté	i
$< Cr$	1	surclass. refusé : j S i ou incompatibilité	j j (indif.)

(" \emptyset " : non pris en compte, "1" : conditions non vérifiées, "0" : conditions vérifiées).

Dans le cas où le surclassement de j par i est refusé, il existe une alternative :

- l'opération j surclasse i

ou

- l'opération j ne surclasse pas i, et donc il y a incomparabilité. Le choix est alors indifférent.

Ainsi l'opération j satisfait les deux cas. Il n'y a donc qu'un seul test à faire pour décider, d'où un gain en temps de calcul.

Remarque : Dans le cas où $e_{ij} - e_{ji} < 0$, on intervertit les indices pour avoir $e_{ji} - e_{ij} \geq 0$

* Ecart limite :

$$\text{Soit } d_{ij} = e_{ij} - e_{ji} = \frac{P^+ - P^-}{P}$$

$$\text{d'où, } d_{ij \min} = 0 \quad (P^+ = P^-)$$

$$d_{ij \max} = 1 \quad (P^+ = P^- = 0), \text{ surclassement absolu}$$

L'intervention des conditions de discordance est réservée au cas où les opérations à comparer sont "proches" l'une de l'autre, relativement aux indices de concordance.

La fixation de la valeur de l'écart limite C_r est aussi subjective. Nous l'avons fixé à :

$$C_r = 0,2 d_{ij \max} = 0,2$$

(en estimant que 20 % de l'écart maximum est une "distance" limite).

4.5.3 SEUILS DE DISCORDANCE

Ils n'interviennent que lorsque les indices de concordance ont des valeurs proches. Ils permettent l'affirmation ou l'infirmité du surclassement.

Dans notre démarche, l'infirmité du surclassement correspond à l'affirmation de son contraire (voir tableau précédent). Ceci afin de permettre une décision en une seule itération et donc de réduire le temps de calcul.

La détermination de ces seuils dépend, d'une part, de l'échelle d'évaluation du critère concerné, et d'autre part, de la proportion de cette échelle devant correspondre au seuil de discordance.

Tous les critères que nous avons retenus dans notre problème sont quantifiables. L'obtention de l'échelle d'évaluation est donc relativement facile. Par contre, la fixation du seuil de discordance reste subjective et donc complexe.

Il est parfois certains critères pour lesquels l'utilité même de ce seuil n'apparaît pas.

a) $J_1(j)$: Nombre de successeurs immédiats de j

L'échelle d'évaluation de ce critère est donné par l'intervalle suivant : $[1, N_{\max}]$

$$\text{où } N_{\max} = \max_j |S(j)|$$

Il est évident que si N_{\max} est de faible valeur, il n'y a pas lieu de prendre en compte un seuil de discordance puisque l'écart sur le nombre de successeurs immédiats entre deux opérations quelconques ne sera jamais important ($< N_{\max}$).



Si par contre N_{\max} est de forte valeur, cet écart risque d'être élevé et justifierait ainsi la discordance.

Nous avons choisi, dans ce cas, de fixer le seuil de discordance à :

$$d(J_1) = 2/3 \cdot N_{\max}$$

Si nous supposons, par exemple, que pour la plupart des produits à réaliser, la valeur de N_{\max} se situe autour de 10, le seuil sera fixé à 6.

b) $J_2(j)$: Nombre de successeurs de j dont la ressource propre est disponible

Ce paramètre est fonction de la présence des ressources propres des opérations successeurs de l'opération j . Il est donc variable puisque les ressources qui sont dans le magasin sont supposées être de disponibilité aléatoire.

Il nous semble alors injustifié de refuser un surclassement à partir d'informations pouvant être remises en cause par la suite.

Ainsi, nous ne proposons pas de seuil de discordance et il n'y aura donc pas de vérifications des conditions de discordance, relativement à ce critère.

c) $J_3(j)$: Coefficient de déblocage

Nous rappelons que :

$$J_3(j) = \max_{k \in S(j)} (D_t(k/j))$$

et

$$D_t(k/j) = 1 / |P_t(k)|$$

Ce critère prend donc ses valeurs dans l'ensemble suivant :

$$\{ 1, 1/2, 1/3, \dots, 1/P(k) \}$$

d'où : $0 < J_3(j) \leq 1$

Si $d(j_3)$ est le seuil de discordance du critère J_3 , on en déduit que :

$$0 < d(J_3) < 1$$

puisque $d(J_3) = J_3(i) - J_3(j) > 0$, lorsque $J_3 \in J_{ji}^-$

et $d(J_3)_{\max} = 1 - 1/P(k) < 1$

Nous avons choisi un seuil de discordance tel que le surclassement est refusé lorsque l'opération dominée a un coefficient de déblocage maximal, c'est-à-dire égal à l'unité.

Il nous semble sensé de refuser un surclassement d'une opération qui après sa réalisation débloque au moins une opération par une autre qui n'en débloque pas.

Pour ce faire, nous avons fixé la valeur du seuil de discordance à :

$$d(J_3) = 0,5$$

d) $J_4(j)$: Temps de réalisation de l'opération j

d.1 Temps de réalisation indépendant des robots

Chaque opération a un temps de réalisation fixé quelque soit le robot qui l'exécute. Il nous semble alors injustifié de refuser un surclassement entre deux opérations pour la simple raison que l'écart entre leurs durées d'exécution dépasse un certain seuil, d'autant plus qu'elles doivent être obligatoirement réalisées.

d.2 Temps de réalisation dépendant des robots

Dans ce cas, on associe à chaque opération j , un écart relatif e_j (§ 422/d.2) entre le temps minimum absolu de réalisation de j et le temps minimum associé au robot le plus performant parmi les robots libres à l'instant de l'affectation.

L'intérêt d'un seuil de discordance, dans ce cas, est évident puisque il permet le refus du surclassement lorsque l'opération dominante est réalisée avec un robot trop lent par rapport à celui qui a été désigné pour l'opération dominée.

Nous avons fixé le seuil de discordance à 30 % dans ce cas.

$$d(J_4) = 0,3$$

e) $J_5(j)$: Temps d'attente moyen d'une ressource propre par l'opération j

Le refus de surclassement ou la discordance entre deux opérations i et j apparaît dès que la différence entre leurs temps d'attente moyen, dépasse un seuil fixé, $d(J_5)$.

Seulement dans ce cas, cette différence ne peut pas traduire à elle seule la discordance. Il est important aussi de connaître la valeur absolue de ces temps.

En effet, il est inutile de comparer deux opérations dont les ressources propres ne sont pas considérées comme rares, c'est-à-dire de temps d'attente faible.

Mais si nous considérons que " t_{mi} faible" correspond à $t_{mi} < d(J_5)$, le problème ne se pose plus.

Nous n'aurons dans ce cas qu'à vérifier si :

- $t_{mj} - t_{mi} < d(J_5)$ Alors le surclassement est accepté.
- $t_{mj} - t_{mi} \geq d(J_5)$ Alors le surclassement est refusé.

La fixation de $d(J_5)$ est délicate. Elle dépend de l'appréciation de l'opérateur.

Pour notre part, nous avons estimé ce seuil à :

$$d(J_5) = 0,3 \cdot \max_j (t_{mj})$$

Après la détermination des seuils de discordance et de l'écart entre les seuils de concordance, il reste l'estimation des pondérations relatives des critères pour pouvoir appliquer la méthode proposée. C'est ce que nous développons maintenant.

4.5.4 PONDERATION DES CRITERES

C'est une opération essentielle dans la méthode d'ordonnement-affectation dynamique proposée puisque celle-ci est très sensible aux variations des valeurs des poids affectés aux critères.

La détermination des pondérations est, dans ce cas, plus complexe que pour les méthodes d'agrégation de critères faisant, par exemple, appel à une fonction d'utilité additive [LAG. 83]. En effet, il n'existe pas, actuelle -

ment, de méthode mathématique générale permettant d'effectuer la désagrégation (i.e partant d'un ensemble d'ordonnements donné, déduire l'ensemble des pondérations qui permettent de les retrouver).

Néanmoins, par apprentissage, on peut obtenir des informations intéressantes sur l'importance relative des critères : l'écart entre les valeurs des poids nous semble plus significatif que les valeurs des poids eux-mêmes.

Notre démarche est la suivante :

Elle consiste, pour un ensemble donné de valeurs des poids, à calculer la performance globale du système associée à toutes les combinaisons possibles de ces pondérations.

Cette étude exhaustive est faite hors-ligne, à l'aide d'un programme de simulation du système.

Parmi toutes ces combinaisons, nous ne retenons que celles qui engendrent la meilleure performance.

Afin de pouvoir juger de l'efficacité de la méthode et donc de comparer le résultat obtenu à la solution optimale calculée (lorsque cela est possible), nous avons été amené à considérer certaines hypothèses de travail :

- les ressources propres associées aux opérations à réaliser sont disponibles dans la cellule (dans le cas où les pièces arrivent d'une manière aléatoire, il est difficile de calculer la solution optimale et donc de juger la méthode) ;
- il n'y a pas de panne sur les robots ;
- il n'y a pas de conflit d'accès à la zone commune entre les robots ;
- les temps d'exécution des opérations ne dépendent pas des robots.

Les exemples que nous présentons sont artificiels.

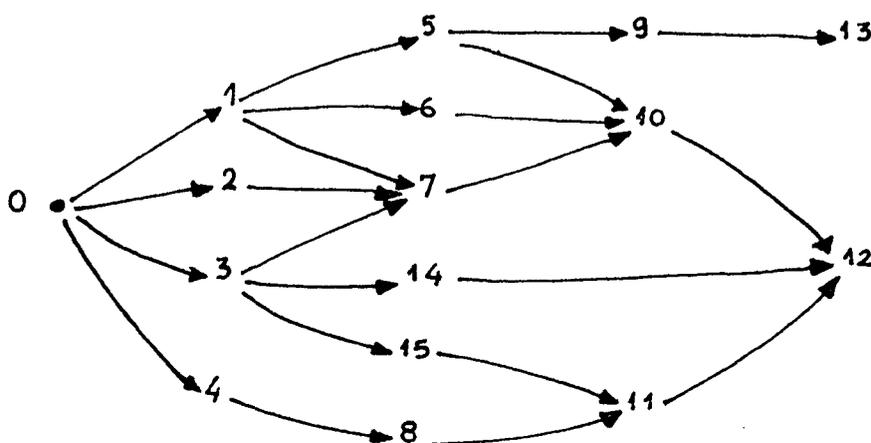
4.5.5 EXEMPLES

Soit une gamme de fabrication composée de 16 opérations. Le tableau des temps d'exécution est donné par :

Op.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R1	-	2	3	-	-	3	5	-	2	-	2	3	-	3	2	-
R2	4	-	3	2	4	-	5	2	-	2	-	3	2	-	2	3

"-" non réalisable par le robot considéré).

Le graphe associé est donné par :



Les valeurs des seuils de discordance et de l'écart entre les indices de concordance sont celles définies auparavant (§ 4.5.2 et § 4.5.3).

Soit le jeu de poids suivant avec un écart constant entre les valeurs, égal à 5.

$$LP1 = (1 \ 6 \ 11 \ 16 \ 21)$$

Après le calcul de l'ordonnement pour les différentes combinaisons correspondant aux permutations dans LP1, seules les suivantes ont conduit à un ordonnancement de durée minimale :

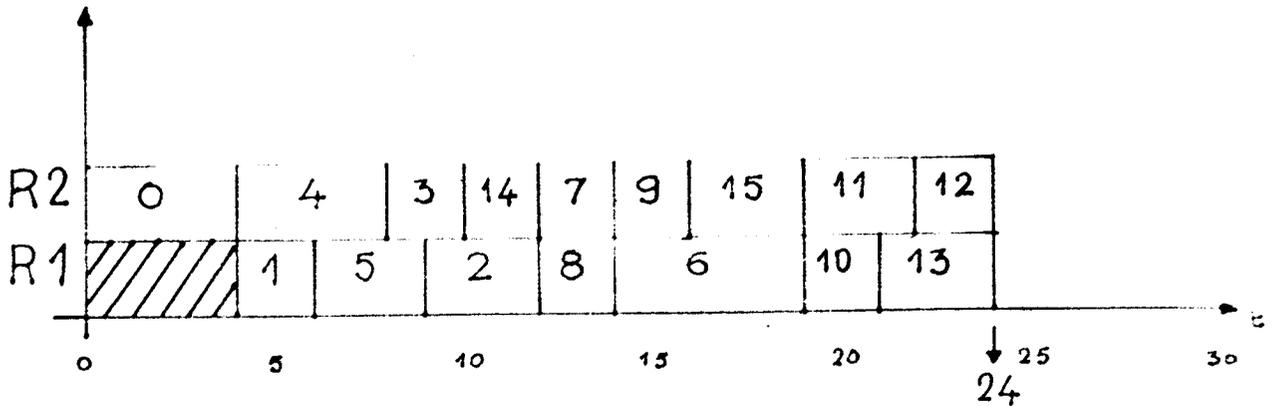
$$\text{Comb 1} = (6 \ 1 \ 16 \ 11 \ 21)$$

$$\text{Comb 2} = (11 \ 1 \ 16 \ 6 \ 21)$$

$$\text{Comb 3} = (1 \ 6 \ 16 \ 11 \ 21)$$

Cette durée est de 24 unités de temps.

Le diagramme de GANTT correspondant, est le suivant :



On constate que dans ce cas, le temps minimal correspond au temps optimal.

Recommençons la même démarche avec un jeu de poids proportionnel au précédent, et un écart entre les valeurs des poids égal à 10.

Soit, $LP2 = (2 \ 12 \ 22 \ 32 \ 42)$

On vérifie que les combinaisons qui donnent l'ordonnancement optimal sont les mêmes aux valeurs des poids près que celles du jeu de poids précédent, LP1.

En effet, nous obtenons :

Comb 11 = (12 2 32 22 42) = 2 x Comb 1

Comb 22 = (22 2 32 12 42) = 2 x Comb 2

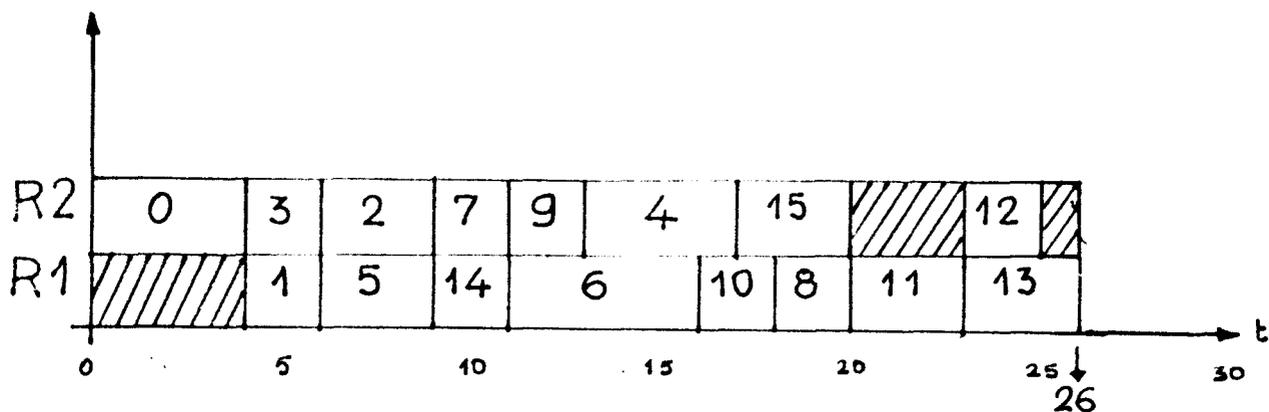
Comb 33 = (2 12 32 22 42) = 2 x Comb 3

Si par contre, nous prenons un jeu de poids où l'écart entre les valeurs des poids est faible, le résultat optimal n'est pas atteint.

Soit, $LP3 = (1 \ 2 \ 3 \ 4 \ 5)$

Le meilleur ordonnancement dans ce cas a une durée de 26 unités de temps.

Son diagramme de GANTT est le suivant :



A partir d'une analyse succincte de ces résultats, on peut tirer une première conclusion.

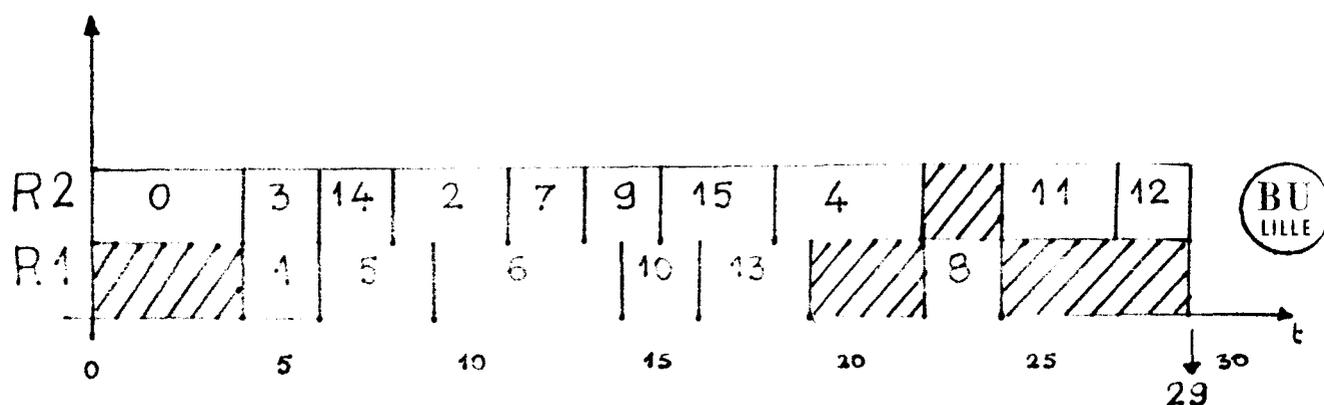
Les critères doivent être différenciés. L'écart entre les valeurs des poids qui leur sont attribués doit être significatif. Mais il peut y avoir pour certaines configurations, des critères ayant une importance équivalente.

Dans notre exemple avec LP2, nous remarquons que J4 et J1 peuvent prendre leurs valeurs dans l'intervalle [12 22] par exemple, sans que les performances de la cellule n'en souffrent.

D'un autre côté, nous avons relevé que la durée maximale de l'ordonnancement est de 29 unités de temps. Une, parmi les combinaisons qui donnent ce résultat, est la suivante :

(6 11 16 21 1) ou (12 22 32 42 2)

Le diagramme de GANTT correspondant, est le suivant :



La conclusion que l'on peut tirer de ce résultat est la sensibilité de la méthode par rapport aux pondérations. Ainsi, il suffit de préférer un critère à un autre pour que les performances se dégradent. Ceci démontre

l'importance et le besoin d'une méthode qui permettrait d'adapter ces pondérations aux différentes situations de fonctionnement de la cellule.

5 - CONCLUSION

Nous avons posé le problème de l'ordonnancement-affectation dans la cellule, comme un problème d'optimisation formulé à l'aide de la programmation dynamique.

L'apparition aléatoire des opérations exécutables rend complexe sa résolution, d'autant plus que nous avons montré qu'une approche stochastique n'était pas envisageable dans un cadre réaliste.

Nous avons donc proposé une méthode de résolution heuristique multicritères, basée sur la notion de surclassement. Nous avons défini un certain nombre de critères qui influencent directement ou indirectement le nombre d'opérations exécutables que nous voudrions accroître.

Les résultats obtenus par l'application de cette méthode, nous semblent très intéressants, puisque l'on arrive, avec un bon choix des pondérations, à obtenir des solutions optimales.

Le choix des pondérations est essentiel dans cette méthode. La part du flou qui intervient dans la fixation de leurs valeurs, nous amène à penser que l'intégration de l'opérateur dans la procédure de réajustement des pondérations, peut être nécessaire, dans le cas bien évidemment, où une dégradation des performances aurait été constatée.

Nous pensons qu'il y a là une voie à approfondir et qui pourrait conduire à l'élaboration d'une méthode d'apprentissage des pondérations en ligne.

CONCLUSION GENERALE

Ce travail concerne l'étude du problème de conduite d'une cellule de production flexible utilisant la coopération entre robots. Il s'agit de réaliser un produit selon une gamme de fabrication donnée. Chaque opération de la gamme nécessite une ressource propre (pièce) et des ressources partageables (robot, espace de travail).

Après la présentation des caractéristiques principales d'une telle cellule, une modélisation par graphes d'états et de transitions est proposée. Compte tenu de cette dernière, un système de conduite organisé de façon modulaire est décrit. Un programme de simulation d'une partie de ce système a été écrit en MULISP sous CP/M 80.

Le problème posé se caractérise particulièrement par la prise en compte de la dynamique de la cellule et notamment par l'apparition aléatoire des opérations exécutables. C'est un problème d'ordonnancement-affectation dynamique, dont l'objectif est la minimisation du temps de réalisation de l'ensemble des opérations à exécuter. Or, ceci conduit à la minimisation du temps d'oisiveté des robots et donc à la maximisation du nombre d'opérations exécutables à tout instant.

Le choix d'une opération qui après son exécution déblocquera le plus grand nombre d'opérations exécutables, n'est pas évident. En effet, les opérations sont caractérisées par plusieurs paramètres. Le problème original monocritère se transforme alors en un problème multicritères.

Pour le résoudre, nous proposons une approche fondée sur la notion de surclassement utilisée dans les méthodes multicritères ELECTRE, développées par B. ROY.

Les résultats que nous avons obtenus sur quelques exemples artificiels sont encourageants, puisque dans la plupart des cas, on arrive à obtenir la solution optimale pour une distribution d'arrivée de ressources propres donnée.

Si l'importance relative des critères est établie, on peut penser que le résultat sera globalement satisfaisant quelle que soit la distribution d'arrivée des ressources propres.

Le point critique de ce type de méthode est la détermination des pondérations des critères choisis. L'approche qui nous semble être une voie

à approfondir consisterait à l'intégration de l'opérateur dans la boucle d'apprentissage de ces poids.

Il est évident que dans le cadre de ce travail, plusieurs problèmes restent posés et méritent d'être approfondis, notamment :

- l'élaboration d'une méthode d'apprentissage des pondérations ;
- la définition précise d'une opération nécessitant plusieurs robots pour sa réalisation, en utilisant les réseaux de Pétri ou le Grafcet ;
- la structure informatique à mettre en œuvre sur le site expérimental, pour s'affranchir des contraintes actuelles dues au fait que LM est monotâche ;
- l'étude de la génération de trajectoires anti-collision ;
- la définition du module "diagnostic".

ANNEXES

PRESENTATION DU PROGRAMME DE SIMULATION

Le programme de simulation de la cellule de production flexible a été écrit sur micro-ordinateur en MULISP sous CP/M 80.

Il repose sur les hypothèses suivantes :

- la décision d'affectation d'une opération n'est prise que lorsqu'un robot sachant la réaliser est libre
- chaque opération de la gamme correspond à une seule action d'un robot
- les paramètres associés à une opération se limitent aux :
 - * temps de réalisation
 - * type de la ressource propre associée
 - * emplacement final de cette ressource dans la zone de travail.
- les temps d'exécution des opérations sont indépendants des robots qui les exécutent
- les temps de stockage d'une pièce ne sont pas pris en compte
- plusieurs opérations d'une gamme peuvent avoir besoin de la même ressource propre. Ainsi lorsque cette dernière n'est pas encore consommée, elle est provisoirement affectée à toutes les opérations qui lui sont associées. Dès qu'elle sera définitivement allouée, une remise à jour de l'état des opérations concernées est effectuée.
- la zone de travail est composée de trois parties :
 - * deux zones propres à chacun des robots
 - * une zone commune.

C'est une simulation à événements discrets. A chaque unité de temps, on teste s'il y a occurrence d'un événement pour le traiter.

Ce programme est organisé autour d'un certain nombre de fonctions dont nous présentons les principales.

1 - Initialisation

C'est la fonction qui initialise toutes les listes intervenant dans le programme.

LRL	<---- NIL, Liste des robots libres
PAR-ROB	<---- NIL, Liste des aptitudes des robots
PAR-MONT	<---- NIL, Liste décrivant la gamme de fabrication
PAR-TACH	<---- NIL, Liste des opérations paramétrées
PAR-TACH-EVAL	<---- NIL, Liste des opérations évaluées par rapport à chacun des critères utilisés
LTB	<---- NIL, Liste des opérations bloquées (non-exécutables)
LTPE (a)	<---- NIL, Liste des opérations pré-exécutables (a)
LTPE (r)	<---- NIL, Liste des opérations pré-exécutables (r)
LTE	<---- NIL, Liste des opérations exécutables
LTEC	<---- NIL, Liste des opérations en cours
LTF	<---- NIL, Liste des opérations finies
LPM	<---- NIL, Liste des pièces dans le magasin
LPS	<---- NIL, Liste des pièces en stock intermédiaire

2 - Communication homme/machine

Cette fonction permet d'une manière interactive, l'acquisition des données de l'opérateur, concernant la gamme de fabrication, notamment :

- les relations de succession entre les opérations sous la forme :

(T_0 T_1 T_2) ; l'opération T_0 a pour successeurs T_1 et T_2

- les paramètres associés à chaque opération :

(T1 (Temps d'exécution type de pièce position finale))

- les aptitudes de chacun des robots :

(R1 (T0) (T1) (T2) ... (Tk))

Elle permet l'affichage à chaque unité de temps, de l'état global de la cellule.

3 - Identification

C'est une fonction dont le rôle est d'identifier les pièces qui entrent dans le magasin et de faire la mise à jour en conséquence.

L'entrée des pièces dans le magasin se fait par la lecture d'une liste de pièces, introduite par l'opérateur, et "décrementée" après chaque lecture.

4 - Pré-exécutabilité

Elle fait évoluer les opérations de la gamme de l'état non-exécutable à l'état pré-exécutable lorsque :

- leurs antécédents sont réalisés

ou

- leurs ressources propres deviennent disponibles.

Elle met à jour les listes des opérations pré-exécutables.

5 - Exécutabilité

Son rôle est de faire évoluer une opération de l'état pré-exécutable à l'état exécutable, tenant compte de l'état courant de la cellule. Toutes les opérations non encore réalisées dont les antécédents sont exécutés et les ressources propres disponibles, sont mises dans la liste des opérations exécutables.

6 - Affectation

C'est la fonction de décision. Elle permet le choix, parmi les opérations exécutables, de la meilleure par rapport aux critères retenus et la méthode de décision utilisée. Dans notre cas, c'est la méthode multicritères présentée dans le chapitre IV.

Ainsi, à chaque opération, est associée l'évaluation de tous les critères pris en compte.

Seules les opérations exécutables dont au moins un robot sachant les réaliser est libre, sont pris en compte par cette fonction.

Lorsque plusieurs robots sont libres en même temps, ils sont ordonnés selon le nombre croissant d'opérations exécutables qu'ils savent réaliser.

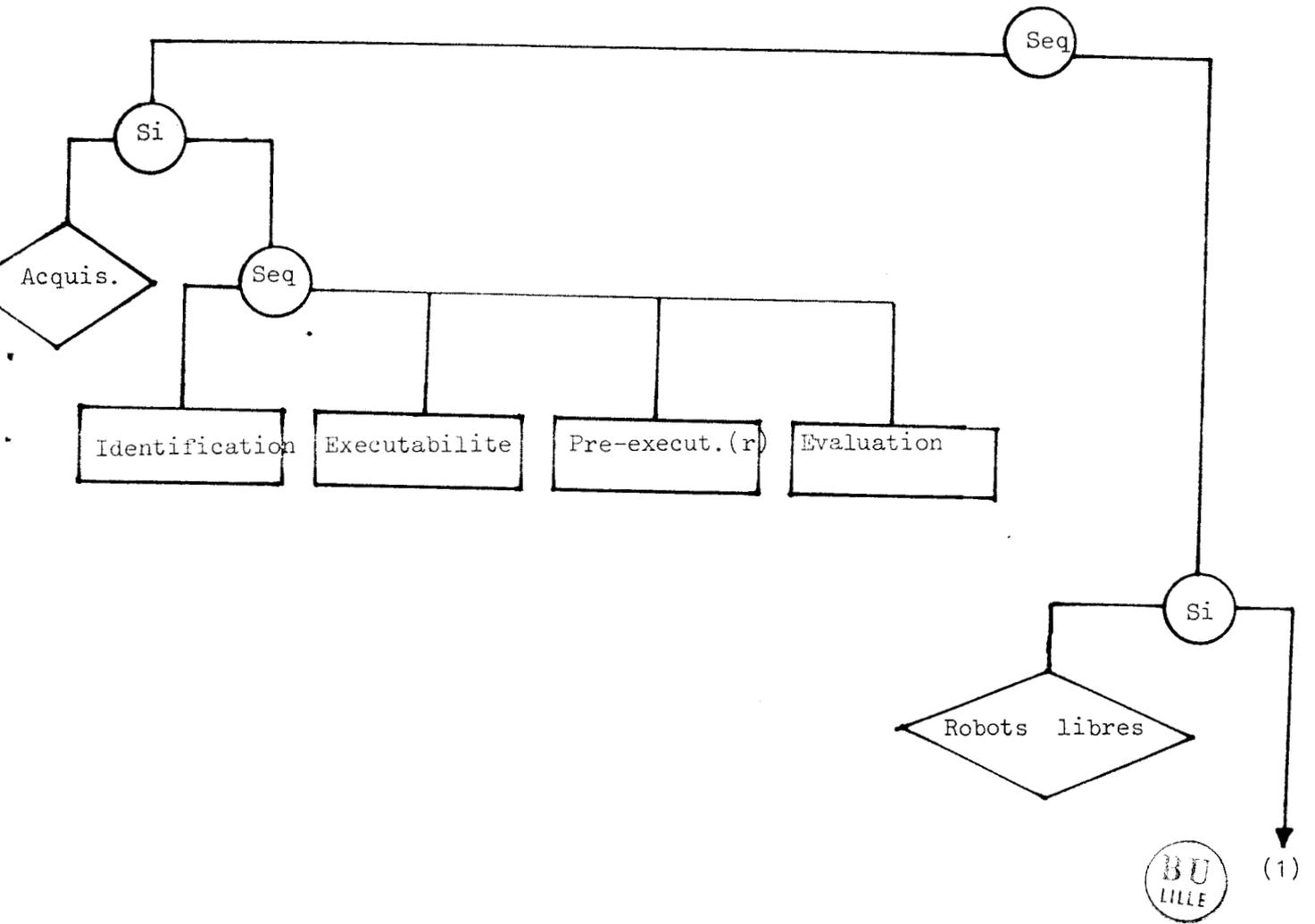
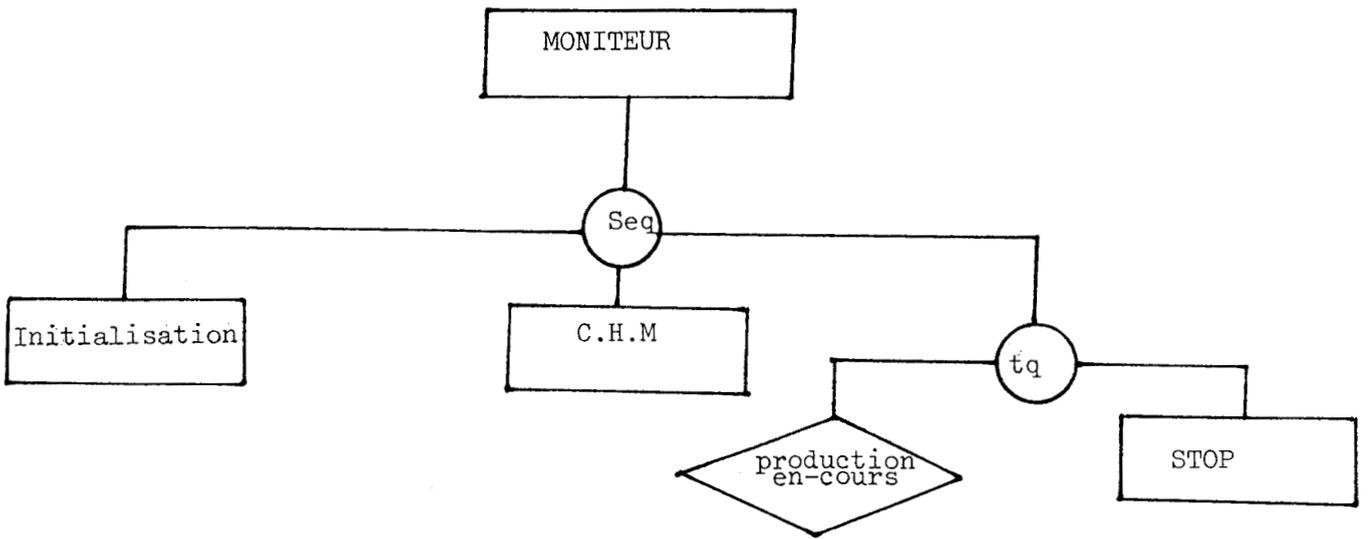
7 - Evaluation

Après chaque changement d'état d'une opération, les valeurs des critères qui lui sont associés, sont réévaluées par cette fonction, et une mise à jour est effectuée.

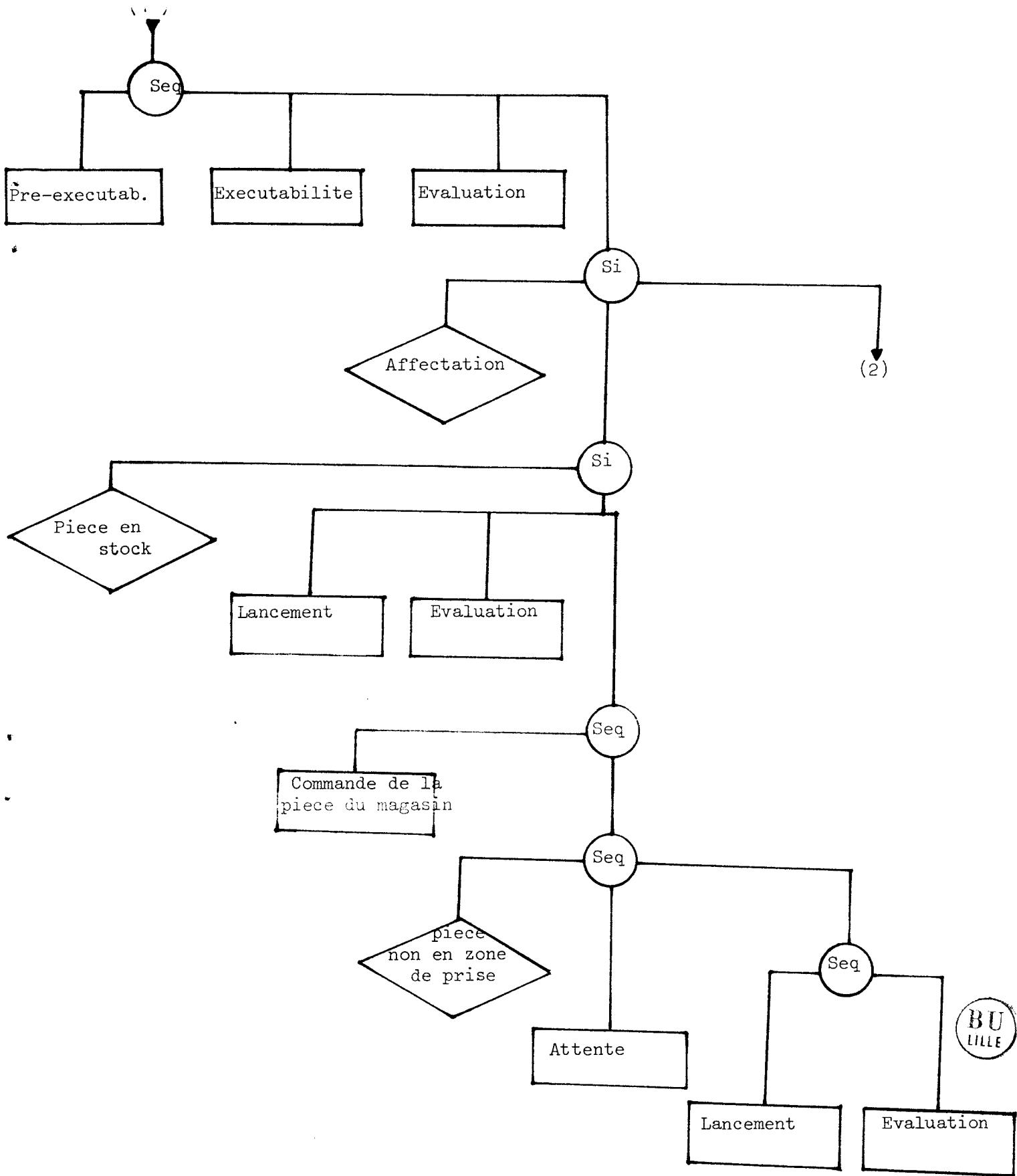
8 - Stockage

Lorsqu'un robot est libre et qu'il n'y a pas d'opérations exécutables qu'il sache réaliser, on l'utilise pour le stockage d'une pièce du magasin en stock intermédiaire (lorsque cela est possible et nécessaire).

Le choix de la pièce obéit aux mêmes règles que le choix d'une opération à exécuter.

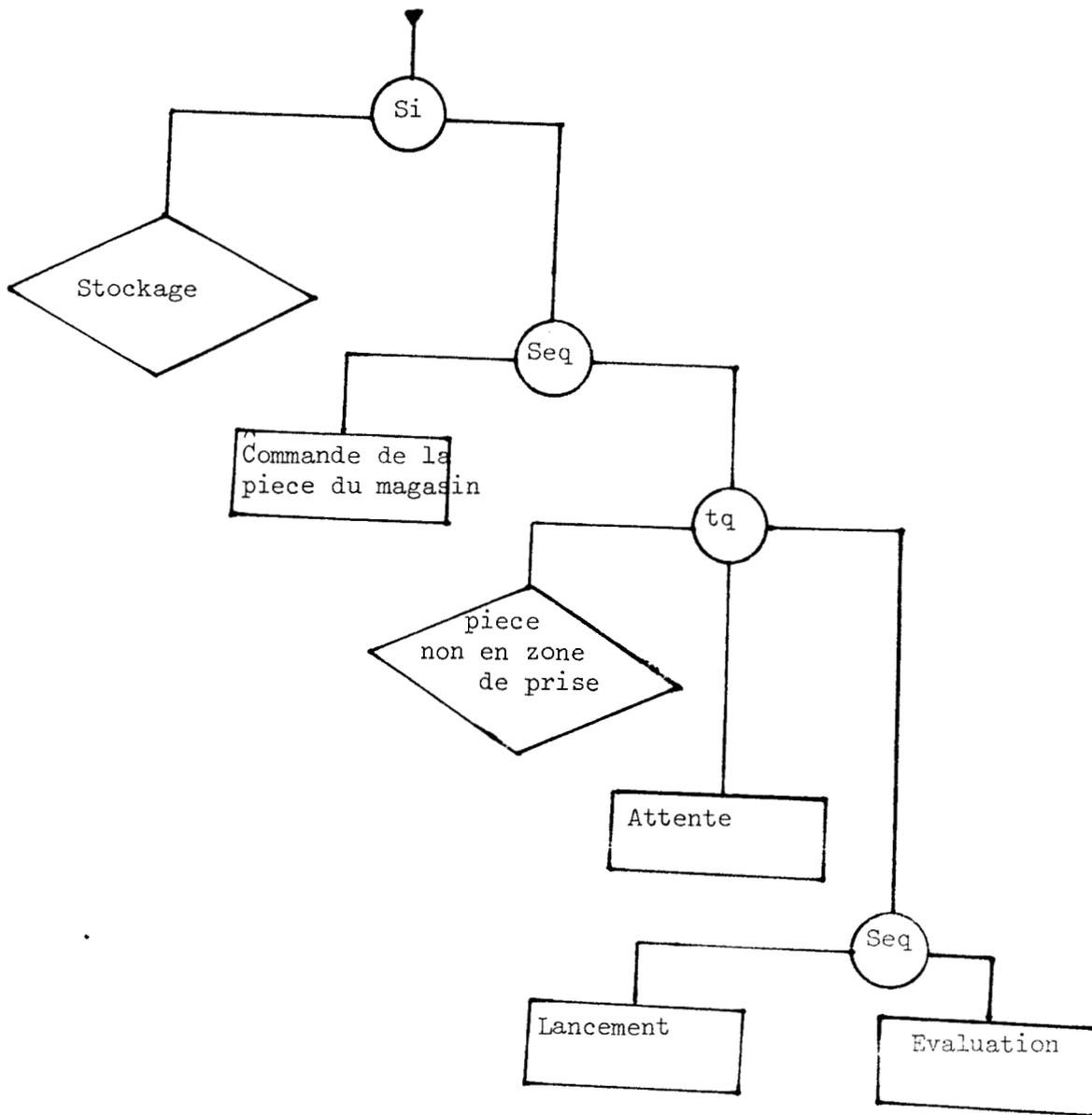


BU
LILLE



BU
LILLE

(2)



PROGRAMME IMPLANTE SUR LE SITE

Le programme implanté sur le site expérimental a été écrit en L.M. Le principal inconvénient de ce langage pour cette manipulation est qu'il est "monotâche". Les robots sont donc activés séquentiellement.

La structure de ce programme correspond en partie à celle du programme de simulation.

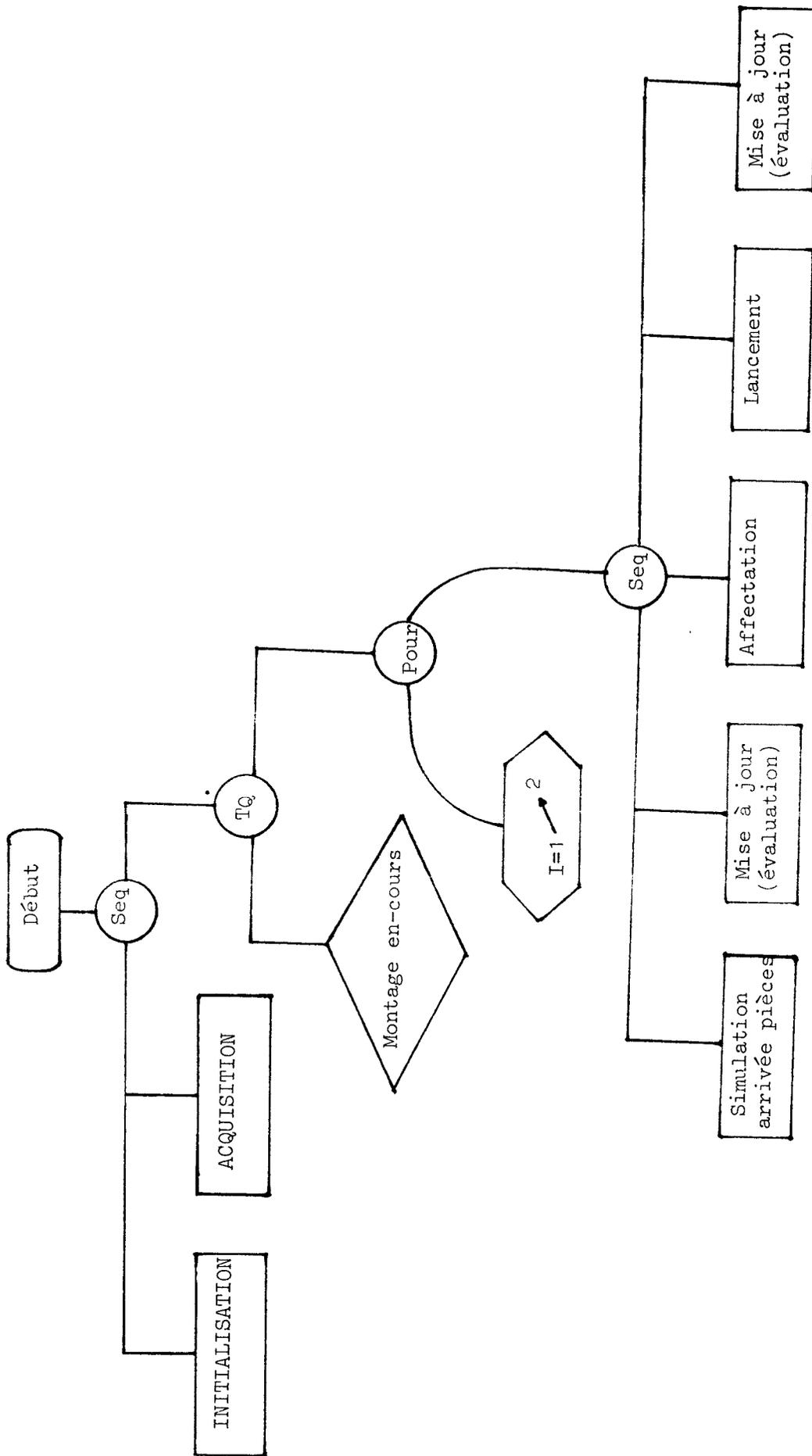
Les différences sont les suivantes :

- les opérations à réaliser se limitent à prendre et à poser (pick and place) ;
- le critère de choix des opérations exécutables est le nombre de successeurs immédiats.

La prise en compte de tous les paramètres présentés dans (IV 4.3.), exigerait un temps de calcul important, ce qui introduirait une longue attente entre deux réalisations d'opérations par les robots.

Ainsi, la manipulation actuelle est très dégradée. La seule possibilité de parallélisme est offerte par l'instruction L.M., "sans attente". Elle permet de lancer un robot juste après un autre, sans attendre la fin de l'action de ce dernier.

L'arbre programmatique est le suivant :



BIBLIOGRAPHIE

- ALA. 83 Un environnement LISP pour la mise en œuvre de systèmes complexes en robotique.
R. ALAMI
Th. D.I. LAAS Toulouse - 1983
- AMM. 85 Scheduling models for aiding real time FMS control
Jane C. AMMONS
SMC Conference IEEE 1985 TUCSON (USA)
- ARA. 84 Rapport A.R.A.
Pôle Ateliers flexibles - 1984
- ✓BEL. 85 Modélisation et simulation de systèmes automatisés de production
G. BEL, D. DUBOIS
A.P.I.I. - 1985 - (RAIRO)
- ✓BEL. 81 Outil de conception d'ateliers flexibles d'usinage
G. BEL, J.B. CAVAILLE, D. DUBOIS
Congrès AFCET - 1981
- BER. 83 Contribution à la conception de structures logicielles pour le pilotage d'Ateliers
C. BERNARD
Th. D.I Bordeaux 1 - 1983
- BERG. 83 Graphes
C. BERGE
3^{ème} Edition - Gauthier Villars - 1983
- BERN. 71 Douze méthodes d'analyse multicritères
G. BERNARD, M.L. BESSON
R.I.R.O. - Vol. 3 - 1971
- ✓BOL. 82 L'analyse multicritère en perspective d'une théorie générale de la gestion des entreprises modernes
G. BOLDUR
RAIRO - R.O. - Vol. 16 - n° 1 - Fev. 82

- BON. 85 Les ateliers flexibles de production
R. BONETTO
Hermes Publishing - 1985
- BOU. 84 "Autonomous Manufacturing : Automating the Job-shop"
D.A. BOURNE, M.S. Fox
Computer - Sept 84 - pp. 76-86
- BOU. 75 Analyse de données multidimensionnelles
J.M. BOUROCHE et P. BERTIER
Presses Universitaires de France - 1975
- BUB. 85 The control in the assembly system with two robots"
Z. BUBNICKI, G. REYMAN, M. STAROSWIECKI, M. DJEGHABA
4th International Conference on Systems Engineering
COVENTRY (ENGLAND) - Sept 1985
- /BUZ. 82 "Optimal" Operating rules for automated Manufacturing
Systems
J.A. BUZACOTT - IEEE Transactions on Automatic Control
Vol. A. 27 n° 1 - Feb. 1982
- /CAR. 82 Un domaine très ouvert : les problèmes d'ordonnement
J. CARLIER, P. CHRETIENNE
RAIRO - Vol. 16 - n° 3 - Août 1983
- COU. 79 Etude de l'existence de solutions pour certains problèmes
d'ordonnement
C. COUZINET-MERCE
Th. D.I., Université P. Sabatier - Toulouse - 1979
- DAL. 82 Simulation of a flexible manufacturing System : case study of
the citroen factory, MEUDON
Y. DALLERY, B. DESCOTTES-GENON, R. BONETTO
A.P.M.S. - 1982 - IFIP CONF. BORDEAUX
- /DEG. 85 Etude d'un système de conduite d'un atelier industriel à flot
unique
P. DEGRAEVE
Th D.I. - U.S.T. Lille - 1985
- /DES. 81 Représentation et exploitation de connaissances "expertes" en
génération de plan d'actions
Application à la conception de gamme d'usinage
Y. DESCOTTE
Th. D.I., I.N.P.G. - Décembre 1981

- DJE. 85 A Management system for a flexible assembly cell using robot cooperation
M. DJEGHABA, M. STAROSWIECKI
S.M.C. Conference - Tucson (U.S.A.) I.E.E.E. - 1985
- DOU. 83 La gestion de Production assistée par ordinateur
G. DOUMEINGTS, D. BREUIL, L. PUN
Hermes Publishing (France) - 1983
- DUP. 82 A Survey of flexible Manufacturing Systems
C. DUPONT-GATELMAND
Journal of Manufacturing System - 1982
- / DUP. 83 Le concept d'atelier flexible : une approche industrielle et une approche par les modeles
C. DUPONT
Journées A.R.A. - Pôle Ateliers Flexibles - 1984
- ERS. 82 Applying new dominance concepts to job schedule optimization
J. ERSCHLER, G. FONTAN, C. MERCE, F. ROUBELLAT
E.J.O.R., North Holland, n° 11, 1982
- ERS. 85 Un nouveau concept de dominance pour l'ordonnancement de travaux sur une machine
J. ERSCHLER, G. FONTAN, C. MERCE
RAIRO, R.O., Vol. 19, n° 1, février 1985, pp. 15-26
- FIN. 85 Network approach to modelling of flexible manufacturing modules and cells.
G. FINKE, A. KUSIAK
A.P.I.I. - 1985 - 19 - 359 - 370
- FON. 80 Notion de dominance et son application à l'étude de certains problèmes d'ordonnancement
G. FONTAN
Thèse de Docteur es-sciences - Université P. Sabatier - 1980
- GUI. 77 Méthodes Multidimensionnelles : Analyse de données et choix à critères multiples.
G.L. GUIGOU
Dunod - 1977
- HAB. 81 Intégration du contrôle d'un robot par un opérateur humain.
A. HABCHI
Th. D.I., U.S.T.L., 1981

- HO. 83 A New approach to the analysis of discrete event dynamic systems.
Y.C. HO and C. CASSANDRAS
Automatica - Vol. 19 - n° 2 - 1983
- / KAL. 85 Conduite décentralisée coordonnée d'atelier
G. KALLEL, X. PELLET, Z. BINDER
A.P.I.I. - 1985 - 19 - pp. 371-387
- / KUS. 85 Planning of flexible manufacturing systems
Andrew KUSIAK
Robotica - Vol. 3 - 1985 - pp. 229-232
- LAG. 78 De la logique d'agrégation des critères à une logique d'agrégation desagrégation des préférences et de jugements
E. JACQUET-LAGREZE - Cahiers des LAMSADE - sep. 78
- LAG 80 Aide à la décision multicritères et systèmes relationnels de préférence
E. JACQUET-LAGREZE, B. ROY
Cahiers du LAMSADE - 1980
- LAG. 83 Méthodes de décision multicritères - Monographie de l'AFCE
E. JACQUET-LAGREZE et J. SISKOS
EDITION HOMMES ET TECHNIQUES - 1983
- LAN. 83 Contribution à l'étude des techniques d'intelligence Artificielle - Réalisation d'un système robotique expérimental à grande autonomie.
A. LANUSSE
Th. D.I. LAAS - TOULOUSE - 1983
- / LEV. 82 Lancement périodique de produits dans un atelier flexible
D. LEVEQUE
Thèse D.I. - INSA TOULOUSE - 1982
- LOZ. 83 Robot programming
T. LOZANO-PEREZ
Proceedings of the I.E.E.E., Vol. 71 - n° 7 - July 1983
- MART 78 Techniques et applications de la recherche opérationnelle
A. MARTEL
Editions GAETAN-MOZIN - 1978

- MAZ. 81 Réalisation d'un support experimental de recherche pour le projet robotique PANDORE
Définition et implantation du langage LM
E. MAZER
Th. 3^{ème} cycle - I.N.P.G. - Janvier 81
- MIR. 84 Le langage LM - manuel de référence
J.F. MIRIBEL - E. MAZER
Editions Cepadues - 1984
- MOU. 81 Les méthodes multicritères comme outil d'aide à la décision dans les problèmes de transport.
MOURMOURIS
Th. 3^{ème} cycle, I.N.P.G. - Janvier 81
- NIC. 79 Contribution à l'étude d'un mécanisme de communication pour un réseau informatique local et son rôle dans la sureté de fonctionnement du système.
Pantelis NICOLOPOULOS
I.N.P.G., Th. D.I., 1979
- OH. 82 Graphic modeling by petri-nets for the production planning.
GIL-ROCK-OH, J. FAVREL, J.-P. CAMPAGNE
A.P.M.S. 82, IFIP working conf. BORDEAUX Août 82
- OKH. 84 Automatic multioperation assembly and application of visual control.
D.E. OKHOTSIMSKY, S.S KAMYNIN, E.I. KUGUSHEV
Artificial intelligence and information control systems of robots 1984
- PARK. 84 State-space representations for coordination of multiple manipulators
W. T. PARK
I.S.I.R. SUEDE - 1984
- REY. 84 General approach to assembly robot control
G. REYMAN
Elsevier science publishery - 1984
- REY. 84 bis Application of patterm recognition algorithms for robotics assembly
G. REYMAN
3th international conference on system engeneering
DAYTON, OHIO, (USA) 1984

- ROY. 64 Les problèmes d'ordonancement : applications et méthodes.
Physionomie et traitement du problème d'ordonancement.
B. ROY
AFIRO, DUNOD, 1964
- ROY. 69 Procédure d'exploitation par séparation et évaluation (PSEP et
PSES)
B. ROY
R.I.R.O. 1969, 3^{ème} année, n°6, pp 61-90
- ROY. 70 Décisions avec critères multiples. Problèmes et méthodes.
B. ROY (Metra)
7^{ème} symposium de LA HAYE, 1970
- ROY. 80 Système relationnels de préférence en présence de critères
multiples avec seuils.
B. ROY, Ph. VINCKE
Cahier du C.E.R.O., Vol 22, 1, 1980
- SAN. 83 Sensor based robotic assembly system. Research and
applications in electronic manufacturing.
A.C. SANDERSON, G. PERRY
Proceeding of IEEE, n°7, 1983
- SCEPTRE - Proposition de noyau normalisé pour les exécutifs temps réel.
Projet développé par le Bureau d'orientation de la
Normalisation en Informatique.
T.S.I. 1984
- SIL. 82 Conception d'un exécutif temps réel modulable pour la
commande de processus par micro-processeurs
M. SILLY, J.P. ELLOY
Proceedings, 5th IASTED, TUNIS - 1982
- SIM. 84 La reconnaissance des formes par algorithmes.
J.C. SIMMON -
1984 - Edition Masson
- SOE. 77 Contribution à l'étude des systèmes de conduite en temps réel
en vue de la commande d'unités de fabrication
R. SOENEN
Th. d'Etat, U.S.T.L., 1977

- / SOU. 81 Un modèle de résolution des problèmes d'ordonnancement
 dynamique.
 J.P. SOUBRIER
 Th.D.I., Université P. Sabatier, Toulouse - 1981
- / SOU. 82 Un algorithme de résolution des problèmes d'ordonnancement
 dynamiques.
 J.P. SOUBRIER
 RAIRO, R.O., Vol. 16, n° 3, Août 1982
- STA. 85 Decisions problems in the flexible assembly cell using robot
 cooperation.
 M. STAROSWIECKI, M. DJEGHABA, M. BAYART, G. REYMAN
 4th International Conference on Systems Engineering
 COVENTRY (ENGLAND) - Sept. 85
- STA. 85 BIS Task scheduling by multicriteria optimization in a flexible
 assembly cell using robot cooperation
 M. STAROSWIECKI, M. DJEGHABA, M. BAYART
 15th International Symposium on Industrial Robots Tokyo
 (Japon) - Sept. 85
- STA. 85 ter Moniteur de coopération entre robots dans une cellule flexible
 d'assemblage.
 M. STAROSWIECKI, M. DJEGHABA, M. BAYART
 Computer Aided Design and applications - IASTED (PARIS) -
 juin 1985
- / THO. 80 Aide à la décision pour l'ordonnancement d'atelier en temps
 réel.
 V. THOMAS
 Th. 3^{ème} cycle, Université P. Sabatier - Toulouse - 1980

