In compliance with the
Canadian Privacy Legislation
some supporting forms
may have been removed from
this dissertation.

While these forms may be included
in the document page count,
their removal does not represent
any loss of content from the dissertation.

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Cette thèse intitulée :

ORDONNANCEMENT INTÉGRÉ DE LA PRODUCTION ET DE LA MANUTENTION

présentée par : EL KHAYAT Ghada

en vue de l'obtention du diplôme de : Philosophiae Doctor

à été dûment acceptée par le jury d'examen constitué de :

M. GAMACHE Michel, Ph.D., président

Mme. RIOPEL Diane, Docteure, membre et directrice de recherche

M. LANGEVIN André, Ph.D., membre et codirecteur de recherche

M. BAPTISTE Pierre, Doctorat, membre

M. GHARBI Ali, Ph.D., membre externe

National Library of Canada
Bibliothèque nationale du Canada

Acquisitions and Bibliographic Services
Acquisisitons et services bibliographiques

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-86442-1
Our file Notre référence
ISBN: 0-612-86442-1

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

Canada

À l'âme de ma grand-mère, source d'inspiration

À mon père, source inépuisable d'encouragement

À mon mari, source d'amour qui m'a permis d'arriver au bout du chemin

À Mariam et Ali, soleil de ma vie

REMERCIEMENTS

Je désire offrir mes sincères remerciements aux professeurs Diane Riopel et André Langevin pour leur soutien hors pair tout au long de mes années de doctorat. Leurs conseils, leur constante disponibilité, leur rigueur scientifique et le support financier, qu'ils m'ont accordés, ont permis la réalisation de cette thèse.

Je remercie également messieurs Michel Gamache, Pierre Baptiste et Ali Gharbi d'avoir accepté d'évaluer cette thèse. Je souhaite, par la même occasion, offrir mes remerciements aux professeurs François Soumis, Mario Godard et Federico Passin pour leurs remarques très constructives lors de la proposition de cette recherche.

Je souhaite remercier l'Association des Femmes d'Affaires du Québec et le Groupe Financier Banque TD, pour m'avoir décerné deux bourses pendant mon doctorat.

Un grand merci à mes collègues qui m'ont soutenue chacun à sa façon. Merci à Ayoub Insea Correa pour nos échanges fructueux, à Nathalie Marcoux pour l'écoute et le partage, à Simon Ledoyen et Alberto Amaya pour l'aide informatique et à Nathalie Perrier pour l'encouragement continuel. Ma vive reconnaissance s'adresse aussi à Rachid Chérifi, à Hinda et Hafida Nacer, mes amis qui étaient toujours là pour moi.

Je désire également offrir mes remerciements à tous mes amis de l'Association Cigogne qui m'ont beaucoup soutenue et encouragée durant les années de doctorat. Toute ma reconnaissance à Christine Husson, à Bianca Veilleux, à Véronique Moreau et à Mireille Chalifour. J'ai passé avec vous des moments inoubliables.

J'exprime ma profonde gratitude à ma mère à qui je dois toute réalisation. C'est elle qui m'a toujours été un excellent modèle. Elle m'appris de persévérer et d'arriver au bout

des mes idées. Une note spéciale de remerciements s'adresse aussi aux membres de ma famille et mes amis qui m'ont encouragée de très loin.

Finalement, j'offre les plus grands remerciements à mes enfants Mariam et Ali qui, en bas âge, ont supporté d'être privés de leur mère pour de longues périodes. Ils m'étaient une source d'inspiration et de motivation inestimable. À ce propos, je ne peux jamais oublier de remercier Madame Florise Cabrouet pour l'amour et le soin qu'elle leur a donnés. Elle ne leur était pas une simple responsable de garde, mais aussi une deuxième mère.

RÉSUMÉ

Les problèmes d'ordonnancement en production sont des problèmes très difficiles à résoudre. Depuis plusieurs décennies, leur complexité a incité les chercheurs à proposer des développements méthodologiques permettant de réduire le temps de résolution. L'enrichissement des problèmes, pour mieux représenter une situation réelle, recevait une attention moindre.

Habituellement, dans le traitement des problèmes d'ordonnancement en production, les machines se trouvent à être la seule ressource contraignante. Quelques cas spéciaux considèrent d'autres ressources spécifiques au contexte étudié. Il existe une multitude de ressources nécessaires à un processus de production. Dans cette thèse nous proposons et résolvons des formulations plus complètes intégrant les ressources relatives au système de manutention.

Les ressources essentielles à l'accomplissement des tâches de manutention sont parmi les plus importantes dans plusieurs contextes de production. Il s'agit des ressources qui sont très coûteuses dans le cas où l'on utilise les chariots automatiques. Il faut en optimiser l'utilisation. Les chariots automatiques circulent dans un réseau d'atelier où les segments sont des ressources contraignantes. Les intersections, aussi, sont des ressources contraignantes car un conflit peut se produire à leur niveau. De même, la capacité limitée des files d'attente qui peuvent accueillir les chariots ou les travaux en traitement dans l'atelier pose une contrainte.

Depuis une époque récente, quelques chercheurs se sont intéressés à l'étude des problèmes d'ordonnancement plus complets qui considèrent le caractère contraignant des ressources en équipements de manutention. Mais, d'une part les approches de résolution sont heuristiques, et d'autre part le reste des ressources : segments, intersections et files d'attente, importantes pour le fonctionnement du système de

manutention, ne sont pas incluses. Il s'ensuit que les alternatives en routage et les conflits ne sont pas considérés.

Une revue exhaustive de la littérature sur le problème d'ordonnancement intégré est présentée dans cette thèse. De plus, la littérature sur les problèmes d'ordonnancement de la manutention, qui traite des aspects complémentaires au problème, a été étudiée. Cette revue a mis en évidence la nécessité du développement de modèles plus élaborés. Les modèles développés dans cette thèse apportent d'une part une contribution à la définition du problème, et d'autre part une contribution au niveau de la résolution optimale.

Dans un premier temps, nous exposons la problématique, à son niveau le plus détaillé, à l'aide d'un diagramme déroulement-matière illustrant les différentes ressources nécessaires à la fabrication d'un horaire réalisable en production. Après cette décomposition des ressources, nous analysons des formulations présentées dans la littérature. Cette analyse confirme le manque de méthodologies pour résoudre le problème intégré.

Par la suite, nous traitons trois problèmes d'ordonnancement intégré de la production et de la manutention dans un contexte d'atelier multigamme. Chacun des trois problèmes présente une contribution importante où nous résolvons à l'optimalité et pour la première fois le problème d'ordonnancement intégré de la production et de la manutention. De plus, la définition des contraintes en ressources change d'un problème à l'autre. Nous présentons des formulations de plus en plus complètes, dont l'objectif est la minimisation de la date de fin de l'horaire. Deux formulations sont proposées pour chaque problème : l'une en programmation linéaire mixte et l'autre en programmation par contraintes. Nous utilisons, pour la première fois, ces deux méthodologies pour résoudre nos problèmes. La résolution a été effectuée à l'aide d'un logiciel commercial.

On considère, dans les trois problèmes, que les machines sont des ressources contraignantes. On ajoute à cela d'autres ressources contraignantes selon le niveau de détail de chacun des problèmes. Au premier problème, on considère uniquement les chariots comme ressources contraignantes. Un temps de plus court chemin est utilisé pour déterminer la durée des tâches de manutention. Les données de la littérature ainsi que d'autres données de plus grande taille, que nous avons générées, ont été utilisées pour valider les modèles. La performance du modèle de programmation par contraintes a été supérieure, notamment, pour les problèmes de grande taille.

Au deuxième problème, les chariots et des routes prédéterminées sont des ressources contraignantes. À partir des mêmes implantations utilisées pour le premier problème, 2 plus courts chemins ont été générés entre toutes les paires source - destination du réseau. Les données du traitement sur les machines restent les mêmes. Les tests effectués ont montré que, malgré la taille, le modèle linéaire mixte a généralement une meilleure performance. La performance était parfois complémentaire. Il était plus facile de trouver une solution par l'une des méthodes pour un jeu donné. Pour d'autres jeux, l'autre méthode était plus performante.

Au troisième problème, les chariots, les segments et les intersections du réseau de manutention ainsi que les files d'attente sont des ressources contraignantes. Les modèles présentés sont très riches et novateurs au niveau des idées de modélisation utilisées que ce soit en programmation linéaire ou en programmation par contraintes. Les mêmes jeux de données des deux premiers problèmes ont été utilisés avec une réduction du nombre de travaux dans l'atelier. Malgré la grande taille des problèmes les résultats sont encourageants. Des raffinements des modèles, notamment, au niveau des structures de données et des stratégies de recherche pourraient améliorer la performance.

Pour une représentation encore plus fidèle de la réalité industrielle, la formulation des objectifs d'optimisation est importante. L'implication de plusieurs ressources dans un

processus de production engendre différents coûts. Une décomposition et une définition des coûts pertinents en production sont proposées dans cette thèse. Ces coûts permettent la formulation d'objectifs intéressants pour les industriels. D'autres objectifs intéressants comprennent les objectifs de juste à temps et de respect des dates d'échéance. Une discussion sur les objectifs nous ramène à revoir la façon dont les problèmes d'ordonnancement sont abordés, et de discuter la relation entre les objectifs et les contraintes.

Les résultats obtenus à travers les différentes expérimentations et analyses nous permettent de tirer des conclusions et de proposer des recherches futures. Sur le plan conceptuel, d'autres enrichissements des problèmes sont possibles tel que la prise en compte des lots de transfert différents des lots de production, ce qui peut engendrer davantage de tâches de manutention. Sur le plan pratique, les outils développés dans la thèse sont utiles non seulement pour résoudre un problème statique mais aussi pour la gestion dynamique de la production, de même que pour l'évaluation des besoins en ressources et la fixation des dates d'échéance. Enfin, sur le plan méthodologique, nous proposons une méthodologie de diagnostic d'un problème d'ordonnancement permettant de l'aborder, à la base, par la formulation la plus appropriée en considérant la criticité des ressources impliquées dans le processus de production.

ABSTRACT

Production scheduling problems are very difficult to solve. For several decades, this fact oriented researcher's efforts to develop better solution methodologies in order to reduce solving time. Less attention was paid to the formulation and the resolution of rich problems representing real situations.

Traditionally, machines are considered to be the only constraining resources when solving a production scheduling problem. In some special cases, other context dependent resources may be considered. Many resources are needed in production. In this thesis, we propose and solve formulations integrating the material handling system resources.

Material handling resources are among the most important in several production settings. They are very costly when automated guided vehicles are used. Optimal utilization is essential. AGVs circulate in a shop network where route segments are constraining resources. Conflicts may result at intersections and hence these are also constraining resources. Buffers that may receive AGVs or pallets are also of limited capacity.

Recently, some researches studied integrated scheduling problems where material handling equipment are constraining resources. However, proposed solution methodologies are all heuristic. On the other hand, other resources conditioning a better utilization of the system are not considered. These include: route segments, intersections and buffers. It follows that routing alternatives and conflicts are not accounted for.

In this thesis, we present an exhaustive literature review on the integrated scheduling problem. Moreover, material handling scheduling literature is also examined. In these problems, interesting complementary aspects are studied. Concluding upon the literature

review, the need to develop more elaborate models is evident. Models developed in this thesis present a double contribution: problem definition and optimal problem solving. First, the detailed problem is presented by an operation process chart that illustrates resources needed in order to produce a feasible production schedule. After which, formulations presented in the literature are analyzed. This analysis confirms the lack of solution methodologies for the integrated scheduling problem.

Next, three job-shop integrated scheduling problems are studied. Every problem presents an important contribution. For the first time in literature, we optimally solve the integrated production and material handling scheduling problem. Moreover, resource constraints definition changes from one problem to another. We present more and more complete formulations with the makespan minimization objective. A mixed integer and a constraint programming model are proposed for each problem. For the first time, both methodologies are used to solve the integrated scheduling problem. Commercial software is used in modeling and solving.

In the three problems, we consider that machines are constraining resources. To this we add other resources in each problem. In the first, AGVs are constraining resources. The shortest path is used to determine the duration of a material handling task or a deadhead. Data from the literature and other larger size problems were generated and used to validate the models. The constraint programming model has a superior performance, especially for larger size problems.

In the second problem, AGVS and predetermined routes are constraining resources. 2 shortest path routes were generated between each origin-destination pair of the network. Processing data on machines are the same as in the first problem. Generally, the mixed integer model had a better performance compared to the constraint programming model. Performance was sometimes complementary. The two models had variable performance according to the test problem.

In the third problem, AGVs, route segments, intersections and buffers are constraining resources. Both models are very rich and present innovative modeling ideas. The same data, used for the previous problems, were used. Fewer jobs were considered for each problem. In spite of the models' size results are encouraging. Enhancement of data structures as well as search strategies might improve performance.

A better representation of an industrial reality entails the formulation of pertinent objective functions. The resources implicated in a production process engender several costs. In this thesis, a decomposition and a definition of production costs are proposed. These allow the formulation of relevant objective functions. Just-in-time and due date objectives are also important in many industrial settings. A discussion on objective functions turns attention to the way production scheduling problems are addressed, and to the relation between objectives and constraints.

Results and analysis obtained through experimentation help provide conclusions and propose future research. Conceptually, more enriching elements can be accounted for. An example is transfer lots that may be different from production lots. This reality creates more handling tasks. Practically, the tools proposed are useful not only for solving a static problem but also for a dynamic production management. They can also be used to evaluate resource needs and in fixing due dates. Finally, we propose a methodology for diagnosis of a scheduling problem that allows tackling the problem, at first, by the most appropriate formulation. This methodology considers critical resources implicated in a production process.

TABLE DES MATIÈRES

DÉDICACE iv
REMERCIEMENTS v
RÉSUMÉ vii
ABSTRACT xi
TABLE DES MATIÈRES xiv
LISTE DES TABLEAUX xviii
LISTE DES FIGURES xx
LISTE DES ANNEXES xxi
LISTE DES SYMBOLES xxii
GLOSSAIRE xxiv
CHAPITRE 1. INTRODUCTION 1
CHAPITRE 2. REVUE DE LA LITTÉRATURE 11
2.1. Préliminaires 11
2.2. Problèmes d'ordonnancement intégré de la production et de la manutention 22
2.2.1. Cas statique 31
2.2.2. Cas dynamique 47
2.3. Problèmes d'ordonnancement de la manutention 53
2.3.1. Cas statique 57
2.3.1.1. Problèmes d'ateliers implantés en ligne 57
2.3.1.2. Problèmes d'ateliers implantés en une seule boucle 59
2.3.1.3. Problèmes d'ateliers implantés en deux boucles 59
2.3.1.4. Problèmes d'ateliers avec une implantation complexe 60
2.3.2. Cas dynamique 63
2.3.2.1. Règles de répartition 64
2.3.2.2. Problèmes d'ateliers implantés en une seule boucle 73
2.3.2.3. Problèmes d'ateliers implantés en plusieurs boucles 73

2.3.2.3. Problèmes d'ateliers avec une implantation complexe 74
2.3.3. Problèmes connexes 77
2.4. Critique et conclusion 78
CHAPITRE 3. PROBLÈMES D'ORDONNANCEMENT INTÉGRÉ : DÉCOMPOSITION DES RESSOURCES 82
3.1. Ressources nécessaires pour un processus de production 88
3.1.1. Ressources pour la transformation ou l'assemblage 89
3.1.2. Ressources pour la manutention 89
3.1.3. Ressources pour la manipulation 90
3.1.4. Ressources en espace : entreposage et circulation 91
3.1.5. Autres ressources 92
3.2. Problèmes traités dans cette thèse : choix des formulations et des méthodes de résolution 94
CHAPITRE 4. PROBLÈME D'ORDONNANCEMENT INTÉGRÉ DE LA PRODUCTION ET DE LA MANUTENTION AVEC DES DURÉES PRÉDÉTERMINÉES 101
4.1. Problème 102
4.2. Formulation en programmation mathématique 102
4.2.1. Notation 103
4.2.2. Modèle 104
4.2.3. Expérimentation 107
4.3. Formulation en programmation par contraintes 116
4.3.1. Notation 121
4.3.2. Modèle 121
4.3.3. Expérimentation 124
4.4. Analyse des résultats 126

CHAPITRE 5. PROBLÈME D'ORDONNANCEMENT INTÉGRÉ DE LA PRODUCTION ET DE LA MANUTENTION SANS CONFLITS AVEC CHOIX DE ROUTES PRÉDÉTERMINÉES 130

5.1. Formulation en programmation mathématique 131

5.1.1. Notation 131

5.1.2. Modèle 133

5.1.3. Expérimentation 138

5.2. Formulation en programmation par contraintes 142

5.2.1. Notation 142

5.2.2. Modèle 145

5.2.3. Expérimentation 154

5.3. Analyse des résultats 157

CHAPITRE 6. PROBLÈME D'ORDONNANCEMENT INTÉGRÉ DE LA PRODUCTION ET DE LA MANUTENTION INCLUANT LE ROUTAGE SANS CONFLITS ET UNE CAPACITÉ LIMITÉE DES FILES D'ATTENTE 160

6.1. Problème 160

6.2. Formulation en programmation mathématique 163

6.2.1. Notation 171

6.2.2. Modèle 174

6.3. Formulation en programmation par contraintes 182

6.3.1. Notation 184

6.3.2. Modèle 187

6.4. Expérimentation et conclusion 193

CHAPITRE 7. PROBLÈMES D'ORDONNANCEMENT EN PRODUCTION: PERTINENCE DES OBJECTIFS D'OPTIMISATION 198

7.1. Objectifs recherchés dans la résolution d'un problème d'ordonnancement 198

7.1.1. Coûts pertinents 204

7.1.2. Estimation des différents composants de coût 208

7.2. Objectifs correspondant à plusieurs niveaux hiérarchiques 209

7.3. Planification et ordonnancement : les échelles de temps 213

7.3.1. Critères influençant les échelles du temps pour les activités de planification et d'ordonnancement 217

7.3.2. Échelles de temps et le lien avec les problèmes d'ordonnancement intégré 219

7.4. Conclusion 220

CHAPITRE 8. CONCLUSION 221

RÉFÉRENCES 230

ANNEXES 263

LISTE DES TABLEAUX

Tableau 2.1. Sommaire des études sur le problème d'ordonnancement intégré de la production et de la manutention 25-28
Tableau 2.2. Critères usuels d'optimisation en ordonnancement de la manutention 56-57
Tableau 2.3. Sommaire des principales règles de répartition des chariots automatiques Règles basées sur l'état des chariots 66
Tableau 2.4. Sommaire des principales règles de répartition des chariots automatiques – Règles basées sur l'état des chariots 67
Tableau 2.5. Sommaire des études sur les règles de répartition des chariots automatiques 73
Tableau 4.1. Modèle complet en programmation mathématique..... 106
Tableau 4.2. Résultats pour les données de la littérature 110-11
Tableau 4.3. Résultats pour les données de la littérature avec temps de manutention ajustés..... 112-113
Tableau 4.4. Résultats pour les données de la littérature avec temps de manutention ajustés et 3 chariots dans l'atelier..... 114-115
Tableau 4.5. Modèle complet en programmation par contraintes 123
Tableau 4.6. Résultats du modèle de programmation par contraintes pour les données de la littérature 124-125
Tableau 4.7. Résultats du modèle de programmation par contraintes pour les données augmentées 126
Tableau 5.1. Modèle complet en programmation mathématique..... 137
Tableau 5.2. Résultats pour la formulation linéaire mixte..... 139-140
Tableau 5.3. Modèle complet en programmation par contraintes..... 152-154
Tableau 5.4. Résultats pour la formulation en programmation par contraintes 155-156

Tableau 7.1. Objectifs usuels utilisés dans la formulation des problèmes d'ordonnancement de la production 202
Tableau 7.2. Taux horaires du personnel de production 209
Tableau 7.3. Exemples de l'ordre de grandeur des intervalles de temps entre la sortie de deux unités de produit 216

LISTE DES FIGURES

Figure 2.1. Séquence de la prise de décisions dans la planification et l'ordonnancement 19
Figure 2.2. Diagramme de flux de l'information en systèmes de production 20
Figure 3.1. Graphique de déroulement-matière pour une gamme de trois opérations 87
Figure 3.2. Module détaillé d'ordonnancement intégré 93
Figure 6.1. Exemple d'implantation 162
Figure 6.2. Représentation des nœuds virtuels nécessaires au flux des voyages à vide 163
Figure 6.3. Transformation des nœuds en arcs virtuels pour modéliser les conflits 165
Figure 6.4. Représentation de flux dans les files d'attente 167
Figure 6.5. Représentation de l'entrepôt dans le réseau de l'atelier 168
Figure 6.6. Représentation d'un entrepôt de produits en cours 169
Figure 7.1. Système de planification manufacturière et de pilotage 211
Figure 8.1. Méthodologie de diagnostic d'un problème d'ordonnancement 223

LISTES DES ANNEXES

ANNEXE A : LES DONNÉES 263
ANNEXE B : EXEMPLES D'ORDONNANCEMENTS 280

LISTE DES SYMBOLES

$\{J_j\}$	Un ensemble de travaux
Brkdwn	Breakdown : panne de machines
C_j	Date de fin d'un travail j
C_{max}	La plus grande date de fin
d_j	Date d'échéance pour un travail j
FBS	Filtered Beam Search : une procédure d'énumération partielle
L_j	Retard algébrique d'un travail j
L_{max}	Retard algébrique maximal
M	Machine
M_j	Ensemble de machines qui peuvent traiter un travail j
Nwt	No wait : sans attente
P	Environnement de machines parallèles à vitesse identique
P/P	Poste de Préparation/Palettisation
P_{ij}	Temps de traitement de travail j sur une machine i
Prec	Precedence : relations d'antériorité
Prmp	Preemption : interruption du traitement d'une tâche par une ressource

Q	Environnement de machines uniformes (parallèles à différentes vitesses)
R	Environnement de machines différentes (parallèles à différentes vitesses qui dépendent aussi du travail traité)
Recrc	Phénomène de recirculation où un travail visite une machine plus qu'une fois
Res	Ressource
R_h	Nombre d'unités (h) en ressource (R)
r_j	Date de disponibilité d'un travail j
S_{ijk}	Temps de réglage d'une machine k pour traiter le travail j après le travail i
S_{jk}	Temps de réglage d'une machine k pour traiter le travail j
SMF	Système Manufacturier Flexible
T_j	Retard d'un travail j
T_{max}	Retard maximal
U_j	Travail en retard : $U_j=1$, si $C_j>d_j$, = 0 sinon
w_j	Poids accordé au travail j
Z	Un ensemble de machines identiques parallèles

GLOSSAIRE

▪ Atelier	Lieu physique dans une usine de fabrication comprenant les postes de travail, et le réseau de manutention les reliant
▪ Atelier aux opérations indépendantes	Atelier de production où les différents travaux visitent toutes les machines. Il n'existe pas d'hypothèse sur les durées de traitement ni sur le routage
▪ Atelier d'assemblage	Atelier de production où il existe des relations d'antériorité entre le traitement de quelques pièces et un produit intermédiaire ou final dont elles feront partie
▪ Atelier monogamme	Atelier de production où les différents travaux ont le même routage et il n'existe pas d'hypothèse sur les durées de traitement
▪ Atelier monogamme de permutation	Atelier de production où les différents travaux ont le même routage et il n'existe pas d'hypothèse sur les durées de traitement. La séquence des travaux sur toutes les machines est identique
▪ Atelier monogamme généralisé	Atelier monogamme où il existe plusieurs machines en parallèles à la place de chaque machine
▪ Atelier multigamme	Atelier de production où les différents travaux ont des routages différents et il n'existe pas d'hypothèse sur les durées de traitement
▪ File d'attente	Espace dans l'atelier permettant l'attente au niveau d'une machine ou dans un lieu plus central
▪ Implantation	Aménagement physique de l'atelier déterminant les lieux des postes de travail et la forme du réseau de manutention
▪ Intersection	Une partie du réseau de manutention représentant une fin ou un début d'un ou plusieurs segments

▪ Manipulation	Déplacement d'un travail sans nécessairement parcourir une distance
▪ Manutention	Déplacement d'un travail d'un poste de travail à un autre plus loin dans l'atelier
▪ Segment	Une partie du réseau de manutention représentant une distance donnée
▪ Tâche	Une opération élémentaire ou des groupements d'opérations élémentaires à effectuer sur un travail
▪ Travail	Une pièce ou un nombre de pièces à traiter sur une ou un ensemble de machines dans un atelier

CHAPITRE 1 :

INTRODUCTION

Cette thèse apporte une contribution à la définition, à la formulation et à la résolution du problème **d'ordonnancement intégré de la production et de la manutention.** Traditionnellement, l'ordonnancement des systèmes de production considérait les travaux sur les machines sans tenir compte de la **capacité des équipements de manutention**. L'ordonnancement des tâches de manutention a également été étudié, mais souvent dans une hiérarchie de décisions qui suppose qu'un ordonnancement de traitement des travaux sur les machines a déjà été établi. Cette approche hiérarchique sous-estime l'importance de la manutention dans un système de production. La bonne conduite du système de manutention permet d'atteindre des objectifs de coût, de qualité et de délais. Le temps que les pièces passent en manutention peut être aussi important, parfois même plus important, que le temps qu'elles passent sur les machines compte tenu de la rapidité accrue de ces dernières. De même, la disponibilité des équipements de manutention, et des autres ressources liées au système de manutention, représente une contrainte du problème. Dans les environnements flexibles, ces équipements sont souvent des chariots automatiques, qui demandent un investissement important et dont il faut optimiser l'utilisation. Plusieurs critères de performance pour la conduite d'un système de manutention tels que la distance totale parcourue, le temps de manutention à vide, et la somme des retards ont été présentés dans la littérature.

Une forte relation d'interdépendance existe entre les machines et les équipements de manutention. Cette relation justifie de faire un ordonnancement intégré. Dans un atelier de production, ce sont les machines qui, après avoir fini leur traitement des tâches, génèrent les missions des équipements de manutention. De même, ce sont ces derniers qui distribuent aux machines les tâches qu'elles vont traiter. Quelques auteurs ont considéré l'ordonnancement intégré de la production et de la manutention. Ces travaux ne sont pas nombreux. Quelques chercheurs ont traité l'ordonnancement intégré dans

des ateliers où les conditions d'opération sont très spécifiques, par exemple, le retour des pièces à une zone de chargement/déchargement après chaque traitement effectué par une machine (Raman et al. 1986), des chariots dédiés à un travail et des chariots dédiés à une machine (Lee et DiCesare 1994). D'autres cas plus généraux ont été traités, mais les approches de solution deviennent moins rigoureuses avec les problèmes de grande taille, souvent rencontrés en industrie. Une approche basée sur la programmation dynamique a été élaborée pour un atelier en boucle à 3 machines et 2 chariots automatiques (Blazewicz et al. 1991). Des procédures d'évaluation et de séparation progressive ont été présentées dans le cas de cellules robotisées à 2 machines et un seul robot (King et al. 1993). Des problèmes d'une taille plus grande ont été abordés par les méthodes de simulation qui utilisent des règles de répartition assez simples (Sabuncuoglu et Hommertzheim 1989 à 1998). Récemment Ulusoy et al. (1997) ont développé un algorithme génétique pour résoudre un problème de 4 machines et 2 chariots automatiques. Pourtant les résultats obtenus par les différents chercheurs laissent place au développement de meilleurs algorithmes, à l'étude de **problèmes plus complets** et de plus grande taille, et qui **considèrent toutes les décisions du problème de façon intégrée,** d'où l'intérêt de cette thèse. Il y a aussi place pour le développement de méthodes d'ordonnancement robustes, rapides et facilement implantables par les praticiens, ce qui a orienté notre choix de méthodes de résolution.

Les contributions en ordonnancement intégré de la production et de la manutention considèrent essentiellement **les temps de manutention**. La **circulation qui peut générer des conflits de routage et la capacité des files d'attente** ne sont pas prises en compte. Sabuncuoglu (1998) étudie un problème où il considère, de façon détournée, les contraintes des files d'attente et des conflits de routage sous forme de blocage. La méthode de résolution développée par l'auteur propose le ré-routage des travaux vers une file d'attente centrale dans le cas où les files d'attente des machines deviendraient remplies, ce qui représente une solution de design qui contourne la contrainte plutôt que de la respecter. Dans sa présentation, la façon selon laquelle son heuristique traite les

conflits n'est pas explicite. Les phénomènes de conflits ont été étudiés dans la littérature sur l'ordonnancement de la manutention sans considérer l'ordonnancement sur les machines.

Il existe aussi une littérature très vaste sur les problèmes d'ordonnancement de la production qui néglige toute contrainte relative au système de manutention. Les modèles utilisés pour décrire les problèmes d'ordonnancement de machines ne servent pas à décrire les problèmes d'ordonnancement intégré. La définition d'une tâche de traitement par machine est telle que la durée du traitement est connue a priori. Même quand il s'agit de machines à plusieurs vitesses, on connaît d'avance les différentes vitesses éventuelles. Dans le cas de manutention, la durée requise pour effectuer une tâche n'est pas connue a priori. Cette durée est composée de deux parties qui sont la durée requise pour qu'un équipement de manutention puisse se rendre à la machine source, et la durée de déplacement entre la machine source et la machine destination. La première partie n'est pas connue a priori parce qu'elle dépend de la disponibilité en équipements de manutention et de leur localisation dans l'espace qui n'est pas connue d'avance. Il s'en suit que la durée de la tâche de manutention au complet n'est pas connue a priori. Ceci représente une distinction fondamentale par rapport aux données des problèmes d'ordonnancement sur les machines. De plus, plusieurs alternatives pour l'affectation aux chariots et aux segments de route rendent ces problèmes plus difficiles.

Les méthodes de résolution développées pour les problèmes d'ordonnancement de la production sont d'une grande diversité. Des revues, traitant d'un grand nombre de problèmes d'ordonnancement, ont été présentées dans des articles synthèse par Davis (1973), Graves (1981), Rodammer et White (1988), et par Brucker et al. (1999). Un livre qui couvre le sujet a été écrit par Pinedo (1995).

Les problèmes d'ordonnancement sont, presque toujours, catégorisés en considérant deux facteurs : l'environnement de machines, et les mesures de performance. Le caractère dynamique des problèmes est déterminé par un troisième critère de catégorisation (Graves 1981), souvent négligé, qui est celui de la génération des besoins en traitement. Le besoin d'étudier un ordonnancement de production est dicté par les commandes des clients ou par les décisions de réapprovisionnement du stock. Cette catégorisation permet quelques distinctions propres aux deux environnements. Les problèmes d'ordonnancement générés par les commandes des clients semblent être les plus simples car ils se réduisent à des problèmes de séquencement sans avoir à traiter d'autres aspects tels que les tailles de lots, les économies d'échelle, les coûts de possession du stock et toute autre spécificité propre à la production pour stockage. Pourtant la pertinence des mesures de performance à considérer dans un problème d'ordonnancement généré par des commandes des clients, ainsi que les phénomènes qui peuvent se reproduire, tels que la nervosité dans la production qui peut résulter d'une introduction ou d'une annulation d'une commande, ne sont pas à négliger. De plus, on se retrouve face à une foule de problèmes tels que la détermination des dates d'échéance et des dates de lancement, les décisions reliées à l'attribution des coûts, les changements dans les ordonnancements établis, l'allocation d'un surplus de ressources, et l'acceptation ou le refus de commandes. Une discussion sur la pertinence des mesures de performance est présentée au chapitre 7.

Dans cette thèse, nous présentons une description du problème d'ordonnancement intégré tel que posé dans un atelier de production. Nous incluons plusieurs hypothèses pour une représentation du problème qui soit proche de la réalité. Par la suite nous traitons trois problèmes d'ordonnancement intégré de la production et de la manutention où la définition des ressources contraignantes change d'un problème à l'autre. Finalement, nous faisons une remise en cause des différentes formulations dans la littérature sur les problèmes d'ordonnancement, et nous présentons une décomposition des différents coûts de production pertinents.

Les trois problèmes traités dans le cadre de cette thèse représentent trois niveaux de détail pour le problème d'ordonnancement intégré de la production et de la manutention. Le premier niveau considère un ensemble de tâches de production et de manutention à ordonnancer sur les machines et les chariots respectivement. La durée des tâches de manutention est prédéterminée. Elle est calculée à partir du plus court chemin entre la machine source et la machine destination de chaque tâche. Ceci est également vrai pour les voyages à vide où l'on considère une durée de plus court chemin entre la machine destination et la machine source des deux tâches de manutention affectées consécutivement au même équipement de manutention. Le problème posé dans la littérature correspond à notre premier modèle présenté au chapitre 4. Les nouveaux éléments pris en compte aux deux autres niveaux concernent les tâches de manutention.

Au deuxième niveau, on introduit des choix de routes prédéterminées pour les tâches de manutention et les voyages à vide à effectuer entre deux machines. Les conflits sont évités en définissant l'ensemble de routes conflictuelles qui partagent un ou plusieurs segments. Si deux tâches doivent utiliser deux routes conflictuelles successivement, la première route à utiliser est complètement mobilisée jusqu'à la fin de la première tâche avant que la deuxième tâche commence. À ce niveau, la durée d'une même tâche de manutention peut varier. Celle-ci est déterminée par la route à prendre et les conflits à éviter. De même, le temps de voyage à vide, pris en compte pour assurer la connectivité des tournées des chariots, peut aussi être variable.

Le troisième niveau utilise directement la structure du réseau de l'atelier pour fabriquer des tournées sans conflits pour les chariots à partir de l'ensemble de segments de route formant le réseau. Aucun choix de routes n'est déterminé a priori. Dans ce modèle, on intègre les contraintes de capacité limitée des files d'attente et aussi un temps de réglage des machines.

Deux formulations ont été proposées pour décrire chaque problème. La première en programmation linéaire mixte et la deuxième en programmation par contraintes. La

résolution des modèles a été effectuée à l'aide d'un logiciel commercial qui incorpore les algorithmes nécessaires pour les deux types de formulation, ce qui a appuyé le choix de ce logiciel. Les deux formulations sont résolues à l'optimalité dans des temps très courts et pour la première fois pour le premier et le deuxième problème. De plus, le fait de présenter deux types de formulations pour le même problème, fait qu'on est en mesure de résoudre par une méthode ou l'autre le problème rapidement. Cela constitue un aspect intéressant pour les praticiens.

Le troisième problème étudié est de très grande taille. La majeure contribution est les deux formulations représentant le problème. Les tests ont été effectués sur des problèmes de petite taille d'un point de vue industriel. Nous faisons référence au nombre de travaux dans l'atelier, nombre de tâches par travail et au nombre de chariots. Cependant, d'un point de vue mathématique, le problème résultant est très complexe. Dans la formulation linéaire mixte, un très grand nombre de variables binaires est nécessaire pour la définition d'ordre de visite des différentes ressources ainsi que pour effectuer un choix parmi les ressources alternatives. De même, un très grand nombre de variables de détermination de dates de débuts pour toutes les tâches sur les ressources occupées est utilisé. Dernièrement, beaucoup de contraintes d'égalité sont utilisées pour modéliser les phénomènes de conflits. La modélisation présentée en programmation par contraintes est beaucoup plus concise mais reste à améliorer ou encore à combiner avec de bonnes stratégies de recherche pour pouvoir résoudre des problèmes de taille réelle.

Pour les formulations en programmation linéaire mixte, les paramètres de résolution sont automatiquement fixés dans Cplex version 8 intégré dans OPLStudio version 3.6. Pour le premier problème, nous accélérons la résolution en définissant une borne inférieure heuristique. Nous introduisons également deux inégalités valides qui accélèrent significativement le temps de calcul. La première impose que les premières tâches de tous les travaux débutent après ou au début de l'horizon. La deuxième impose que chaque chariot ait au moins une mission de manutention à effectuer.

Pour les formulations en programmation par contraintes, des stratégies de recherche et des procédures spécifiques sont définies pour chaque problème et elles sont discutées dans les chapitres correspondants. De plus, la contribution présentée par les formulations en programmation par contraintes repose sur des idées intéressantes et flexibles, comme la définition des durées variables des tâches selon les ressources requises, et les différentes définitions des ressources qui sont parfois alternatives. Le type d'une ressource impose ses contraintes de disjonction. Cela est fait sans pour autant avoir un très grand nombre de variables comme dans le cas de programmation mathématique.

Un autre aspect qui a été exploré dans les formulations en programmation par contraintes est la redondance des contraintes et même des variables qui a amélioré de façon significative le temps de calcul pour le deuxième problème, formulé dans la chapitre 5, qui inclut des choix de routes. Un troisième aspect qui rend l'écriture des modèles plus expressive et la définition des variables plus compacte est la flexibilité dans la définition des types de variables. Cet aspect a permis une formulation très élégante des problèmes.

Les implantations d'ateliers utilisés pour générer les premiers jeux de données sont tirés de Bilge et Ulusoy (1995). Les auteurs ont utilisé quatre implantations d'atelier pour générer les données sur les durées de tâches de manutention. Ils ont proposé aussi dix ensembles de travaux à traiter avec les gammes respectives. Les données pour les tâches de manutention ont été utilisées avec les données sur les gammes pour générer quarante problèmes tests. À notre connaissance, il s'agit des seuls problèmes tests reproductibles présentés dans la littérature pour le problème d'ordonnancement intégré dans le contexte d'atelier multigamme. Ces 40 problèmes ont été résolus à l'optimalité pour le premier modèle formulé considérant un plus court chemin pour les tâches de manutention et pour les voyages à vide. De plus, un problème test parmi les plus difficiles de la littérature sur le problème d'ordonnancement d'atelier multigamme (mt10 de Fisher et Thompson),

a été utilisé pour valider le premier modèle. Il est disponible à la bibliothèque électronique de recherche opérationnelle (Beasley). Les données ont été augmentées par les données de manutention nécessaires. Les résultats dans le cas du modèle de programmation par contraintes sont meilleurs. En fait pour les problèmes de grande taille, la programmation par contraintes fonctionne beaucoup mieux pour ce problème comparativement à la programmation linéaire mixte.

Le deuxième problème formulé, présenté au chapitre 5, où nous introduisons des choix de routes pour les tâches de manutention et les voyages à vide a été également résolu à l'optimalité. Les mêmes quarante jeux de données ont été utilisés pour valider le modèle. Cependant, les jeux ont été augmentés par les données nécessaires pour représenter le choix de routes en définissant deux plus courts chemins entre chaque paire de machines. Bien que cette formulation soit a priori sous-optimale car elle restreint les choix de routes, elle comporte pour un même atelier un nombre moindre des variables par rapport au problème où l'on considère tous les segments de route dans l'atelier. De plus la solution de ce problème représente une borne supérieure du problème où l'on considère tous les segments de route dans l'atelier. De façon générale, la formulation résolue en programmation linéaire mixte a donné de meilleurs résultats par rapport au modèle formulé à l'aide de la programmation par contraintes. Cependant, la rapidité de la résolution de ces modèles dépend d'une bonne définition des procédures de recherche qui méritent beaucoup d'étude. Le raffinement des procédures de recherche pourra constituer l'objet des recherches futures.

Le troisième problème présenté dans cette thèse a un intérêt double : la représentation réaliste des problèmes d'ordonnancement intégré tel que posé dans un atelier de production et aussi la résolution, même si les problèmes sont de petite taille. Les formulations peuvent être éventuellement utiles pour quelques cas spéciaux ou pour des contextes plus stratégiques où les ordonnancements doivent être déterminés quelque temps à l'avance. Deux formulations du problème sont présentées, l'une en

programmation linéaire mixte et l'autre en programmation par contraintes. Les jeux de données utilisés pour tester les deux formulations sont développés à partir des données initiales du premier problème fournies par Bilge et Ulusoy (1995). Mais on considère un nombre moindre de travaux dans l'atelier. La longueur des segments de route est déterminée, et chaque machine est dotée d'une file d'attente qui peut accueillir quatre palettes au maximum, à un instant donné.

Le caractère dynamique de l'environnement de production d'aujourd'hui est à ne pas négliger. La modification ou même, le changement d'un ordonnancement de production établi, est une réalité toujours présente. Il faut définir les politiques d'opération qui vont imposer des limites aux perturbations qui peuvent être introduites. Ces politiques d'opération ainsi que les informations nécessaires pour une prise de décision bien fondée sont vaguement discutées dans la littérature. Il n'y existe pas de méthodologie claire pour gérer la production dans cet environnement volatile. On ne se sert pas non plus des données réelles dans un atelier de production pour étudier les systèmes dynamiques. Il existe surtout des approches de simulation qui testent des règles de décision myopes ou des algorithmes en ligne qui possèdent une vision à court terme. Ces contributions supposent une arrivée continue des travaux suivant une certaine distribution. Raman et Talbot (1993) présentent une autre approche. Ils proposent de traiter un problème statique d'ordonnancement à chaque fois que l'état du système change. Mais la méthodologie d'intervention, les limites des perturbations et la performance à long terme ne sont pas discutées. De plus, la résolution des différents problèmes statiques considère que toutes les machines sont disponibles au temps zéro. Ceci n'est pas vrai quand les travaux sont en cours de production dans un atelier. Il existe forcément **des machines qui ne sont pas disponibles au temps zéro.**

Dans nos modèles, on définit le début d'un horizon de planification par un paramètre dont la valeur peut changer. La rapidité de la résolution, pour le premier et le deuxième problème, permet alors de re-résoudre un problème statique à partir de différents

moments dans le temps et de considérer les tâches en question. Il y a donc possibilité d'utiliser l'outil dans un environnement dynamique. De plus, les formulations en programmation par contraintes peuvent se résoudre très rapidement comme des problèmes de satisfaction de contraintes sans avoir forcément à optimiser un objectif donné. Ces résolutions permettent d'avoir une solution réalisable implantable dans un atelier pour remédier à une situation d'interruption de production par exemple, et donc l'outil proposé est bénéfique dans les cas dynamiques.

Cette thèse est organisée comme suit. Après cette introduction, un chapitre de revue de littérature est présenté. La revue porte sur les problèmes d'ordonnancement intégré de la production et de la manutention et les problèmes d'ordonnancement de la manutention qui contiennent des aspects complémentaires à notre problème. Le chapitre 3 discute en détail du problème d'ordonnancement intégré, des ressources nécessaires et des formulations reliées au problème. Les chapitres 4, 5 et 6 exposent les modèles et l'expérimentation des trois problèmes d'ordonnancement intégré discutés plus haut. Le chapitre 7 offre une discussion sur les objectifs d'optimisation pour les problèmes d'ordonnancement ainsi qu'une décomposition des coûts pertinents en production. La conclusion de cette thèse est présentée au chapitre 8.

CHAPITRE 2 :
REVUE DE LA LITTÉRATURE

Dans ce chapitre nous présentons une revue sur deux classes de problèmes. Les problèmes d'ordonnancement intégré de la production et de la manutention sont d'abord présentés. La catégorisation est faite selon les deux cas, statique et dynamique, et l'analyse est axée sur l'intégration des différentes décisions relatives au problème, ainsi que l'identification des conditions spéciales d'opération du système. La littérature sur les problèmes d'ordonnancement de la manutention est ensuite examinée, où d'autres aspects complémentaires au problème, qui n'ont pas été étudiés dans le cadre des problèmes d'ordonnancement intégré, sont considérés. Les cas statique et dynamique sont examinés et la catégorisation est faite selon le type d'implantation qui influence largement le fonctionnement du système de manutention. Nous présentons d'abord quelques définitions dans la section suivante.

2.1. Préliminaires

Les problèmes d'ordonnancement sont d'une diversité étonnante. On les rencontre dans beaucoup de secteurs mais souvent on adopte le langage du secteur manufacturier pour les décrire. Le problème général de l'ordonnancement déterministe est un problème d'optimisation qu'on peut énoncer comme suit : on considère l'allocation dans le temps de ressources (ex. machines) à des activités appelées produits ou travaux. Un travail est souvent composé d'un nombre de pièces constituant un lot de production. En général, n **travaux** sont traités sur un groupe de m **machines**, de façon à optimiser un **objectif** de gestion et ceci sous un ensemble de **contraintes**. Les travaux sont décomposables en **tâches** qui peuvent être des opérations élémentaires ou des groupements d'opérations selon que l'on veuille aller plus ou moins loin dans les détails. Pour chaque travail j, on définit une date de disponibilité (r_j). Pour chacune des tâches, les notions suivantes ont une signification concrète digne d'intérêt : la durée du traitement qui peut dépendre de la

machine *i* (P_{ij}), et les ressources (autres que les machines) qu'on utilise lors de la période de traitement (R_h) où h va de 1 au nombre de ressources. Dans certains cas, le volume des ressources affectées à une même tâche peut varier. Souvent les ressources ne sont pas prises en compte dans la résolution des problèmes d'ordonnancement sur les machines. Il existe pourtant un nombre de contributions qui considèrent les problèmes d'ordonnancement avec contraintes de ressources. Une grande revue sur ces derniers se trouve dans Blazewicz et al. (1986). Des contributions plus récentes ont été présentées surtout dans le contexte des problèmes d'ordonnancement de projets qui considèrent les contraintes de ressources (Brucker et al., 1999).

Selon la définition avancée, trois déterminants du problème d'ordonnancement sont retenus : l'environnement de machines, les contraintes et les objectifs. Ces trois déterminants sont discutés ci-dessous. Le premier est **l'environnement de machines**. Il détermine la complexité du problème d'ordonnancement sur les machines ainsi que le type d'implantation qui a une grande influence sur la manutention. Plusieurs environnements de machines ont été étudiés dans le cadre des problèmes d'ordonnancement. **L'environnement d'une seule machine** est le cas le plus simple qui représente un cas spécial de tous les autres environnements plus complexes. Chaque travail *j* requiert une seule opération sur la machine *i*. En général, il n'existe pas de relations d'antériorité entre les travaux. **Dans l'environnement de *m* machines identiques en parallèle (P)**, le travail *j* requiert une seule opération et peut être traité sur n'importe laquelle des *m* machines ou sur un sous-ensemble des *m* machines qui ont toutes la même vitesse. Les *m* machines parallèles peuvent aussi avoir différentes vitesses de traitement v_i. Dans ce cas, elles sont appelées les **machines uniformes (Q)**. Les **machines différentes en parallèle (R)** sont une généralisation de l'environnement de machines uniformes. Les vitesses des machines sont dépendantes des travaux ($v_{ij} \neq v_i$ $\forall i, \forall j$).

Des environnements plus complexes se trouvent dans les ateliers. **L'environnement d'atelier** est d'un intérêt spécial pour les problèmes d'ordonnancement intégré. Il donne naissance à des tâches de manutention plus complexes. Un atelier consiste en un ensemble de m $\geq$groupes de machines [Z_1, Z_2, Z_3, Z_4, ..., Z_m]. Chaque groupe Z_z possède $m_i \geq 1$ machines identiques parallèles. Un nombre de travaux $\{J_j\}$ $1 \leq j \leq n$, sont traités dans un atelier. Trois types d'atelier sont identifiés. **L'atelier monogamme** consiste en *m* machines en série. Chaque travail doit être traité sur chacune des *m* machines. Tous les travaux ont le même routage. En général, on trouve des **ateliers monogammes de permutation** (prmu) où la séquence des travaux sur les différentes machines est la même. Les ateliers monogammes sont appelés **ateliers monogammes généralisés** si à la place de *m* machines en série, on a *S* étapes en série avec un nombre de machines en parallèle à chacune des étapes. Les travaux doivent être traités sur une des machines à chaque étape. **L'atelier aux opérations indépendantes** consiste en *m* machines. Chaque travail doit être traité sur chacune des *m* machines. Quelques temps de traitement peuvent être zéro. Les différents travaux peuvent avoir des routages différents et il n'existe pas de contraintes d'antériorité parmi les différentes tâches. **L'atelier multigamme** consiste en *m* machines. Chaque travail a son propre routage. On distingue 2 types d'atelier multigamme, celui où chaque travail visite les différentes machines pour une seule fois au maximum et celui où un travail peut visiter une seule machine plus qu'une fois. Ce phénomène est appelé recirculation (recrc).

Les ateliers discutés précédemment se trouvent dans les environnements traditionnels ainsi que dans les environnements automatisés de production. Ces derniers sont typiquement des systèmes manufacturiers flexibles qui consistent en un nombre de machines à commande numérique connectées par un système automatisé de manutention, souvent des chariots automatiques. Les machines dans ces systèmes sont capables d'effectuer différentes opérations de transformation de pièces. Chaque opération demande des outils différents. Les outils sont stockés dans un carrousel d'outils à capacité limitée. S'il est nécessaire d'ajouter ou d'enlever des outils du

carrousel, la machine peut être arrêtée pour un réglage. Dans quelques cas, la régénération d'outils se fait en temps masqué.

Le deuxième déterminant du problème d'ordonnancement est **l'ensemble de contraintes** auxquelles la réalisation peut être soumise. Elles comprennent la disjonctivité et la disponibilité de ressources (res), l'antériorité technologique (prec) et la localisation temporelle, et l'interdiction de l'attente des travaux entre deux machines qui se succèdent (nwt), cas qui peut se reproduire dans les ateliers monogammes. Un autre phénomène qui peut se reproduire dans les environnements d'atelier monogamme est le blocage (block). Si dans l'atelier on a une file d'attente limitée entre deux machines qui se succèdent, il se peut lorsque la file est remplie ou lorsqu'il n'existe pas de file d'attente, qu'on ne puisse libérer la première machine. Le travail complété doit rester sur la machine en l'empêchant de faire le traitement d'un autre travail. Dans ce cas, on fait souvent l'hypothèse que les machines fonctionnent selon le principe du premier arrivé premier servi et que le blocage (block) implique la permutation (prmu).

D'autres contraintes sont le respect d'une date limite d'un travail (d_j), et les contraintes sur l'éligibilité de machines (M_j) dans le cas de machines parallèles. M_j est l'ensemble de machines qui peuvent faire le traitement d'un travail *j*. Une contrainte sur la panne de machines, par exemple pour la maintenance, peut aussi exister (brkdwn) dans le cas d'ordonnancement déterministe et elle implique la non disponibilité des machines pendant quelques périodes connues d'avance.

Considérer les contraintes sur les temps de réglage qui dépendent de la séquence (S_{jk}) amène plus de difficultés. Généralement les temps de réglage sont ajoutés aux temps de traitement. Le temps de réglage entre les travaux *j* et *k* peut aussi dépendre de la machine *i* (S_{ijk}). La préemption du traitement d'une tâche sur une machine (prmp) peut, dans certains cas, faciliter l'analyse du problème, comme elle peut la compliquer car l'espace de solutions réalisables sera agrandi de beaucoup (Pinedo 1995).

Les facteurs profonds de différenciation entre les problèmes d'ordonnancement se trouvent dans la nature des contraintes les plus restreignantes (Roy 1964). À titre d'exemple, dans le contexte manufacturier on rencontre des problèmes où les contraintes de ressources sont plus importantes que les contraintes d'antériorité technologique. Par contre, dans quelques problèmes d'ordonnancement de projets, on trouve des contraintes d'antériorité technologique qui sont plus importantes que les contraintes de ressources.

Le troisième déterminant qui complète la définition du problème est l'objectif **à optimiser**. Le choix d'un ordonnancement est guidé par quelques critères (objectifs) qui auront des expressions analytiques dans les formulations mathématiques qui décrivent les problèmes d'ordonnancement. Deux types d'objectifs sont identifiés, les **objectifs système** et les **objectifs de respect des dates d'échéance** (d_j). Les deux types sont présentés dans les lignes qui suivent. L'objectif usuel qui cherche à maximiser l'utilisation du système est celui de la minimisation de la date de fin de l'ordonnancement (C_{max}). Un autre objectif est celui de la minimisation de la durée totale de séjour ($\sum C_j$) où C_j est la date de fin du travail *j*. Dans le cas d'ordonnancement de la manutention, on considère la minimisation de la distance totale parcourue pour une meilleure utilisation du système de manutention. Les objectifs de respect des dates d'échéance comprennent la minimisation du retard T_{max} (T_j = max (C_j-d_j, 0), du retard algébrique L_{max} (L_j= C_j -d_j) et du nombre des travaux tardifs $\sum U_j$ (U_j=1, si C_j>d_j, =0 sinon). Dans quelques cas, on attribue un poids (w_j) à chacun des travaux et on s'en sert dans la formulation des objectifs. Quelques exemples sont la minimisation du nombre pondéré des travaux tardifs ($\sum w_j U_j$), la minimisation de la somme des dates de fin pondérée ($\sum w_j C_j$), et la minimisation de la somme des retards pondérée ($\sum w_j T_j$). Les fonctions objectifs qu'on vient de présenter sont très courantes, et elles sont appelées des mesures régulières de performance. Il s'agit des fonctions qui sont croissantes en la date de fin des travaux (C_j). D'autres mesures qui ne respectent pas ce critère sont appelées les mesures irrégulières de performance. Une de ces mesures est la minimisation de la somme des retards et des avances.

Une fois que la définition du problème est complétée, une approche de solution est appliquée. Différents types d'ordonnancement sont utilisés dans la résolution des problèmes. Beaucoup de termes deviennent de plus en plus répandus. Nous reprenons les différentes définitions de Sabuncuoglu et Hommertzheim (1992 b) pour ne pas créer d'ambiguïtés plus tard dans le document. *L'ordonnancement en ligne* consiste à ordonnancer les opérations, une à la fois, à chaque fois que le besoin se crée. Il s'agit d'une vision à court terme. *L'ordonnancement hors ligne* est une approche plus globale et consiste à ordonnancer toutes les opérations des travaux disponibles pour la totalité de la période d'ordonnancement. *L'ordonnancement en temps réel ou l'ordonnancement dynamique* est un processus de prise de décision à court terme qui génère et met à jour l'ordonnancement établi en se basant sur l'état actuel du système et sur les exigences du système au complet. C'est un terme plus large que celui *d'en ligne* ou de *hors ligne*. Ce type d'ordonnancement peut être effectué avec une méthode *hors ligne* ou une méthode *en ligne* ou bien une combinaison des deux. Si les méthodes *hors ligne* sont utilisées, le processus d'ordonnancement devient de l'ordonnancement ou du ré-ordonnancement. Dans le cas d'ordonnancement *en ligne*, la décision de l'ordonnancement est prise quand l'état du système change, comme au moment de la fin d'un travail, l'arrivée des travaux, etc. L'idéal serait d'alterner entre ces deux extrêmes pour profiter des avantages des deux.

Dans la littérature, la catégorisation des problèmes d'ordonnancement se fait en considérant un ou plusieurs des critères suivants :

- l'objectif à optimiser;
- l'environnement de machines;
- la génération des besoins en traitement : il s'agit du cas où le besoin est généré par les décisions du réapprovisionnement du stock ou celui où il est généré par les commandes des clients. Cette catégorisation donne naissance aux deux modes d'ordonnancement statique et dynamique respectivement;

- le traitement à une seule étape ou le traitement multi-étape : le traitement muli-étape considère une multitude de niveaux de production par lesquels un produit doit passer. À chaque niveau, des contraintes technologiques ainsi que des ressources sont définies. Chaque nouvelle étape peut nécessiter le calcul de nouveaux lots de production, ainsi que la disponibilité d'un ensemble de pièces produites aux étapes précédentes qui feront partie d'un seul produit fini.

Sauf indication contraire, la plupart des problèmes présentés dans la littérature traitent des problèmes statiques, mono-objectif et à une seule étape. Un problème d'ordonnancement est généralement décrit par un triplet $\alpha|\ \beta\ |\ \gamma$. Ces trois champs représentent l'environnement de machines, les caractéristiques du traitement et les contraintes, et l'objectif du problème respectivement. Cette représentation n'est pas suffisante pour bien décrire les problèmes d'ordonnancement intégré. La représentation classique des trois champs des problèmes d'ordonnancement en production est étendue par Liu et MacCarthy (1996). Les auteurs définissent 5 champs $A|B|C|D|E$, pour décrire les problèmes d'ordonnancement dans les systèmes manufacturiers flexibles (SMFs). Les attributs sont A : le type du SMF, B : les contraintes de capacité, C : les caractéristiques des travaux, D : l'environnement de production et E : les objectifs d'ordonnancement. Cette représentation décrit en détail le problème d'ordonnancement considéré, mais elle n'est pas utilisée dans la littérature.

Comme pour la représentation habituelle du problème, la définition générale du problème d'ordonnancement ne peut malheureusement pas englober les problèmes d'ordonnancement intégré de la production et de la manutention. La difficulté principale est le fait de ne pas avoir des tâches et des temps de traitement définis a priori pour les voyages de manutention. De plus, le problème d'ordonnancement intégré comprend trois décisions au lieu d'une seule, comme dans la définition classique :

- la répartition des équipements de manutention;
- le routage des équipements de manutention;

- l'établissement des ordonnancements des machines ainsi que des équipements de manutention.

La répartition consiste à faire l'affectation des tâches aux chariots. Le routage consiste à choisir le parcours que devra suivre un chariot pour accomplir une tâche. L'établissement des ordonnancements consiste à déterminer à quels moments les chariots vont parcourir chacun des segments de leur route, et à quels moments le traitement sur les machines va commencer.

Avant de présenter les études faites sur les problèmes d'ordonnancement intégré de la production et de la manutention, il serait important de préciser qu'il existe différentes représentations des systèmes de production. Il est important de se situer par rapport à une certaine représentation du système de production pour pouvoir définir les niveaux de décision concernés par notre problème ainsi que les ressources contraignantes considérées. Les différents types de modèles représentant les systèmes de production, trouvés dans la littérature sont présentés au chapitre trois. Hax et Meal (1975) ont présenté l'approche hiérarchique fondamentale de la planification de la production (figure 2.1). Pinedo (1995) fournit une représentation plus récente des systèmes de production (figure 2.2).

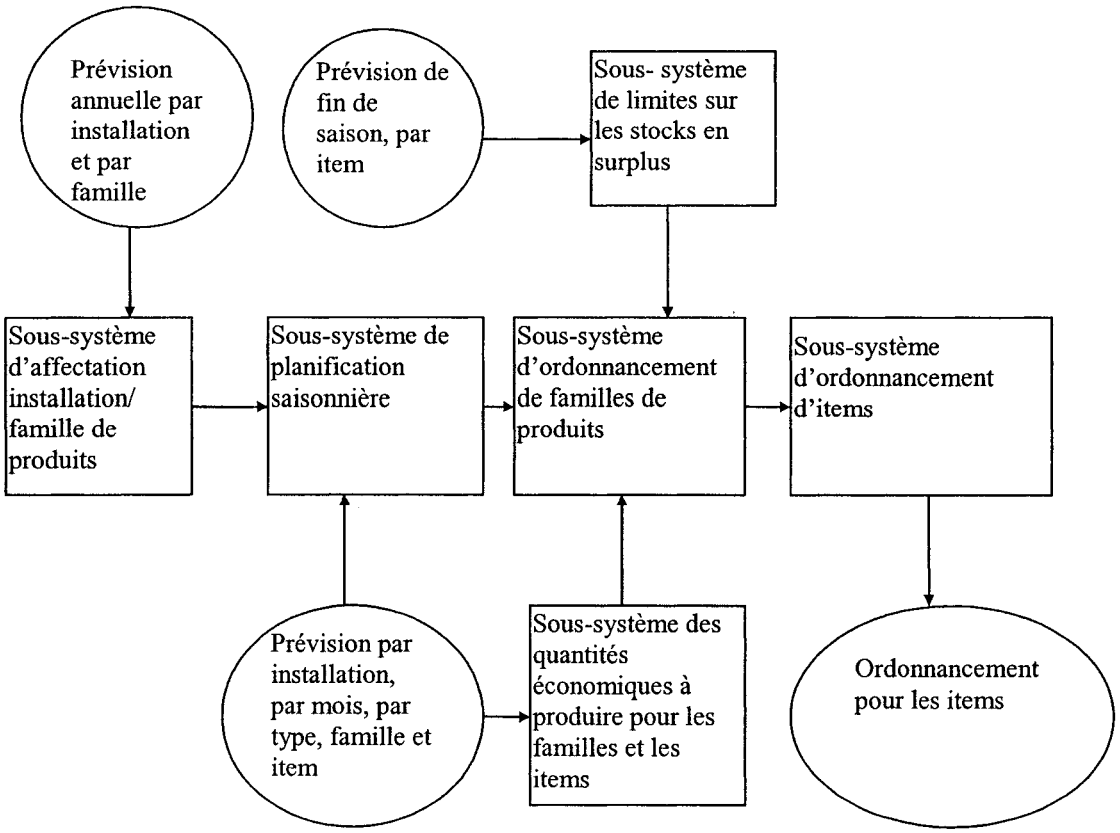


Figure 2.1. Séquence de la prise de décisions dans la planification et l'ordonnancement (Hax et Meal 1975)

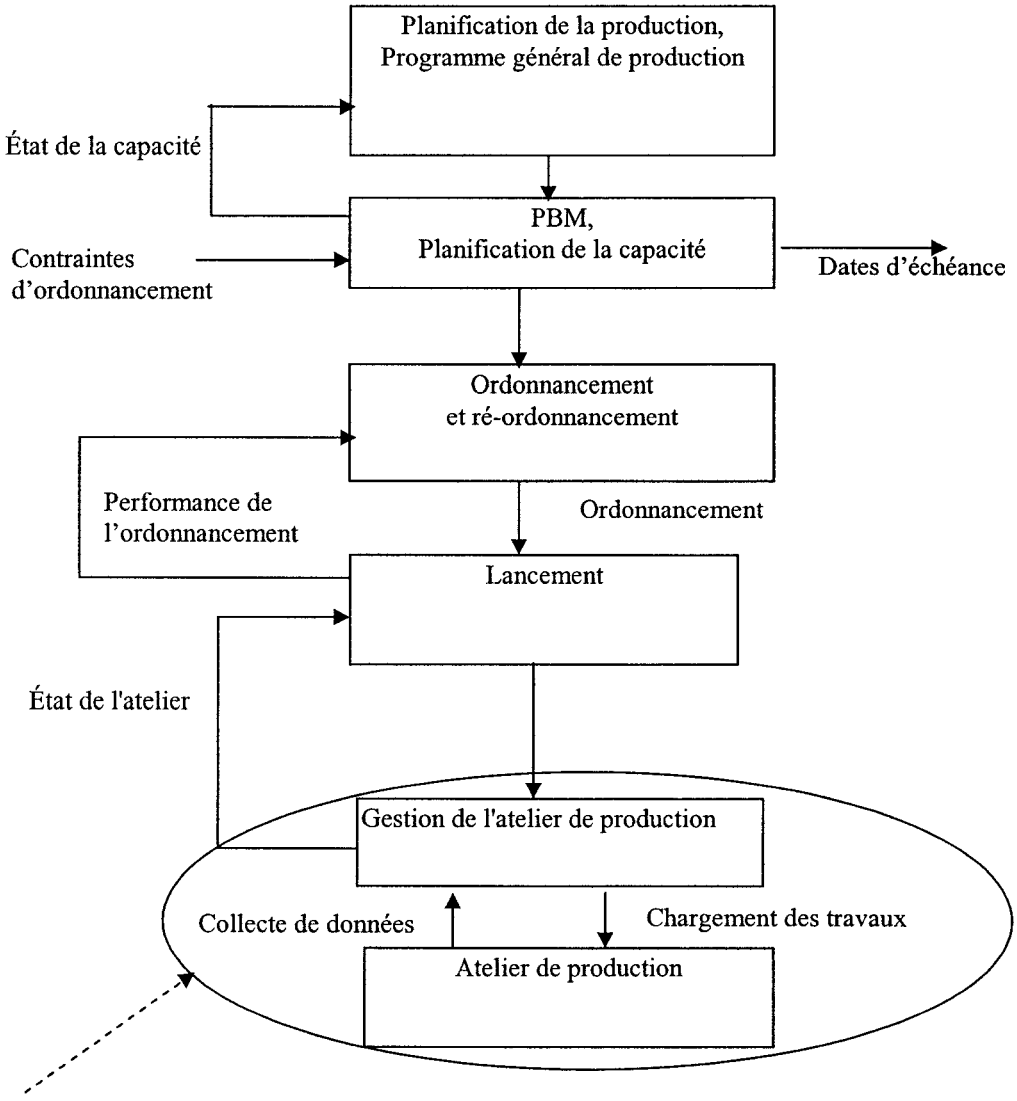


Figure 2.2. Diagramme de flux de l'information en systèmes de production (M. Pinedo, 1995)

Un avantage de l'approche hiérarchique est de pouvoir traiter des problèmes de taille raisonnable. Par contre, il faut noter que cette approche peut résulter en une sous-optimisation. L'idéal serait d'intégrer les niveaux de décision quand cela s'avère possible et nécessaire et aussi, à un même niveau, considérer tous les facteurs influents. Quelques auteurs ont considéré l'intégration de plusieurs niveaux. Des exemples suivent.

Un problème qui considère l'intégration de deux niveaux hiérarchiques est celui de la détermination simultanée des tailles des lots et des ordonnancements. Une revue de ce problème se trouve dans Elmaghraby (1978) pour les problèmes d'une seule machine. Le problème de déterminer simultanément les tailles et la séquence dynamique des lots de plusieurs produits dans un environnement d'une seule machine avec des contraintes de capacité, a été traité par Salomon et al. (1991), Cattrysse et al.(1993) et par Glass (1992) qui considère seulement le cas de trois produits. Pinto et Rao (1992) étudient également le même problème mais dans le contexte d'un atelier monogamme avec des contraintes de capacité. Des méthodes heuristiques sont proposées pour la résolution.

Dans le même ordre d'idées, le problème d'atelier multigamme a aussi été abordé dans la littérature avec une approche plus intégrée. Wein et Chevalier (1992) considèrent trois décisions à optimiser à la fois : la fixation des dates d'échéance des travaux, le lancement des travaux dans l'atelier de production et la détermination de la séquence. Un atelier de deux machines est étudié. Le système est modélisé comme un réseau de file d'attente. Pour le même type d'atelier, Lasserre (1992) considère un modèle intégré qui traite simultanément les problèmes de planification et d'ordonnancement dans un atelier multigamme. Une procédure de décomposition est utilisée pour la résolution. Elle fait l'alternance entre les deux problèmes en question. Des expériences numériques sont présentées.

2.2. Problèmes d'ordonnancement intégré de la production et de la manutention

Si l'intégration se fait entre les différents niveaux de prise de décision dans une hiérarchie de production, il serait encore plus pertinent de la considérer à un seul niveau de prise de décision, en tenant compte des facteurs influents. Dans notre analyse, nous retenons la représentation du système de production fournie par Pinedo (1995). Nous nous intéressons ici au niveau de prise de décision concerné par l'ordonnancement. Les décisions prises à ce niveau sont influencées par plusieurs contraintes relatives à la disponibilité des ressources telles que les équipements de manutention, les files d'attentes et les outils. Traditionnellement, l'ordonnancement de la production a toujours été effectué en considérant une seule ressource représentée par les machines de transformation. Cette vision suppose une capacité infinie de toutes les autres ressources nécessaires au processus de production, et encore, sous-estime les relations d'interdépendance qui existent parmi les différentes ressources impliquées dans ce processus.

Dans les différents modèles hiérarchiques proposés dans la littérature, on remarque qu'il n'existe pas un découpage fin définissant les différentes décisions à prendre pour effectuer un ordonnancement. Les décisions relatives à la gestion des systèmes de manutention et d'outillage n'y sont pas énumérées de façon explicite. Par conséquent, on néglige un grand ensemble de contraintes et donc de décisions à prendre lors de la résolution des problèmes d'ordonnancement. Les contraintes négligées sont parfois considérées dans le cadre de l'étude d'un même problème ou encore, traitées séparément dans le cadre d'autres problèmes. Dans les deux cas, ceci se fait souvent en un deuxième temps après qu'un ordonnancement sur les machines soit établi.

Les décisions relatives au système d'outils peuvent être étudiées en se servant des contributions présentées dans le domaine de l'ordonnancement de la production avec contraintes de ressources. Ces contributions considèrent les ressources qui doivent être

utilisées en même temps que les machines, telles que les outils et les matières premières. Ce corps de littérature ne sert pas dans l'étude des problèmes d'ordonnancement intégré. Les machines et les ressources du système de manutention possèdent des relations d'interdépendance d'une nature différente. Les équipements de manutention, les files d'attente et les segments de route sont utilisés avant ou après les tâches de transformation sur les machines. Des contraintes d'antériorité sont donc à respecter.

Avec l'émergence et l'étude des SMFs, plus d'attention a été accordée aux problèmes de gestion du système de manutention. Notamment, les problèmes de design et d'opération des systèmes de chariots automatiques, qui représentent des investissements coûteux et des ressources de goulot d'étranglement dans plusieurs cas, ont reçu une attention particulière. Les problèmes d'opération des systèmes de chariots automatiques touchent aux aspects d'ordonnancement, de routage et de répartition. Les différentes contributions sont présentées dans la section 2.3. Dans ces problèmes, les requêtes de manutention sont dictées par les ordonnancements des machines déjà établis. Dans quelques cas, le problème se réduit à un problème de tournées de véhicules qui a été excessivement étudié et qui est résolu efficacement pour les problèmes de petite taille.

L'objectif à optimiser considéré dans la plupart des problèmes d'ordonnancement présentés dans la littérature est celui de la minimisation de la date de fin de l'ordonnancement (C_{max}). Dans le but d'arriver à des ordonnancements encore plus courts, quelques auteurs se sont intéressés à l'étude des problèmes plus réalistes qui considèrent des lots de transfert qui ne sont pas égaux aux lots de production. Il s'agit de diviser les lots des travaux en sous-lots de telle façon que les opérations puissent se chevaucher. Potts et Baker (1989), étudient les lots de transfert pour un problème d'atelier monogamme dans le cas d'un et de plusieurs produits. Vickson et Alfredsson (1992) considèrent ce problème pour un atelier monogamme à deux et trois machines. Ils étudient également l'objectif de minimisation de la durée totale de séjour. Trietsch et Baker (1993) étudient un problème de lots de transfert en considérant les contraintes

relatives à la capacité de l'équipement de manutention. Glass (1994) étudie le cas d'un seul produit dans des ateliers monogamme, multigamme et aux opérations indépendantes. Sriskandarajah et Wagneur (1998) considèrent ce problème pour un atelier monogamme à deux machines traitant plusieurs produits et sous la contrainte de « sans attente ». Esaignani et al. (1999) considèrent le même problème dans le cas d'atelier aux opérations indépendantes. Langevin et al. (1999) calculent les lots de transfert pour un atelier monogamme implanté en boucle, avec l'objectif de minimiser l'ensemble des coûts pertinents. À l'exception de la contribution de Trietsch et Baker (1993) qui considère la capacité finie de l'équipement de manutention, les auteurs en général ne considèrent pas les contraintes relatives au système de manutention utilisé pour le déplacement de ces lots de transfert. Langevin et al. (1999) considèrent un coût associé à l'utilisation d'un équipement de manutention, sans contraintes de disponibilité. De même, les temps de manutention et la capacité des files d'attente, ne sont généralement pas considérés.

Récemment, **l'ordonnancement intégré** de la production et de la manutention a reçu l'attention d'un certain nombre d'auteurs. Dans le tableau 2.1, nous présentons les contributions les plus importantes sur le problème d'ordonnancement intégré de la production et de la manutention. La majorité de ces contributions sont dans un contexte d'atelier multigamme, ce qui correspond au contexte manufacturier considéré dans l'étude des problèmes de cette thèse. Quelques-unes sont présentées dans un contexte d'atelier monogamme. Toutefois, elles représentent un apport important soit au niveau de la taille du problème traité ou la méthodologie proposée. Les différentes contributions présentées dans le tableau et d'autres sont détaillées dans le texte qui suit.

Tableau 2.1. Sommaire des études sur le problème d'ordonnancement intégré de la production et de la manutention

Légende

AGV	Chariot automatique	ORD-Hier	ORDonnancement Hiérarchique
C/D	Poste de Chargement/Déchargement	ORD-Int	ORDonnancement Intégré
C_{max}	Date de fin de l'ordonnancement	PD	Programmation Dynamique
DE	Distribution Exponentielle	P_i	Temps de traitement sur machines
DP	Distribution Poisson	PLM	Programme Linéaire Mixte
DU	Distribution Uniforme	PNE	Programme en Nombres Entiers
Dyn	Dynamique	PNLM	Programme Non Linéaire Mixte
ÉE	Étude Empirique	Seg-bidir	Segment bidirectionnel
ESP	Évaluation et Séparation Progressive	Seg-unidir	Segment unidirectionnel
MoG	Atelier MonoGamme	SMF	Système Manufacturier Flexible
MuG	Atelier MultiGamme	Sta	Statique
N/D	Non Disponible	T_{ij}	Durée d'une tâche de manutention

Nb.	Auteur	Problème	Options en routage	Durée traitement/ manutention	Nb. Mach.	Nb. et type d'équip. de manutention	Implantation	Nb. d'opérations / travail	Nb. de travaux	Méthodologie	Objectif	Caract. spéciales
1	Raman et al. 1986	ORD-Int SMF-MoG-Sta SMF-MoG-Dyn	N/D	N/D	N/D	N/D	N/D	N/D	N/D	PNE- ESP, approche étendue pour le cas dynamique.	Min $\sum T_j$	Retour au C/D après chaque tâche de manutention
2	Sabancuoglu et Hommertzhein (1989)	ÉE-Règles d'ordonnance-ment des machines et des AGV-MuG	N/D	N/D	6	2 AGVs	En cellules	1 à 6	N/D	Simulation sous plusieurs conditions	Objectifs de respect des dates d'échéance	Nombre et contenu des travaux simulés

Nb.	Auteur	Problème	Options en routage	Durée traitement/ manutention	Nb. Mach.	Nb. et type d'équip. de manutention	Implantation	Nb. d'opérations / travail	Nb. de travaux	Méthodologie	Objectif	Caract. spéciales
3	Jaikumar et Solomon (1990)	SMF-Dyn	Seg-bidir	N/D	33 cellules	20 AGVs	50 cellules d'assemblage, d'usinage et de réception/ expédition, implantées en deux grands carrés	N/D	200 types de pièces	1. Ordonnan-cement pour machines 2. Réseau espace-temps. 3. Une décomposition minimale des chaînes du réseau détermine les affectations et le nombre des AGVs.	**1.** Max l'utilisation de machines **2.** Min le temps total de manutention **3.** Min la flotte des AGVs	Retour au C/D après chaque tâche d'usinage
4	Blazewicz et al. (1991)	ORD-Hier	Seg-unidir	P_i=1 a 2 T_{ij}>0.5 <3	3	2 AGVs	En boucle	1	9	1.Ordonnance-ment pour machines 2.Résolution d'un problème de tournées de véhicules avec fenêtres de temps.	Min C_{max}	Machines identiques
5	Blazewicz et al. (1991)	ORD-Int	Seg-unidir	P_i=1 a 2 T_{ij}>0.5 <3	3	2 AGVs	En boucle	1	9	Algorithme pseudo-polynomial basé sur la PD	Min C_{max}	Machines identiques.
6	Sabancuoglu et Hommertzhein (1992a)	ÉE-Règles d'ordonnance-ment des machines et des AGVs MuG.	Seg-unidir	Distribution normale	6	2 AGVs	En cellules	1 à 6	3000 simulés	Investigation par simulation. Plusieurs conditions testées	Durée moyenne de séjour « Mean flow time ».	Arrivée des travaux avec DE
7	Sabancuoglu et Hommertzhein (1992b)	ÉE-Règles d'ordonnance-ment des machines et des AGVs- MuG.	Seg-unidir	Distributions uniformes et exponentielles	6	2 AGVs	En cellules	1 à 6	N/D	Un algorithme de répartition « en ligne » utilisant plusieurs règles de priorités	Durée moyenne de séjour, et retard positif moyen	Arrivée des travaux avec DE

Nb.	Auteur	Problème	Options en routage	Durée traitement/ manutention	Nb. Mach.	Nb. et type d'équip. de manutention	Implantation	Nb. d'opérations / travail	Nb. de travaux	Méthodologie	Objectif	Caract. spéciales
8	Sabuncuoglu et Hommertzheim (1993)	ÉE-Règles d'ordonnance-ment des machines et des AGVs – MuG.	Seg-unidir	P_i :Plusieurs distributions T_{ij} : Plusieurs vitesses	6	2 AGVs	En cellules	1 à 6	3000 simulés	Simulation	Objectifs de respect des dates d'échéance	Arrivée des travaux avec DP
9	King et al. (1993)	ORD-Int dans une cellule- MoG	Seg-unidir	P_i= 1 à 9 DU T_{ij}= 3 à 7 DU	2	1 Robot	En série	2	5 à 25	PLM-ESP	Min C_{max}	Un nombre fini de travaux
10	Lee et DiCesare (1994)	ORD-Int MuG	Seg-bidir Deux liaisons directes entre les machi-nes	P_i= 1 à 23 T_{ij}=8 à 20	3+ 1 Robot d'usinage	5 AGVs	N/D	<=3	<=4	Modèle de réseau Petri.	Min C_{max}	AGV dédié à un travail. Retour à C/D
11	Lee et DiCesare (1994)	ORD-Int MuG	Seg-bidir Deux liaisons directes entre les machi-nes	P_i= 1 à 23 T_{ij}=8 à 20	3+ 1 Robot d'usinage	5 AGVs	N/D	<=3	<=4	Model de réseau Petri.	Min C_{max}	AGV dédié à une machine 2 AGV dédiés à C/D
12	Bilge et Ulusoy (1995)	ORD-Int MuG	Seg-unidir	P_i= 3 à 21 T_{ij}= 2 à 20	4	2 AGVs	Par processus	<=5	5 à 8	PNLM et une heuristique de fenêtre de temps.	Min C_{max}	
13	Ioachim et Sanlaville (1996)	ORD-Int Répétitif et plusieurs pièces	-----	Approche applicable pour toutes les données	M	1 Robot	Cellule robotisée Atelier en série	-----	------	Formulation comme problème central répétitif de Carlier et Chretienne (1988). Algorithme de plus long chemin	Min temps de cycle (Max débit)	Résultats étendus pour les cellules multi-robots.

Nb.	Auteur	Problème	Options en routage	Durée traitement/ manutention	Nb. Mach.	Nb. et type d'équip. de manutention	Implantation	Nb. d'opérations / travail	Nb. de travaux	Méthodologie	Objectif	Caract. spéciales
14	Geiger et al. (1997)	ORD-Int	-----	DU Moyenne= 6.67	12 bassins	1 Robot	Atelier en série	6 à 12	10 à 50	Une méthode Tabou.	Min C_{max}	Contraintes de sans attente, blocage et files d'attente limitées
15	Anwar et Nagi (1997)	ORD-Int Assemblage	N/D	N/D	N/D	AGVs	En cellules	N/D	N/D	Heuristique	Juste à temps	Aucun résultat numérique
16	Crama (1997)	Revue sur l'intégration de manutention, outillage et usinage	----	-----	-----	-----	----	-----	------	Propositions de quelques modèles d'optimisation combinatoire.	Plusieurs	Une revue représentative.
17	Ulusoy et al. (1997)	ORD-Int MuG	Seg-unidir	P_i= 8 à 20 T_{ij}= 5 à 21	4	2 AGVs	3 Par processus	2 à 4	5 à 30	Algorithmes génétiques	Min C_{max}	Déviation moyenne de l'optimal =2.53% pour les problèmes test.
18	Anwar et Nagi (1998)	ORD-Int Assemblage	Seg-unidir	Pi= 5 à 10 d'une DU Tij = Plusieurs ratios	2à 9 cellules à 1 à 3 machines	1 à 3 AGVs	En cellules	5 à 10 pour pièces à fabriquer	12 à 239 pièces au total 12 à 126 pièces à fabriquer	Heuristique	Min C_{max} Objectifs de coûts	Pas de détails sur les gammes
19	Sabuncuoglu et Karabuk (1998)	ORD-Int MuG	N/D	Distribution 2-Erlang	6	3 AGVs	En cellules	5 à 6	25	Énumération partielle	Min C_{max}	File d'attente centrale pour contourner le blocage
20	Sabuncuoglu (1998)	ÉE-Règles d'ORD-Int des machines et des AGVs- MuG	Seg-unidir	DE	6	2 AGVs	En cellules	1 à 6	2000 simulés	Simulation	Durée moyenne de séjour « Mean flow time ».	Arrivée des travaux avec une DE. Pannes considérées

Les équipements de manutention qu'on peut considérer dans les problèmes d'ordonnancement intégré sont nombreux. Cette revue porte majoritairement sur les contributions qui considèrent les chariots automatiques comme étant les équipements de manutention. Toutefois, on souligne que plusieurs des résultats obtenus dans le cadre de l'étude des problèmes d'un certain type d'équipements de manutention sont applicables pour d'autres.

Notons, à titre d'exemple, que les résultats obtenus pour résoudre les problèmes de cellules robotisées sont applicables aux cas des cellules desservies par des chariots automatiques bidirectionnels qui effectuent les mêmes tâches de manutention, assumées par le robot. De même, ils sont applicables aux problèmes d'atelier monogamme implanté en ligne et desservi par des chariots automatiques bidirectionnels (Ganesharajah 1998). Un résultat obtenu par Sethi et al. (1992) pour le routage d'un robot dans une cellule robotisée (m=2), a également été utilisé par Kise, Shioyama et Ibaraki (1991) pour le routage d'un chariot automatique unidirectionnel.

Avant de présenter les différentes contributions qui traitent du problème d'ordonnancement intégré de la production et de la manutention, nous présentons une définition de ce problème. On présente cette définition de façon détaillée et générale en énumérant les différentes décisions que considère le problème et les contraintes auxquelles il est soumis. Il s'agit de déterminer :

- la date de début de chacune des opérations des différents travaux, sur les différentes machines qu'ils doivent visiter;
- la séquence des machines à visiter si celle-ci n'est pas fixée a priori, (par exemple, dans le contexte d'atelier aux opérations indépendantes);
- la date de début de la manutention des pièces d'un poste de travail à un autre poste sur lequel le prochain traitement doit s'effectuer;
- la charge à manutentionner si celle-ci n'est pas égale au lot de production;
- la route que va suivre l'équipement de manutention;

- l'affectation d'un équipement de manutention aux tâches de manutention.

Ces décisions ne sont pas prises de façon simultanée, dans les différentes contributions traitant ce problème. Par exemple, quelques méthodes de résolution procèdent par la fabrication des tournées, et puis elles font l'affectation des équipements de manutention à chacune des tournées. De même, les conditions d'opération et les contraintes changent d'une contribution à l'autre. Quelques auteurs considèrent des conditions spéciales. Par exemple, le retour des pièces à une zone de chargement/déchargement après le traitement sur chacune des machines. Ou encore, les chariots dédiés à un travail qui sont utilisés comme des tables d'assemblage. Souvent dans les problèmes d'atelier monogamme automatisé et les cellules robotisés, une séquence des mouvements du chariot ou du robot est définie a priori et le problème se réduit à celui du séquencement d'entrée des pièces dans le système. L'idéal serait de considérer la flexibilité des ressources pour les exploiter au mieux et non de leur imposer un mode de fonctionnement, à moins que ce ne soit dicté par le contexte du problème. Les contraintes principales sont la capacité finie des machines et la capacité finie des équipements de manutention. D'autres contraintes comprennent la capacité finie des files d'attentes et les conflits de routage.

Le but, dans l'étude de cette classe de problèmes, est l'optimalité globale par rapport à un certain objectif en tenant compte de la relation d'interdépendance qui se trouve entre les deux ressources : les machines et les équipements de manutention. Les problèmes d'ordonnancement de la production et les problèmes d'ordonnancement des chariots automatiques traités séparément sont souvent des problèmes NP-difficiles. Il s'en suit que le problème d'ordonnancement intégré l'est aussi. La littérature contient en général des méthodes heuristiques pour la résolution. Pour faciliter le traitement du problème, quelques auteurs favorisent l'approche qui considère une ressource avant l'autre. En général, deux problèmes d'ordonnancement sont résolus de façon itérative : un problème

d'ordonnancement de la production, et un problème d'ordonnancement de la manutention.

Nous nous intéressons dans cette revue à examiner **le degré d'intégration** considéré par les différents auteurs ainsi que **la liberté d'exploitation des ressources**. Voici les deux définitions :

- le degré d'intégration se réfère au nombre de décisions prises de façon simultanée pour faire l'ordonnancement intégré;
- la liberté dans l'exploitation des ressources se réfère aux conditions particulières d'opération qui peuvent imposer un mode de fonctionnement restrictif au système.

On remarque dans les différentes contributions que quelques auteurs caractérisent leurs problèmes comme étant des problèmes d'ordonnancement intégré de la production et de la manutention, alors que toutes les décisions de l'ordonnancement intégré ne sont pas prises simultanément. De même, il existe des contributions qui considèrent la prise des décisions relatives au problème sans mentionner de façon explicite qu'il s'agit de l'ordonnancement intégré. Alors, plus précisément, nous présentons dans ce qui suit les problèmes d'ordonnancement qui considèrent une ou plusieurs données sur le système de manutention, en plus d'effectuer un ordonnancement sur les machines. Les problèmes sont catégorisés selon les deux cas statique et dynamique.

2.2.1. Cas statique

Les premiers travaux d'ordonnancement intégré de la production et de la manutention représentaient une extension normale du problème d'atelier monogamme à deux machines résolu à l'optimalité par Johnson (1954). La suite normale de ces travaux était d'essayer de résoudre ce problème en considérant les temps de manutention, la disponibilité des équipements de manutention et les contraintes de files d'attente. Maggu et al. (1981), Stern et Vitner (1990), Panwalker (1991) et Levner et al. (1995)

étudient ce problème en considérant l'objectif de minimisation de la date de fin de l'ordonnancement (C_{max}). Ces travaux ne considèrent pas de files d'attente ou bien considèrent des files d'attente de capacité infinie. Les problèmes d'ateliers monogammes aux files d'attente de capacité limitée sont NP-difficiles au sens fort (Papadimitroiu et Kanellakis 1980). Ils sont rarement étudiés.

Maggu et al. (1981) étudient un problème d'atelier monogamme à deux machines. Un nombre infini d'équipements de manutention est considéré de même qu'une file d'attente de capacité infinie entre les deux machines. Ce problème est équivalent au problème traité par Johnson (1954). Il suffit d'ajouter le temps de manutention au temps de traitement sur les machines et d'appliquer l'algorithme de Johnson (1954). Seul le temps de manutention entre les deux machines est considéré. Le temps de retour de l'équipement de manutention de la deuxième machine à la première pour introduire une autre pièce dans le système n'est pas considéré. Stern et Vitner (1990) considèrent le cas d'un seul équipement de manutention, sans file d'attente entre les deux machines et, un temps de manutention qui dépend du travail. Le problème est formulé comme un problème de voyageur de commerce asymétrique qui est NP-difficile. Les petits jeux sont résolus optimalement et une procédure approximative O (n log n) est développée pour les problèmes de grande taille. La procédure utilise une borne inférieure de Gilmore et Gomory (1964) calculée pour un atelier monogamme de deux machines avec les contraintes de « sans attente ». Seuls les ordonnancements de permutation sont considérés. Les travaux ont la même séquence sur les deux machines.

Le cas d'un seul équipement et une file d'attente d'entrée de capacité infinie pour la deuxième machine, est traité par Panwalker (1991). Les temps de manutention ne dépendent pas des travaux. Une politique d'opération est proposée pour l'équipement de manutention. L'auteur développe un algorithme polynomial basé sur les résultats de Johnson (1954) pour résoudre un problème de séquencement des pièces dans le système. Levner, Kogan et Levin (1995) généralisent ce résultat en considérant un problème où

les temps de manutention dépendent des travaux. Ils considèrent également les temps de chargement et de déchargement des pièces. Ces différents problèmes ont une implantation d'atelier monogamme, généralement faite en ligne.

Une autre variante des problèmes décrits précédemment est celle où les implantations sont faites en boucle. Kise, Shioyama et Ibaraki (1991) considèrent le cas de deux machines implantées en boucle et un chariot automatique. Il n'existe pas de file d'attente dans leur problème. Une politique de routage est proposée. Le chariot s'arrête seulement dans le cas où il n'existerait aucune tâche de manutention à effectuer. Un algorithme $O(n^3)$ basé sur celui de Gilmore et Gomory (1964) est utilisé pour faire l'ordonnancement des travaux sur les machines avec l'objectif de minimisation de la date de fin de l'ordonnancement (C_{max}). Il s'agit d'un problème de séquencement de pièces en considérant un cycle spécifique du mouvement du chariot choisi a priori. Les résultats sont étendus pour le cas où l'implantation de l'atelier serait faite en boucle.

Levner, Kogan et Maimon (1995) présentent une généralisation de Kise, Shioyama et Ibaraki (1991) en développant un algorithme $O(n^3)$ pour le cas où les temps de manutention dépendent des différents travaux. Ils considèrent également les temps de réglage des machines.

Agnetis, Pacciarelli et Rossi (1994) considèrent un atelier monogamme de deux machines implanté en une seule boucle comme celui étudié par Kise, Shioyama et Ibaraki (1991). Contrairement aux contributions présentées pour les systèmes semblables, les auteurs considèrent explicitement l'ordonnancement des lots des travaux identiques où chaque travail est manutentionné seul et non pas dans un lot. Toutes les pièces dans un lot doivent être traitées de façon consécutive. La manutention est effectuée par un chariot unidirectionnel et une politique d'opération est suggérée. L'objectif est de trouver une séquence d'entrée des lots qui minimise la date de fin de l'ordonnancement (C_{max}). Il s'agit d'un cas spécial du problème de voyageur de

commerce résolu en temps polynomial dans le cas où la taille du lot *i* serait supérieure à une certaine valeur b_i*. La technologie de l'atelier permet le chargement et le déchargement rapide entre la machine et le chariot (swapping devices). Une formule de calcul de la taille optimale du lot bi* est fournie.

Blazewicz et al. (1991) considèrent l'ordonnancement intégré de la production et de la manutention. Ils considèrent un SMF avec une disponibilité des ressources renouvelables (outils) et non renouvelables (matières premières). Les machines sont les ressources de goulot d'étranglement et elles sont identiques à l'exception de leur localisation (elles peuvent faire toutes les opérations), et elles sont connectées par des chariots automatiques qui servent surtout à manutentionner les matières premières aux machines et à manutentionner les produits finis des machines. Les files d'attente sont permises. L'objectif à optimiser est celui de la minimisation de la date de fin de l'ordonnancement (C_{max}). Les trois façons d'aborder le problème sont présentées dans les paragraphes suivants.

La première façon suppose que l'ordonnancement des machines est déjà connu, et que l'on cherche à établir un ordonnancement pour les équipements de manutention. L'objectif est de trouver un ordonnancement réalisable pour les chariots qui respecte l'ordonnancement des machines. Les données sont l'ordonnancement de production (possiblement optimal) avec les dates au plus tard d_j^i pour la livraison des matières premières nécessaires pour la fabrication de la pièce T_j à la machine P_i, et les temps de manutention du poste d'entrée aux machines particulières τ_i. Ce problème peut être résolu en temps polynomial. Il s'agit d'un problème équivalent au problème de tournées de véhicules avec fenêtres de temps (Desrochers et al. 1988). Un algorithme polynomial est développé pour vérifier la condition de faisabilité d'un ordonnancement des chariots par rapport à l'ordonnancement de production donné, et il donne la solution réalisable si elle existe. L'exemple numérique donné est un système de 3 machines, 9 opérations et 2 chariots automatiques.

Si cette approche ne donne pas d'ordonnancement réalisable pour les équipements de manutention, la deuxième façon consiste à changer l'ordonnancement de production et on reprend la même méthode. Si on n'arrive pas à trouver une solution pour les équipements de manutention avec les nouvelles fenêtres de temps, on passe à l'ordonnancement intégré de la production et de la manutention. Il s'agit d'un problème NP-difficile au sens ordinaire. L'algorithme proposé est pseudo-polynomial pour un petit nombre de machines et il est basé sur la programmation dynamique. Le système étudié est composé de 2 chariots automatiques et 4 machines. Les auteurs considèrent la congestion en proposant un délai entre le début de deux voyages consécutifs qui commencent à partir de la file d'attente des matières premières.

Rajendran (1999) présentent des formulations et des heuristiques pour le problème d'ordonnancement dans les ateliers monogammes et qui tiennent compte des activités de manutention. L'auteur considère plusieurs objectifs.

Dans le même ordre d'idées, King et al. (1993) considèrent l'ordonnancement d'un ensemble des travaux dans une cellule robotisée de 2 machines et un robot. Les files sont permises devant les machines. Les déplacements se font dans une seule direction. Le nombre des travaux dans les jeux d'essai varie entre 5 et 25 et les temps de traitement et de manutention sont tirés d'une distribution uniforme avec des valeurs qui varient entre 1 et 9 unités de temps. Le problème est formulé comme programme mixte entier avec l'objectif de minimiser la date de fin de l'ordonnancement (C_{max}), et une procédure de séparation et d'évaluation progressive est proposée pour la résolution. L'algorithme détermine la séquence d'entrée des pièces et la séquence des mouvements des robots qui sont définis comme étant des tâches distinctes. Il s'agit d'un atelier monogamme de permutation où la séquence des travaux sur les deux machines est la même.

À l'opposé des problèmes des cellules robotisées qui considèrent un ensemble de travaux à ordonnancer, une autre classe de problèmes de cellules robotisées considère un

ordonnancement répétitif. Il s'agit des problèmes où, théoriquement, un nombre infini d'une même pièce est produit en subissant les mêmes tâches de traitement et en suivant les mêmes routes. L'objectif normal devient de maximiser le débit. Une cellule robotisée est constituée d'un seul robot et d'un nombre de machines *m*. Il n'existe pas de files d'attente devant les machines. Il existe *m!* différentes routes potentiellement optimales (Sethi et al. 1992). Les problèmes d'ordonnancement répétitif considèrent les contraintes de disponibilité des machines et du robot ainsi que les temps de traitement sur les machines et les temps de manutention dans le calcul du cycle minimum. De même, les contraintes de « sans attente » font que le traitement sur les machines et le mouvement du robot doivent être synchronisés. Dans quelques cas, le découpage des tâches au niveau des mouvements du robot peut être plus détaillé que dans le cas d'autres équipements de manutention. Ces problèmes ont été étudiés par, entre autres, Hall et al. (1995,1997), Ioachim et Soumis (1995), Ioachim et Sanlaville(1996), Logendran et Sriskandarajah (1996), Sriskandarajah et al. (1996), et Crama et Van De Klundert (1997). Les différentes contributions sont présentées dans les paragraphes suivants.

Ioachim et Soumis (1995) ont développé deux méthodes basées sur les réseaux pour évaluer le temps de cycle associé à un cycle fixe de mouvements de robot, dans le cas de *m* machines. Ioachim et Sanlaville (1996) modélisent le problème comme une instance de la version cyclique du problème central de l'ordonnancement étudié par Carlier et Chrétienne (1988). Ils proposent un algorithme polynomial détaillé basé sur la recherche du plus long chemin dans un graphe pour calculer le temps de cycle minimum. Ne faisant pas appel à une recherche dichotomique, l'algorithme donne la valeur exacte du temps de cycle quelles que soient les données. Les résultats introduits pour un seul robot sont ensuite étendus à des cellules multi-robots. L'algorithme considère aussi le cas où différents types de pièces sont produits. Il suffit de faire un traitement préliminaire qui détermine l'ordre d'entrée des pièces, puis d'appliquer l'algorithme. Crama et Van De Klundert (1997) donnent également un algorithme polynomial qui

résout optimalement le cas général de *m* machines pour le problème d'ordonnancement répétitif d'un seul type de pièces dans une cellule robotisée.

En général, dans les problèmes de cellules robotisées et quand il s'agit de différents types de pièces, une politique d'opération qui fixe la route du robot est définie. Il s'agit par la suite de déterminer la séquence d'entrée des pièces et ce, habituellement, pour minimiser la date de fin de l'ordonnancement (C_{max}). Contrairement à cette logique, Hall, Kamoun et Sriskandarajah (1997) considèrent le problème d'ordonnancement de différents types de pièces dans une cellule robotisée de 2 machines où le cycle du robot est modifié en combinant deux cycles élémentaires ou en alternant entre deux, et ce selon la pièce en traitement. Cette méthode maximise le débit. Un algorithme O (n^4) a été développé pour la résolution. L'algorithme considère les deux décisions de déterminer la séquence d'entrée des pièces et les mouvements des robots simultanément. D'autres résultats sont obtenus quant à la comparaison des politiques de lots pour maximiser le débit. Il s'avère que la production répétitive d'une seule pièce dans une cellule de 3 machines domine les cycles complexes de la production de 2 pièces ou plus, et maximise le débit à long terme. Pour la même cellule de trois machines, 4 cycles de robot parmi les 6 potentiellement optimaux sont étudiés. Quelques cas spéciaux sont identifiés pour le séquencement optimal de différents types de pièces dans la cellule en utilisant chacun des 4 cycles.

Aneja et Kamoun (1999) étudient le même problème d'ordonnancement des pièces et des mouvements de robots dans une cellule robotisée, étudié par Hall et al. (1997). Ils considèrent également les deux décisions de déterminer la séquence d'entrée des pièces et des mouvements du robot simultanément. L'objectif est de maximiser le débit. Les pièces sont produites avec une proportion correspondante à leurs demandes de façon répétitive. Le problème est formulé comme un cas spécial du problème de voyageur de commerce et un algorithme plus rapide, O (n log n), est développé pour la résolution.

D'autres variantes des problèmes des cellules robotisées sont étudiées. Kamoun et al. (1993) étudient un problème de plusieurs cellules connectées par une file d'attente centrale, où chacune des cellules est desservie par un robot. Un problème d'ordonnancement cyclique qui compare différentes alternatives pour maximiser le débit est présenté par Sriskandarajah et al. (1999). Un seul type de pièces est considéré. L'auteur compare le débit du système dans trois scénarios par opposition au scénario traditionnel du robot à un seul préhenseur. Le premier scénario est l'utilisation d'un robot à deux préhenseurs, le deuxième est le routage aux opérations indépendantes et le troisième est l'utilisation d'une file d'attente. Mak et al.(1998) présentent un algorithme pour l'ordonnancement cyclique des mouvements des robots dans l'industrie des circuits imprimés avec des contraintes spécifiques au problème. Kats et al. (1999) proposent une méthode exacte (Sieve) pour l'ordonnancement d'une cellule robotisée avec plusieurs types de pièces. Yang et al. (2001) traitent le problème d'ordonnancement de cellules robotisées à plusieurs robots. Une méthode heuristique basée sur le recuit simulé est proposée.

Des problèmes d'atelier monogamme ont été traités dans d'autres contextes de fabrication. Notons ceux de l'industrie des semi-conducteurs où les contraintes de manutention sont rigoureuses. Les tâches de traitement (en bassins), et les tâches de manutention doivent être synchronisées. En général, il s'agit d'ateliers où on doit respecter des contraintes de « sans attente » et de blocage, tout en considérant les limites du système de manutention. Parmi d'autres, Geiger et al. (1997) traitent ce problème. L'objectif considéré est celui de la minimisation de la date de fin de l'ordonnancement (C_{max}). La logique des mouvements des robots est programmée d'avance et il faut la respecter. Il s'agit de déterminer le séquencement des pièces dans le système. Une méthode *Tabou* est proposée pour la résolution. Le problème traité est de 12 bassins et 1 robot de manutention. Les jeux d'essais sont composés de 10 à 50 travaux qui sont composés de 6 à 12 opérations. Les temps de traitement et de manutention sont tirés d'une distribution uniforme avec une moyenne de 6,67 unités de temps.

Les grands ateliers d'assemblage composés de plusieurs lignes et plusieurs boucles ont été considérés pour un problème d'ordonnancement de la production et de la manutention ainsi que pour la détermination de la taille optimale de la flotte de chariots automatiques. Jaikumar et Solomon (1990) considèrent l'ordonnancement de la production et de la manutention dans une hiérarchie à deux niveaux. Ils étudient un système réel de l'industrie japonaise. L'opération du système est telle que les pièces sont retournées à un entrepôt après chaque opération de traitement sur machine. L'objectif primaire est la maximisation de l'utilisation des machines et l'objectif secondaire est la minimisation du temps total de manutention des chariots automatiques. L'objectif tertiaire est la minimisation de la taille de la flotte de chariots automatiques. D'abord, les différentes pièces sont affectées aux différentes machines avec l'objectif de maximiser l'utilisation des machines et de minimiser le temps de manutention des chariots. Ensuite, les dates d'arrivée du chariot automatique à une machine j sont fixées comme étant un ensemble ordonné $[a_{j,1} \leq a_{j,2} \leq \ldots]$ et un réseau espace-temps est créé comme suit : les nœuds sont $(j,a_{j,i})$, i=1,2,..., j=1,2,... Un arc entre les nœuds $(j_1, a_{1,1})$ et $(j_2, a_{2,2})$, dans le réseau, existe si c'est réalisable pour le même chariot de voyager de la machine j_1 au dépôt central et après à la machine j_2 à l'intérieur de la fenêtre de temps $(a_{1,1}, a_{2,2})$. Les affectations optimales des chariots automatiques sont trouvées à partir d'une décomposition minimale des chaînes du réseau décrit. Chaque chaîne correspond à une séquence des routes de chariots automatiques couvertes par le même chariot. La décomposition minimale de chaînes qui atteint la taille minimale de la flotte peut être trouvée en temps polynomial en résolvant un problème de flot maximum (Shapiro 1979). Cette flotte assure le niveau ciblé d'utilisation de machines. Un algorithme polynomial de séquencement de machines est également fourni. L'algorithme donne la séquence des pièces affectées à chaque machine. Il est basé sur l'intuition que les travaux doivent être ordonnés sur chaque machine dans un ordre qui permet à un chariot automatique d'être affecté à la machine au moment de ou à un moment avant la date de fin du travail correspondant. Le but est de réaliser une utilisation complète des machines

(à 100%) avec la taille de flotte déterminée. Ceci peut être réalisé sous un ensemble de conditions déterminées par les auteurs.

La logique hiérarchique associée à ce problème représente une approche différente de ce que devrait être un problème d'ordonnancement intégré. La taille du problème (50 cellules et 20 chariots), le caractère critique des ressources en machines de même que la formulation des objectifs imposent le choix de la démarche. Bien que l'ordonnancement ne soit pas fait de façon intégrée, les temps et les contraintes relatives aux systèmes de manutention et de machines sont pris en compte.

Raman et al. (1986) considèrent l'objectif de minimiser le retard total pour effectuer un ordonnancement intégré dans le contexte d'un atelier monogamme automatisé. Une condition particulière d'opération considérée par les auteurs est le retour des chariots à une zone de chargement/déchargement après chaque tâche de manutention entre deux machines. Le problème statique est formulé comme un programme mathématique en nombres entiers. La formulation utilisée est semblable à celle utilisée pour modéliser le problème d'ordonnancement de projets avec contraintes de ressources. Ils proposent une méthode de séparation et d'évaluation progressive pour le problème statique, et ils recommandent que cette approche soit applicable sur un horizon fuyant pour le cas dynamique. La condition d'opération considérée résulte en des temps de manutention qui ne dépendent pas de la séquence, ce qui a permis d'aborder le problème avec des modèles d'ordonnancement de projets. Il s'agit d'un cas spécial, et les mêmes modèles ne seront pas applicables au cas général où les temps de manutention dépendent de la séquence.

Ulusoy et Bilge (1993) considèrent un problème d'ordonnancement intégré de la production et de la manutention avec l'objectif de minimiser la date de fin de l'ordonnancement (C_{max}). Un atelier multigamme automatisé est étudié. Contrairement à la contribution présentée par Raman et al. (1986), les chariots automatiques du système

de manutention ne retournent pas à une zone de chargement/déchargement après chaque tâche de manutention entre deux machines, ce qui représente des temps de manutention qui dépendent de la séquence et ceci pose une difficulté supplémentaire. Les auteurs fournissent un programme mixte entier non-linéaire pour formuler le problème. La formulation proposée consiste en un ensemble de contraintes qui représentent un sous-problème d'ordonnancement sur les machines, et un ensemble de contraintes qui représentent un sous-problème d'ordonnancement de chariots. Les deux sous-problèmes interagissent à travers un ensemble de contraintes de fenêtre de temps indiquant les dates de début éventuelles des voyages de manutention.

Le problème est par la suite décomposé en deux sous-problèmes et résolu en utilisant une heuristique itérative. Celle-ci est composée de trois parties :
a. une procédure qui génère des ordonnancements pour les machines.
b. une procédure qui génère une solution pour le problème d'ordonnancement des chariots à partir de l'ordonnancement des machines déjà généré.
c. une structure itérative qui lie ces deux procédures et qui facilite la recherche d'une bonne solution.

À chaque itération, un nouvel ordonnancement des machines est généré par une procédure heuristique. Les dates de fin des opérations obtenues par l'ordonnancement sont utilisées pour construire les fenêtres de temps pour les voyages de manutention et on cherche une solution réalisable pour le deuxième sous-problème d'ordonnancement de la manutention.

Les algorithmes utilisés pour la génération des ordonnancements des machines sont des modifications de Giffler et Thompsons (1960), et de French (1982). Pour les appliquer, les temps de traitement sont définis comme étant les temps de traitement sur machines plus les temps de voyages chargés des équipements de manutention. Les deux

algorithmes utilisent trois règles de décision : travail maximum restant (MWR), travail minimum restant (LWR), et temps de traitement le plus court (SPT).

L'heuristique de tournées de véhicules avec les fenêtres de temps glissantes (STW) est utilisée pour résoudre le sous-problème d'ordonnancement des chariots. Elle est basée sur le concept de coût d'opportunité présenté par Ferland et Fortin (1989). Il s'agit d'identifier les paires de tâches qui offrent un meilleur coût d'opportunité et ceci en modifiant les dates de début dans les fenêtres des temps si nécessaire. L'heuristique (STW) peut résulter en une nouvelle construction des fenêtres des temps si on ne trouve pas une solution réalisable avec le nombre de chariots automatiques dont on dispose. Des exemples numériques ont été testés. Les jeux d'essai sont composés de quatre machines et deux chariots automatiques. Quatre différentes implantations ont été testées. Les résultats montrent que l'implantation influence considérablement la qualité de la solution donnée par l'heuristique.

Les heuristiques développées pour les deux sous-problèmes ne se servent pas de la formulation proposée. Celle-ci est épineuse à cause de la taille et de la non-linéarité. En effet, il s'agit de deux problèmes d'ordonnancement connus, celui du $J_m|C_{max}$ qui est NP-difficile selon Lenstra et Rinnooy Kan (1978), et un problème de tournées de véhicules qui est NP-difficile selon Orloff (1976). Les deux sont liés par des fenêtres de temps.

En 1995, Bilge et Ulusoy étudient à nouveau le même problème en utilisant la même approche de solution. Cette référence est plus détaillée et contient plus d'exemples numériques. Les auteurs mentionnent que leur approche est éventuellement applicable à plusieurs types de problèmes, tels que :

- les problèmes d'ordonnancement de projets avec des ressources qui possèdent des temps de réglage qui dépendent de la séquence;
- les problèmes de machines parallèles où les temps de réglage dépendent de la séquence et où les opérations possèdent des dates de disponibilité et d'échéance;

- les problèmes de transport à la demande où les dates de début du transport sont incluses à l'intérieur des fenêtres de temps.

Ulusoy et al. (1997) reprennent le même problème et ils proposent de le résoudre en utilisant les algorithmes génétiques. Les résultats obtenus sont meilleurs que ceux obtenus par leur approche heuristique basée sur les fenêtres de temps. L'algorithme a résolu optimalement 60% des jeux d'essai considérés. Il s'agit des mêmes jeux d'essai considérés auparavant. La différence fondamentale entre cette approche et leur approche précédente est que la solution de l'algorithme génétique est représentée de sorte qu'elle contienne l'information sur l'ordonnancement sur les machines et l'ordonnancement sur les équipements de manutention de façon simultanée dans les chromosomes contrairement à l'approche itérative qui considère les deux systèmes séparément. Dans les différentes contributions présentées par les auteurs, il n'existe pas de considérations de réseau routier et une matrice donne les temps de manutention entre les différents postes. Les files d'attentes possèdent une capacité infinie.

Lee et DiCesare (1994) étudient le problème d'ordonnancement intégré de la production et de la manutention dans un contexte de fabrication d'atelier multigamme. Le réseau routier est tel qu'il existe des liaisons directes entre les machines formées à partir des routes droites. Une paire de routes, chacune dans une direction, existe entre les différents postes pour faciliter la gestion des chariots. Les routes sont des ressources contraignantes que les chariots doivent partager. Si la route est longue, des nœuds virtuels peuvent la diviser et donc elle peut accueillir plus qu'un chariot sans conflit. Alors, les conflits sous forme de blocage sont considérés et ceux sous forme de collision sont contournés (par l'existence d'une paire de routes). Un modèle de réseaux de Petri est présenté et une méthode heuristique de recherche qui donne des solutions optimales ou près de l'optimal est développée. L'objectif considéré est la minimisation de la date de fin de l'ordonnancement (C_{max}). L'atelier considéré est composé de 3 machines et de 1 robot pour effectuer les tâches de transformation et de 5 chariots automatiques pour

effectuer les tâches de manutention. Deux cas sont étudiés. Dans le premier le chariot automatique est dédié à un travail. Il l'accompagne dans son routage à l'intérieur de l'atelier, puis il retourne à une zone de chargement/déchargement. Dans le deuxième, un chariot est dédié à une machine pour déplacer ses travaux complétés, et 2 chariots sont dédiés à la zone de chargement/déchargement. Dans ces modèles, les décisions d'affectation ne sont pas considérées puisqu'elles sont déterminées par la politique d'opération.

La méthode a également été appliquée à un autre problème qui ne suppose pas de politique d'opération (3 machines, 1 chariot et 3 travaux). Cette méthode d'ordonnancement intégré a donné de meilleurs résultats comparativement aux règles de (file de longueur minimum (SQL), temps de traitement le plus court (SPT), durée minimale d'attente (SWT), appliquées dans une simulation. Dans les exemples, les temps de traitement sur les machines varient entre 1 et 23 et les temps de manutention varient entre 8 et 20 unités de temps. Le nombre de travaux considéré est ≤ 4 avec un nombre d'opérations ≤ 3 par travail.

À l'opposé des contributions précédentes, Sabuncuoglu et Karabuk (1998) étudient le problème d'ordonnancement intégré en considérant la capacité limitée des files d'attente des machines. Ils utilisent une approche d'énumération partielle. La méthode est celle de « Filtered Beam Search ». L'atelier hypothétique considéré est du type multigamme et comprend 6 machines, et un poste de chargement/déchargement. La manutention est effectuée par 3 chariots automatiques. La contrainte sur la capacité des files d'attente n'est pas une contrainte rigide car elle est contournée par une solution de design qui suppose qu'une file d'attente centrale de capacité infinie existe et que l'on peut dégager les routes dans le cas de blocage en déplaçant quelques travaux à cette file d'attente.

Les auteurs indiquent également que l'algorithme considère les conflits sous forme de blocage et que les nœuds qui amèneront à une solution qui génère un conflit ne sont pas

explorés davantage. Pourtant, ils ne montrent pas de façon explicite comment les conflits sont considérés dans leur approche de solution. Chaque segment de route dans le réseau décrit mesure cinq unités de distance. Les jeux d'essai sont composés de 25 travaux avec un nombre d'opérations de 5 ou 6 par travail. Les temps de traitement sont tirés d'une distribution 2-Erlang. La performance de l'algorithme est meilleure que celle de différentes règles d'ordonnancement des machines et des chariots, et ce pour plusieurs critères de performance : durée moyenne de séjour, retard moyen et date de fin de l'ordonnancement. Des conditions d'expérimentation, telles que les niveaux de charges et flexibilité dans le routage et la séquence, ont également été modifiées pour vérifier la robustesse de l'algorithme.

Smith et al. (1999) étudient un problème d'ordonnancement intégré dans un contexte d'atelier multigamme. Les auteurs sont les premiers à considérer non seulement les activités de manutention, mais aussi celles de chargement et déchargement de façon explicite. Alors, une non disponibilité en équipements de manutention après qu'une machine finit le traitement d'un travail implique le blocage de la machine jusqu'à ce qu'un équipement de manutention devienne libre pour le déchargement. Deux approches de solution sont présentées. Elles sont décrites ci-après.

La première est plus globale et considère toutes les activités à ordonner dans un seul problème. Il s'agit d'une heuristique de recherche aléatoire qui requiert un temps de calcul assez long. L'algorithme utilisé est de l'ordre ($n^2 \log n$) par recherche, où n est le nombre d'activités à ordonner. La deuxième approche procède en deux étapes. Premièrement une heuristique est utilisée pour résoudre un problème d'ordonnancement des travaux sur les machines. Dans un deuxième temps, les activités de manutention et de chargement/déchargement sont ajoutées à l'ordonnancement obtenu dans la première étape de sorte que la solution soit respectée autant que possible. Il n'existe pas de réseau routier de l'atelier décrit. Les chariots circulent en mouvement libre et les conflits ne sont pas considérés. De même, les files d'attente possèdent une capacité suffisante pour

empêcher tout blocage. Les résultats numériques montrent que la première approche donne de meilleurs résultats.

Toutes les contributions présentées ont traité un système de production à une seule étape. Un atelier d'assemblage, à deux étapes, composé de 4 cellules à machines identiques est étudié dans un problème d'ordonnancement intégré, par Anwar et Nagi (1998). Seulement, les distances intercellulaires sont considérées. La distinction principale entre les ateliers à une seule étape et les ateliers multi-étapes est dans la nature des contraintes d'antériorité technologique. Dans les grands ateliers d'assemblage on trouve des relations d'antériorité technologique entre les opérations des différentes pièces qui constitueraient un même produit fini. Le problème ne considère pas un réseau routier, et les conflits ne sont donc pas pris en compte. Ce sont les temps de manutention entre les cellules qui sont considérés. Les auteurs considèrent des files d'attente de capacité infinie.

Anwar et Nagi (1997), étudient le même atelier d'assemblage en considérant les conflits de circulation. Une méthodologie heuristique de résolution est présentée. Elle procède en sens inverse où la dernière opération est ordonnancée en premier et les autres opérations sont ordonnancées par la suite tout en respectant les contraintes de capacité et d'antériorité technologique. Si l'activité ordonnancée est une activité de manutention, un sous-problème de routage de chariots génère les r plus courts chemins de chaque chariot pour effectuer cette tâche. Les plus courts chemins sont ordonnés selon un ordre non décroissant de la date de fin de l'opération. Les conflits sont vérifiés. Par la suite le plus court chemin sans conflit et le chariot correspondant sont choisis. Les résultats numériques ne sont pas rapportés.

2.2.2. Cas dynamique

Quelques contributions en ordonnancement intégré de la production et de la manutention considèrent une arrivée continue des travaux au lieu d'un ensemble de travaux disponibles au début de la période d'ordonnancement. Les règles d'ordonnancement myopes sont généralement utilisées et la simulation sert d'outil de validation. Sabancuoglu et Hommertzhein (1989) ont étudié les règles d'ordonnancement des machines et des chariots automatiques dans un SMF. En 1992, ils ont étudié l'ordonnancement intégré de la production et de la manutention pour un SMF où la production est du type atelier multigamme. À l'aide d'une simulation, ils ont évalué la performance de différentes règles de décision. L'objectif à optimiser est celui de la minimisation de la durée moyenne de séjour. Cette durée, pour chacun des travaux, est composée du :

- temps total de traitement;
- temps total d'attente dans la zone de chargement/déchargement;
- temps total d'attente dans les files d'attente d'entrée/sortie;
- temps total d'attente dans la file d'attente centrale;
- temps total de manutention;
- temps de réglage;
- temps de chargement et de déchargement.

Les contraintes du problème sont les capacités finies des machines, des équipements de manutention et des files d'attente des encours de production. Un modèle de simulation est développé pour évaluer l'interaction des différentes ressources et ceci en utilisant différentes règles d'ordonnancement pour les machines ainsi que pour les équipements de manutention. Le nombre de machines qu'un travail doit visiter est entre 1 et 6, et le système possède 2 chariots automatiques. La recirculation n'est pas permise. Plusieurs règles d'ordonnancement ont été testées sous les conditions suivantes:

- variation des charges des machines et des chariots;

- différentes capacités des files d'attente;
- variation dans les vitesses des chariots.

Les meilleures règles pour les machines s'avéraient être SPT (temps de traitement le plus court), SPT.TOT (temps de traitement le plus court multiplié par le temps total d'opération) combinées avec n'importe quelle règle d'affectation des chariots automatiques. Les règles SPT/TOT (temps de traitement le plus court divisé par le temps total d'opération) et LWKR (travail minimum restant) ont également une bonne performance. Les meilleures règles pour les chariots automatiques sont STD (distance de manutention la plus courte), LQS (longueur maximale de la file) dans les différentes combinaisons avec les règles d'ordonnancement pour les machines. En 1993, les auteurs reprennent cette investigation expérimentale des règles d'ordonnancement des machines et des chariots automatiques en considérant plusieurs objectifs des dates d'échéance.

En 1992, Sabuncuoglu et Hommertzheim traitent le problème d'ordonnancement intégré en proposant un algorithme en ligne appliqué en temps réel. Comme dans leurs contributions précédentes, les auteurs supposent que les outils et les montages d'usinage sont continuellement disponibles, et que les temps de réglage sont de zéro pour toutes les opérations. L'algorithme proposé est basé sur l'idée qu'un travail ne doit pas être ordonnancé sur une machine (ou un chariot automatique) s'il doit attendre pour une machine ou pour un chariot pour effectuer l'activité suivante. L'algorithme utilise plusieurs règles de priorité et il est du type en-ligne. Il ne produit pas un seul ordonnancement complet qui donne les dates de début et de fin de tous les travaux. Il fait plutôt l'ordonnancement des travaux sur les machines ou sur les chariots une opération à la fois quand le besoin d'une décision d'ordonnancement apparaît. L'algorithme consiste en deux grandes parties :

- un ensemble de procédures pour ordonner les travaux sur les machines. La logique associée à l'ordonnancement des travaux sur les machines est appliquée à chaque fois qu'une machine complète le traitement d'une pièce et devient disponible pour

d'autres pièces dans la file. Plusieurs indices de priorités sont définis selon les différents critères d'optimisation.

- un ensemble de procédures pour ordonner les travaux sur les chariots. La logique associée à l'ordonnancement des travaux sur les chariots est appliquée à chaque fois qu'un chariot complète sa mission courante et devient disponible pour l'affectation à une deuxième mission. Cette partie consiste en quatre niveaux hiérarchiques:

1. logique « push » pour vérifier les machines critiques (avec blocage ou files remplies);
2. logique file d'attente pour vérifier les pièces dans la file d'attente centrale;
3. logique « pull » pour vérifier la machine inactive;
4. logique « push-pull » pour identifier la machine la plus appropriée et la pièce à servir.

Pour tous les niveaux, des règles de priorité sont utilisées. L'algorithme a été testé sous un nombre de conditions d'expérimentation :

- variation des niveaux de charges;
- différents critères d'ordonnancement;
- différents types de distributions des temps de traitement (distributions exponentielle et normale).

L'algorithme a été comparé avec d'autres règles d'ordonnancement intégré pour les deux systèmes (machines et manutention) qui sont d'une utilisation répandue. Ces règles sont :

- MOD/SDT (la plus petite date d'échéance modifiée d'une opération/ distance de manutention la plus courte);
- MOD/LQS (la plus petite date d'échéance modifiée d'une opération/ la file la plus longue);
- SPT/SDT (temps de traitement le plus court/ distance de manutention la plus courte);
- SPT/LQS (temps de traitement le plus court/ la file la plus longue).

Par rapport aux règles, l'algorithme a toujours donné de meilleurs résultats selon les tests numériques.

Sabuncuoglu (1998) utilise la simulation pour étudier les règles d'ordonnancement des machines et des chariots. L'objectif est d'étudier la performance des différentes règles dans le cas de bris et pour différentes distributions du temps de traitement. Le critère d'optimisation est celui de la minimisation de la durée moyenne de séjour.

L'atelier hypothétique étudié par l'auteur dans ses différentes contributions est composé de 8 postes de travail (6 de machines de transformation) avec des files d'attente limitées, et un poste d'entrée/sortie, deux files d'attente centrales pour éviter le blocage et 2 chariots qui circulent sur un réseau unidirectionnel. À l'exception de la dernière contribution qui considère le cas de bris, les différentes études supposent que les machines ainsi que les chariots automatiques sont continuellement opérationnels. Les autres hypothèses sont décrites ci-après :

- chaque machine peut effectuer au plus une seule opération;
- l'interruption des tâches n'est pas permise;
- le temps de réglage est inclus dans le temps d'opération;
- quand une machine et un chariot automatique deviennent libres, le travail qui suit dans la file est traité immédiatement « non-delay-scheduling »;
- le temps de chargement/déchargement des pièces, des chariots automatiques est considéré comme une constante;
- le temps de transfert des pièces par un chariot automatique est directement proportionnel à la distance parcourue (vitesse constante du chariot);
- les capacités des files d'attente de toutes les machines sont finies et égales. Donc une pièce complétant l'opération courante attend dans la file au poste de travail jusqu'à ce que la file de la prochaine machine et un chariot soient disponibles;
- il n'existe pas de contraintes de capacité sur la file d'attente centrale ni sur le carrousel d'entrée/sortie;
- le chariot transporte une seule pièce à la fois;
- aux points d'intersection du réseau routier des chariots automatiques, le chariot chargé est prioritaire par rapport au chariot qui voyage vide. Dans le cas où les deux

seraient chargés, le droit de passage aux intersections est déterminé sur une base de premier arrivé, premier servi.

- quand un travail est complété à n'importe quelle machine, s'il se trouve plus d'un chariot disponible pour le manutentionner vers la prochaine machine, le chariot le plus proche de la machine source est sélectionné;
- le système d'outillage n'est pas modélisé, et un nombre infini de palettes est considéré.

Jawahar et al. (1998) étudient le problème d'ordonnancement intégré de la production et de la manutention. Ils présentent une heuristique qui utilise des règles de répartition des chariots qui tiennent compte des conflits. Les règles de répartition considérées sont temps de traitement le plus court (SPT), temps de traitement le plus long (LPT), temps de manutention le plus long (LTT) et temps de manutention le plus court (STT). Le critère d'optimisation est celui de minimiser la date de fin de l'ordonnancement (C_{max}). La règle du temps de manutention le plus court (STT) donne les meilleurs résultats.

Les différentes contributions présentées jusque-là présentent les distances inter machines comme étant les distances les plus importantes à parcourir par l'équipement de manutention, et qui vont éventuellement influencer les ordonnancements. Lee et Maneesavet (1999) présentent une contribution où les activités de manutention qui influencent les ordonnancements se font entre le poste de chargement/déchargement et la machine. Les auteurs développent et évaluent des stratégies de répartition pour des chariots guidés par rails dans une zone de chargement/déchargement d'un SMF. Le déplacement s'effectue de façon bidirectionnelle. L'intérêt est porté à la zone du chargement/déchargement du système, mais la combinaison des règles de répartition des chariots et les règles d'ordonnancement des travaux est également explorée pour donner la priorité aux travaux qui sont dans la file d'attente qui se trouvent entre la zone de chargement/déchargement et la machine. Cinq règles de répartition de chariots sont combinées avec des règles habituelles d'ordonnancement de travaux et l'évaluation est

faite à travers une simulation. Les résultats montrent que ces règles de répartition combinées avec la priorité accordée aux pièces ont amélioré le débit et la durée de séjour. L'influence du choix des règles d'ordonnancement des travaux sur la performance du système dépend des caractéristiques de la file d'attente. Dans un contexte de système manufacturier flexible, un ordonnancement en ligne a été proposé par Chan et Chan (2001) où on alterne entre les règles de répartition pour optimiser parmi trois critères.

Les problèmes d'ordonnancement intégré reconnaissent l'impact que peut avoir la mauvaise gestion du système de manutention sur les ordonnancements de production. D'autres approches qui reconnaissent cet impact ont été présentées par un nombre d'auteurs sans forcément traiter des problèmes d'ordonnancement intégré. Seidman et Tenenban (1994) proposent des politiques de routage de pièces qui réduisent la charge de travail du système de manutention dans le cadre d'un problème d'ordonnancement d'un SMF. Plusieurs approches de modélisation sont présentées et plusieurs heuristiques sont proposées. L'objectif du problème est la maximisation du débit. Dans un contexte dynamique, Wu et Zhou (2001) proposent des politiques de contrôle pour éviter le blocage et la pénurie aux machines dans un système manufacturier automatisé.

Crama (1997) présente une revue représentative sur quelques modèles d'optimisation combinatoire pour les problèmes d'ordonnancement intégré des SMFs. Les systèmes de manutention ainsi que ceux d'outillage sont considérés dans cette revue. L'auteur établit des relations entre ces modèles et des modèles développés pour des systèmes de production traditionnels.

Suh et Lee (1998), traitent dans un contexte de ports, l'ordonnancement intégré de la manutention et des opérations de déchargement et autres opérations sur les navires et ils présentent un système expert hiérarchique.

Tel que présenté, les contributions, qui portent sur le problème d'ordonnancement intégré de la production et de la manutention, trouvées à partir de nos recherches ne sont pas nombreuses. Il existe pourtant une littérature très vaste sur les problèmes d'ordonnancement en production qui considère les machines de transformation comme étant la seule ressource contraignante. D'autre part, l'ordonnancement de la manutention attirait l'attention de beaucoup de chercheurs et il a été étudié séparément dans plusieurs contextes. Ces problèmes sont particulièrement intéressants, car des dimensions qu'on juge complémentaires pour le problème d'ordonnancement intégré, y sont considérées. Un exemple est les phénomènes de conflits de routage. Il serait donc intéressant d'examiner la littérature sur les problèmes d'ordonnancement de la manutention. La prochaine section présente les différentes contributions.

Quant aux modèles développés pour les problèmes d'ordonnancement de la production, ils ne servent pas à décrire les problèmes d'ordonnancement intégré vu la nature différente des tâches à ordonnancer et des ressources contraignantes. Par contre la littérature sur les problèmes d'ordonnancement de la production contient un grand nombre de méthodes de résolution, dont quelques-unes peuvent être explorées dans des recherches futures.

2.3. Problèmes d'ordonnancement de la manutention

Traditionnellement, l'ordonnancement en production considérait l'ordonnancement des travaux sur les machines. Les temps de manutention ainsi que la capacité des équipements de manutention étaient négligés jusqu'à une date récente. Depuis les années 1980, un nombre de chercheurs ont étudié les problèmes d'ordonnancement des tâches en manutention dans le contexte manufacturier. Le besoin d'effectuer cet ordonnancement se manifestait suite à l'établissement d'un ordonnancement de traitement de travaux sur les machines. La littérature sur ces problèmes est relativement récente et moins vaste que la littérature sur l'ordonnancement de la production. Il existe

également un corps de littérature sur les problèmes de tournées et d'ordonnancement de véhicules qui peut être applicable à un nombre de problèmes d'ordonnancement de la manutention. Une classification des différents problèmes de routage et d'ordonnancement de véhicules, est présentée par Bodin et Golden (1981). Kusiak (1983) présente des formulations pour différentes variantes du problème d'ordonnancement de véhicules de même que deux heuristiques pour la résolution.

Bien que le traitement séparé de l'ordonnancement de la production et de l'ordonnancement de la manutention ne soit pas la meilleure approche pour réaliser une optimalité globale, l'attention accordée à l'ordonnancement de la manutention se justifiait par le fait que le système de manutention soit un système pour lequel il faut optimiser un certain objectif de gestion. Il représente des ressources limitées et des investissements coûteux dont il faut optimiser l'utilisation. La gestion des systèmes de manutention pose donc des défis (Maxwell et Muckstadt 1982). Comme dans la section précédente, cette revue sera majoritairement axée sur les problèmes des chariots automatiques. Une revue sur le sujet a été présentée par Co et Tanchoco (1991). Une revue complète sur les différents problèmes de design et d'opération des chariots automatiques est présentée par Ganesharajah et al. (1998).

Il existe un nombre de ressources contraignantes qui font que la gestion des équipements de manutention devient une tâche particulièrement complexe. Parmi ces ressources, il y a :

- les files d'attente de capacité limitée ;
- la capacité et la disponibilité des équipements de manutention ;
- les segments de routes à partager par les différents équipements de manutention.

Si ces contraintes n'existaient pas, on aurait additionné le temps de manutention au temps de traitement sur machines, et on aurait appliqué les mêmes méthodes développées pour l'ordonnancement sur les machines. Mais, en plus de la décision sur

les ordonnancements de tournées de manutention, ces contraintes imposent des décisions d'affectation des équipements de manutention aux tâches et aux segments de routes. On remarque que les contributions qui traitent l'ordonnancement de la manutention ne considèrent pas les mêmes décisions. Ou encore, elles ne les considèrent pas de façon intégrée. Kusiak et Cyrus (1985) étudient la prise des décisions de répartition et d'établissement des ordonnancements. La résolution d'un problème d'ordonnancement de la manutention qui tient compte des trois décisions de répartition, routage et établissement des ordonnancements, relatives au système de façon intégrée a été présentée par Langevin et al. (1996) pour la première fois en utilisant une approche de programmation dynamique. La répartition, le routage et l'ordonnancement sont considérés simultanément dans un système de production incluant deux chariots automatiques et 12 postes de travail (7 machines de transformation et 5 stations de service).

La catégorisation de la présentation qui suit tiendra compte de la définition du problème, axée sur l'implantation de l'atelier qui influence largement l'opération du système de manutention. Les deux cas statique et dynamique sont présentés. La plupart des contributions importantes seront couvertes vu leur nombre relativement restreint par rapport à celles présentées pour les problèmes d'ordonnancement sur les machines. L'implantation de l'atelier détermine beaucoup de phénomènes de conflits ou de blocage qui peuvent se reproduire et dont il faut tenir compte en effectuant l'ordonnancement de la manutention. Ces phénomènes représentent une autre particularité des problèmes d'ordonnancement de la manutention car souvent les décisions d'ordonnancement sont combinées avec la suggestion des politiques d'opération pour éviter les problèmes de conflit. Des technologies de détection et de vision sont également utilisées à cette fin. On utilise aussi les méthodes de contrôle par zone qui divisent le réseau routier en plusieurs zones et chacune d'elles doit être occupée par un chariot au maximum. Les problèmes d'ordonnancement de la manutention sont souvent résolus en combinant les solutions technologiques avec les techniques de recherche opérationnelle. Plusieurs

objectifs ont été considérés pour la résolution de ces problèmes. La maximisation de l'utilisation de l'équipement de manutention, la minimisation de la durée moyenne de séjour pour les travaux, de la longueur maximale des files d'attente aux points d'entrée/sortie et du retard moyen des travaux sont parmi les mesures de performance considérées dans la littérature sur les chariots automatiques. Différents objectifs sont présentés dans le tableau qui suit.

Tableau 2.2. Critères usuels d'optimisation en ordonnancement de la manutention

Critère d'optimisation	**Parmi les contributions**
Maximisation du débit	Egbelu et Tanchoco (1984) Han et Mc Ginnis (1989) Occena et Yakota (1991)
Minimisation du temps de manutention à vide	Klein et Kim (1996)
Maximisation de l'utilisation, ou de l'utilisation moyenne de l'équipement de manutention.	Jaikumar et Solomon (1990) Mahadevan et Narendran (1994)
Minimisation du nombre d'équipements utilisés	Jaikumar et Solomon (1990) Egbelu (1987)
Minimisation de la durée moyenne de séjour pour les travaux	Mahadevan et Narendran (1994) Bozer et Yen (1996)
Maximisation du volume ou du volume moyen de production (nombre moyen de travaux finis)	Raju et Chetty (1993) Mahadevan et Narendran (1994)
Minimisation de la longueur maximale des files d'attente	Bozer et Yen (1996) Klein et Kim (1996)
Minimisation du temps moyen d'attente	Bartholdi et Platzman (1989) Mahadevan et Narendran (1994)
Minimisation de la distance totale parcourue = Minimisation du temps de	Blair et al. (1987)

Critère d'optimisation	Parmi les contributions
transport	
Minimisation de la date de fin des travaux	Klein et Kim (1996)
Minimisation de la somme des retards	Akiturk et Yilmaz (1996)
Minimisation de la date de fin de l'ordonnancement	Langevin et al. (1996)
Minimisation du nombre de travaux en retard	Agnetis et al. (1997)
Minimisation des encours	Agnetis et al. (1997) Occena et Yakota (1991)

2.3.1. Cas statique

Les sections suivantes présentent les contributions sur les problèmes statiques d'ordonnancement de la manutention catégorisés par le type d'implantation.

2.3.1.1. Problèmes d'ateliers implantés en ligne

Parmi les premiers travaux sur l'ordonnancement de la manutention dans les ateliers monogammes implantés en ligne, on trouve les travaux d'ordonnancement intégré présentés dans la section précédente. Les travaux en question considéraient l'atelier monogamme de deux machines résolu optimalement par Johnson (1954). L'objectif considéré était la minimisation de la date de fin de l'ordonnancement (C_{max}). La différence parmi les différentes contributions était au niveau des contraintes prises en compte, relatives au système de manutention telles que le nombre d'équipements de manutention disponibles et l'existence d'une file d'attente entre les deux machines.

Dans le milieu industriel, les ateliers monogammes sont plus grands que cette version souvent étudiée dans la littérature. On les rencontre fréquemment dans l'industrie automobile. Ils sont de très grande taille et demandent une gestion judicieuse des tâches

de manutention. En général, ils sont divisés en sous-ateliers responsables de plusieurs sous-assemblages. Dans ce contexte de production, les chariots automatiques peuvent être dédiés aux lignes d'assemblage, ce qui immobilise la ressource pour une longue période. En plus de la minimisation de la date de fin, la minimisation du nombre de chariots utilisés est un objectif souhaité. Le chariot peut éventuellement être affecté ailleurs dans l'atelier. Ceci donne des objectifs conflictuels. On voudra donc avoir un équilibre entre le temps mort des machines ainsi que le temps mort des chariots. Bien que ces problèmes considèrent le problème de séquencement de travaux dans le système, nous les catégorisons comme étant des contributions en ordonnancement de la manutention. La raison étant que les auteurs considèrent explicitement un objectif relié au système de manutention en plus de l'objectif général (min C_{max}) souhaité par le problème d'ordonnancement de travaux sur les machines. Ce dernier est résolu indépendamment des données sur le système de manutention. De plus, les décisions de d'affectation font que les travaux et les équipements de manutention deviennent physiquement une seule entité, et tout phénomène que va subir l'équipement de manutention dans son routage va largement influencer le fonctionnement du système. Ici, seul la décision de l'établissement de l'ordonnancement est considérée.

Egbelu et Roy (1988) et Grasso (1990) considèrent ce problème de grands ateliers aux machines et chariots multiples. Egbelu et Roy (1988) estiment les dates de lancement « **launching** » des chariots pour réaliser l'équilibre souhaité. Grasso (1990) propose trois politiques d'opération pour les chariots automatiques : un lancement « **launching** » au plus tôt, un lancement au plus tard et une politique de « sans attente » devant les machines. L'étude montre que la politique de lancement au plus tard est la meilleure. L'auteur utilise une heuristique pour l'ordonnancement des machines, basée sur les travaux de Dannenbring (1977), qui considère uniquement les données relatives au système de machines. L'ordonnancement est combiné avec une politique d'opération des chariots pour déterminer une solution au problème selon l'ordre d'importance des objectifs souhaités.

Quand les équipements de manutention sont dédiés à une seule ligne de production, un objectif pertinent du problème devient de maximiser le débit. Han et Mc Ginnis (1989) étudient ce problème. Il est équivalent au problème d'ordonnancement des dates d'échéance d'une seule machine avec des temps de traitement qui dépendent de la séquence. Dans leur problème la machine est un transporteur et le temps de traitement est le temps de manutention de la pièce. Ce temps dépend du travail. Ce qui complique le problème est que la date d'échéance dépend également de la séquence. Celle-ci, pour une machine source, est définie comme étant la date de la première occurrence du blocage à la machine source ou de la disponibilité de la machine destination moins le temps de manutention entre les deux machines. Une heuristique est proposée. Le problème d'ordonnancement de la production n'est pas traité.

2.3.1.2. Problèmes d'ateliers implantés en une seule boucle

Tel que présenté dans la section précédente, plusieurs problèmes où l'implantation est faite en boucle étaient traités pour un ordonnancement intégré de la production et de la manutention. Blazewicz et al. (1991) considèrent uniquement l'ordonnancement de la manutention en étudiant un problème de minimisation de la date de fin de l'ordonnancement. Toutefois, ils tiennent compte du respect de la date de fin obtenue pour le problème d'ordonnancement des travaux sur les machines. Un algorithme polynomial détermine si une solution de manutention existe qui respecte l'ordonnancement des travaux sur les machines. L'algorithme utilise la règle de la marge la plus petite.

2.3.1.3. Problèmes d'ateliers implantés en deux boucles

Blazewicz et al. (1994) étudient le même problème en considérant un système à deux boucles de routage des chariots automatiques. Les conflits de routage sont donc considérés. Deux stratégies de routage sont proposées : une qui considère un temps

d'attente unitaire avant le lancement d'un deuxième chariot, et une autre qui calcule ce temps de décalage en fonction des temps des cycles nécessaires pour que le chariot parcoure les deux boucles. De même, ils développent une approche basée sur la programmation dynamique pour vérifier si pour un ordonnancement de production donné il existe un ordonnancement réalisable pour le système de manutention.

2.3.1.4. Problèmes d'ateliers avec une implantation complexe

Les implantations complexes se trouvent généralement dans les environnements de production conçus pour une production du type atelier multigamme où une grande variété de pièces est produite. Elles sont caractérisées par l'existence de plusieurs chemins disponibles pour la manutention d'une charge d'une source à une destination, ainsi que l'existence de plusieurs combinaisons origine-destination à cause de la diversité des plans de traitement des pièces. Le cas où les équipements de manutention sont nombreux est encore plus complexe car les conflits de routage deviennent possibles.

Dans un environnement statique, le routage sans conflit des chariots automatiques correspond à un cas particulier du problème de tournées de véhicules ou encore au problème de tournées de véhicules avec fenêtres de temps (Ganesharajah et al. 1998). L'objectif traité est souvent de minimiser la somme des longueurs des tournées. Les approches de solution applicables pour ce problème sont utilisées pour résoudre des problèmes de chariots. Voir Golden et Assad (1986) qui présentent une revue sur les différentes heuristiques, et Solomon (1987) et Kolen et al. (1987) pour le cas avec fenêtres de temps. Un autre type de problème semblable est celui du transport à la demande (Stein 1978). La distinction principale entre ces problèmes et les problèmes de routage de chariots réside dans les conflits qui peuvent se reproduire sur les arcs, les intersections et les points d'entrée/sortie. De plus, les contraintes supplémentaires sont différentes de celles du contexte des problèmes de tournées de véhicules. Par exemple, les contraintes de capacité de véhicules ne sont pas traitées de la même façon dans le cas

de manutention car celle-ci est rarement effectuée sous forme de collecte. De plus, il n'existe généralement pas de contraintes sur une longueur maximale de tournée et ceci est plutôt déterminé par l'objectif d'optimisation. Dans le cas de manutention, on n'impose pas non plus que le chariot retourne au dépôt initial où il a commencé son voyage. Il suffit juste de satisfaire les requêtes de manutention. Les différentes contributions sur les problèmes d'implantation complexe sont présentées ci-après.

Dans un SMF, Blair et al. (1987) considèrent le routage optimal des chariots automatiques avec l'objectif de minimisation de la longueur maximale des tournées des chariots. La séquence des tâches de manutention affectées à un chariot constitue une tournée. Les conflits ne sont pas considérés. Le problème est formulé comme un problème de plusieurs commis-voyageurs. Les nœuds représentent les demandes en manutention et les arcs bidirectionnels définissent leur séquence. Ils proposent une heuristique de deux phases basée sur les méthodes de séparation et d'évaluation progressive. L'heuristique détermine l'ensemble des tournées qui donneront une longueur totale minimale.

Fuji et Sandoh (1987) considèrent les conflits dans l'étude d'un problème semblable. Ils proposent un algorithme qui détermine le $K^{ième}$ plus court chemin qui ne contient pas de conflit pour chaque chariot. Fuji, Sandoh et Hohzaki (1988) proposent un modèle de contrôle qui minimise les interférences et les temps d'inactivité de n chariots. Lorsqu'un chariot libre reçoit une nouvelle requête de transport, les autres chariots continuent vers leurs destinations préalablement fixées. On décrit ensuite deux approches de routage. Tout d'abord, dans la méthode d'assignation, un itinéraire est produit pour le chariot qui a reçu la nouvelle requête de façon à éviter tout conflit avec les itinéraires déjà fixés des autres chariots. Deuxièmement, dans la méthode de révision chaque chariot ira au prochain nœud dans son parcours et par la suite, on considère de nouveaux itinéraires pour tous les chariots. Dans les deux cas, il s'agit de déterminer pour chacun des chariots dont l'itinéraire doit être établi, les k plus courts chemins. Pour une

combinaison de tels plus courts chemins, (1 par chariot) on doit résoudre un programme linéaire qui fixe les délais maximaux sur ces chemins afin d'éviter tout conflit. Il y a donc k combinaisons dans le cas de la méthode de fixation et k^n combinaisons dans le cas de la méthode de révision impliquant chacune la résolution d'un programme linéaire. Cette approche est semi-dynamique car à chaque point de décision elle résout de nouveaux problèmes statiques de détermination de routes complètes soit pour un ou plusieurs chariots.

Haines (1985) développe un algorithme de routage pour un réseau de design relativement simple. Il n'existe pas plus de deux chemins sortant d'une intersection, et il existe un chemin dans le réseau qui passe une seule fois par l'intersection, la file d'attente et la machine.

Krishnamurthy et al. (1993) étudient le routage sans conflit pour les chariots bidirectionnels. Ils fournissent une formulation en nombres entiers et proposent une heuristique basée sur la génération de colonnes pour la résolution. L'heuristique donne des solutions à 5% de la borne inférieure considérée. Le problème maître de leur formulation est un problème de partitionnement d'ensemble. L'ensemble contient les routes possibles des chariots. Les contraintes sont celles de partitionnement d'ensemble et d'élimination des conflits. Les variables de la formulation sont des routes des chariots. Les variables duales optimales du problème maître relaxé sont transmises à un sous-problème de plus court chemin qui cherche à déterminer une route avec un coût réduit favorable. Cette route est ajoutée à la liste appropriée des routes d'un chariot. La procédure se termine quand on ne peut plus trouver de telles routes pour aucun chariot.

D'autres approches sont présentées pour résoudre ces problèmes. Chung et Fischer (1995) donnent une heuristique basée sur les réseaux de neurones et le recuit simulé pour trouver le plus court chemin sans conflit dans un réseau. Leur méthode donne des résultats près de l'optimum pour quelques exemples.

Akturk et Yilmaz (1996) considèrent le routage sans conflits des chariots automatiques pour minimiser les retards aux moments de ramassage et de livraison de charges. L'objectif est d'effectuer des tâches de ramassage et de livraison juste à temps. Ils proposent une heuristique myope. La performance de l'heuristique a été mesurée en la comparant avec les règles de répartition suivantes : lancement au plus tôt, temps de manutention le plus court (STT), et date d'échéance la plus tôt (EDD). Les résultats montrent que l'heuristique proposée possède une meilleure performance.

Langevin et al. (1996) étudient ce problème en considérant les conflits pour des chariots bidirectionnels. L'objectif est la minimisation de la date de fin de l'ordonnancement (C_{max}). L'objectif de la minimisation des retards est également étudié. Une méthode de programmation dynamique est proposée pour résoudre à l'optimalité les petites instances. La méthode peut être appliquée comme une heuristique pour les problèmes de grande taille de l'industrie. Villeneuve et al. (1999) utilisent une méthode basée sur la génération de colonnes pour résoudre ce même problème. Leur objectif est de minimiser la somme des (C_{max}) de chacune des machines.

Une méthodologie générale basée sur le partage de ressources « **Analytic process of resource** » est présentée par Wu (1998) pour résoudre les problèmes d'ordonnancement des chariots automatiques, les ressources étant les chariots, les segments de route et les files d'attente. Les différents états que peuvent avoir les différentes ressources sont définis. Un algorithme de recherche locale est proposé pour la résolution.

2.3.2. Cas dynamique

Dans un environnement dynamique, les travaux arrivent de façon aléatoire. Il s'en suit qu'on ne peut pas planifier les ordonnancements des machines ni des chariots. Le problème devient plus complexe. D'autres approches sont utilisées telles que la théorie des graphes et les réseaux de Petri. Beaucoup de règles de répartition sont également

proposées ainsi que les systèmes experts et la simulation. Nous consacrons la section suivante aux règles de répartition, vu leur usage répandu dans la résolution des cas dynamiques. Toutefois, ces règles peuvent également être utilisées pour les problèmes statiques.

2.3.2.1. Règles de répartition

Dans les systèmes où les machines, les chariots automatiques et les routages possibles sont nombreux, le pilotage du système devient extrêmement difficile. Des décisions sont à prendre quand plusieurs machines demandent un seul chariot disponible et inversement quand il y a plusieurs chariots disponibles et une seule machine à servir. Ces deux décisions constituent le problème de répartition de chariots. Les principales règles de répartition sont présentées dans les tableaux 2.2. et 2.3. Ces règles sont divisées en deux groupes : celles qui tiennent compte de l'état des chariots et celles qui tiennent compte de l'état des machines (Egbelu et Tanchoco 1984). Les principales études et leurs résultats sont également présentés dans le tableau 2.4.

Des classifications supplémentaires des règles de répartition ont été présentées dans la littérature sur les chariots automatiques. Yim et Linn (1993) font la classification des règles comme étant des règles « **push** » ou « **pull** ». Une règle **push** commence par choisir un travail situé à une machine source et puis détermine la machine destination à laquelle il sera acheminé. Une fois la source et la destination déterminées, un chariot automatique est choisi pour effectuer le mouvement. À ce stade, on peut appliquer les règles qui tiennent compte de l'état du chariot et celles qui tiennent compte de l'état des machines. Une règle **pull** commence par choisir une machine qui peut éventuellement recevoir un travail, et puis détermine une liste des travaux, qui se trouvent dans les files d'attente de sortie d'autres machines, et qui peuvent être acheminés à la machine destination choisie. Une règle de choix d'un travail est appliquée à cette liste pour en choisir un.

Klein et Kim (1996) font la distinction entre les règles de répartition à un seul attribut et les règles de répartition multi-attributs. Une règle de répartition multi-attributs peut combiner des règles mono-attributs avec une pondération associée à chacune d'elles. Ils font une étude qui prouve que les règles de répartition multi-attributs ont une meilleure performance par rapport à celles mono-attributs.

Hwang et Kim (1998) considèrent des règles de répartition plus sophistiquées. Elles tiennent compte de trois informations à la fois : l'état des files d'attente d'entrée/sortie des machines, la durée du voyage du chariot et les encours de production.

Seifert et al. (1998) considèrent le choix des règles de répartition pour un problème donné. Les auteurs évaluent plusieurs règles par simulation avant d'en choisir une. Hoff et al. (1998) présentent une revue sur les règles de répartition et un autre problème connexe qui est celui du design du réseau.

Le choix d'une règle appropriée de répartition est une tâche importante. Il n'existe pas une seule règle qui aura toujours la meilleure performance pour tous les critères d'optimisation et pour toute implantation d'atelier. Un mauvais choix de règles de répartition peut créer des problèmes de blocage. Ces derniers se produisent quand les règles de répartition sont des fonctions des mesures de distance (temps de manutention le plus court, temps de manutention le plus long). Dans ce cas, quelques départements peuvent ne pas être bien servis. La performance de ces règles de répartition dépend largement de l'implantation du système en question qui détermine la localisation des machines et des points d'entrée/sortie. Éventuellement, les files de sortie peuvent devenir complètement remplies et par la suite les machines seront bloquées. Ceci est suivi par un blocage des chariots qui font des livraisons dans ces départements quand la file d'entrée sera à sa capacité maximale. Cette réaction en chaîne peut se propager à tous les chariots dans l'atelier, ce qui résultera en un blocage total de tout le flot des matériaux. Egbelu et Tanchoco (1984) expliquent en détail le phénomène de blocage

d'atelier. Yim et Linn (1993) expliquent que pour éviter le blocage en atelier il faut diminuer le nombre de travaux dans le système. Le fait d'avoir beaucoup de travaux dans l'atelier résulte en une congestion dans l'atelier de production. De même, avoir peu de travaux dans le système diminue le taux moyen de sortie.

Tableau 2.3. Sommaire des principales règles de répartition des chariots automatiques - Règles basées sur l'état des chariots

Abréviation anglaise	**Définition anglaise de la règle**	**Définition française de la règle**	**Abréviation française**
RW	Random Work center	Poste de Travail au Hasard	PTH
STT	Shortest Travel Time	Temps de Manutention le plus Court	TMC
LTT	Longest Travel Time	Temps de Manutention le plus Long	TML
MOQS	Maximum Outgoing Queue Size	Longueur Maximale de la File de Sortie	LMFS
MROQS	Minimum Remaining Outgoing Queue Size	Longueur Minimale du Restant de la File de Sortie	LMRFS
FCFS	First Come First Served	Premier Arrivé Premier Servi	PAPS
MFCFS	Modified First Come First Served	Modification du Premier Arrivé Premier Servi	MPAPS
LQS	Largest Queue Size	File la Plus Longue	FPL
VLFW	Vehicle Looks For Work	Chariot Cherchant du Travail	CCT
LWT	Longest Waiting Time	Temps d'Attente le Plus Long	TAPL
LIT	Longest Inter-arrival Time	Temps Entre Arrivées le Plus Long	TEAPL
MWQ	Minimum Work-in-Queue	Travail Minimum en File	TMF
MD	Maximum Demand	Demande Maximale	DM
FEFS	First Encountered First Served	Premier Rencontré Premier Servi	PRPS
ULSAT	Unit Load Shop Arrival Time	Temps d'Arrivée de l'Unité de Charge en Atelier	TAUCA

Tiré, sans traduction, de Ganesharajah et al. (1998).

Tableau 2.4. Sommaire des principales règles de répartition des chariots automatiques - Règles basées sur l'état des machines

Abréviation anglaise	Définition anglaise de la règle	Définition française de la règle	Abréviation française
RV	Random Vehicle	Chariot au Hasard	CH
NV	Nearest Vehicle	Chariot le Plus Proche	CPP
FV	Farthest Vehicle	Chariot le Plus Loin	CPL
LIV	Longest Idle Vehicle	Chariot avec un Temps Mort Maximum	CTMM
LUV	Least Utilised Vehicle	Chariot le Moins Utilisé	CMU
FAFS	First Available vehicle	Chariot Disponible en Premier	CDP
MIT	Most cumulative Idle Time	Cumul MaXimum des Temps Morts	CMXTM
LIT	Least cumulative Idle Time	Cumul MiNimum des Temps Morts	CMNTM

Tiré, sans traduction, de Ganesharajah et al. (1998).

Un sommaire des études sur les règles de répartition des chariots automatiques a été présenté par Ganesharajah et al. (1998). Dans le tableau (2.5), nous y ajoutons plusieurs autres études.

Tableau 2.5. Sommaire des études sur les règles de répartition des chariots automatiques

Auteurs	Politique	Méthodologie	Objectif	Résultats Principaux
Egbelu et Tanchoco (1984)	Chariot le plus proche	Simulation	Maximisation du débit	Phénomène de blocage identifié Même performance pour toutes les règles qui dépendent de l'état des machines
	Chariot le plus loin			
	Chariot avec un temps mort maximum			
	Longueur maximale de la file de sortie			
	Temps de manutention le plus court			
	Temps de manutention le plus long			
	Longueur minimale du restant de la file de sortie			
	Modification du premier arrivé premier servi			
Russel et Tanchoco (1984)	La file la plus longue	Simulation Q-Gert	Minimisation de la durée moyenne de séjour Maximisation de l'utilisation des machines Maximisation de l'utilisation des chariots Minimisation de la longueur maximale des files d'attente	Politique n'influence pas la durée moyenne de séjour, ni le taux d'utilisation de machines et des chariots automatiques Elle influence la longueur des files de sorite
	Charge avec la plus longue durée d'attente			
	Charge la plus proche			
	Affectation au hasard			

Auteurs	Politique	Méthodologie	Objectif	Résultats Principaux
Cheng (1987)	Chariot disponible en premier	Simulation		Première et quatrième règles sont supérieures Deuxième et troisième règles n'ont pas une bonne performance Cinquième règle a une performance moyenne La vitesse des chariots influence largement la performance
	Chariot avec un temps mort maximum			
	Chariot avec un temps mort minimum			
	Chariot le plus proche			
	Chariot le plus loin			
Egbelu (1987)	Règles Push et Pull	Simulation	Maximisation du débit Minimisation du C_{max} Minimisation du nombre des chariots	Compétition entre les règles push et pull
Hodgson et al. (1987)	Servir le poste avec le poids le plus élevé	Processus décisionnel semi-markovienne	Maximisation du débit	Performance près de l'optimal. Performance meilleure que celle de la règle (chariot cherche travail)
Occena et Yokota (1991)	Règle de demande maximale	Simulation	Maximisation du débit Minimisation de la moyenne des encours	Meilleure que les règles utilisées précédemment pour les systèmes juste à temps

Auteurs	Politique	Méthodologie	Objectif	Résultats Principaux
Sabuncuoglu et Hommertzheim (1992)	Premier arrivé premier servi Temps de manutention le plus court Longueur maximale de la file de sortie La file la plus longue Travail minimal restant Nombre minimal d'opérations restantes	Simulation	Minimisation de la durée moyenne de séjour	Temps de manutention le plus court, et la file la plus longue possèdent la meilleure performance
Sabuncuoglu et Hommertzheim (1992)	Hiérarchie à quatre niveaux	Simulation	Minimisation du retard moyen Minimisation de la durée moyenne de séjour	Performance meilleure par rapport aux règles étudiées précédemment
Sabuncuoglu et Hommertzheim (1993)	Premier arrivé, premier servi La file la plus longue	Simulation	Plusieurs objectifs des dates d'échéance	La file la plus longue possède une meilleure performance par rapport au premier arrivé premier servi
Raju et Chetty (1993)	Chariot le moins utilisé Chariot le plus proche Chariot avec un temps mort maximum Chariot au hasard	Réseaux Petri	Maximisation de l'utilisation des machines Maximisation de l'utilisation des chariots Minimisation de la longueur maximale des files d'attente Maximisation des volumes de production	Chariot le plus proche est meilleure Chariot avec un temps mort maximum, et Chariot le moins utilisé, réalisent l'équilibre des charges pour les chariots

Auteurs	Politique	Méthodologie	Objectif	Résultats Principaux
Yim et Linn (1993)	Règles Push et Pull	Réseaux Petri et Simulation	Maximisation du taux moyen de sortie	Réduction de la taille de la palette pour éviter le blocage dans l'atelier
Mahadevan et Narendran (1994)	Chariot moins utilisé	Simulation	Minimisation de la durée moyenne de séjour	Chariot le plus proche possède une meilleure performance Répartition sur des chemins fixes possède une mauvaise performance
	Chariot le plus proche		Minimisation de la Longueur moyenne de la file d'attente	
	Chariot disponible en premier		Maximisation de l'utilisation moyenne des chariots	
	Répartition sur des chemins fixes		Minimisation de la durée moyenne d'attente pour les chariots	
	Répartition sur des chemins fixes modifiés		Maximisation du volume moyen de production	
Shang (1995)	Premier arrivé, premier servi Distance de manutention la plus courte La file la plus longue	Simulation Méthodes Taguchi (Méthode de surface de réponse)	Maximisation du débit	La file la plus longue >>> Distance de manutention la plus courte>>> Premier arrivé, premier servi

Auteurs	Politique	Méthodologie	Objectif	Résultats Principaux
Bozer et Yen (1996)	Modification du temps de manutention le plus court Des règles basées sur plusieurs informations	Simulation	Maximisation de l'utilisation de l'équipement Minimisation de durée moyenne de séjour Minimisation de la durée moyenne d'attente en file Minimisation de la longueur maximale de la file	Les deux ont une meilleure performance par rapport au temps de manutention le plus court
Klein et Kim (1996)	Règles d'un seul attribut et règles multi-attributs	Théorie des ensembles floues et Simulation	Minimisation de la longueur moyenne et maximale de la file d'attente Minimisation de la durée moyenne et maximale d'attente d'une charge dans une file d'attente Minimisation de la durée des voyages à vide du chariot Minimisation de la date de fin des travaux	Les règles multi-attributs ont une meilleure performance
Agnetis et al. (1997)	Règles Pull et Push et Matching	Simulation d'une usine réelle	Minimisation du nombre de travaux tardifs Minimisation des encours	Matching ont une meilleure performance. Nécessité de modifier système d'information
Hoff et al. (1998)	Une revue sur les règles de répartition	----	----	----
Seifert et al. (1998)	Évaluation des stratégies de routage statique et dynamique	Simulation	Minimisation du retard relatif	L'approche dynamique est supérieure

Auteurs	Politique	Méthodologie	Objectif	Résultats Principaux
Sabuncuoglu et Hommertzheim (1998)	Premier arrivé premier servi Temps de manutention le plus court Longueur maximale de la file de sortie La file la plus longue Travail minimal restant Nombre minimal d'opérations restantes	Simulation dans le cas de bris de machines/ chariots	Minimisation de la durée moyenne de séjour	Au cas de bris, le système est très sensible au choix des règles. Leur performance dépend des conditions d'opérations et de la distribution des temps d'opération

Augmenté à partir des travaux de Ganesharajah et al. (1998).

2.3.2.2. Problèmes d'ateliers implantés en une seule boucle

Bartholdi et Platzman (1989) considèrent un atelier dynamique implanté en une seule boucle et servi par un seul chariot automatique. L'atelier est composé de plusieurs machines et plusieurs points d'entrée/sortie. L'objectif considéré est de minimiser le temps moyen d'attente. Ils ont développé deux règles de décision : premier arrivé, premier servi et premier rencontré, premier servi. Cette dernière a montré une meilleure performance.

2.3.2.3. Problèmes d'ateliers implantés en plusieurs boucles

Des ateliers monogammes d'un et de plusieurs boucles sont étudiés par Ganesharajah et al. (1996a). Ils considèrent des chariots unidirectionnels. Trois objectifs sont considérés : la maximisation du débit, la minimisation de l'utilisation des chariots automatiques et la minimisation de la taille de la flotte en chariots automatiques. La performance du système est étudiée en appliquant les trois règles de répartition proposées par Grasso (1990) et présentées à la section 2.3.1.1. Des conditions sont identifiées où les configurations tandems deviennent favorables.

Agnetis et al. (1997) étudient l'ordonnancement des chariots automatiques dans un problème d'une ligne d'assemblage flexible dans l'industrie automobile. Les travaux arrivent de façon dynamique et les chariots automatiques sont gérés par trois types de règles de répartition qui sont les règles **pull, push** et **matching**. Les règles **matching** font partie des règles **push** et elles tiennent compte non seulement de l'état des machines à l'étape suivante mais aussi de l'état des pièces qui sont en cours de traitement. La ligne d'assemblage est composée de plusieurs boucles qui constituent différents sous-assemblages. Par la suite, les sous-produits fournis par ces boucles sont assemblés sur la ligne centrale d'assemblage. Des retards sont identifiés dans quelques boucles et les nouvelles règles de répartition sont appliquées dans ces boucles pour améliorer la performance de la ligne au complet.

2.3.2.4. Problèmes d'ateliers avec une implantation complexe

Dans les implantations complexes, conçues pour traiter différents types de travaux, le problème est très difficile surtout si on considère les conflits de circulation. Kim et Tanchoco (1991,1993), Huang et al. (1993), Dowsland et Greaves (1994), et Taghaboni-Dutta et Tanchoco (1995) considèrent le problème où un plan de routage pour un chariot est déjà établi pour un problème statique. Ils considèrent sa modification quand un nouveau travail est introduit dans l'atelier.

Kim et Tanchoco (1991) utilisent la théorie des graphes pour résoudre le problème. Un graphe de fenêtres de temps est construit de telle façon que les conflits sont pris en compte. Les arcs existent seulement entre les nœuds physiques si un conflit éventuel ne peut pas se reproduire. Les nœuds représentent les fenêtres de temps libres obtenues par l'ordonnancement établi pour un ensemble initial de travaux. Le routage d'un chariot est effectué selon le plus court chemin dans le graphe, trouvé en temps polynomial par l'algorithme de Dijkstra (1959).

Kim et Tanchoco (1993) fournissent une stratégie myope dans laquelle un seul chariot est considéré à la fois. Le chariot est affecté à la dernière demande de service seulement quand il devient inactif. Les conflits sont réglés par la méthode de Kim et Tanchoco (1991).

Huang et al. (1993) présentent une approche similaire à celle de Kim et Tanchoco (1991). Elle permet de traiter le problème de conflit de façon simultanée. Dowsland et Greaves (1994) calculent k plus courts chemins dans le réseau physique avec l'algorithme de Yen (1971). Pour éviter les conflits, ils utilisent les « **spurs** » et ils considèrent un délai entre le lancement consécutif des chariots. Les k plus courts chemins qui donnent le temps total le plus court en considérant les délais d'attente sont retenus. Cette approche est heuristique car les chemins ne sont pas tous explorés.
Taghaboni-Dutta et Tanchoco (1995) proposent une stratégie myope. Elle procède par le choix des segments de routes que va parcourir le chariot, un à la fois, jusqu'à l'arrivée à la destination.

D'autres approches sont proposées. Zeng et al. (1991) présentent des contributions qui utilisent les réseaux attribués de Petri pour un problème où les conflits de routage sont considérés. Lee et Lin (1995) utilisent les réseaux colorés de Petri qui permettent d'incorporer les aspects dynamiques. Dhouib et Kadi (1994) utilisent les systèmes experts pour un problème dynamique en implantation complexe. Ils proposent des stratégies pour éviter les conflits.

Hao et Lai (1996) utilisent les réseaux de neurones pour un problème de chariots en mouvement libre. L'approche est quasi-temps réel car les routes sont proposées à compter d'une deuxième demande de service. Ceci se fait dans le but de fabriquer de meilleures routes en se basant sur plus d'informations (2 demandes de service au lieu d'une seule). Ils étudient quatre cas :

- un seul chariot et pas de files d'attente ;

- plusieurs chariots et pas de files d'attente ;
- un seul chariot et des files d'attente ;
- plusieurs chariots et des files d'attentes.

Yeh et Yeh (1998) utilisent les graphes orientés pour prévoir et éviter, en temps réel, les conflits de routage dans le cas de chemins unidirectionnels. Lee et al. (1998) utilisent des modèles de réseaux pour calculer k-chemins plus courts pour en choisir un, sans conflit, en temps réel. Narasimhan et al. (1999) analysent à travers une simulation le ré-routage des chariots qui rencontrent des perturbations.

Des méthodes orientées objet, robustes dans un environnement où la production se fait à partir des commandes des clients sont présentées par Shah et al. (1997). Les auteurs considèrent la détermination dynamique des routes tout en tenant compte des conflits de routage.

Les deux approches statique et dynamique utilisées pour l'ordonnancement des systèmes de chariots automatiques sont discutées par Rajotia et al. (1998). Ils proposent également un algorithme semi-dynamique de fenêtre de temps pour fabriquer les routes des chariots dans un réseau bidirectionnel. L'algorithme proposé tient compte de l'état actuel de la circulation pour calculer les plus courts chemins sans conflits. Il s'agit d'un algorithme dynamique car il considère des routes alternatives pour effectuer une mission de manutention donnée. Contrairement à une approche statique qui choisit une seule route fixe entre les mêmes deux points, l'algorithme proposé peut choisir différentes routes reliant deux points selon l'état de la circulation. Le caractère semi-dynamique de l'algorithme est dû au fait que les routes (constituées à partir d'un ensemble de segments) déjà sélectionnées pour des missions données ne seront pas changées plus tard. Comme présenté précédemment, dans un algorithme complètement dynamique, les routes seraient fabriquées progressivement à tous les nœuds pour construire un chemin pour un chariot et non pas seulement au moment du début du voyage.

Contrairement aux contributions précédentes qui supposent l'existence d'un réseau routier, Taghaboni et Tanchoco (1988) ont proposé une stratégie de contrôle en temps réel pour un problème de chariots automatiques en mouvement libre « free ranging ». Elle consiste à utiliser des règles heuristiques de répartition. De plus, on prévoit une route pour accomplir la nouvelle mission de façon à éviter les conflits routiers avec les autres véhicules dont les itinéraires ont déjà été fixés. La résolution de ces conflits s'effectue de manière à minimiser les délais occasionnés.

Kim et Hwang (1999) proposent un processus évolutionnaire pour la répartition adaptative des chariots autoguidés qui tient en compte les changements dans le système.

2.3.3. Problèmes connexes

Un problème connexe relié aux problèmes d'ordonnancement de la manutention est celui du **positionnement d'un chariot en attente d'une demande de service**. Ce problème a été étudié par Egbelu (1993) pour le cas d'un seul et de plusieurs chariots pour un atelier en boucle. L'auteur considère les deux cas : unidirectionnel et bidirectionnel. Une hypothèse de son étude est que les demandes de service pour les chariots sont statiques. Chang et Egbelu (1995) considèrent le cas où les demandes de service changent de façon dynamique à chaque poste de travail. Kim et Kim (1997) étudient ce problème pour le cas de plusieurs chariots et une implantation en boucle unidirectionnelle. Une procédure est suggérée pour déterminer où localiser les chariots inactifs dans le but de minimiser le temps moyen de réponse à une requête de livraison arbitraire. Ce problème devient très difficile à résoudre dans les implantations complexes et dans le cas de plusieurs chariots (Egbelu 1993).

Le problème de positionnement de chariots inactifs apparaît parce que l'établissement d'ordonnancements des chariots considère généralement un ensemble fini de travaux à ordonnancer ou encore une entrée dynamique des travaux. Il s'en suit qu'après que

l'ordonnancement complet soit établi ou après qu'un travail soit lancé, on ne sait pas quel serait le meilleur positionnement des chariots. Quand la nature des tâches des chariots est répétitive, le problème est analysé autrement. Zaremba et al. (1997) présentent une approche pour le pilotage distribué des systèmes des chariots automatiques répétitifs. Le (max-algebra) est utilisé pour modéliser le système. La méthode exploite le caractère répétitif du flot des chariots. Le système est traité comme un système distribué composé de flots autonomes et répétitifs pour les différents chariots.

Une autre façon à aborder le problème d'ordonnancement des chariots dans les implantations complexes est de considérer diverses configurations. Les configurations tandem sont proposées par Bozer et Srinivasan (1991) pour faciliter la gestion des chariots automatiques. Elles empêchent les problèmes de conflits qui sont difficiles à résoudre. Il s'agit d'un nombre de boucles fermées dans lesquelles un seul chariot est utilisé. Il y a une file d'attente entre les boucles pour permettre les transferts de charges. Mais, cette approche possède aussi des inconvénients. À titre d'exemple, la manutention d'une charge nécessite plus d'un chariot à cause des limites de zones d'opération de chacun des chariots. Un temps d'attente plus long est donc prévu dans les files d'attente. Il existe aussi le problème de choix du chariot ou bien de la boucle quand on veut transférer une charge d'une boucle à une autre. Ce problème est appelé « load routing » et a été traité par, entre autres, Lin, Chang et Liu (1994) et Lin et Dgen (1994). Des comparaisons entre les deux types de configurations : conventionnelles et tandems ont été effectuées par Bozer et Srinivasan (1992) et Wang et Hafeez (1994).

2.4. Critique et conclusion

Étant donné que les travaux sur le problème d'ordonnancement intégré sont relativement récents, on remarque qu'ils traitent un nombre restreint d'objectifs. Des critères

réguliers d'optimisation sont toujours considérés. L'objectif habituel est celui de la minimisation de la date de fin de l'ordonnancement.

Les approches analytiques sont utilisées pour les problèmes statiques uniquement. Les problèmes dynamiques sont abordés par une méthode en ligne qui possède une vision à très court terme (utilisation d'une information partielle) et dont la performance n'est pas étudiée à l'aide de données réalistes. Dans les diverses contributions étudiées, une voie de recherche future suggérée par les auteurs est d'étudier la performance des algorithmes dans un environnement dynamique. Shah et al. (1997) présentent des méthodes orientées objet pour l'ordonnancement de la manutention dans un environnement où la production se fait sur commande des clients. Une simulation d'un cas réel a démontré la bonne performance de la méthode sauf que l'ordonnancement des travaux sur les machines n'est pas considéré de façon intégrée.

Les temps de chargement/déchargement sont souvent négligés ou bien ajoutés au temps de manutention, s'ils sont considérés. Dans un cas pratique, ils peuvent dépendre du travail ainsi que de la machine. Les temps de réglage des machines ne sont considérés que par Levner et al. (1995). De plus, les contributions apportées dans la littérature traitent beaucoup de cas où les conditions d'opérations sont particulières, comme le retour des travaux à une zone de chargement/déchargement après chaque traitement.

Il n'existe pas une formulation efficace pour décrire le problème. Bilge et Ulusoy (1995) combinent deux formulations classiques d'ordonnancement de machines et d'ordonnancement de véhicules par des contraintes de fenêtres de temps. Dans le cas où les pièces sont retournées à un dépôt après chaque traitement sur les différentes machines les durées des tâches de manutention ne dépendent pas de la séquence. Ce cas est modélisé par Raman et al. (1986) en utilisant une formulation semblable à celle utilisée dans les problèmes d'ordonnancement de projets avec contraintes de ressources.

Une autre formulation a été présentée par Anwar et Nagi (1998) pour un problème d'atelier d'assemblage.

En général, une seule option en routage est considérée. On n'a souvent pas les données sur le design du réseau routier. Ce sont les temps de manutention qui sont considérés plutôt que des tournées pour lesquelles il faut choisir un ensemble de segments de routes en évitant les conflits. Les conflits de routage sont considérés dans peu de cas (Anwar et Nagi, 1997). Il s'en suit que l'applicabilité des méthodes de résolution aux réseaux uni/bidirectionnels n'est pas prise en compte dans les différentes contributions.

La contrainte sur la capacité des files d'attente est présentée seulement par Sabuncuoglu et al. (1998). Elle est en fait contournée par l'ajout d'une file d'attente centrale vers laquelle on déplace les travaux dans le cas où les files d'attente des machines seraient remplies.

Bilge et Ulusoy (1995) ont considéré une approche de solution itérative où on alterne entre l'établissement d'un ordonnancement des machines et l'établissement d'un ordonnancement des équipements de manutention. Cette approche n'intègre pas toutes les décisions relatives au problème à la fois. Par contre, elle permet de profiter des avancements algorithmiques dans les deux domaines d'ordonnancement de la production et de la manutention.

Dans les systèmes opérés selon une politique particulière, les décisions d'affectation sont considérées a priori et elles ne sont pas intégrées dans la résolution, ce qui contredit l'essence de l'ordonnancement intégré.

Les contributions en ordonnancement de la manutention, bien qu'éclaircissant plusieurs aspects du problème réel, présentent, pour la majorité des cas, des solutions approximatives.

Les problèmes académiques d'ordonnancement intégré sont d'une nature différente de ceux rencontrés en industrie. Plus de difficultés sont présentes en milieu industriel. Parmi les raisons, il y a :

- l'opération d'un grand nombre de chariots;
- les lots de production qui ne sont pas égaux aux lots de transfert. Des considérations de volumes et de poids sont prises en compte en industrie;
- les conditions d'opération et les spécificités technologiques quant aux vitesses des chariots, les chariots en mouvement libre, et les voies d'évitement;
- quelques machines non disponibles au temps zéro;
- les commandes qui viennent perturber un ordonnancement de production, et pour lesquelles il faut déterminer des dates de lancement.

Il est important de développer des méthodes efficaces et exactes pour résoudre le problème d'ordonnancement intégré de la production et de la manutention. Cette étape réalisée, il sera par la suite plus facile d'essayer d'intégrer d'autres aspects du problème réel.

Tel que présenté, plusieurs outils de résolution ont été utilisés pour aborder le problème d'ordonnancement intégré, mais il reste des méthodes qui n'ont pas été étudiées dans ce contexte comme la programmation dynamique, la génération de colonnes, quelques méta-heuristiques et le « Shifting bottleneck procedure » présenté par Adams et al. (1988). La programmation dynamique peut être intéressante car elle a du potentiel en ce qui concerne les applications en temps réel où les ordonnancements sont établis de façon séquentielle. Les deux outils de résolution présentés dans cette thèse : programmation mathématique et programmation par contraintes, sont utilisés pour la première fois pour résoudre ce problème.

Au chapitre suivant, après une présentation des différentes ressources nécessaires au processus de production, nous discutons de notre choix méthodologique pour les problèmes traités dans cette thèse.

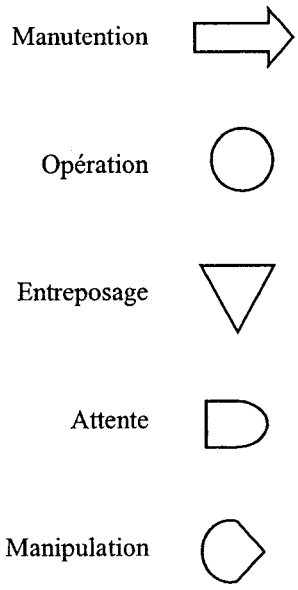
CHAPITRE 3 :

PROBLÈMES D'ORDONNANCEMENT INTÉGRÉ : DÉCOMPOSITION DES RESSOURCES

Dans ce chapitre nous discutons du problème d'ordonnancement intégré en détail. Nous présentons une description du problème d'ordonnancement intégré tel que posé dans un atelier de production. Nous incluons plusieurs hypothèses pour une représentation du problème qui soit proche de la réalité. Cette représentation réaliste met en évidence la nécessité de la disponibilité de différentes ressources pour s'assurer de produire un horaire réalisable. Les différentes ressources nécessaires pour un processus de production sont présentées dans la section 3.1. Dans les chapitres suivants, nous traitons trois problèmes d'ordonnancement intégré de la production et de la manutention où la définition des ressources contraignantes change d'un problème à l'autre.

Comme le problème d'ordonnancement intégré au sens large, peut se poser dans plusieurs environnements de production, nous avons choisi le contexte d'un atelier multigamme où un lot composé d'une ou de plusieurs pièces est acheminé pour passer par plusieurs transformations. Ces pièces ne passent pas par des étapes d'assemblage. Pour représenter le cheminement des pièces dans l'atelier, son graphique déroulement-matière détaillé apparaît à la figure 3.1. Chaque étape présentée sur le graphique est consommatrice de temps et toutes les étapes doivent être prises en compte pour produire un horaire réaliste et faisable. Il existe aussi d'autres activités qui peuvent influencer l'horaire de production mais celles-ci ne sont pas effectuées sur les pièces à produire et donc ne figurent pas sur le graphique déroulement-matière. Le réglage des machines en est un exemple.

Les cinq symboles standard de génie industriel sont utilisés pour construire le graphique déroulement-matière. Ils sont listés ci-après.



Sur ce graphique, nous précisons également la ressource sollicitée par chacune des activités. À titre d'exemple une gamme de 3 opérations (machines) est présentée. Quand plusieurs gammes sont traitées dans le système, le même cheminement est considéré pour chacune des gammes selon le nombre d'opérations à effectuer. Les différentes opérations des différents produits vont avoir un ordre quelconque sur les différentes machines pour déterminer un certain horaire. Selon ce graphique général, il ne s'agira pas de déterminer l'ordre de traitement des lots des différents types de pièces sur chacune des machines. Un ordre de traitement déterminera plutôt l'ordre de chacun des lots de transfert sur les différentes machines. Ainsi, les sous-lots des différents types de pièces peuvent alterner sur les machines de façon à obtenir l'horaire le plus court possible, ce qui peut augmenter le nombre de missions de manutention. Les sous-lots sont introduits pour permettre le chevauchement entre les opérations et donc le raccourcissement de l'horaire.

Dans cette thèse, nous faisons l'hypothèse que la taille des sous-lots est déterminée a priori selon la capacité des plateaux de transfert ainsi que d'autres critères de décision, et qu'elle est égale à la taille du lot de production. Chaque lot correspond à une seule palette. Sur le graphique, x et y possèdent alors une valeur de 1.

Il est à noter que si l'on déterminait la taille du sous-lot, simultanément, en résolvant un problème d'ordonnancement intégré, cela aurait pu donner de meilleurs résultats que dans le cas où l'on considère une taille de lot préétablie. Mais, l'introduction des variables correspondant aux nombres et tailles de sous-lots complexifie le problème de façon importante. La détermination de la taille du sous-lot dans un problème d'ordonnancement intégré peut constituer une recherche future, et se justifiera selon le contexte du problème. Pour les fins d'une modélisation, nous pouvons traiter les sous-lots comme étant des travaux distincts dont la gamme est identique, et donc un lot de production est le même qu'un lot de transfert.

Pour une représentation plus proche du réel, nous supposons qu'un entrepôt de produits en cours, qui ne représente pas une ressource contraignante, est disponible dans l'atelier. Tel que présenté sur le graphique déroulement-matière, l'entrepôt peut être utilisé pour remédier aux situations de blocage lors du processus de fabrication. Cette éventualité n'est pas incluse dans les modèles présentés dans cette thèse, mais l'idée de sa modélisation est présentée au chapitre 6.

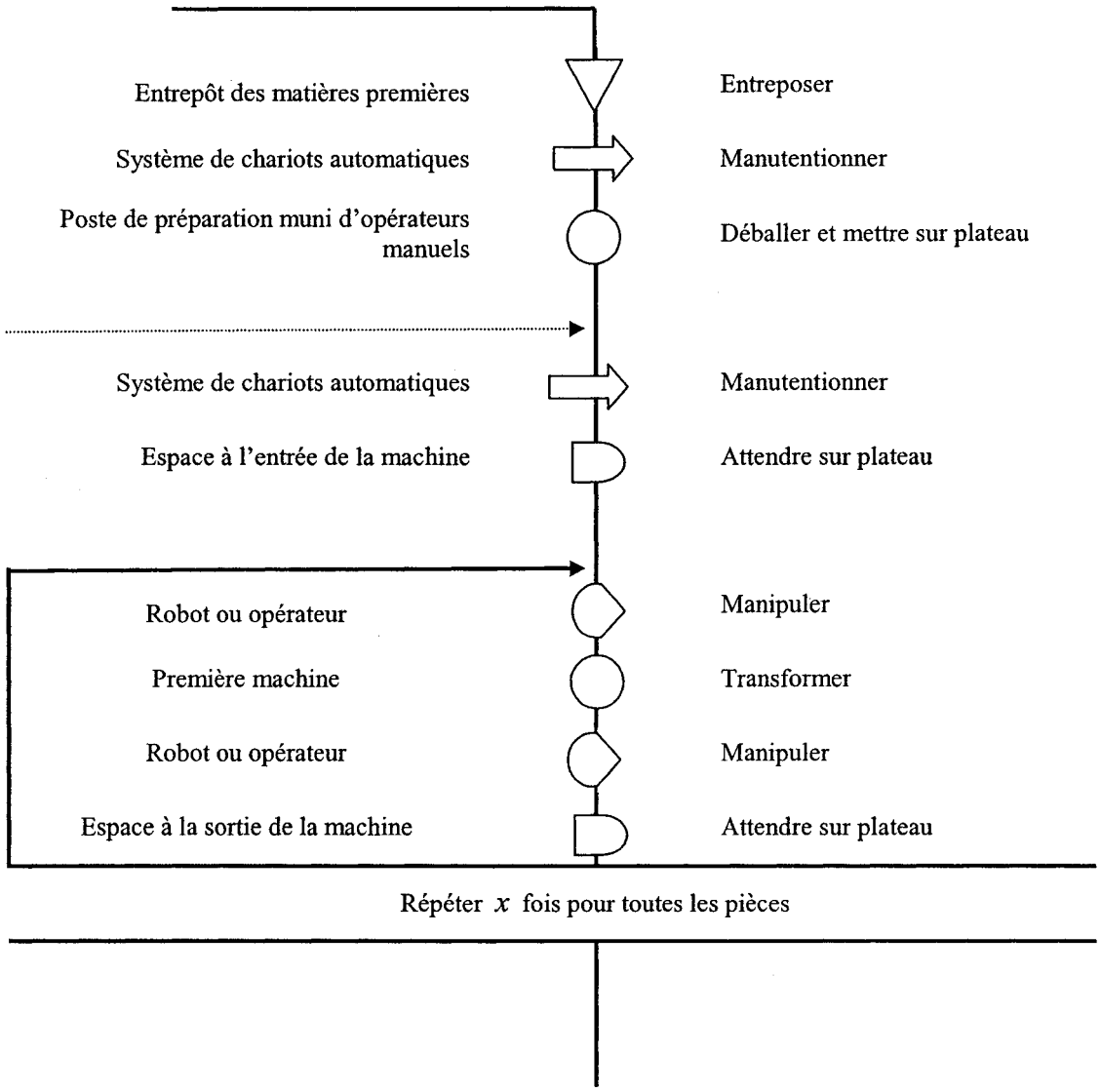


Figure 3.1. Graphique de déroulement-matière pour une gamme de trois opérations

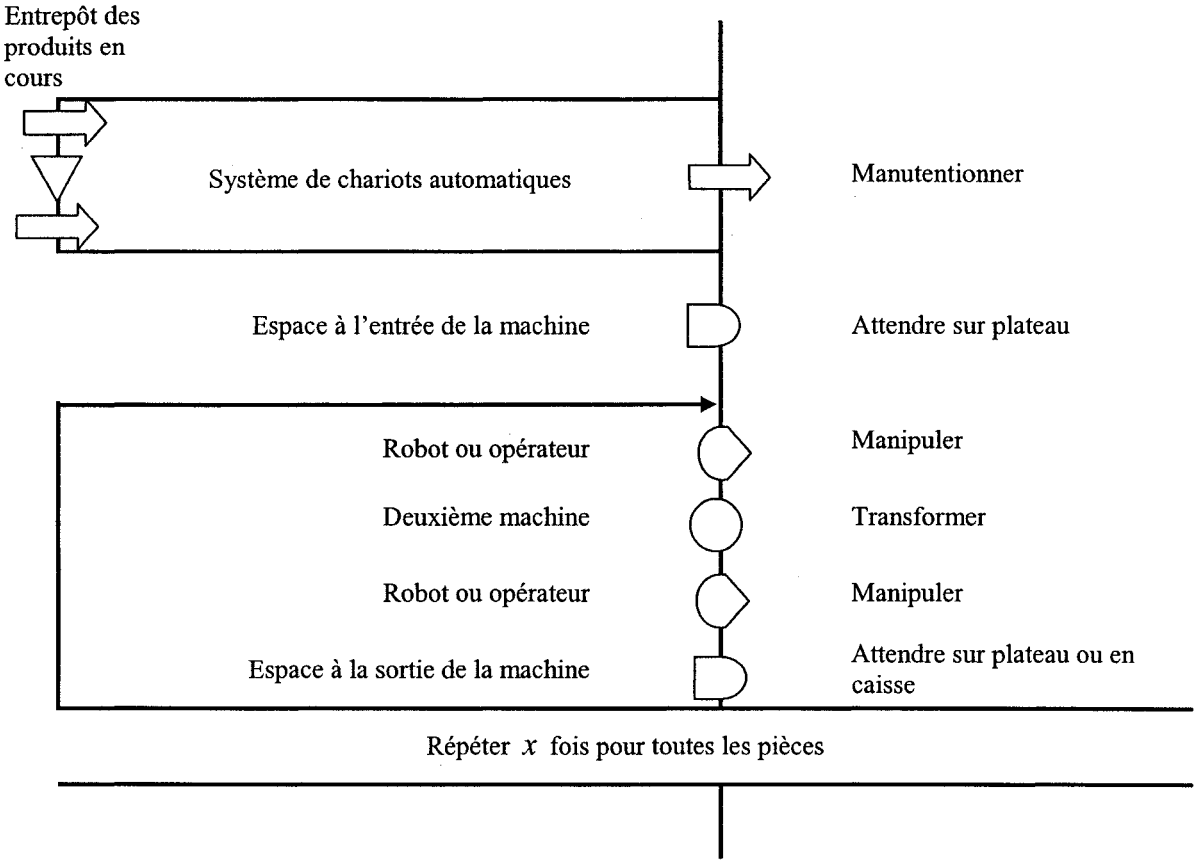


Figure 3.1. Graphique de déroulement-matière pour une gamme de trois opérations (suite)

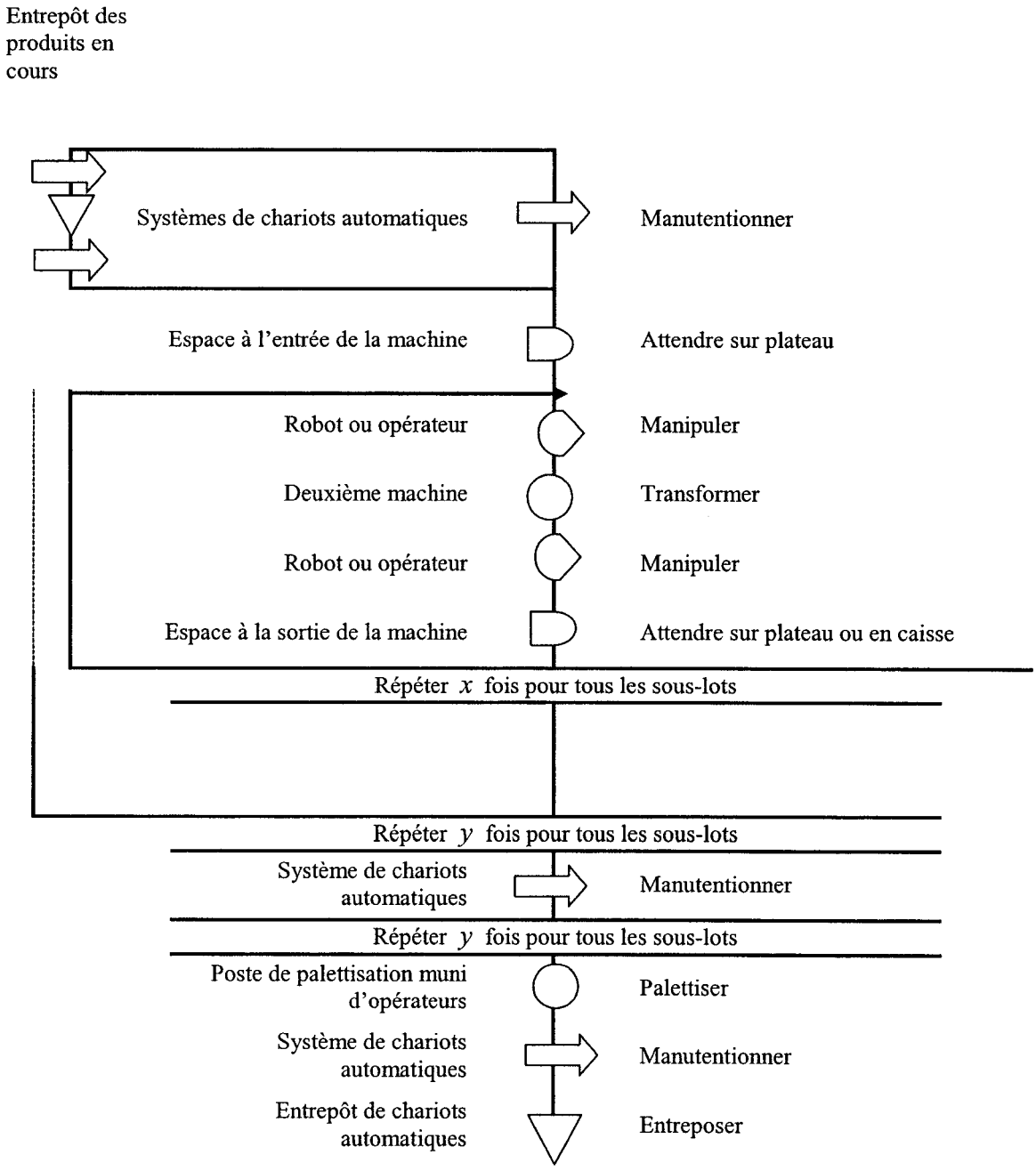


Figure 3.1. Graphique de déroulement-matière pour une gamme de trois opérations (suite et fin)

3.1 Ressources nécessaires pour un processus de production

Les décisions à prendre pour résoudre un problème d'ordonnancement intégré, où les machines, les chariots automatiques, les segments de route et les files d'attentes sont des ressources contraignantes, sont les suivantes :

- la date de début de chacune des opérations de tous les travaux;
- la date de début de la manutention des pièces d'un poste de travail à un autre poste;
- les routes que vont suivre les équipements de manutention avec les dates de début d'occupation de chacun de segments de route ;
- l'affectation des équipements de manutention aux tâches de manutention;
- les durées de séjour dans les files d'attente à l'entrée/sortie des machines.

Les différentes décisions sont à prendre de façon intégrée et elles détermineront l'horaire détaillé. D'autres informations telles que les délais d'attente sur quelques ressources, sont également importantes dans la représentation d'un horaire réel.

Nous reprenons l'ensemble de décisions considérées par notre problème et nous essayons de placer chacune des décisions à son niveau correspondant dans un des schémas conventionnels utilisés pour représenter les systèmes de production. Nous retenons celui de Pinedo (1995) présenté au chapitre 2 à la page 20. Il n'y a que la décision sur la fabrication de l'horaire des machines qui est incluse au niveau ordonnancement. Les autres décisions ne trouvent pas leur place dans le schéma. En fait, les décisions correspondantes à un certain nombre de ressources négligées sont prises à un niveau de décision inférieur dans la hiérarchie dans une zone non précisée encerclée dans le schéma.

L'ensemble de ressources négligées comprend les équipements de manutention, les files d'attente et les segments de routes. Plusieurs autres ressources sont contraignantes dans un système de production. Une ressource peut être définie à partir de l'activité effectuée.

Plusieurs ressources peuvent être considérées du même type si celles-ci sont utilisées pour effectuer une même activité dans un système.

Les ressources peuvent donc être catégorisées en équipements ou opérateurs nécessaires pour effectuer les tâches représentées sur le graphique déroulement-matière. Quelques données peuvent changer telles que les temps standards d'exécution dans les cas de ressources manuelles et automatisées. Les différentes catégories sont présentées dans la section suivante.

3.1.1. Ressources pour la transformation ou l'assemblage

Les activités de transformation ou d'assemblage sont les seules activités qui se déroulent dans un atelier de production pendant lesquelles une valeur s'ajoute aux différents produits. Pour les fins de cette recherche, nous retenons le cas de transformation. Contrairement aux activités de manutention et aux attentes dans un atelier, ces activités ne sont pas à minimiser mais à bien affecter dans le temps, pour produire un horaire qui réalisera les objectifs souhaités. Les activités autres que transformation représentent aussi des coûts au système de production et sont donc à minimiser autant que possible.

La transformation et l'assemblage sont des activités effectuées par des machines conventionnelles, des machines automatisées ou encore par des opérateurs manuels. Dans la majorité des industries, on considère que ces ressources sont statiques et que le produit circule en atelier pour les visiter selon sa gamme.

3.1.2. Ressources pour la manutention

Les activités de manutention assurent la disponibilité du produit au niveau des différentes ressources de : transformation, assemblage, préparation, palettisation ou entreposage. Elles doivent être effectuées de la façon la plus économique pour minimiser

leur coût d'utilisation et en même temps assurer leur contribution à la réalisation des objectifs globaux souhaités par le système de production au complet.

Les ressources nécessaires pour effectuer la tâche de manutention sont matérialisées par plusieurs technologies. Ces dernières sont très variées selon l'industrie, la charge de travail et la nature du produit. On y trouve les technologies *fixes* telles que les transporteurs et les convoyeurs qui effectuent les tâches de manutention continue et les technologies *mobiles* qui circulent dans un atelier, effectuant des tâches de manutention discrète, telles que les chariots. Il existe également des opérateurs humains qui effectuent des tâches de manutention de façon manuelle. Les opérateurs sont également utilisés conjointement avec une technologie matérielle pour effectuer des tâches de manutention comme dans le cas des chariots pilotés par un opérateur humain. Quelques technologies de manutention mobiles sont complètement automatiques et circulent seules sans un opérateur, par exemple, les systèmes de chariots automatiques. Un opérateur lointain, un contrôleur de système, supervise et peut contrôler les mouvements des chariots. Un programme informatique peut également être utilisé pour effectuer ce contrôle.

3.1.3. Ressources pour la manipulation

Les ressources de manipulation assurent le déplacement du produit d'une position actuelle, souvent une file d'attente, pour le placer sur la ressource de transformation, sur une palette pour préparer une charge à manutentionner, ou sur un équipement de manutention. L'équipement de manipulation assure également le déplacement du produit de la ressource après que l'opération soit effectuée. Par exemple, la dépose des pièces dans une file d'attente à la sortie d'une machine. La manipulation peut se faire de façon manuelle par des opérateurs humains ou encore de façon automatisée. Les robots représentent souvent la technologie de manipulation automatisée.

Une tâche de manipulation est essentiellement composée d'une prise et d'une dépose du produit et la distance parcourue est souvent négligeable. Dans le cas de cellules robotisées, les tâches de manipulation et de manutention sont généralement combinées et entièrement assumées par le robot.

3.1.4. Ressources en espace : entreposage et circulation

Les espaces d'entreposage sont les files d'attente de différentes machines qui sont capables d'accueillir, à un instant donné, un certain nombre de pièces ou encore des lots de transfert qui sont en cours de traitement. Elles se trouvent à l'entrée/sortie des différentes machines. Elles sont occupées par des plateaux ou des caisses qui sont utilisées pour effectuer les tâches de manutention à l'intérieur de l'atelier quand le produit est en cours de transformation. D'autres espaces d'entreposage pour les produits en cours peuvent également exister et ils sont utilisés pour dégager les routes ou les machines dans le cas de blocage. Ces espaces sont appelés les files d'attente centrales ou les entrepôts de produits en cours. Une file d'attente centrale est considérée à capacité infinie dans la plupart des cas. En d'autres termes, elle possède une capacité suffisante pour éviter tout blocage éventuel.

Les deux autres espaces d'entreposage nécessaires pour le processus de production sont l'entrepôt des matières premières et celui des produits finis.

Les ressources en espace permettant une circulation sont nécessaires pour effectuer les missions de manutention dans un atelier. Il s'agit des segments de routes prédéfinis sur lesquels un chariot peut circuler. Ces ressources peuvent avoir différentes capacités. Un segment de route peut accueillir un ou plusieurs chariots selon la technologie. Il peut aussi permettre une circulation dans une ou deux directions.

3.1.5. Autres ressources

D'autres ressources comprennent des **opérateurs manuels** qui peuvent effectuer les activités de réglage des machines ou d'autres activités spécialisées.

Les **services** tels que l'électricité, le gaz et l'eau constituent également des ressources qui sont sollicitées à différents degrés selon le type d'industrie. Le profil de la consommation peut également varier selon l'évolution dans le temps du processus de production.

Les **outils et les outillages** sont également des ressources contraignantes qui demandent une gestion judicieuse pour minimiser les problèmes dus à l'usure et à la dégénération. Plusieurs articles qui traitent des problèmes d'outils dans les systèmes manufacturiers flexibles ont été présentés dans la littérature. Les problèmes les plus traités sont ceux de la minimisation du nombre de changements d'outils dans le carrousel d'une machine, et la minimisation du nombre d'outils changés par opération de changement (Crama 1997).

Plusieurs contraintes sont reliées à l'exploitation des ressources en production et elles influencent les horaires de production. Parmi celles-ci il y a :

- la maintenance préventive prévue pour les machines;
- le chargement des batteries des chariots automatiques;
- l'usure des outils;
- les conditions de travail des opérateurs humains.

Dans la littérature, le module qui paraît souvent dans les schémas pour représenter l'activité d'ordonnancement contient uniquement les machines comme la seule ressource contraignante.

Selon nos hypothèses, nous représentons les différentes ressources contraignantes dans un module d'ordonnancement. Les différentes décisions rattachées à chacune des ressources sont indiquées en italique.

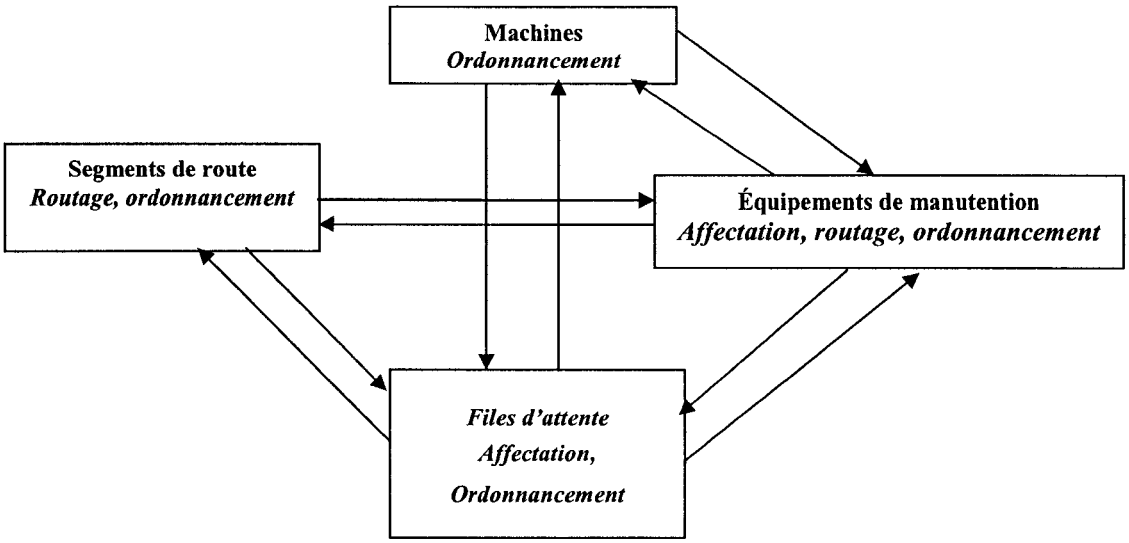


Figure 3.2. Module détaillé d'ordonnancement intégré

L'horaire de la file d'attente indique le séjour des différentes pièces dans la file d'attente à des moments différents. Cette information peut être déduite de l'horaire des machines et des chariots dans le cas où la file d'attente aurait une capacité infinie et donc les décisions d'horaire des machines et des chariots ne peuvent pas être influencées. Dans le cas d'une file d'attente de capacité finie, une machine peut être bloquée à cause d'un manque d'espace dans une file d'attente et par conséquent son horaire sera influencé. Par la même logique, un chariot peut être amené à enlever des pièces dans une file d'attente pour éviter ou remédier à une situation de blocage. En réalité, on veut minimiser les durées de séjour des pièces dans une file d'attente.

Les flèches sur le schéma indiquent une interdépendance parmi les différentes décisions. Les files d'attente se réfèrent aux files d'attente à l'entrée/sortie des machines. Elles peuvent également inclure une file d'attente centrale (entrepôt des produits en cours) qui peut être utilisé dans un cas de risque de blocage. Cet entrepôt ne représente généralement pas une ressource contraignante. Cependant son existence influence la gestion et l'horaire de production dans l'atelier.

3.2. Problèmes traités dans cette thèse : choix des formulations et de méthodes de résolution

Dans cette thèse, nous ne définissons pas a priori un secteur d'activités pour en résoudre des problèmes d'ordonnancement intégré. Par contre, nous précisons qu'il s'agit d'un contexte manufacturier où :

- l'implantation est faite par processus dans un atelier type multigamme conventionnel. Les machines peuvent être entièrement automatiques;
- un réseau de manutention est utilisé pour la circulation des chariots automatiques;
- les chariots automatiques sont de type à une charge. Ils s'auto-chargent et interfacent avec plusieurs équipements.

Plusieurs domaines d'application peuvent être identifiés. Trois problèmes traités dans cette thèse sont décrits au chapitre 4 et présentés en détail aux chapitres 4, 5 et 6. Dans les trois problèmes, les contraintes relatives au système de manutention sont prises en compte à différents niveaux de détail.

Dans un premier temps nous avons examiné la littérature pour trouver des formulations qui peuvent représenter notre problème, au niveau le plus détaillé, dans sa totalité ou en partie. Ceci a été fait dans le but de représenter le problème par une formulation qui a été prouvée efficace au niveau de la résolution. Il est fort probable que, pour une

formulation présentée pour un autre problème, une estimation de certaines données doit être faite pour faire le transfert d'une réalité à une autre.

Le problème d'ordonnancement intégré de la production et de la manutention peut être ramené à d'autres problèmes d'allocation de ressources qui ont été présentés dans différents contextes, notamment en gestion de projets. Cependant, plusieurs distinctions existent. On note, par exemple, la durée variable de séjour dans les files d'attente, la durée variable des tâches de manutention dépendamment des routes et de l'affection de chariots, et aussi la possibilité de permettre l'attente des chariots sur les segments pour éviter un conflit éventuel.

Dans un premier temps, si on considère que les files d'attente et les segments ne sont pas des ressources contraignantes, le problème revient à celui traité dans la littérature par Sabuncuoglu et al. (1998) et par Ulusoy et Bilge (1995) ainsi que par Anwar et Nagi (1998). Les différentes contributions de ces auteurs présentent des solutions heuristiques. Les auteurs ne se servent pas d'un modèle dans la résolution du problème. Nous présentons et résolvons des formulations en programmation mathématique et en programmation par contraintes pour nos trois problèmes.

Des formulations des problèmes d'ordonnancement ont été résolues à l'optimalité par les techniques de génération de colonnes. Gélinas (1997) présente des formulations pour les problèmes d'ordonnancement sur les machines et les problèmes d'ordonnancement de projets. Villeneuve (1999) présente une formulation du problème d'ordonnancement et de routage sans conflits des chariots automatiques.

La modélisation présentée dans le cadre des problèmes d'ordonnancement sur les machines dans la thèse de Gélinas (1997) suppose que l'on possède à chacune des étapes de traitement un nombre de machines parallèles toutes capables de traiter les travaux en question. Les machines parallèles sont identiques (un seul temps d'opération pour

chaque travail). Il n'existe pas d'hypothèse sur le type de relations d'antériorité mais on suppose qu'elles sont peu nombreuses.

Le type de machine est indicé k et donc les k correspondants à des machines ont un nombre n_k=1 et les k correspondants à des chariots ont un n_k= nombre de chariots. Dans un problème réel, une mission de manutention est effectuée par un chariot qui provient d'un certain poste et une fois la mission finie le chariot est repéré à partir du poste de destination. Il reste alors au même point jusqu'à ce qu'il devienne sollicité. Si on relaxe cette hypothèse on considère que l'on rentre à un dépôt central après avoir effectué la tâche de manutention, le problème devient plus simple. On peut alors énumérer les missions de manutention à effectuer par les chariots qui seront considérés comme des tâches sur des machines parallèles.

Mais, on rencontre un problème : la mission de manutention pour un travail ne peut pas commencer à moins que le traitement soit fini sur une machine. Pendant que le chariot vide se rend à la machine source de la mission de manutention, la machine ne doit pas être en traitement du travail. Dans le cas où les allées seraient étroites et le chariot est en déplacement à vitesse maximale, cette représentation ne sera pas très pénalisante.

La formulation proposée par la même auteure (Gélinas, 1997) pour le problème d'ordonnancement de projets, propose de faire le cumul de ressources dans des périodes données, mais les ressources ne sont pas substituables entre elles. Elles sont considérées identiques et divisibles. L'essentiel est de ne pas dépasser la capacité totale permise. Cette formulation ne peut donc pas représenter les choix parmi les ressources en chariots pour les missions de manutention. On pourra donc rencontrer le problème d'une seule mission de manutention partagée entre plusieurs chariots. Il faut donc imposer des contraintes pour que les ressources ne soient pas partagées, et pour la connectivité des chemins.

La formulation des problèmes d'ordonnancement de projets avec contraintes de ressources avec plusieurs modes d'affectation de ressources peut bien représenter le problème d'ordonnancement intégré. Nudtasomboon et Randahawa (1997) traitent ce problème en contexte multi-projet (on considère que chaque travail peut correspondre à un projet). Il existe plusieurs modes de production pour chaque tâche et évidemment les contraintes de ressources correspondantes. La formulation présentée est très claire et représentative pour notre problème incluant les tâches présentées sur le graphique déroulement-matière. Elle permet la prise en compte des différentes ressources contraignantes. Plusieurs modes de production correspondent à plusieurs alternatives pour effectuer les tâches de manutention. Des méthodes heuristiques basées sur le branchement ont été utilisées dans la résolution. Encore ici la connectivité des chemins des chariots est à ajouter à la formulation. L'horizon est discrétisé en de petites unités de temps. La formulation comprend alors un très grand nombre de variables binaires correspondant aux périodes d'accomplissement des tâches.

Dans la formulation proposée par Villeneuve (1999) pour le problème de chariots, des relations d'antériorité sont définies entre les missions de manutention. En même temps des relations d'antériorité sont définies entre la collecte et la livraison pour une palette donnée. Au niveau des machines, un temps d'attente paramétré m est considéré pour éviter les conflits. On suppose l'existence d'un horaire préétabli des machines. L'ajout des familles de contraintes pour le problème d'ordonnancement sur la machine complète la formulation et la rend représentative du problème d'ordonnancement intégré sans garantir la résolution par la méthode proposée par l'auteur.

Le type de relations d'antériorité dans un problème d'ordonnancement intégré ressemble au type des contraintes d'antériorité des problèmes d'ateliers d'assemblage. Une tâche t de manutention à plein est successeur d'une tâche de production sur une machine, et est aussi successeur d'une tâche de manutention à vide correspondant à l'arrivée d'un chariot à la machine où commence cette tâche de manutention t. Il existe cependant une

contrainte supplémentaire. Aux étapes de production correspondant à des voyages de manutention vides ou chargés, il existe plusieurs *machines parallèles* (non identiques, correspondant aux chariots) capables d'effectuer l'opération. Une contrainte supplémentaire est donc nécessaire. Il faut imposer que si l'opération (voyage à vide) est effectuée par un chariot v, l'opération (voyage chargé) successeur doit être effectuée par le même chariot v. Les chemins admissibles sont à énumérer à l'avance. Cette représentation d'atelier d'assemblage (avec une contrainte supplémentaire) peut nous inciter à adapter les méthodes développées pour résoudre les problèmes d'ordonnancement dans le contexte d'atelier d'assemblage. Toutefois, si pour une méthode donnée, on ne considère pas de machines parallèles, il va falloir apporter des modifications. Rhamudin et Marier (1996) ont présenté une contribution pour la résolution heuristique du problème d'ordonnancement dans le contexte d'atelier d'assemblage. Agrawal et al. (1996) ont également présenté des méthodes heuristiques pour l'ordonnancement d'ateliers d'assemblage.

On peut aussi penser à représenter le problème d'ordonnancement intégré, comme un problème de gammes alternatives. Dans ce cas-là, en changeant un chemin ou un chariot pour les missions de manutention, on a des gammes alternatives pour un même travail. Pourtant, dans ces problèmes on suppose que les relations d'antériorité sont comme celles du contexte d'atelier multigamme, ce qui n'englobe pas les vraies contraintes du problème d'ordonnancement intégré. Ce dernier ressemble plus à un problème d'assemblage comme mentionné plus haut. De plus, à notre connaissance, les contributions présentées dans la littérature ne traitent pas de façon simultanée le problème de confection d'horaire et de choix de gamme de fabrication. Les contributions présentées traitent plutôt, de façon simultanée, les problèmes de groupement de pièces et de choix de gamme de fabrication dans un contexte de systèmes manufacturiers flexibles automatisés.

Dans cette thèse, nous avons opté pour des méthodes de résolution exacte dans la recherche d'une solution. D'une part une formulation fournit une abstraction formelle de la réalité qui représente les problèmes de façon élégante. D'autre part, nous avons jugé important d'explorer les limites des formulations et de voir jusqu'où on peut aller dans la recherche de solutions optimales sans juger a priori de la difficulté du problème et opter pour les méthodes heuristiques.

Deux formulations sont proposées pour chacun des problèmes. Une formulation en programmation mathématique et une autre en programmation par contraintes. Selon Lustig et Puget (1999), cette technique s'est avérée très efficace dans la résolution des problèmes d'ordonnancement dans le contexte d'atelier multigamme, surtout dans le cas où l'on intègre plusieurs contraintes réelles au problème conventionnel d'ordonnancement de la production. Comme précisé au début du chapitre, vu la complexité du problème, nous considérons l'objectif de minimisation de la date de fin de l'horaire

Pour la résolution, nous avons opté pour l'utilisation d'un logiciel commercial (OPLStudio). OPLStudio permet la résolution des problèmes de programmation mathématique par Cplex et CplexMIP et des problèmes de programmation par contraintes par les algorithmes appropriés (Solver et Scheduler). Il s'agit d'un logiciel convivial agréable à utiliser par les praticiens. OPLStudio s'interface avec d'autres logiciels pour importer les données et pour exporter les résultats de façon très élégante.

Même s'il ne s'agit pas d'une méthode hybride intégrant la programmation mathématique et la programmation par contraintes, le développement de deux méthodes de résolution tire profit des avantages différents de ces deux techniques. Les deux méthodes ne s'appliquent pas aussi bien pour un même jeu de données parfois. Le praticien trouvera alors au moins une méthode efficace pour résoudre son problème dans

la plupart des cas. Les chapitres suivants présentent les trois problèmes d'ordonnancement intégré traités dans cette thèse.

CHAPITRE 4 :

PROBLÈME D'ORDONNANCEMENT INTÉGRÉ DE LA PRODUCTION ET DE LA MANUTENTION AVEC DES DURÉES PRÉDÉTERMINÉES

Le problème traité dans le cadre de cette thèse est abordé en considérant trois niveaux de détail. L'ordonnancement intégré de la production et de la manutention est traité à chaque niveau. Le premier niveau considère un ensemble de tâches de production et de manutention à ordonnancer sur les machines et les chariots respectivement. La durée des tâches de manutention est prédéterminée. Elle est calculée à partir du plus court chemin entre la machine source et la machine destination de chaque tâche. Les nouveaux éléments pris en compte aux deux autres niveaux concernent les tâches de manutention. Au deuxième niveau, on introduit des choix de routes prédéterminées pour les tâches de manutention et les voyages à vide à effectuer entre deux machines. Les conflits sont évités en définissant l'ensemble de routes conflictuelles qui partagent un ou plusieurs segments. Si deux tâches doivent utiliser deux routes conflictuelles successivement, la première route à utiliser est complètement mobilisée jusqu'à la fin de la première tâche et libérée pour que la deuxième tâche commence. Ceci est également vrai pour les voyages à vide. Le troisième niveau utilise directement la structure du réseau de l'atelier pour fabriquer des tournées sans conflits pour les chariots à partir de l'ensemble de segments de route formant le réseau. Aucun choix en routes n'est déterminé a priori. À ce dernier niveau, on considère également que les files d'attente sont à capacité finie.

Au deuxième et troisième niveau de détail, la durée d'une même tâche de manutention peut varier. Ceci est déterminé par la route à prendre et les conflits à éviter. De même,

le temps de voyage à vide, pris en compte pour assurer la connectivité des tournées des chariots, est aussi variable.

4.1. Problème

Dans nos modèles, on considère un ensemble de travaux à traiter sur un ensemble de machines selon la gamme de chaque travail. Chaque travail est constitué de plusieurs pièces formant un seul lot palettisé. Le traitement d'un travail sur une machine constitue une tâche de production. Les tâches de production génèrent des tâches de manutention qui consistent à prélever une palette, après que le traitement du travail soit fini sur une machine, et la manutentionner pour la livrer à la prochaine machine de sa gamme.

Dans le premier modèle, on considère une durée prédéterminée, correspondant à l'utilisation du plus court chemin sur le réseau pour les tâches de manutention et les voyages à vide. Les voyages à vide dépendent de la séquence des tâches de manutention affectées à un chariot. Les conflits de routage ne sont pas pris en compte. Les files d'attente possèdent une capacité infinie. Deux modèles sont formulés pour décrire cette réalité, l'un en programmation mathématique et l'autre en programmation par contraintes. Dans cette thèse, pour les trois modèles, on considère l'objectif de minimisation de la date de fin de l'horaire.

4.2. Formulation en programmation mathématique

Le modèle de programmation mathématique est un programme linéaire mixte. Plusieurs formulations de ce type ont été présentées dans la littérature pour les problèmes d'ordonnancement dans un contexte d'atelier multigamme. Blazewicz et al. (1991) présentent une revue sur les formulations du problème d'ordonnancement de la production. Dans notre modèle, on ajoute des contraintes relatives au système de manutention.

Le modèle proposé comprend les variables et les contraintes définissant le problème d'ordonnancement de la production dans un atelier multigamme. Cette partie du modèle considère les ressources en machines. Nous y ajoutons la partie qui considère les ressources en chariots. Cela inclut les contraintes d'antériorité pour les tâches de manutention, les contraintes d'affectation des tâches de manutention aux chariots et les contraintes de disjonction des chariots qui, en même temps, assurent que les voyages à vide sont bien pris en compte. Le modèle utilise une grande valeur (H) dans la formulation des contraintes de disjonction. A priori, nous n'avons pas considéré des alternatives en machines. Cette éventualité est possible avec une petite modification du modèle. Dans le modèle, un paramètre (ϕ_{jw}) est utilisé pour indiquer qu'une tâche de production *j* est effectuée sur une machine *w*. Ce paramètre remplacé par une variable permet l'utilisation du modèle dans le cas de machines alternatives. Dans ce cas là, il devient équivalent au modèle présenté par Anwar et Nagi (1998) dans un contexte d'atelier d'assemblage. Les auteurs ne résolvent pas la formulation et présentent une heuristique pour résoudre le problème. Bilge et Ulusoy (1995) fournissent aussi une formulation non linéaire pour le problème et proposent une heuristique pour la résolution.

4.2.1. Notation

O	Ensemble des tâches ($j \in O$), chaque tâche appartient à un travail spécifique
$O^L \subset O$	Ensemble des dernières tâches de tous les travaux
$O^F \subset O$	Ensemble des premières tâches de tous les travaux
W	Ensemble de tous les postes de travail : machines et chariots $w \in W$
$C \subset W$	Ensemble des chariots
$M \subset W$	Ensemble des machines
O_w	Ensemble des tâches sollicitant le poste de travail w
$n(j)$	Tâche successeur de la tâche j

S	Date de début de l'horizon de planification
t_j	Temps de traitement de la tâche j
ϕ_{jw}	Paramètre ayant la valeur 1 si la tâche de production j est effectuée sur la machine w
k_{ij}	Temps de voyage à vide de la machine destination de la tâche de manutention i à la machine source de la tâche de manutention j
H	Une grande valeur
Ψ_{ij}	Variable ayant la valeur 1 si la tâche *i* précède la tâche *j* sur un poste de travail, 0 sinon
Φ_{jw}	Variable ayant la valeur 1 si la tâche de manutention *j* est affectée au chariot w, 0 sinon
S_j	Variable définissant la date de début de la tâche *j*
C_{max}	Variable définissant la date de fin de l'horaire

4.2.2. Modèle

La formulation du problème en programme linéaire mixte est présentée ci-après. L'objectif du modèle est de minimiser la date de fin de l'horaire. Les contraintes sont suivies d'explications.

Objectif

$$\text{Min } C_{max} \qquad (1)$$

L'objectif du modèle consiste à minimiser la date de fin de l'horaire (C_{max}) en respectant les contraintes suivantes :

Relations d'antériorité

$$C_{max} \geq S_j + t_j \qquad \forall j \in O^L \qquad (2)$$

La date de fin de l'horaire (C_{max}) doit être plus grande ou égale à la date de fin ($S_j + t_j$) de toutes les dernières tâches de tous les travaux.

Relations d'antériorité

$$S_{n(j)} \geq S_j + t_j \qquad \forall j \in O \backslash O^L \qquad (3)$$

La date de début d'une tâche successeur ($S_{n(j)}$) d'une tâche (*j*) doit être plus grande ou égale à sa date de fin.

Détermination d'ordre

$$\Psi_{ij} + \Psi_{ji} = 1 \qquad \forall i,j \in O_w \mid i \neq j, \forall w \qquad (4)$$

Contrainte déterminant l'ordre de visite des tâches aux postes de travail sollicités. Pour deux tâches (*i*) et (*j*) sollicitant un poste de travail w, $\Psi_{ij} = 1$ indique que la tâche (*j*) suit la tâche (*i*) sur le poste de travail. Inversement, $\Psi_{ji} = 1$ indique que la tâche (*i*) suit la tâche (*j*) sur le poste de travail. La contrainte assure alors que l'une de ces deux relations est satisfaite.

Disjonction des machines

$$S_j \geq (S_i + t_i) + (\Psi_{ij} + \phi_{iw} + \phi_{jw} - 3)\, H \qquad \forall w \in M, \forall i,j \in O_w \mid i \neq j \qquad (5)$$

Si la tâche de production *j* succède à la tâche de production *i* ($\Psi_{ij} = 1$) sur une machine ($\phi_{iw} = \phi_{jw} = 1$), il faut que la première tâche *i* soit finie avant que la deuxième *j* ne commence.

Disjonction des chariots

$$S_j \geq S_i + t_i + k_{ij} + (\Psi_{ij} + \Phi_{iw} + \Phi_{jw} - 3)\, H \qquad \forall w \in C, \forall i,j \in O_w \mid i \neq j \qquad (6)$$

Si la tâche de manutention *j* succède à la tâche de manutention *i* ($\Psi_{ij} = 1$) sur un chariot ($\Phi_{iw} = \Phi_{jw} = 1$), il faut que la première tâche soit finie et que le chariot ait effectué un voyage à vide pour se rendre à la machine source de la deuxième tâche, avant que celle-ci ne commence. Le voyage à vide peut être d'une durée (k_{ij}) de zéro si la machine destination de la première tâche de manutention est la machine source de la deuxième tâche.

Relations d'antériorité

$S_j \geq S$ $\forall j \in O^F$ (7)

La date de début de chaque tâche est plus grande ou égale au début de l'horizon (S).

Affectation aux chariots

$\sum_w \Phi_{jw}=1$ $\forall w \in C, \forall j \in O_w$ (8)

Une tâche de manutention j est affectée à un seul chariot w quand $\Phi_{jw}=1$.

Le modèle complet présenté au tableau 4.1 minimise la date de fin de l'horaire (1), sous les contraintes (2) à (8). Les contraintes (9-10,11) définissent les variables binaires et continues respectivement.

$\Psi_{ij} \in \{0,1\}$ (9)

$\Phi_{jw} \in \{0,1\}$ (10)

$S_j \geq 0$ (11)

Tableau 4.1. Modèle complet en programmation mathématique

Min C_{max}		(1)
$C_{max} \geq S_j + t_j$	$\forall j \in O^L$	(2)
$S_{n(j)} \geq S_j + t_j$	$\forall j \in O \backslash O^L$	(3)
$\Psi_{ij} + \Psi_{ji} = 1$	$\forall i,j \in O_w \mid i \neq j, \forall w$	(4)
$S_j \geq (S_i + t_i) + (\Psi_{ij} + \phi_{iw} + \phi_{jw} - 3) H$	$\forall w \in M, \forall i,j \in O_w \mid i \neq j$	(5)
$S_j \geq S_i + t_i + k_{ij} - (3 - \Psi_{ij} - \Phi_{iw} - \Phi_{jw}) H$	$\forall w \in C, \forall i,j \in O_w \mid i \neq j$	(6)
$S_j \geq S$	$\forall j \in O$	(7)
$\sum_w \Phi_{jw} = 1$	$\forall w \in C, \forall j \in O_w$	(8)
$\Psi_{ij} \in \{0,1\}$		(9)
$\Phi_{jw} \in \{0,1\}$		(10)
$S_j \geq 0$		(11)

La contrainte numéro 6 impose que si deux tâches de manutention se suivent sur un même chariot, il faut que le temps de voyage à vide entre les deux tâches soit écoulé. Plus précisément, il a fallu imposer que si deux tâches de manutention se suivent directement, il faut que la contrainte soit prise en compte. Le modèle devient plus difficile à résoudre avec une contrainte de ce type. Cependant, que les tâches soient des successeurs directs ou pas, la contrainte numéro 6 est forcément vérifiée. Dans un contexte de plus court chemin, quand trois tâches de manutention *i*, *j*, et *l* se suivent sur un chariot, la relation suivante est respectée :

$k_{il} \leq k_{ij} + k_{jl}$

k_{ij} étant le temps de voyage à vide entre la machine destination de la tâche de manutention *i* et la machine source de la tâche de manutention *j*.

Comme k_{il} est le plus court chemin entre la machine destination de la tâche de manutention *i* et la tâche de manutention *l*, n'importe quel autre chemin d'accès entre ces deux machines incorporant une tâche de manutention *j* accomplie entre les deux tâches *i* et *l*, sera plus long ou égal au plus court chemin.

La grande valeur utilisée dans la formulation des contraintes de disjonction (5) et (6), doit être choisie soigneusement pour éviter les problèmes de précision numérique.

4.2.3. Expérimentation

Les implantations d'ateliers utilisés pour générer les premiers jeux de données sont tirés de Bilge et Ulusoy (1995). Les auteurs ont utilisé quatre implantations d'atelier pour établir les données sur les durées de tâches de manutention. Ils ont proposé aussi dix ensembles de travaux à traiter avec leurs gammes respectives. Avec les données pour les tâches de manutention, tirées à partir des implantations, et les données sur les gammes quarante jeux de données ont été générés. À notre connaissance, il s'agit des seuls problèmes tests reproductibles présentés dans la littérature pour le problème d'ordonnancement intégré dans le contexte d'atelier multigamme. Nous avons utilisé les

mêmes jeux de données pour la validation de nos modèles. Les quatre implantations d'atelier sont présentées à l'annexe A.

Anwar et Nagi (1998) et Sabuncuoglu et al. (1989-1998) ont travaillé avec des données qui n'ont pas été présentées en détail dans leurs travaux et que nous ne pouvions pas reproduire.

Ces jeux de données que nous utilisons comprennent dix ensembles de travaux avec des gammes différentes. Ces dix ensembles utilisés avec quatre implantations donnent quarante problèmes test. Les matrices de distances pour les quatre implantations, les gammes et les durées de traitement sont données en annexe A. Les temps de traitement sur machines sont d'une durée moyenne de 9 unités de temps. Les temps de manutention ou encore les temps pour effectuer un voyage à vide sont d'une durée moyenne de 3.5 unités de temps.

Dans un premier temps, ces différents jeux de données présentés par Bilge et Uusoy (1995) ont été utilisés sans aucune modification pour tester ce premier modèle. Par la suite, des modifications à ces jeux ont été introduites. Les modifications comprennent :

- l'ajustement des temps de manutention : les jeux de la littérature présentent un temps de manutention comparable à celui du traitement sur machines, sinon plus long. Ce cas est très restrictif car souvent les temps de manutention seront inférieurs à ceux du traitement sur machines. Un chariot automatique voyage à une vitesse moyenne de 3 km/h. Considérant les dimensions habituelles d'un atelier, et une production par lots, le temps de manutention doit être inférieur à celui présenté. Un temps de manutention ajusté a été obtenu en divisant les temps de manutention de la littérature par deux. Ce temps ajusté a été utilisé dans les tests des trois problèmes.

- la généralisation en réseau bidirectionnel : l'article de Bilge et Ulusoy propose un réseau unidirectionnel et donc le temps de manutention entre les machines x et y n'est pas nécessairement égal au temps de manutention entre les machines y et x. Cette hypothèse n'est pas importante car les conflits de routage ne sont pas pris en compte, et donc un réseau unidirectionnel ou bidirectionnel ne présente pas de différence. On considère dans les jeux modifiés (problèmes du tableau 4.7, et les problèmes des chapitres suivants) que le réseau est bidirectionnel. Plus d'alternatives en routage sont donc possibles.

- l'agrandissement de l'atelier : les jeux présentés sont de petite taille. Pour élaborer des jeux de données d'une plus grande taille, nous avons choisi d'augmenter le nombre de machines en agrandissant la quatrième implantation. Cette dernière est éventuellement plus intéressante dans l'étude des phénomènes de conflits. Pour générer les jeux du tableau 4.7, l'atelier de l'implantation 4 a été modifié en ajoutant 4 autres machines pour un total de 8 pour donner l'implantation 5. Le nombre de travaux a été également doublé pour atteindre un total de 16.

Les tests ont été effectués sur un ordinateur personnel dont les spécifications sont les suivantes :

Processeur	Intel Pentium 4, 2.53GHz
Carte maîtresse	Intel D850E Audio/Lan
Mémoire	4x 256 MB RIMM PC800 (1 GB)

Les tableaux de résultats sont présentés ci-après. Le tableau 4.2 présente les résultats pour les mêmes problèmes de la littérature sans aucun ajustement (Bilge et Ulusoy, 1995). Le modèle a été écrit en langage OPL et résolu à l'optimalité dans des temps très courts sauf deux jeux qui ont pris un temps de calcul relativement long (3700 et 1549 secondes). Un troisième a été résolu en 66 secondes. Le reste des jeux est résolu en moins de 27 secondes. Tous les jeux de données sont résolus à l'optimalité.

Tableau 4.2. Résultats pour les données de la littérature

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-1-1	21	4	2	551	339	0,11	72	344	1187
PM-1-2	21	4	2	551	339	0,03	72	5	71
PM-1-3	21	4	2	551	339	0,03	72	20	143
PM-1-4	21	4	2	551	339	0,03	70	13	112
PM-2-1	24	4	2	701	433	0,80	80	2328	61515
PM-2-2	24	4	2	701	433	0,11	72	99	713
PM-2-3	24	4	2	701	433	0,16	78	58	551
PM-2-4	24	4	2	701	433	2,02	82	6913	19210
PM-3-1	27	4	2	811	522	0,19	82	120	584
PM-3-2	27	4	2	811	522	0,13	74	22	323
PM-3-3	27	4	2	811	522	0,09	78	11	189
PM-3-4	27	4	2	811	522	0,44	82	833	1779
PM-4-1	33	4	2	1259	968	26,47	84	47329	155093
PM-4-2	33	4	2	1259	968	66,97	74	116263	413777
PM-4-3	33	4	2	1259	968	6,09	74	10395	34593
PM-4-4	33	4	2	1259	968	17,72	83	32831	95476
PM-5-1	21	4	2	551	338	0,13	59	306	797
PM-5-2	21	4	2	551	338	0,06	58	119	303
PM-5-3	21	4	2	551	338	0,11	56	22	157
PM-5-4	21	4	2	551	338	0,11	59	112	376
PM-6-1	30	4	2	1055	736	0,72	102	713	4391
PM-6-2	30	4	2	1055	736	0,61	94	532	3125
PM-6-3	30	4	2	1055	736	0,61	98	476	3238
PM-6-4	30	4	2	1055	736	11,42	101	25344	71982
PM-7-1	31	4	2	1055	643	3700	79	967748	6142831
PM-7-2	31	4	2	1055	643	0,52	66	178	1223
PM-7-3	31	4	2	1055	643	12,02	69	14846	50725
PM-7-4	31	4	2	1055	643	1549	83	3917591	12782333
PM-8-1	34	4	2	1331	998	13,14	153	6777	45247
PM-8-2	34	4	2	1331	998	2,77	145	1062	8841
PM-8-3	34	4	2	1331	998	5,95	149	3129	17896

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-8-4	34	4	2	1331	998	5,08	155	4035	21606
PM-9-1	29	4	2	991	718	1,55	97	1088	4204
PM-9-2	29	4	2	991	718	0,70	91	221	1304
PM-9-3	29	4	2	991	718	0,72	93	232	1492
PM-9-4	29	4	2	991	718	15,94	97	17945	54180
PM-10-1	36	4	2	1481	1115	7,22	127	4385	18321
PM-10-2	36	4	2	1481	1115	3,78	123	3938	11609
PM-10-3	36	4	2	1481	1115	1,28	126	720	4159
PM-10-4	36	4	2	1481	1115	24,47	130	29867	68709

Légende

PM-nb.1-nb.2

PM	Programmation mathématique
no. 1	numéro du jeu de données
no.2	numéro de l'implantation

Trois stratégies ont été employées pour réduire le temps de calcul. L'effet a été remarquable particulièrement pour les problèmes PM-7-1 et PM-7-4. Ces problèmes demandaient plusieurs jours de temps de calcul avant de prouver l'optimalité sans l'emploi de ces stratégies.

- Une borne supérieure heuristique : il suffit de lancer la résolution et prendre la solution entière obtenus à la fin des 30 premières secondes et l'introduire dans le modèle par la suite. Malgré le fait que les bornes sont automatiquement mises à jour dans Cplex, cela aide dans la réduction du temps de calcul.
- La contrainte 7 du modèle améliore le temps de calcul de façon significative. Dans les problèmes tests, l'horizon débute à zéro. On sait d'avance, à cause des contraintes de non négativité qu'on ne peut pas avoir une date de début négative dans l'horaire. Cependant, cette inégalité valide aide à réduire le temps de calcul. La contrainte présente aussi un aspect pratique, celui de pouvoir débuter l'ordonnancement à différents moments. Il serait intéressant d'indicer le

paramètre indiquant la date de début au plus tôt par les tâches pour pouvoir représenter la disponibilité des machines correspondantes.

- Une inégalité valide a été introduite pour imposer qu'au moins une seule tâche de manutention soit affectée à chaque chariot de la flotte.

$$\sum_{i \in Ow} \Phi_{iw} \geq 1 \qquad \forall w \in C$$

Les jeux de données avec des temps de manutention ajustés à la baisse (divisés par deux) où les chariots sont moins sollicités sont plus faciles à résoudre. Tous les jeux sont résolus en moins de 20 secondes.

Tableau 4.3. Résultats pour les données de la littérature avec temps de manutention ajustés

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. Contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-MA-1-1	21	4	2	551	336	0,06	63	20	123
PM-MA-1-2	21	4	2	551	336	0,02	61	3	72
PM-MA-1-3	21	4	2	551	336	0,13	65	8	93
PM-MA-1-4	21	4	2	551	336	0,14	62	8	93
PM-MA-2-1	24	4	2	701	431	0,22	77	48	365
PM-MA-2-2	24	4	2	701	431	0,14	74	38	400
PM-MA-2-3	24	4	2	701	431	0,11	77	36	359
PM-MA-2-4	24	4	2	701	431	0,11	76	20	316
PM-MA-3-1	27	4	2	811	520	0,22	75	45	323
PM-MA-3-2	27	4	2	811	520	0,20	72	41	461
PM-MA-3-3	27	4	2	811	520	0,25	74	50	409
PM-MA-3-4	27	4	2	811	520	0,14	73	33	264
PM-MA-4-1	33	4	2	1259	966	9,58	61	13257	30758
PM-MA-4-2	33	4	2	1259	966	0,34	58	50	307
PM-MA-4-3	33	4	2	1259	966	0,31	58	48	312
PM-MA-4-4	33	4	2	1259	966	0,31	61	51	385

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. Contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-MA-5-1	21	4	2	551	336	0,16	52	10	122
PM-MA-5-2	21	4	2	551	336	0,09	53	23	127
PM-MA-5-3	21	4	2	551	336	0,03	52	10	95
PM-MA-5-4	21	4	2	551	336	0,03	51	0	94
PM-MA-6-1	30	4	2	1055	734	0,44	94	209	1316
PM-MA-6-2	30	4	2	1055	734	0,44	91	195	1108
PM-MA-6-3	30	4	2	1055	734	0,42	92	165	1060
PM-MA-6-4	30	4	2	1055	734	0,33	93	55	516
PM-MA-7-1	31	4	2	1055	641	0,28	66	60	534
PM-MA-7-2	31	4	2	1055	641	0,33	66	133	693
PM-MA-7-3	31	4	2	1055	641	0,33	66	89	772
PM-MA-7-4	31	4	2	1055	641	0,70	67	775	3014
PM-MA-8-1	34	4	2	1331	996	4,38	147	4118	17447
PM-MA-8-2	34	4	2	1331	996	19,08	143	20740	79123
PM-MA-8-3	34	4	2	1331	996	8,83	145	9561	36738
PM-MA-8-4	34	4	2	1331	996	4,69	148	4673	20022
PM-MA-9-1	29	4	2	991	716	0,38	88	127	757
PM-MA-9-2	29	4	2	991	716	0,38	88	154	914
PM-MA-9-3	29	4	2	991	716	0,31	88	83	713
PM-MA-9-4	29	4	2	991	716	0,36	87	106	809
PM-MA-10-1	36	4	2	1481	1113	1,33	121	900	4009
PM-MA-10-2	36	4	2	1481	1113	0,88	119	432	2002
PM-MA-10-3	36	4	2	1481	1113	1,30	121	857	3933
PM-MA-10-4	36	4	2	1481	1113	0,95	120	422	2235

Les contraintes de ressources affectent significativement le temps de calcul. Pour tester cette hypothèse, nous avons résolu les mêmes jeux de données en augmentant le nombre de chariots de 2 à 3. Les résultats sont présentés ci-après.

Tableau 4.4. Résultats pour les données de la littérature avec temps de manutention ajustés et 3 chariots dans l'atelier

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-MA-1-1	21	4	3	573	448	0,06	65	11	127
PM-MA-1-2	21	4	3	573	448	0,14	61	7	97
PM-MA-1-3	21	4	3	573	448	0,17	65	11	116
PM-MA-1-4	21	4	3	573	448	0,05	65	11	145
PM-MA-2-1	24	4	3	726	575	0,14	77	98	491
PM-MA-2-2	24	4	3	726	575	0,17	74	60	399
PM-MA-2-3	24	4	3	726	575	0,28	77	95	514
PM-MA-2-4	24	4	3	726	575	0,31	76	101	572
PM-MA-3-1	27	4	3	838	700	0,31	75	55	345
PM-MA-3-2	27	4	3	838	700	0,25	72	43	299
PM-MA-3-3	27	4	3	838	700	0,27	74	49	339
PM-MA-3-4	27	4	3	838	700	0,28	73	68	366
PM-MA-4-1	33	4	3	1293	1330	0,44	57	60	309
PM-MA-4-2	33	4	3	1293	1330	0,56	54	120	464
PM-MA-4-3	33	4	3	1293	1330	0,52	55	110	460
PM-MA-4-4	33	4	3	1293	1330	0,61	57	197	729
PM-MA-5-1	21	4	3	573	448	0,08	52	11	117
PM-MA-5-2	21	4	3	573	448	0,05	53	10	119
PM-MA-5-3	21	4	3	573	448	0,16	52	8	111
PM-MA-5-4	21	4	3	573	448	0,05	51	20	145
PM-MA-6-1	30	4	3	1086	998	0,77	95	246	1839
PM-MA-6-2	30	4	3	1086	998	0,75	91	269	2373
PM-MA-6-3	30	4	3	1086	998	0,76	92	291	2070
PM-MA-6-4	30	4	3	1086	998	0,67	93	236	1402
PM-MA-7-1	31	4	3	1086	861	0,50	66	139	741
PM-MA-7-2	31	4	3	1086	861	0,41	66	59	394
PM-MA-7-3	31	4	3	1086	861	0,41	66	81	484
PM-MA-7-4	31	4	3	1086	842	0,94	67	421	2920
PM-MA-8-1	34	4	3	1366	1360	7,80	147	6093	20186

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-MA-8-2	34	4	3	1366	1360	16,95	143	11687	59454
PM-MA-8-3	34	4	3	1366	1360	5,63	145	3390	17517
PM-MA-8-4	34	4	3	1366	1360	2,88	148	1741	7287
PM-MA-9-1	29	4	3	1021	980	0,42	88	133	750
PM-MA-9-2	29	4	3	1021	980	41,03	88	62202	67998
PM-MA-9-3	29	4	3	1021	980	0,38	88	110	749
PM-MA-9-4	29	4	3	1021	980	0,44	87	127	807
PM-MA-10-1	36	4	3	1518	1533	1,28	121	502	2833
PM-MA-10-2	36	4	3	1518	1533	71,06	119	74705	94860
PM-MA-10-3	36	4	3	1518	1533	3,81	121	2098	5600
PM-MA-10-4	36	4	3	1518	1533	1,73	120	800	4189

D'après les résultats, nous avons remarqué que les problèmes relativement difficiles à résoudre prennent moins de temps de résolution en ajoutant un troisième chariot, par exemple, les problèmes PM-MA-4-1 et PM-MA-8-2. Par contre quelques problèmes facilement résolus avec deux chariots prennent un peu plus de temps avec 3 chariots. C'est le cas du problème PM-MA-9-2 où le temps de calcul passe de 0,38 secondes à 41,03 secondes. De même, pour le problème PM-MA-10-2 le temps de calcul passe de 0,88 secondes à 71,06 secondes. La résolution du même problème dans les cas de deux et trois chariots permet de savoir si la flotte est sous dimensionnée (obtention d'un objectif moindre avec 3 chariots), ou si les contraintes de ressources en chariots sont très serrées (obtention du même objectif avec 3 chariots mais en moins de temps).

Les jeux à 8 machines/16 travaux, présentés en détail en annexe A, sont très difficiles à résoudre et ils demandent plusieurs heures, voire plusieurs jours, pour prouver l'optimalité. Cependant on atteint la solution optimale en quelques minutes.

On note qu'il est difficile de définir les limites du modèle plus précisément en indiquant le nombre maximal de variables et de contraintes. Le temps de calcul change

dépendamment des données. C'est une combinaison de facteurs qui fait que la résolution devient difficile. Nous avons essayé de tester les modèles avec un jeu de données « base de comparaison » de la littérature sur les problèmes d'ordonnancement. Il s'agit du problème 10x10 de Fisher et Thompson dont le nom est (mt10). Le jeu a cependant été augmenté par les données nécessaires pour représenter les contraintes de manutention. Pour ce faire, nous avons choisi les données de l'implantation quatre et on a considéré que la flotte est de deux chariots. Le modèle présenté plus bas en programmation par contraintes a donné d'excellents résultats par rapport au problème formulé en programmation linéaire mixte. Nous avons atteint l'optimal en 30 minutes avec une preuve d'optimalité en 2 heures. Le même problème formulé en programmation mathématique a tourné pendant une semaine sans atteindre la valeur optimale. Nous avons atteint un résultat de 984 unités de temps pour l'horaire alors que l'optimal se situe à 948 unités de temps.

Il faut également préciser qu'à la base, il s'agit d'un problème difficile d'ordonnancement sur les machines. Les machines sont des ressources très contraignantes dans ce problème. L'ajout des contraintes de manutention au problème ne complique pas la résolution. Cependant, cela nous permet de produire un horaire réaliste et implantable en atelier pour ce jeu de données.

4.3. Formulation en programmation par contraintes

Le modèle formulé à l'aide de la programmation par contraintes utilise des fonctions d'un logiciel commercial (OPL Studio, version 3.6) qui font appel au solveur (Scheduler) spécialement conçu pour résoudre les problèmes d'ordonnancement en se servant des algorithmes de propagation de la programmation par contraintes. Le nombre de variables et de contraintes est beaucoup moindre par rapport à la formulation mathématique mixte. De plus, l'écriture est plus expressive et les données sont entrées de façon plus compacte.

Avant de décrire la formulation, nous définissons quelques concepts de la programmation par contraintes intégrés dans le logiciel OPLStudio version 3.6. Pour une introduction plus complète à la programmation par contraintes, le lecteur est invité à consulter Marriot et Stucky (1998).

Les ressources unaires

Une ressource est dite unaire si elle ne peut pas être partagée par deux tâches à un même instant. Elle possède une capacité de 1 en tout temps.

Les ressources alternatives

Des ressources sont dites alternatives si elles sont équivalentes du point de vue d'une tâche. On peut donc utiliser l'une ou l'autre. On peut avoir autant qu'on veut des ressources alternatives dans un modèle. L'instruction **ActivityHasSelectedRessource** peut être utilisé avec des arguments appropriés pour désigner la ressource qui a été choisie par une tâche, et pour formuler des contraintes globales.

Les ressources discrètes

Les ressources discrètes représentent un concept fondamental offert par OPL pour les applications en ordonnancement. Elles sont utilisées pour modéliser des ressources disponibles en plusieurs unités et qui sont considérées équivalentes et interchangeables. Elles sont souvent utilisées pour modéliser les budgets et les magasins à outils où les outils sont identiques. Les tâches peuvent les consommer ou simplement les solliciter durant leur exécution pour les libérer par la suite. Il est généralement plus facile de trouver une solution pour un problème dont les ressources sont discrètes comparativement à un problème dont les ressources sont unaires.

Les contraintes globales

Les contraintes globales sont des contraintes sur les contraintes (Van Hentenryk, 1989). Elles sont souvent formulées avec des relations d'implication. Ceci veut dire que la

satisfaction d'une ou de plusieurs contraintes implique la satisfaction d'une ou de plusieurs autres contraintes. L'appellation globale est utilisée par opposition à locale. En général, la propagation des contraintes en programmation par contraintes suit le principe de localité. Selon ce principe, la propagation de chaque contrainte est considérée seule selon les domaines admissibles des variables. Dans les dernières années, il a été montré dans plusieurs contributions que les contraintes globales, où plusieurs contraintes sont prises en compte ensemble pour la propagation peuvent améliorer significativement la performance des modèles et donc réduire les temps de calcul (Van Hentenryk, 1989).

Pour mieux comprendre la notion de contrainte globale, nous donnons une définition des différents types de contraintes tel que présentée par Van Hentenryk (1989). En programmation par contraintes, il faut distinguer deux types de contraintes : élémentaire et composite. Les contraintes élémentaires sont soit des contraintes primitives de type ($>$, $=$,$\neq$, $\geq$) , ou des programmes logiques qui sont soit soumis à une déclaration soit utilisées comme des choix.

Les contraintes composites sont de deux types, contraintes de génération et contraintes globales. Les contraintes de génération sont utiles si on veut utiliser le même programme avec plusieurs jeux de données. On génère alors les contraintes élémentaires à partir des données du problème. Le deuxième type est les contraintes globales. Ces contraintes représentent une relation entre plusieurs types de variables. Pour mieux comprendre ce type de contraintes, on les divise encore en deux catégories.

Contraintes globales déterministes

Une contrainte globale déterministe est une conjonction de contraintes élémentaires. L'exemple classique pour expliquer ce type de contraintes est la contrainte (alldifferent) qu'on utilise si on veut que n variables prennent toutes des différentes valeurs. Ce type de contraintes n'est pas géré comme une contrainte élémentaire. Il génère un ensemble

de contraintes élémentaires. C'est alors équivalent à la conjonction de contraintes élémentaires, mais la propagation ne se fait pas de la même façon.

Contraintes globales non déterministes

Ce sont des contraintes qui ne peuvent pas être exprimées en tant qu'une conjonction de contraintes élémentaires. Ces contraintes sont nécessairement définies en tant qu'un prédicat non déterministe et sont par la suite utilisées comme un point de choix. La contrainte globale, numéro 5, utilisée dans notre modèle tombe dans cette catégorie.

La formulation du problème en programmation par contraintes, a comme objectif la minimisation de la date de fin de l'horaire. On définit dans le modèle un ensemble de tâches de production à exécuter de même qu'un ensemble de tâches de manutention. Les voyages à vides ne sont pas des tâches distinctes à accomplir mais ils sont générés selon l'ordre d'affectation des tâches de manutention aux chariots. Elles sont prises en compte dans la contrainte globale de disjonction des chariots.

Le modèle comprend les contraintes d'antériorité. Les contraintes d'affectation aux chariots ne sont pas nécessaires car on définit les chariots comme étant des ressources alternatives, et donc une tâche de manutention est affectée à l'un ou l'autre des chariots. Les contraintes de disjonction sont également omises car les ressources sont définies comme étant **unaires.** De même, les ressources sont définies comme étant **discrètes** dans le but de faciliter la recherche de solutions. Une fois qu'une solution est trouvée, elle est par la suite transformée par le logiciel en une solution qui respecte les contraintes de ressources unaires. Une contrainte globale est formulée pour tenir compte du temps de voyage à vide. Elle impose que si deux tâches de manutention se succèdent dans l'affectation à un chariot, il faut que le temps de voyage à vide soit pris en compte.

À notre connaissance, ce problème n'a pas été formulé ni résolu en utilisant la programmation par contraintes. Cependant, des problèmes d'ordonnancement de la

production (Jain et Grossmann, 2001) ainsi que des problèmes de tournées de véhicules (Pesant et al., 1996) ont été résolus en utilisant la programmation par contraintes et les résultats ont été prometteurs.

Pour résoudre notre problème il faut utiliser une bonne procédure de recherche ainsi qu'une stratégie de recherche parallèlement au modèle. Plusieurs procédures de recherche sont fournies dans le langage de programmation utilisé (OPLStudio). On peut également définir d'autres procédures de recherche pour les différents problèmes de façon très flexible. En fait, une procédure de recherche consiste à développer une heuristique, plus ou moins élaborée, qui guidera le logiciel dans la recherche de solutions. Elle est considérée complémentaire au modèle. D'autre part, le logiciel offre la possibilité de fixer des stratégies de recherche standard ou dichotomique et plusieurs alternatives pour guider la recherche dans l'arbre de solutions. Dans notre cas, une stratégie dichotomique combinée avec IDFS (interleaved depth first search) donne les meilleurs résultats.

Le IDFS consiste à simuler une recherche parallèle sur un ordinateur monoprocesseur. Cela est fait dans le but d'éviter la perte de temps causée par les erreurs commises très tôt dans la recherche.

Des procédures de recherche simples fournies dans le logiciel sont utilisées pour résoudre notre problème. Il s'agit de **setTimes** qui affecte des dates de début aux tâches. Cette procédure est efficace dans le cas où les ressources sont discrètes et où la durée de la tâche est fixe. De même, la procédure **assignAlternatives** qui propose des alternatives en choix de chariots est utilisée. Il s'agit d'une instruction non-déterministe qui affecte les tâches à une ressource parmi les ressources alternatives sollicitées.

4.3.1. Notation

Le modèle formulé en programmation par contraintes à l'aide du logiciel commercial requiert la déclaration des entités suivantes :

Machines	Les machines
Chariots	Les chariots
Tasks	L'ensemble des tâches de production et de manutention
Tasksmach $\subset$ Tasks	Tâches de production
Tasksman $\subset$ Tasks	Tâches de manutention
setOfPrecedences (t,q)| t et q $\in$ Tasks	Les couples de tâches qui possèdent une relation d'antériorité (t précédant q).
d[t]	Temps de traitement de chacune des tâches de production ou de manutention t
resourcem[t]	Machine sollicitée par la tâche de production t
k[t,q]	Temps de voyage à vide entre les tâches de manutention t et q affectées au même chariot
S[t]	Variable indiquant la date de début de la tâche t
Usedby[t]	Variable indiquant le chariot effectuant une tâche de manutention
S[makespan]	Date de fin de l'horaire

4.3.2. Modèle

La formulation du problème est présentée ci-après. L'objectif est de minimiser la date de fin de l'horaire. Les contraintes sont suivies d'explications.

Objectif

Minimiser S[makespan] (1)

L'objectif consiste à minimiser la date de fin de l'horaire en respectant les contraintes suivantes :

Relations d'antériorité

$S[q] \geq S[t] + d[t] \quad \forall (t,q) \in setOfPrecedences$ (2)

La date de début d'une tâche *q* successeur d'une tâche *t* est plus grande à la date de début de cette dernière plus sa durée de traitement.

Relations d'antériorité

$S[makespan] \geq S[t] + d[t] \quad \forall t \in Tasks$ (3)

La date de début de la tâche fictive « makespan » est plus grande ou égale à la date de début de toutes les tâches *t* plus leurs durées de traitement. Ces contraintes sont redondantes (déjà écrites dans la série de contraintes précédentes pour toutes les tâches sauf les dernières dans les gammes. Les contraintes redondantes peuvent aider dans la recherche de solutions.

Disjonction des machines

$S[q] \geq S[t] + d[t] \vee S[t] \geq S[q] + d[q]$

$\forall m \mid resourcem[t] = resourcem[q] = m$ (4)

Dans le modèle écrit sous OPL Studio, ces contraintes sont prises en compte par la définition des ressources unaires qui assurent qu'une machine ne sera pas occupée par deux tâches en même temps. Les contraintes de disjonction en programmation par contraintes sont des contraintes logiques qui sont formulées à l'aide d'un opérateur « ou ». Elles disent que pour deux tâches *t* et *q* sollicitant une même machine soit la tâche *t* commence après la fin de la tâche *q*, soit la tâche *q* commence après la fin de la tâche *t*.

Disjonction des chariots et connectivité des chemins

$$\text{Usedby}[t] = \text{Usedby}[q] \Rightarrow S[q] \geq S[t] + d[t] + k[t,q] \vee S[t] \geq S[q] + d[q] + k[q,t] \quad \forall\, t,q \in \text{Tasksman} \qquad (5)$$

Cette contrainte utilise l'instruction **ActivityHasSelectedRessource** d'OPL Studio qui sert à formuler une contrainte globale présentant une relation d'implication. C'est une contrainte logique qui assure la disjonction des chariots ainsi que de connectivité de chemins. Si deux tâches de manutention *t* et *q* sont affectées à un chariot, soit que *t* commence après que *q* soit terminée et que le chariot a parcouru une distance de voyage à vide entre la machine destination de la tâche *q* et la machine source de la tâche *t*, soit que *q* commence après que *t* soit terminée et que le chariot a parcouru une distance de voyage à vide entre la machine destination de la tâche *t* et la machine source de la tâche *q*.

Le modèle complet minimise la date de fin de l'horaire (1), sous les contraintes de (2) à (5).

Tableau 4.5. Modèle complet en programmation par contraintes

Minimiser S[makespan]		(1)
$S[q] \geq S[t] + d[t]$	$\forall\, (t,q) \in$ setOfPrecedences	(2)
$S[\text{makespan}] \geq S[t] + d[t]$	$\forall\, t \in$ Tasks	(3)
$S[q] \geq S[t] + d[t] \vee$ $S[t] \geq S[q] + d[q]$	$\forall\, m \mid \text{resourcem}[t] = \text{resourcem}[q] = m$	(4)
$\text{Usedby}[t] = \text{Usedby}[q] \Rightarrow$ $S[q] \geq S[t] + d[t] + k[t,q] \vee$ $S[t] \geq S[q] + d[q] + k[q,t]$	$\forall\, t,q \in$ Tasksman	(5)

4.3.3. Expérimentation

Les mêmes jeux résolus pour la formulation mathématique ont été repris. Les résultats sont très encourageants, surtout pour les problèmes de grande taille où l'on trouve la solution optimale en quelques secondes. Les résultats sont présentés ci-après :

Tableau 4.6. Résultats du modèle de programmation par contraintes pour les données de la littérature

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variable	Nb. contraintes	Temps secondes	Objectif	Échecs	Points de choix
PC-1-1	21	4	2	154	170	0,05	72	65	176
PC-1-2	21	4	2	154	170	0,00	72	4	69
PC-1-3	21	4	2	154	170	0,00	72	4	63
PC-1-4	21	4	2	154	170	0,05	70	32	188
PC-2-1	24	4	2	175	210	0,38	80	188	1372
PC-2-2	24	4	2	175	210	0,05	72	4	106
PC-2-3	24	4	2	175	210	0,06	78	21	405
PC-2-4	24	4	2	175	210	0,72	82	812	1275
PC-3-1	27	4	2	189	252	0,27	82	119	868
PC-3-2	27	4	2	189	252	0,05	74	9	195
PC-3-3	27	4	2	189	252	0,14	78	6	110
PC-3-4	27	4	2	189	252	0,36	82	204	1158
PC-4-1	33	4	2	238	458	37,98	84	19246	36323
PC-4-2	33	4	2	238	458	9,70	74	4993	7763
PC-4-3	33	4	2	238	458	1,24	74	647	1622
PC-4-4	33	4	2	238	458	7,56	83	4103	11043
PC-5-1	21	4	2	154	170	0,06	59	38	132
PC-5-2	21	4	2	154	170	0,02	58	4	57
PC-5-3	21	4	2	154	170	0,02	56	10	77
PC-5-4	21	4	2	154	170	0,03	59	50	142
PC-6-1	30	4	2	217	348	0,25	102	55	442
PC-6-2	30	4	2	217	348	0,05	94	8	53
PC-6-3	30	4	2	217	348	0,05	98	10	135

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variable	Nb. contraintes	Temps secondes	Objectif	Échecs	Points de choix
PC-6-4	30	4	2	217	348	0,97	101	675	842
PC-7-1	31	4	2	217	302	294	79	733628	733648
PC-7-2	31	4	2	217	302	0,39	66	223	700
PC-7-3	31	4	2	217	302	0,23	69	42	690
PC-7-4	31	4	2	217	302	1816	83	4942737	4942747
PC-8-1	34	4	2	245	460	0,33	153	36	1135
PC-8-2	34	4	2	245	460	0,14	145	5	75
PC-8-3	34	4	2	245	460	0,06	149	7	75
PC-8-4	34	4	2	245	460	0,38	155	1502	1519
PC-9-1	29	4	2	210	346	0,22	97	37	379
PC-9-2	29	4	2	210	346	0,08	91	9	114
PC-9-3	29	4	2	210	346	0,09	93	14	120
PC-9-4	29	4	2	210	346	0,27	97	105	182
PC-10-1	36	4	2	259	522	0,44	127	99	477
PC-10-2	36	4	2	259	522	0,33	123	64	417
PC-10-3	36	4	2	259	522	0,30	126	39	446
PC-10-4	36	4	2	259	522	0,67	130	203	1067

Légende

PC-nb.1-nb.2

PC	Programmation par contraintes
no. 1	numéro du jeu de données
no.2	numéro de l'implantation

Dans le tableau, les points de choix réfèrent aux différentes étapes de recherche d'une solution. Cela correspond aux nœuds de l'arbre de branchement pour un problème en nombre entier. Les échecs sont les points de choix qui ont été explorés dans la recherche et qui correspondent à des solutions irréalisables ou moins bonnes par rapport à la dernière solution trouvée.

Ces premiers résultats présentent une performance légèrement meilleure par rapport aux résultats obtenus avec la formulation en programmation linéaire mixte. Cependant, les

jeux de données à 8 machines sont résolus à l'optimalité en quelques secondes (maximum de 60) contrairement à la formulation en programmation linéaire mixte. Ces problèmes présentent un nombre plus élevé de contraintes et de variables, mais ils sont résolus plus facilement. Les problèmes avec un temps de manutention ajusté et un nombre de chariots plus élevé ont également présenté une très bonne performance.

Tableau 4.7. Résultats du modèle de programmation par contraintes pour les données augmentées

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variable	Nb. contraintes	Temps secondes	Objectif	Échecs	Points de choix
PC-11-5	PC-11-5	42	8	2	301	596	0,09	76	220
PC-12-5	PC-12-5	48	8	2	343	744	0,17	138	108
PC-13-5	PC-13-5	52	8	2	371	904	1,64	94	4939
PC-14-5	PC-14-5	62	8	2	441	1450	60	131	21055
PC-15-5	PC-15-5	42	8	2	301	596	0,16	84	10
PC-16-5	PC-16-5	54	8	2	385	1410	1,70	131	864
PC-17-5	PC-17-5	60	8	2	427	1088	0,13	132	83
PC-18-5	PC-18-5	68	8	2	483	1704	0,86	205	97
PC-19-5	PC-19-5	64	8	2	455	2261	11,13	114	3979
PC-20-5	PC-20-5	60	8	2	427	1272	3.63	134	7566

4.4. Analyse des résultats

La programmation par contraintes est une technique qui se trouve au carrefour entre trois sciences majeures : l'informatique, l'intelligence artificielle et la recherche opérationnelle. L'expérience d'utiliser cette technique pour résoudre le problème formulé dans ce chapitre a été réussie et prometteuse. Cependant, l'utilisation d'un solveur commercial fait qu'on ignore les algorithmes implantés dans le logiciel et donc la performance n'est pas prédictible dans tous les cas. Cette difficulté a été confirmée par Marriott et Stuckey (1998). De ce fait, choisir le modèle le plus appropriée devient un exercice empirique et donc, pour un même problème la performance des modèles

peut varier selon le logiciel commercial utilisé. La correction de ce problème sort du cadre de cette thèse. Il incombe aux spécialistes en programmation par contraintes d'étudier ces phénomènes.

Pour le coté pratique, les développements de ce chapitre présente un outil double où le problème est résolu à l'optimalité avec une technique ou l'autre. La recherche opérationnelle et la programmation par contraintes ne sont pas nécessairement des techniques appropriées pour un même type de problèmes. Les problèmes d'ordonnancement posent la difficulté d'avoir pour un même nombre de tâches, des niveaux de difficultés différents, que ce soit en recherche opérationnelle ou en programmation par contraintes. Cela est dû d'une part aux contraintes d'antériorité technologique qui imposent des contraintes temporelles aux variables de dates de début, et qui sont différents d'un jeu de données à un autre. D'autre part, les durées de traitement sur les ressources combinées avec les contraintes de disjonction influencent grandement le caractère contraignant d'une ressource. Dans les essais effectués pour valider notre modèle, nous avons remarqué que pour quelques jeux de données, la programmation par contraintes fonctionne mieux par rapport à la programmation mathématique, et pour d'autres jeux de données l'inverse est vrai.

Les contraintes en ressources d'équipements de manutention influencent le temps de calcul pour quelques jeux de données. Quand on augmente le nombre de chariots, le temps de calcul baisse rapidement. Ceci est le cas pour les deux techniques de résolution utilisées. L'ajout d'un équipement de manutention dans quelques problèmes ne modifie pas la valeur de l'objectif. Dans d'autres problèmes, la valeur de l'objectif a changé à la baisse. Si cela se reproduit dans la résolution de la majorité des jeux de données pour un même système, cela justifie l'ajout d'un équipement de manutention. Donc, l'outil d'ordonnancement peut être utilisé dans l'évaluation des besoins en équipements de manutention. La rapidité de la résolution avec cet outil permet alors l'évaluation des besoins en chariots en testant plusieurs scénarios. Bien que le taux d'utilisation des

chariots soit le premier déterminant pour dimensionner la flotte, des demandes ponctuelles pour les chariots peuvent influencer la longueur d'un ordonnancement.

Le caractère contraignant des ressources en machines peut également influencer le temps de calcul. Les deux formulations présentées peuvent être adaptées pour représenter des choix de ressources en machines.

Bien que les jeux résolus paraissent de petite taille, le problème est assez difficile. Il combine un problème d'ordonnancement d'atelier multigamme, un problème d'ordonnancement sur des machines parallèles et un problème de tournées de véhicules qui sont tous NP-difficile (Garey et Johnson ,1979). Nous pensons que les résultats sont prometteurs.

Nous nous sommes contentées, pour les fins de cette thèse, d'utiliser les données de la littérature plus les quelques ajouts proposés. Plus d'agrandissements d'ateliers sont possibles surtout dans le cas de programmation par contraintes où le nombre de variables ne change pas avec l'augmentation du nombre de ressources. Les variables correspondent majoritairement au nombre de tâches à ordonnancer. Cependant, il serait intéressant d'étudier le degré d'étranglement de chaque ressource en proposant une formule de calcul, et de déterminer la performance du modèle en fonction du degré d'étranglement.

La génération de plusieurs jeux de données de façon aléatoire sans pour autant comprendre ce qui fonctionne bien et ce qui ne fonctionne pas ne semble pas être la stratégie la plus appropriée pour valider les outils d'ordonnancement. Un problème d'ordonnancement à 200 tâches peut se résoudre en très peu de temps, alors qu'un autre avec le même nombre de tâches peut se résoudre très lentement. Conceptuellement, l'idée des ressources de goulot d'étranglement a été avancée par Adams et al. (1988) qui ont développé la méthode de résolution de « Shifting Bottleneck procedure » basée

sur ce concept. Leur méthode a permis la résolution à l'optimum du problème test (10*10) présenté par Thompson et qui est resté sans résolution pendant 10 ans. Des problèmes de plus grande taille ont été également résolus par leur heuristique. Cette méthode a été généralisée par Ramuhdin et Marier (1996) pour résoudre plusieurs classes de problèmes d'ordonnancement. Cette généralisation a prouvé être performante.

Il serait intéressant d'extrapoler ces idées à notre problème. L'objectif n'est pas simplement de résoudre notre problème avec cette méthode. Il est aussi important de pouvoir caractériser les problèmes en fonction de la criticité des ressources, ce qui peut guider le choix des méthodes de résolution. Ceci constitue un créneau de recherche futur.

CHAPITRE 5 :

PROBLÈME D'ORDONNANCEMENT INTÉGRÉ DE LA PRODUCTION ET DE LA MANUTENTION SANS CONFLITS AVEC CHOIX DE ROUTES PRÉDÉTERMINÉES

Le deuxième modèle présenté introduit des choix en routage dans l'atelier. Pour chacune des tâches de manutention un ensemble de routes admissibles est défini. C'est également le cas pour les voyages à vide. Les routes sont alors des ressources sollicitées par les tâches de manutention et les voyages à vide au même titre que les chariots. Les tâches de manutention sont définies de la même façon que les tâches de production. Une tâche de manutention agrégée consiste à effectuer une collecte à une machine suivie d'une livraison à la machine suivante dans la gamme du travail manutentionné. Son exécution est sujette au respect des contraintes d'antériorité technologique la reliant aux tâches de production. Les voyages à vide sont des tâches éventuelles qui peuvent se reproduire selon l'ordre de visite des tâches de manutention aux chariots. Ils assurent que le flot de chariots soit conservé. Ils ne sont pas tous à effectuer. Cependant, nous les définissons tous au départ puisqu'on ne connaît pas a priori l'affectation aux équipements de manutention. La dernière tâche d'un travail est une tâche de production. Elle peut être la tâche à exécuter sur la dernière machine dans la gamme d'un travail. Dans ce cas, les chariots s'immobilisent au niveau des machines après avoir effectué leurs dernières tâches de manutention. Pour conserver les mêmes notations et définitions, cette dernière tâche dans la gamme d'un travail peut aussi correspondre à la visite de l'entrepôt de produits finis, où le temps de traitement du travail est de zéro. Dans ce cas, les chariots à la fin de l'horaire s'immobilisent au niveau de l'entrepôt de produits finis. Les files d'attente possèdent une capacité

suffisante pour accueillir les travaux en traitement et les chariots qui peuvent attendre dans l'atelier.

5.1. Formulation en programmation mathématique

Le modèle en programmation mathématique est un programme linéaire mixte. Il comprend les variables et les contraintes définissant le problème d'ordonnancement de la production dans un atelier multigamme, c'est-à-dire, la partie qui considère les ressources en machines. Une autre partie du modèle considère les ressources en chariots et en routes. Cela comprend les contraintes d'antériorité pour les tâches de manutention, les contraintes d'affectation des tâches de manutention et des voyages à vide aux chariots et les contraintes d'affectation des tâches de manutention et des voyages à vide aux routes. Finalement, les contraintes de disjonction des chariots et des routes sont formulées en tenant compte de la connectivité des chemins.

5.1.1. Notation

O	Ensemble de toutes les tâches, y compris les voyages à vide éventuels et une tâche de fin fictive
O^R	Ensemble des tâches de production et de manutention $O^R \subset O$
O^M	Ensemble des tâches de manutention $O^M \subset O$
O^P	Ensemble des tâches de production $O^P \subset O$
O^F	Ensemble des premières tâches de tous les travaux $O^F \subset O$
O^L	Ensemble des dernières tâches de tous les travaux $O^L \subset O$
O^V	Ensemble des voyages à vide $O^V \subset O$
O^{MV}	Ensemble des tâches de manutention et des voyages à vide $O^{MV} \subset O$
W	Ensemble de tous les postes de travail $w \in W$
M	Ensemble des machines $M \subset W$

C	Ensemble des chariots $C \subset W$
R	Ensemble des routes $r \in R$
O^w	Ensemble des tâches sollicitant le poste de travail *w*
O^{si}	Ensemble des voyages à vide pouvant suivre la tâche de manutention *i*
O^{pj}	Ensemble des voyages à vide pouvant précéder la tâche de manutention *j*
R^j	Ensemble des routes pouvant accueillir une tâche *j*
R^C	Ensemble des paires de routes conflictuelles partageant un ou plusieurs segments
O^r	Ensemble des tâches sollicitant une route *r*
d_r	Longueur de la route *r*
n(j)	Tâche successeur de la tâche *j*
s	Début de l'horizon de planification
t_j	Temps de traitement de la tâche de production ou fictive *j*
ϕ_{jw}	Paramètre ayant la valeur 1 si la tâche de production *j* est traitée sur la machine *w*
H	Une grande valeur
Ψ_{ij}	Variable ayant la valeur 1 si la tâche *i* précède la tâche *j* , 0 sinon
Φ_{jw}	Variable ayant la valeur 1 si la tâche de manutention *j* est affectée au chariot *w*, 0 sinon
Π_{jr}	Variable ayant la valeur 1 si la tâche de manutention ou le voyage à vide *j* est affecté à la route *r*
Ω_{ij}	Variable ayant la valeur 1 si la tâche de manutention ou le voyage à vide *i* précède la tâche de manutention ou le voyage à vide *j* sur une route, 0 sinon
C_{max}	Variable indiquant la date de fin de l'horaire
S_j	Variable indiquant la date de début de la tâche de production, de manutention ou du voyage à vide *j*

5.1.2. Modèle

La formulation du problème en programme linéaire mixte est présentée ci-après. L'objectif du modèle est de minimiser la date de fin de l'horaire. Les contraintes sont suivies d'explications.

Objectif

Min C_{max} (1)

L'objectif consiste à minimiser la date de fin de l'horaire en respectant les contraintes suivantes :

Relations d'antériorité entre la date de fin de l'horaire et les dernières tâches

$C_{max} \geq S_j + t_j \qquad \forall j \in O^L$ (2)

La date de fin de l'horaire est plus grande ou égale à la date de fin de toutes les dernières tâches de tous les travaux.

Relations d'antériorité entre une tâche de production et une tâche successeur de manutention

$S_{n(j)} \geq S_j + t_j \qquad \forall j \in O^P$ (3)

La date de début d'une tâche de manutention successeur d'une tâche de production est plus grande ou égale à sa date de fin.

Relations d'antériorité une tâche de manutention et une tâche successeur de production

$S_{n(j)} \geq S_j + \Sigma_r \; d_r \Pi_{jr} \qquad \forall j \in O^M$ (4)

La date de début d'une tâche de production successeur d'une tâche de manutention est plus grande ou égale à sa date de fin. La durée de la tâche de manutention est égale à la durée requise pour parcourir la route choisie.

Relations d'antériorité entre les tâches et le début de l'horizon

$$S_j \geq s \qquad \forall j \in O^F \qquad (5)$$

La date de début de toutes les tâches est plus grande ou égale à la date de début des tâches sans prédécesseurs. Cette contrainte est redondante mais elle influence positivement le temps de calcul.

Détermination d'ordre de visite des machines et chariots

$$\Psi_{ij} + \Psi_{ji} = 1 \qquad \forall w, \forall i,j \mid i \neq j,\ i,j \in O^w \qquad (6)$$

Contrainte déterminant l'ordre de visite des tâches aux machines et aux chariots sollicités. Si la tâche *j* succède à la tâche *i* sur une machine ou un chariot, la variable Ψ_{ij} est égale à 1. Si l'inverse est vrai, la variable Ψ_{ji} est égale à 1.

Détermination d'ordre de visite des routes

$$\Omega_{ij} + \Omega_{ji} = 1 \qquad \forall r, \forall i,j \mid r \in R^j, r \in R^i, i \neq j,\ i,j \in O^{MV} \qquad (7)$$

Contrainte déterminant l'ordre de visite des tâches aux routes sollicitées. Si la tâche *j* succède à la tâche *i* sur une route, la variable Ω_{ij} est égale à 1. Si l'inverse est vrai, la variable Ω_{ji} est égale à 1.

Affectation aux chariots

$$\sum_w \Phi_{jw} = 1 \qquad \forall w \in C, \forall j \mid j \in O^w \qquad (8)$$

Une tâche de manutention est affectée à un seul chariot.

Affectation aux routes

$$\sum_r \Pi_{jr} = 1 \qquad \forall j \in O^{MV} \mid r \in R^j \qquad (9)$$

Chaque tâche de manutention ou voyage à vide utilise une seule route admissible. Une route admissible est une route qui peut accueillir la tâche de manutention ou le voyage à vide.

Disjonction des machines

$$S_j \geq (S_i+t_i) + (\Psi_{ij} + \phi_{iw} + \phi_{jw} - 3)\, H \qquad \forall w \in M,\ \forall i,j \mid i \neq j,\ i,j \in O^w \qquad (10)$$

Si la tâche de production *j* succède à la tâche de production *i* sur une machine, il faut que la première tâche soit finie avant que la deuxième ne commence.

Disjonction des chariots et connectivité des chemins

$$S_j \geq S_i + \Sigma_r\, d_r\, \Pi_{ir} + \Sigma_s\, d_s\, \Pi_{vs} + (\Pi_{ir} + \Pi_{vs} + (\Sigma_{k \neq} \Psi_{ik} - \Sigma_{k \neq} \Psi_{jk}) + \Phi_{iw} + \Phi_{jw} - 5)\, H$$

$$\forall r \in R^i,\ \forall s \in R^v,\ \forall w \in C,\ \forall i,j \mid i \neq j,\ i,j \in O^w,\ v \in O^{si} \cap O^{pj} \qquad (11)$$

Si la tâche de manutention *j* succède immédiatement à la tâche de manutention *i* sur un chariot, il faut que la première tâche soit finie et que le chariot ait parcouru un voyage à vide pour se rendre à la machine source de la deuxième tâche de manutention, avant que celle-ci ne commence. La durée du voyage à vide dépend de la route choisie. Le voyage à vide peut être d'une durée zéro si la machine destination de la première tâche de manutention est, en même temps, la machine source de la deuxième tâche de manutention.

Disjonction des routes

$$S_j \geq S_i + d_s + (\Pi_{jr} + \Pi_{is} + \Omega_{ij} - 3)\, H$$

$$\forall (r,s) \in R^C,\ \forall i,j \mid i \neq j,\ i,j \in O^{MV},\ s \in R^i,\ r \in R^j \qquad (12)$$

Si une tâche de manutention ou un voyage à vide *j* suit une tâche de manutention ou un voyage à vide *i* sur deux routes conflictuelles (r,s) partageant un ou plusieurs segments, il faut que la première soit finie avant que la deuxième ne commence. Évidemment, une route est conflictuelle avec elle-même.

Relations d'antériorité entre les tâches de manutention et les voyages à vide

$$S_v \geq S_i + d_s + ((\Sigma_{k \neq} \Psi_{ik} - \Sigma_{k \neq} \Psi_{jk}) + \Phi_{iw} + \Phi_{jw} + \Pi_{is} - 4)\, H$$

$$\forall w \in C,\ v \in O^{si} \cap O^{pj},\ \forall s \in R^i,\ \forall i,j \mid i \neq j,\ i,j \in O^M \qquad (13)$$

Si une tâche de manutention *i* précède immédiatement une tâche de manutention *j* sur un chariot, le voyage à vide résultant *v*, commence après, ou au moment où finit la tâche *i*. La tâche *i* est d'une durée correspondante au temps requis pour parcourir la route qu'elle entreprend.

Relations d'antériorité entre les tâches de manutention et les voyages à vide

$$S_j \geq S_v + d_s + ((\sum_{k \neq} \Psi_{ik} - \sum_{k \neq} \Psi_{jk}) + \Phi_{iw} + \Phi_j + \Pi_{vs} - 4)\, H$$
$$\forall w \in C,\ \forall i,j,\ v \in O^{si} \text{ et } v \in O^{pj},\ \forall s \in R^v \mid i \neq j,\ i,j \in O^M \qquad (14)$$

Si une tâche de manutention *i* précède immédiatement une tâche de manutention *j* sur un chariot, cette dernière commence après, ou au moment où finit le voyage à vide *v* résultant de cette affectation. Le voyage à vide *v* est d'une durée correspondante au temps requis pour parcourir la route qu'il entreprend.

Le modèle complet minimise la date de fin de l'horaire (1), sous les contraintes de (2) à (14). Les contraintes suivantes (15-18,19) définissent les variables binaires et continues respectivement.

$$\Psi_{ij} \in \{0,1\} \qquad (15)$$

$$\Phi_{jw} \in \{0,1\} \qquad (16)$$

$$\Pi_{jr} \in \{0,1\} \qquad (17)$$

$$\Omega_{ij} \in \{0,1\} \qquad (18)$$

$$S_j \geq 0 \qquad (19)$$

Tableau 5.1. Modèle complet en programmation mathématique

Min C_{max}		(1)
$C_{max} \geq S_j + t_j$	$\forall j \in O^L$	(2)
$S_{n(j)} \geq S_j + t_j$	$\forall j \in O \backslash O^L \mid j \text{ in } O^P$	(3)
$S_{n(j)} \geq S_j + \Sigma_r\ d_r\ \Pi_{jr}$	$\forall j \in O \backslash O^L \mid j \in O^M$	(4)
$S_j \geq s$	$\forall j \in O^F$	(5)
$\Psi_{ij} + \Psi_{ji} = 1$	$\forall w, \forall i,j \mid i \neq j,\ i,j \in O^w$	(6)
$\Omega_{ij} + \Omega_{ji} = 1$	$\forall r, \forall i,j \mid r \in R^j, r \in R^i, i \neq j,\ i,j \in O^{MV}$	(7)
$\sum_w \Phi_{jw} = 1$	$\forall w \in C, \forall j \mid j \in O^w$	(8)
$\Sigma_r\ \Pi_{jr} = 1$	$\forall j \in O^{MV} \mid r \in R^j$	(9)
$S_j \geq (S_i + t_i) + (\Psi_{ij} + \phi_{iw} + \phi_{jw} - 3)\ H$	$\forall w \in M, \forall i,j \mid i \neq j,\ i,j \in O^w$	(10)
$S_j \geq S_i + \Sigma_r\ d_r\ \Pi_{ir} + \Sigma_s\ d_s\ \Pi_{vs} + (\Pi_{ir}$	$\forall r \in R^i, \forall s \in R^v, \forall w \in C, \forall i,j \mid i \neq j,\ i,j \in O^w,$	(11)
$+ \Pi_{vs} + (\sum_{k \neq} \Psi_{ik} - \sum_{k \neq} \Psi_{jk}) + \Phi_{iw}$	$v \in O^{si} \cap O^{pj}$	
$+ \Phi_{jw} - 5)\ H$		
$S_j \geq S_i + d_s + (\Pi_{jr} + \Pi_{is} + \Omega_{ij} - 3)\ H$	$\forall (r,s) \in R^C, \forall i,j \mid i \neq j,\ i,j \in O^{MV}, s \in R^i, r \in R^j$	(12)
$S_v \geq S_i + d_s + ((\sum_{k \neq} \Psi_{ik} - \sum_{k \neq} \Psi_{jk})$	$\forall w \in C, v \in O^{si} \cap O^{pj}, \forall s \in R^i, \forall i,j \mid i \neq j,$	(13)
$+ \Phi_{iw} + \Phi_{jw} + \Pi_{is} - 4)\ H$	$i,j \in O^M$	
$S_j \geq S_v + d_s + ((\sum_{k \neq} \Psi_{ik} - \sum_{k \neq} \Psi_{jk})$	$\forall w \in C, \forall i,j, v \in O^{si} \cap O^{pj}, \forall s \in R^v \mid i \neq j,$	(14)
$+ \Phi_{iw} + \Phi_j + \Pi_{vs} - 4)\ H$	$i,j \in O^M$	
$\Psi_{ij} \in \{0,1\}$		(15)
$\Phi_{jw} \in \{0,1\}$		(16)
$\Pi_{jr} \in \{0,1\}$		(17)
$\Omega_{ij} \in \{0,1\}$		(18)
$S_j \geq 0$		(19)

5.1.3. Expérimentation

Les jeux de données utilisés pour tester le modèle présenté au chapitre 4 ont été augmentés par l'information qui manquait pour le choix de routes et l'évitement de conflits. Nous nous sommes servis des 4 premières implantations présentées en annexe A pour générer le choix de routes. Cependant, comme pour le problème présenté au chapitre 4, les données de manutention ont été ajustées en divisant la durée des plus courts chemins par deux pour les rendre plus réalistes. Nous avons aussi considéré un réseau bidirectionnel contrairement aux données de la littérature. Cela veut dire que la distance pour accéder à la machine x à partir de la machine y est identique à la distance pour accéder à la machine y de la machine x.

Dans ce deuxième problème, deux choix de routes sont permis pour effectuer une tâche de manutention ou un voyage à vide. L'évitement de conflits est assuré en affectant une seule tâche à une route pendant toute la durée d'exécution. La route est donc mobilisée au complet pendant l'exécution de la tâche. De plus, toutes les routes qui partagent un ou plusieurs segments avec une route utilisée sont mobilisées pendant toute la durée d'utilisation de cette dernière. Ce modèle ne permet pas une utilisation optimale des ressources en segments. Cependant, il génère une solution réalisable, sans conflits, implantable dans un atelier. Cette formulation est résolue à l'optimalité. De plus, la solution de ce modèle représente une borne supérieure pour un problème plus détaillé où tous les segments sont considérés séparément.

La formulation en programmation linéaire mixte comporte un très grand nombre de variables et de contraintes. Pour quelques jeux, on atteint un nombre de variables près de 147 000 et jusqu'à presque 10 000 contraintes. Les variables d'affectation aux différentes ressources et surtout les variables de détermination de l'ordre de visite augmentent significativement avec l'augmentation du nombre de tâches de manutention, qui génèrent plusieurs voyages à vide éventuels.

Il est à noter que plusieurs variables définies dans le modèle vont impérativement avoir une valeur qui n'influencera pas la solution recherchée. Il s'agit des variables de dates de début de tous les voyages à vide éventuels qui n'auront pas lieu. Ces variables comprennent aussi les variables d'affectation de ces voyages à vide aux ressources sollicitées en routes, et les variables de détermination d'ordre. Toutefois, l'affectation aux chariots n'étant pas déterminée a priori, cette définition de variables est inévitable.

Les problèmes tests ont été résolus à l'optimalité à l'aide du logiciel OPLStudio version 3.6 et dans des temps très courts pour la majorité des jeux de données. Les résultats sont présentés ci-après.

Tableau 5.2. Résultats pour la formulation linéaire mixte

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-CR-1-1	21	4	2	16995	6583	4,86	63	70	620
PM-CR-1-2	21	4	2	16995	9111	5,64	61	37	1232
PM-CR-1-3	21	4	2	16995	8711	5,45	66	70	1208
PM-CR-1-4	21	4	2	16995	8207	6,31	62	100	1780
PM-CR-2-1	24	4	2	25401	1896	1,94	74	47	287
PM-CR-2-2	24	4	2	25401	1972	2,36	71	55	427
PM-CR-2-3	24	4	2	25401	1968	2,17	73	37	266
PM-CR-2-4	24	4	2	25401	1940	2,14	73	35	209
PM-CR-3-1	27	4	2	35940	2271	2,66	75	5	163
PM-CR-3-2	27	4	2	35940	2373	3,17	72	2	111
PM-CR-3-3	27	4	2	35940	2363	3,13	74	1	81
PM-CR-3-4	27	4	2	35940	2327	2,91	74	11	160
PM-CR-4-1	33	4	2	113388	4465	20 *	62	---	---
PM-CR-4-2	33	4	2	113388	4686	15,31	56	40	385
PM-CR-4-3	33	4	2	113421	4671	25*	59	---	---
PM-CR-4-4	33	4	2	113408	4608	6*	57	---	---
PM-CR-5-1	21	4	2	16995	1438	1,55	52	0	148
PM-CR-5-2	21	4	2	16995	1502	1,86	49	0	149

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variables	Nb. contraintes	Temps secondes	Objectif	Noeuds	Itérations
PM-CR-5-3	21	4	2	16995	1490	1,80	53	0	111
PM-CR-5-4	21	4	2	16995	1460	1,73	50	9	142
PM-CR-6-1	30	4	2	66768	3371	9,27	95	446	4762
PM-CR-6-2	30	4	2	66768	3507	10,69	91	427	4214
PM-CR-6-3	30	4	2	66768	3491	10,84	94	508	3907
PM-CR-6-4	30	4	2	66768	3439	9,30	93	211	2376
PM-CR-7-1	31	4	2	51021	2876	6,95	66	645	2839
PM-CR-7-2	31	4	2	51021	2986	7,14	66	155	801
PM-CR-7-3	31	4	2	51021	2986	72	68	37831	175756
PM-CR-7-4	31	4	2	51021	2942	6,64	66	141	957
PM-CR-8-1	34	4	2	114556	4567	18,59	147	1151	9419
PM-CR-8-2	34	4	2	114556	4735	22,03	143	1124	9904
PM-CR-8-3	34	4	2	114556	4735	20,70	149	949	9886
PM-CR-8-4	34	4	2	114556	4643	28,97	148	2548	27267
PM-CR-9-1	29	4	2	65911	3329	8,11	88	271	2135
PM-CR-9-2	29	4	2	65911	3459	10,25	84	440	2926
PM-CR-9-3	29	4	2	65911	3459	10,22	89	501	3160
PM-CR-9-4	29	4	2	65911	3397	16,36	87	946	4230
PM-CR-10-1	36	4	2	146475	5232	36,66	121	1561	9768
PM-CR-10-2	36	4	2	146475	5446	29,94	116	1689	11488
PM-CR-10-3	36	4	2	146475	5394	28,77	119	1318	11337
PM-CR-10-4	36	4	2	146475	5310	38,14	120	2106	15098

* Temps nécessaire pour trouver la solution sans preuve d'optimalité.

Légende

PM-CR-no.1-no.2

PM	Programmation Mathématique
CR	Choix en Routes
no.1	numéro du jeu de données
no.2	numéro de l'implantation

Les cases grises dans le tableau correspondent aux problèmes qui n'ont pas été résolus à l'optimalité.

Il est à noter que le temps, que le logiciel prend, comprend le temps de compilation, de résolution et d'affichage de résultats. Les résultats sont obtenus dans des temps très acceptables avec preuve d'optimalité (temps moyen de 13 secondes) sauf pour les problèmes PM-CR-4-1, PM-CR-4-3 et PM-CR-4-4, qui sont indiqués en gris dans le tableau, où le temps nécessaire pour prouver l'optimalité était très long. Les résultats obtenus pour le premier et le deuxième problème ne sont pas optimaux. L'optimal du premier problème se situe à 61 et est obtenu en résolvant la formulation en programmation par contraintes. Pour le deuxième, la résolution de la formulation en programmation par contraintes présentée plus loin donne un objectif de 58 sans preuve d'optimalité. Nous obtenons un objectif de 57 pour le troisième problème sans preuve d'optimalité. Notons aussi que les données de ces problèmes présentent un phénomène de recirculation où les travaux visitent une même machine plus qu'une fois.

On remarque que les problèmes comportent un très grand nombre de contraintes et surtout de variables, et les résultats sont très encourageants. Nous avons effectué des essais avec des problèmes d'ordonnancement ayant ce même nombre de variables et même moins, sans inclure la manutention. Nous avons remarqué que le problème d'ordonnancement intégré se résout mieux pour un même nombre de variables. En d'autres termes, la prise en compte des contraintes relatives au système de manutention ne complique pas nécessairement la résolution. La criticité des ressources et les choix multiples parmi les ressources influencent le temps de résolution du problème.

Tel que précisé au chapitre dernier, le choix de la grande valeur H doit être effectué très soigneusement pour éviter les problèmes de précision numérique. Ceci veut dire que la valeur de H doit être juste un peu plus grande que la longueur estimée de l'horaire.

C'est une conséquence typique pour ce type de modèles qui a été d'ailleurs remarquée pendant les tests.

5.2. Formulation en programmation par contraintes

Le modèle formulé à l'aide de la programmation par contraintes est basé sur des idées de modélisation intéressantes et flexibles et qui sont élégamment transcrites dans le logiciel OPLStudio. La durée de l'activité est définie comme étant variable. Elle prend la longueur de la route à laquelle une tâche de manutention ou un voyage à vide est affecté. Plusieurs variables du type *énuméré* sont définies pour la détermination de l'ordre et l'affectation aux ressources. Par exemple, une variable de type « chariot » est définie pour déterminer le chariot auquel une activité sera affectée. De même, une variable de type « tâche de manutention » est définie pour déterminer quelle tâche de manutention suivra quelle autre tâche de manutention sur un chariot. Chaque tâche de manutention est suivie directement par une seule autre tâche et une tâche suivante ne doit pas suivre plus d'une tâche précédente. Ces variables permettent la formulation des contraintes relatives aux voyages à vide de façon très concise en utilisant des relations d'implication.

Pour construire des chemins pour chaque chariot, on crée des tâches de manutention fictives. Deux tâches fictives sont créées pour chaque chariot, une tâche de début et une tâche de fin. Le trajet d'un chariot commence par une tâche fictive suivie par un nombre de tâches de manutention qui y sont affectées, et finit par une autre tâche fictive. Le nombre total de tâches fictives créées est égal au double du nombre de chariots.

5.2.1. Notation

Machines	Ensemble de machines
Tasks	Ensemble de tâches

Tasksmach	Ensemble de tâches de production $\subset Tasks$
Tasksman	Ensemble de tâches de manutention $\subset Tasks$
Chariots	Ensemble de Chariots
MANfictif	Ensemble de tâches de manutention fictives représentant la fin d'un trajet
MANorigine	Ensemble de tâches de manutention fictives représentant le début d'un trajet
TasksmanFIC	Ensemble des tâches de manutention et les tâches fictives de manutention représentant la fin d'un trajet $Tasksman \cup MANfictif$
TasksmanSour	Ensemble des tâches de manutention et les tâches fictives de manutention représentant le début d'un trajet $Tasksman \cup MANorigine$
TasksmanSourFIC	Ensemble des tâches de manutention et les tâches fictives de manutention représentant le début et la fin d'un trajet $Tasksman \cup MANfictif \cup MANorigine$
SetOfPrecedences	Ensemble de paires de tâches possédant une relation d'antériorité
Vav	Ensemble de paires de tâches de manutention représentant un voyage à vide éventuel (tâche de manutention avant et tâche de manutention après)
Routes	Ensemble de routes
mayhost[t]	Ensembles de routes pouvant accueillir une tâche $t \in Tasks$
mighthost[v]	Ensembles de routes pouvant accueillir un voyage à vide $v \in vav$
RoutesConflictuelles	Ensemble de paires de routes conflictuelles (r1,r2) partageant un ou plusieurs segments $r1 \in Routes, r2 \in Routes$
Duration [t]	Temps de traitement sur machines $t \in Tasksmach$

Duree [r]	Temps requis pour parcourir une route $r \in Routes$
resourcem [t]	Données de type « Machines » indiquant quelle machine est sollicitée par une tâche $t \in Tasksmach$

Variables

FollowsonChariot[t]	Variable de type « Tasks » indiquant quelle tâche suivra une tâche de manutention t ou encore une tâche de début sur un chariot $t \in TasksmanSour$
occupiedBy [v]	Variable de type « Routes » indiquant quelle route va accueillir un voyage à vide v, $v \in vav$
carries [t]	Variable de type « Chariots » indiquant quel chariot va manutentionner une tâche de manutention t ou encore quelle tâche fictive de début ou de fin de trajet t $t \in TasksmanSourFIC$
hosts [t]	Variable de type « Routes » indiquant quelle route va accueillir une tâche de manutention t, $t \in Tasks$
dureeVIDE[v]	Variable indiquant la durée d'un voyage à vide v, $v \in vav$
S[t]	Variable indiquant la date de début d'une tâche de production ou de manutention t, $t \in Tasks$
SV[v]	Variable indiquant la date de début du voyage à vide v, $v \in vav$
S[makespan]	Date de fin de l'horaire, correspondant à la date de début d'une tâche de fin fictive

Dans le modèle, les machines, les chariots et les routes sont définies comme étant des **ressources unaires**. En même temps, les chariots représentent des choix multiples en ressources pour l'ensemble des tâches de manutention. Cependant, la définition de la variable « carries » élimine le besoin de définir les chariots comme des **ressources**

alternatives. Les sous-ensembles des routes pouvant accueillir une tâche de manutention ou un voyage à vide constituent des choix multiples pour les tâches admissibles. La définition des variables « hosts » et « occupiedBy » élimine le besoin de définir les routes comme étant des **ressources alternatives.**

La définition des **ressources discrètes** en OPL est un aspect important qui a été utilisé dans ce modèle. Il est parfois plus facile de trouver une solution pour un problème d'ordonnancement à ressources discrètes et par la suite transformer la solution pour respecter les contraintes de **ressources unaires.** Une **ressource discrète** est une ressource à capacité discrète. La capacité, qui peut varier dans le temps, représente le nombre de copies disponibles d'une même ressource. Les ressources discrètes peuvent être utilisées pour modéliser un budget, des opérateurs ou des machines identiques du point de vue d'une tâche. Contrairement aux **ressources unaires**, une tâche peut être affectée à différentes ressources à des moments différents. Il suffit juste de ne pas dépasser la capacité des ressources disponibles.

Parallèlement à la définition des ressources en OPLStudio, le mot clé **requires** est utilisé pour désigner les besoins en ressources. Si une activité requiert une **ressource unaire,** les contraintes de disjonction sont générées et on n'a pas besoin de les écrire explicitement.

Le modèle comprend un nombre de contraintes redondantes qui améliorent le temps de résolution. Au début des travaux, nous avons développé un modèle plus compact, à l'image de celui présenté au chapitre dernier. Malheureusement, la performance n'était pas bonne. Le modèle développé par la suite, et présenté dans la section suivante, présente une meilleure performance, cependant moins bonne que celle du modèle de programmation linéaire mixte.

5.2.2. Modèle

Le modèle a l'objectif de minimiser la date de fin de l'horaire. Les contraintes sont suivies d'explications.

Objectif

Minimiser

S [makespan] (1)

L'objectif est de minimiser la date de fin de l'horaire en respectant les contraintes suivantes :

Relations d'antériorité entre la date de fin de l'horaire et les tâches

$S[makespan] \geq S[t] + duration[t] \quad \forall t \in Tasks$ (2)

La date de fin de l'horaire est plus grande ou égale à la date de début de chacune des tâches plus sa durée de traitement.

Des contraintes redondantes y sont incluses

Seule la dernière tâche de chaque travail doit être finie avant le début de la tâche fictive « makespan ». Les autres contraintes sont assurées par les contraintes (4).

Relations d'antériorité entre la date de fin de l'horaire et les voyages à vide

$S[makespan] \geq SV[v] + dureeVIDE[v] \quad \forall v \in vav$ (3)

La date de fin de l'horaire est plus grande ou égale à la date de début de chacun des voyages à vide plus sa durée de traitement.

Contraintes redondantes

Les contraintes d'antériorité pour les voyages à vide sont assurées par les contraintes (5) et (7).

Relations d'antériorité entre deux tâches successeurs

$S[q] \geq S[t] + duration[t] \quad \forall (t,q) \in setOfPrecedences$ (4)

La date de début d'une tâche *q* successeur d'une tâche *t* est plus grande ou égale à la date de début plus la durée de traitement de cette dernière.

Génération de relations d'antériorité entre les tâches de manutention et les voyages à vide

followsonChariot [p] = q $\Rightarrow$ S[p] + Duration [p] $\leq$ SV[v]

$\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q (5)

Si deux tâches de manutention se suivent sur un chariot, ceci implique que la date de début de la première plus sa durée soit inférieure ou égale à la date de début du voyage à vide résultant de cette affectation de tâches de manutention au chariot.

Génération de relations d'antériorité entre les tâches de manutention

followsonChariot [p] = q $\Rightarrow$ S[p] + Duration [p] $\leq$ S[q]

$\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q (6)

Si deux tâches de manutention se suivent sur un chariot, ceci implique que la date de début de la première plus sa durée soit inférieure ou égale à la date de la deuxième.

Contraintes redondantes

Ces contraintes sont assurées par les contraintes (7).

Génération de relations d'antériorité entre les tâches de manutention et les voyages à vide

followsonChariot [p]= q $\Rightarrow$ SV[v] + dureeVIDE[v] $\leq$ S [q]

$\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q (7)

Si deux tâches de manutention se suivent sur un chariot, ceci implique que la date de début de la deuxième soit supérieure ou égale à la date de début du voyage à vide résultant de cette affectation de tâches de manutention au chariot plus sa durée.

Détermination d'ordre de visite des chariots

alldifferent (followsonChariot) (8)

Une tâche de manutention peut suivre directement une seule autre tâche de manutention (tâches fictives incluses) sur un chariot.

Détermination d'ordre de visite des chariots

followsonChariot [t] = q $\Rightarrow$ t <> q

$\forall$(t,q)|p $\in$TasksmanSour et q $\in$ TasksmanFIC (9)

Une tâche de manutention ne peut pas se suivre.

Détermination d'ordre de visite des chariots

followsonChariot [p]= q $\Rightarrow$ q in TasksmanFIC

$\forall$ p $\in$ TasksmanSour, q $\in$ Tasks (10)

Si deux tâches de manutention se suivent sur un chariot, ceci implique que la tâche successeur soit une tâche admissible. Une tâche admissible est soit une tâche de manutention soit une tâche fictive de fin.

Définition du début et de fin de trajet pour les chariots

carries [Debut] = carries[fin] (11)

carries [Debut] <> carries[Source] (12)

carries[Source] = carries[Pasdetache] (13)

carries [Pasdetache] <> carries[fin] (14)

Contraintes de définition de début et de fin de trajet pour chaque chariot. Les tâches en argument de la variable « carries » sont les tâches fictives de début et fin de trajet.

Définition des durées des tâches de production et de manutention

duree [hosts [t]] = duration [t] $\forall$ t $\in$ Tasksmach $\cup$ Tasksman (15)

La durée d'une tâche de production ou de manutention est égale à la durée requise pour parcourir la route qu'elle entreprend; les tâches de production prennent des routes fictives dont la durée de parcours est égale à la durée de traitement sur les machines.

Définition des durées pour les voyages à vide

followsonChariot[p]=q $\Rightarrow$ dureeVIDE[v]=duree [occupiedBy[v]] (16)

$\forall$ r $\in$ routes, $\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q

Si de tâches de manutention se suivent sur un chariot, ceci implique que le voyage à vide résultant de cette affectation a une durée égale à la longueur de la route que prend ce voyage à vide.

Affectation des tâches de manutention aux routes

$hosts[t] = r \Rightarrow r \text{ in } mayhost[t] \qquad \forall t \in Tasksman, \forall r \in Routes \qquad (17)$

Si une route *r* accueille une tâche de manutention *t*, ceci implique que cette route fasse partie de l'ensemble des routes pouvant accueillir cette tâche.

Affectation des voyages à vide aux routes

$occupiedBy[v] = r \Rightarrow r \text{ in } mighthost[v] \qquad \forall v \in vav, \forall r \in Routes \qquad (18)$

Si une route *r* accueille un voyage à vide *v*, ceci implique que cette route fasse partie de l'ensemble des routes pouvant accueillir ce voyage à vide.

Besoin et disjonction des machines

$$S[q] \geq S[t] + d[t] \vee S[t] \geq S[q] + d[q]$$
$$\forall t \in Tasksmach, \forall m \mid resourcem[t] = resourcem[q] = m; \qquad (19)$$

Pour deux tâches *t* et *q* sollicitant une même machine, soit la tâche *t* commence après la fin de la tâche *q*, soit la tâche *q* commence après la fin de la tâche *t*.

Besoin en chariots pour les tâches de manutention

$carriers[t] = c \Rightarrow t$ **requires** $c \qquad \forall t \in TasksmanSourFIC, \forall c \in Chariots \qquad (20)$

Si une tâche de manutention *t* est affectée à un chariot *c*, cela implique que le chariot soit mobilisé pendant la durée de la tâche. La disjonction est assurée par la déclaration de ressources unaires.

Besoin en routes pour les tâches de manutention

$hosts[t] = r \Rightarrow t$ **requires** $r \qquad \forall t \in Tasksman, \forall r \in mayhost[t] \qquad (21)$

Si une tâche de manutention *t* est affectée à une route *r*, cela implique que la route soit mobilisée pendant la durée de la tâche. La disjonction est assurée par la déclaration de ressources unaires.

Besoin en routes pour les voyages à vide

followsonChariot[p] = q & occupiedBy [v]= r $\Rightarrow$ vide[v] **requires** r (22)

$\forall$ r $\in$ routes, $\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ Tasksman, v après p et avant q

Si deux tâches de manutention se suivent sur un chariot, et que le voyage à vide résultant de cette affectation prend la route *r*, ceci implique que la mobilisation de cette route pendant la durée de ce voyage à vide.

Besoin en chariots pour les tâches de manutention

followsonChariot[p] = q $\Rightarrow$ carries [p]=carries [q] (23)

$\forall$ p $\in$ TasksmanSour, q $\in$ Tasks

Si deux tâches de manutention se suivent sur un chariot, ceci implique que les deux tâches soient affectées à ce même chariot.

Besoin en chariots pour les voyages à vide

followsOnChariot [p] = q & carries [p] = c $\Rightarrow$ vide[v] **requires** c (24)

$\forall$ c $\in$ Chariots, $\forall$ p $\in$ TasksmanSour, $\forall$ q $\in$ TasksmanFIC,

$\forall$ v $\in$ vav | v après p et avant q

Si deux tâches de manutention sont affectées successivement à un chariot, ce dernier est mobilisé pendant la durée du voyage à vide résultant de cette affectation.

Contraintes de conflits pour les tâches de manutention et les voyages à vide

OccupiedBy [v] =r2 & hosts[t]= r1 $\Rightarrow$ a[t].start $\geq$vide [v].end $\vee$ vide[v].start $\geq$a[t].end

$\forall$ (r1,r2) $\in$ RoutesConflictuelles, $\forall$ t $\in$ Tasksman,

$\forall$ v $\in$ vav| r1 $\in$ mayhost[t]& r2 $\in$ mighthost[v] (25)

Si le voyage à vide *v* et la tâche de manutention *t* occupent deux routes conflictuelles *r1* et *r2* respectivement, il faut que les contraintes de disjonction soient respectées.

Contraintes de conflits pour les tâches de manutention et les voyages à vide

OccupiedBy [v] = r1 & hosts[t]= r2 $\Rightarrow$ a[t].start $\geq$vide [v].end $\vee$ vide[v].start $\geq$a[t].end

$\forall$ (r1,r2) $\in$ RoutesConflictuelles, $\forall$ t $\in$ Tasksman,

$\forall$ v $\in$ vav| r1 $\in$ mayhost[t]& r2 $\in$ mighthost[v] (26)

Si le voyage à vide *v* et la tâche de manutention *t* occupent deux routes conflictuelles *r2* et *r1* respectivement, il faut que les contraintes de disjonction soient respectées.

Contraintes de conflits pour les voyages à vide

OccupiedBy[v] = r2 & OccupiedBy[f] = r1 $\Rightarrow$ vide[f].start $\geq$ vide [v].end $\vee$ vide[v].start $\geq$vide[f].end (27)

$\forall$ (r1,r2) $\in$ RoutesConflictuelles, $\forall$ v $\in$ vav,

$\forall$ f $\in$ vav| r1 $\in$ mighthost[f] & r2 $\in$ mighthost[v]

Si deux voyage à vide *v* et *f* occupent deux routes conflictuelles *r1* et *r2*, il faut que les contraintes de disjonction soient respectées.

Contraintes de conflits pour les tâches de manutention

hosts [q] = r2 & hosts [t]= r1 $\Rightarrow$ a[t].start $\geq$a[q].end $\vee$ a[q].start $\geq$a[t].end (28)

$\forall$ (r1,r2) $\in$ RoutesConflictuelles, $\forall$ t $\in$ Tasksman, $\forall$ q $\in$ Tasksman | r1 $\in$ mayhost[t] & r2 $\in$ mayhost[q]

Si deux tâches de manutention *q* et *t* occupent deux routes conflictuelles *r1* et *r2*, il faut que les contraintes de disjonction soient respectées.

On ajoute à ces familles de contraintes des contraintes d'élimination de sous-tours au besoin, pour s'assurer que uniquement des chemins correspondants au nombre de chariots sont construits. En général des contraintes éliminant des cycles de trois sont suffisantes. L'ajout de ces contraintes ne complique généralement pas la solution. Un exemple de ces contraintes suit :

followsonChariot [p]= q & followsonChariot [q]= j $\Rightarrow$ not followsonChariot [j]= p

$\forall$ p,q,j $\in$ Tasks

Tableau. 5.3. Modèle complet en programmation par contraintes

Minimiser S [makespan]	(1)
S[makespan] $\geq$ S[t] + duration[t] $\forall$ t $\in$ Tasks	(2)
S[makespan] $\geq$ SV[v] + dureeVIDE [v] $\forall$ v $\in$ vav	(3)
S[q] $\geq$ S[t] + duration [t] $\forall$(t,q) $\in$ setOfPrecedences	(4)
followsonChariot [p]= q $\Rightarrow$ S[p] + Duration [p] $\leq$SV[v] $\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q	(5)
followsonChariot [p]= q $\Rightarrow$ S[p] + Duration [p] $\leq$S[q] $\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q	(6)
followsonChariot [p]= q $\Rightarrow$ SV[v] + dureeVIDE[v] $\leq$S [q] $\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q	(7)
alldifferent (followsonChariot)	(8)
followsonChariot [t] = q $\Rightarrow$ t <> q $\forall$(t,q)|p $\in$TasksmanSour et q $\in$ TasksmanFIC	(9)
followsonChariot [p]= q $\Rightarrow$ q in TasksmanFIC $\forall$ p $\in$ TasksmanSour, q $\in$ Tasks	(10)
carries [Debut] = carries[fin]	(11)
carries [Debut] <> carries[Source]	(12)
carries[Source] = carries[Pasdetache]	(13)
carries [Pasdetache] <> carries[fin]	(14)
duree [hosts [t]]=duration [t] $\forall$ t $\in$ Tasksmach $\cup$ Tasksman	(15)
followsonChariot[p]=q $\Rightarrow$ dureeVIDE[v]=duree [occupiedBy[v]] $\forall$ r $\in$ routes, $\forall$ v $\in$ vav, $\forall$(p,q)| p $\in$ Tasksman, q $\in$ TasksmanFIC, v après p et avant q	(16)

hosts[t]= r ⇒ r in mayhost[t]; ∀t ∈ Tasksman, ∀ r ∈ Routes	(17)
occupiedBy[v]=r ⇒ r in mighthost [v] ∀v ∈ vav, ∀ r ∈ Routes	(18)
S[q] ≥S[t] + d[t] ∨ S[t] ≥S[q] +d[q] ∀ t ∈ Tasksmach, ∀ m | resourcem[t]= resourcem[q]= m	(19)
carriers[t]= c ⇒ t **requires** c ∀ t ∈ TasksmanSourFIC, ∀ c ∈ Chariots	(20)
hosts[t]= r ⇒ t **requires** r ∀ t ∈ Tasksman, ∀ r ∈ mayhost [t]	(21)
followsonChariot[p]=q & occupiedBy [v]= r ⇒ vide[v] **requires** r ∀ r ∈ routes, ∀ v ∈ vav, ∀(p,q)| p ∈ Tasksman, q ∈ Tasksman, v après p et avant q	(22)
followsonChariot[p]=q ⇒ carries [p] = carries [q] ∀ p ∈ TasksmanSour, q ∈ Tasks	(23)
followsonChariot[p]=q & carries [p]=c ⇒ vide[v] **requires** c ∀ p ∈ TasksmanSour, ∀ q ∈ TasksmanFIC, v après p et avant q,∀ v ∈ vav, ∀ c ∈ Chariots	(24)
OccupiedBy [v] =r2 & hosts[t]= r1 ⇒ a[t].start ≥vide [v].end ∨ vide[v].start ≥ a[t].end ∀ (r1,r2) ∈ RoutesConflictuelles, ∀ t ∈ Tasksman, ∀ v ∈ vav| r1 ∈ mayhost[t]& r2 ∈ mighthost[v]	(25)
OccupiedBy [v] = r1 & hosts[t]= r2 ⇒ a[t].start ≥vide [v].end ∨ vide[v].start ≥a[t].end ∀ (r1,r2) ∈ RoutesConflictuelles, ∀ t ∈ Tasksman, ∀ v ∈ vav| r1 ∈ mayhost[t]& r2 ∈ mighthost[v]	(26)
OccupiedBy [v] = r2 & OccupiedBy [f] = r1 ⇒ vide[f].start ≥vide[v].end ∨ vide[v].start ≥vide[f].end ∀ (r1,r2) ∈ RoutesConflictuelles, ∀ v ∈ vav, ∀ f ∈ vav| r1 ∈ mighthost[f] & r2 ∈ mighthost[v]	(27)

hosts [q] = r2 & hosts [t]= r1 $\Rightarrow$ a[t].start $\geq$a[q].end $\vee$ a[q].start $\geq$a[t].end $\forall$ (r1,r2) $\in$ RoutesConflictuelles, $\forall$ t $\in$ Tasksman, $\forall$ q $\in$ Tasksman | r1 $\in$ mayhost[t] & r2 $\in$ mayhost[q] *Contraintes de sous-tours au besoin*	(28)

5.2.3. Expérimentation

La procédure de recherche utilisée pour le modèle en programmation par contraintes fonctionne de façon aléatoire. Il s'agit de l'instruction **generate** qui fait partie des procédures de génération en OPLStudio. Les procédures de génération reçoivent une variable discrète ou un tableau arbitraire de variables discrètes, et génèrent des valeurs pour ces variables. L'instruction **generate** génère des valeurs pour les variables en argument en considérant la variable avec le plus petit domaine en premier.

Une stratégie de recherche complémentaire au modèle est utilisée. L'instruction **generate** est employée pour générer des valeurs pour différentes variables. De plus, une stratégie de recherche dichotomique combinée avec « Slice Based Search » est utilisée. Cette dernière est aussi appelée « Limited Discrepancy Search ». Elle présume l'existence d'une bonne heuristique pour résoudre un problème. L'intuition derrière l'utilisation d'une telle stratégie est que l'heuristique échoue car probablement elle devait prendre un nombre restreint de décisions différentes à certaines étapes. Dans ce cas là, elle aurait peut-être trouvée une solution. Les décisions différentes de celle de l'heuristique, sont appelées « discrepancy » ou incompatibilité. La stratégie de recherche fait l'exploration de l'arbre en incrémentant le nombre d'incompatibilités permises jusqu'à ce qu'on trouve une meilleure solution ou jusqu'à ce que l'arbre soit complètement exploré.

Les mêmes jeux de données utilisés pour tester la formulation linéaire mixte ont été utilisés pour tester le modèle en programmation par contraintes. Les résultats de l'expérimentation sont présentés au tableau 5.4.

Tableau 5.4. Résultats pour la formulation en programmation par contraintes

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variable	Nb. contraintes	Temps secondes	Objectif	Échecs	Points de choix
PC-CR-1-1	21	4	2	971	5569	15	63	17638	18815
PC-CR-1-2	21	4	2	971	6778	26	61	24500	26176
PC-CR-1-3	21	4	2	971	5569	10	66	11324	12531
PC-CR-1-4	21	4	2	960	6773	14	62	19048	20522
PC-CR-2-1	24	4	2	1189	8829	21	74	24679	26329
PC-CR-2-2	24	4	2	1189	11284	27	71	10261	12966
PC-CR-2-3	24	4	2	1189	8813	28	73	26710	28307
PC-CR-2-4	24	4	2	1189	10363	83	73	32038	42846
PC-CR-3-1	27	4	2	1419	11258	55	75	57208	59316
PC-CR-3-2	27	4	2	1419	14207	30	72	12	12027
PC-CR-3-3	27	4	2	1419	13641	18	74	9	7069
PC-CR-3-4	27	4	2	1419	12763	27	74	9	14060
PC-CR-4-1	33	4	2	2353	29655	1739	61	118529	123133
PC-CR-4-2	33	4	2	2353	40484	250*	56	8	19504
PC-CR-4-3	33	4	2	2353	37843	15524*	58	156715	164530
PC-CR-4-4	33	4	2	2353	--	2711*	66	--	--
PC-CR-5-1	21	4	2	971	5605	2,13	52	48	2938
PC-CR-5-2	21	4	2	971	6911	3,9	52	48	4176
PC-CR-5-3	21	4	2	971	6697	4,7	53	27	5979
PC-CR-5-4	21	4	2	971	6359	2,3	50	6	2953
PC-CR-6-1	30	4	2	1801	18780	78*	95	231	19638
PC-CR-6-2	30	4	2	1801	26912	400*	91	890	29844
PC-CR-6-3	30	4	2	1801	26184	170	94	206	13486
PC-CR-6-4	30	4	2	1801	24240	143	93	201	12074

Données	Nb. tâches	Nb. machines	Nb. chariots	Nb. variable	Nb. contraintes	Temps secondes	Objectif	Échecs	Points de choix
PC-CR-7-1	31	4	2	1691	16383	30	66	41	11879
PC-CR-7-2	31	4	2	1691	21345	109	66	14	15965
PC-CR-7-3	31	4	2	1691	20967	120	68	42	16651
PC-CR-7-4	31	4	2	1961	19706	92	66	33	15084
PC-CR-8-1	34	4	2	2363	68344	500*	147	17	----
PC-CR-8-2	34	4	2	2363	37112	600*	143	468	38339
PC-CR-8-3	34	4	2	2363	37664	500*	149	54	32746
PC-CR-8-4	34	4	2	2363	33124	650*	148	555	37091
PC-CR-9-1	29	4	2	1935	18661	60	88	138	15451
PC-CR-9-2	29	4	2	1935	25027	235	84	13	28741
PC-CR-9-3	29	4	2	1935	24664	90*	89	137	11911
PC-CR-9-4	29	4	2	1935	23188	125*	87	340	11712
PC-CR-10-1	36	4	2	2674	39961	194*	128	17	7923
PC-CR-10-2	36	4	2	2674	54631	2010*	138	61095	71982--
PC-CR-10-3	36	4	2	2674	52047	2287*	131	80649	91292
PC-CR-10-4	36	4	2	2674		2480*	122	---	---

* Temps nécessaire pour trouver la solution sans preuve d'optimalité.

Légende

PM-CR-no.1-no.2

PM	Programmation Mathématique
CR	Choix en Routes
no.1	numéro du jeu de données
no.2	numéro de l'implantation

Le temps de résolution donné pour les problèmes PC-CR-6-1, PC-CR-6-2, les problèmes PC-CR-8-1, PC-CR-8-2, PC-CR-8-3, PC-CR-8-4, et les problèmes PC-CR-9-3, PC-CR-9-4 est le temps nécessaire pour atteindre la solution optimale sans preuve d'optimalité. Cela a été très long (plusieurs heures voire jours). Nous n'avons pas trouvé la solution

optimale pour les problèmes PC-CR-4-2, PC-CR-4-3, PC-CR-4-4, PC-CR-10-1, PC-CR-10-2, PC-CR-10-3, PC-CR-10-4, indiqués en gris dans le tableau. Les résultats obtenus sont inscrits au tableau 5.4.

5.3. Analyse des résultats

À l'exception des problèmes PM/PC-CR-4-3 et PM/PC-CR-4-4, les jeux de données présentés dans ce chapitre ont été résolus à l'optimalité tel que présenté dans les sections présentant l'expérimentation des deux modèles. Cependant, une solution réalisable est obtenue très rapidement. À noter également que le temps inscrit dans les tableaux représente le temps de calcul et aussi d'affichage de résultats.

Tel que présenté plus haut, le réseau considéré est bidirectionnel. Cela veut dire qu'on a choisi les deux plus courts chemins sans contraintes d'orientation d'arcs. Les données de la littérature utilisées dans le chapitre précédent considèrent un réseau unidirectionnel où la distance entre deux postes peut considérablement accroître selon la direction. La justification pour ce faire était d'éviter les conflits à base. Dans un réseau unidirectionnel, d'une part nous évitons les collisions frontales et d'autre part le nombre de tâches de manutention sollicitant un chemin en particulier est moindre. Cependant, il n'y a aucune garantie que les conflits soient évités. Les résultats de ce chapitre justifient l'approche intégrée et montrent que quand le réseau est bidirectionnel et que les contraintes de conflits sont respectées, on peut avoir un horaire plus court (une meilleure utilisation des ressources) par rapport à un réseau unidirectionnel où nous éliminons les contraintes de conflits. Des exemples d'horaires sont fournis dans l'annexe B.

Le développement du modèle en programmation par contraintes était un exercice intéressant et empirique. Au début des travaux, nous avons développé un modèle compact qu'on pensait être efficace. Il n'a pas fourni de bons résultats. Le modèle retenu, et présenté dans ce chapitre, comporte plus de variables et de contraintes, que nous avons pensé inclure au début. Pourtant, il présente une meilleure performance.

La performance de ce modèle comparée à celle de la formulation linéaire mixte est moins bonne. La formulation linéaire mixte possède un très grand nombre de variables (jusqu'à 147 000) et de contraintes (jusqu'à presque 10 000) et la résolution est très efficace. Les résultats sont très satisfaisants. En fait, les contraintes de conflits, de détermination d'ordre et d'affectation présentent des inégalités qui réduisent l'espace réalisable par rapport au problème d'ordonnancement de la production. Le même nombre de contraintes et de variables décrivant un problème d'ordonnancement multigamme conventionnel peut résulter en des jeux plus difficiles à résoudre et cela a même été testé lors des travaux. Nous précisons que les données influencent la difficulté du problème d'ordonnancement avec ou sans contraintes de manutention.

Nous avons utilisé des procédures de recherche fournies par le logiciel pour améliorer la performance du modèle en programmation par contraintes. Règle générale une stratégie de recherche dichotomique combiné avec « Slice-based search » fournit les meilleures résultats. La flexibilité du solveur commercial permet aussi la définition d'autres procédures de recherche taillées sur mesure pour un problème spécifique. Une telle procédure peut possiblement permettre une recherche efficace de solutions et améliorer le temps de calcul. Donc, nous ne pouvons pas à ce stade confirmer que la programmation par contraintes n'est pas adaptée à ce problème. De plus, pour tous les problèmes tests utilisés, si une solution optimale est relativement longue à trouver au moins une *bonne* solution est obtenue dans un temps très court (moins d'une minute). Seuls les jeux de données 4 et 10 présentent des difficultés surtout avec la deuxième formulation en programmation par contraintes. Par contre, cette formulation ne présente pas de problèmes de précision numérique comme c'est le cas de la formulation linéaire mixte qui utilise des grandes valeurs.

Selon la littérature sur l'ordonnancement (Baker, 1978), une méthode très utile pour la détermination des valeurs des variables de date de début pour les problèmes d'ordonnancement c'est la détermination des dates de début plus tôt et des dates de

début au plus tard. Cela est fait dans le but de cerner l'éventail des valeurs potentielles pour les variables. Cette idée a également été avancée par des spécialistes de la programmation par contraintes (Baptiste et al., 2001). En programmation par contraintes, l'ajout de ce type de contraintes à notre modèle peut théoriquement améliorer la performance du modèle. Le calcul des dates de début au plus tôt et au plus tard se fait à partir de l'information fournie dans les données du problème. Ces contraintes redondantes réduisent de façon importante les domaines de variables et donc l'espace de recherche (Marriot et Stuckey, 1998).

Cet effort de programmation peut améliorer le programme et le rendre plus pratique. Cependant, la détermination des dates de début au plus tôt et au plus tard ne constitue pas une tâche facile. Elle doit se faire avec beaucoup de précision pour ne pas éliminer des solutions potentiellement optimales. C'est une avenue à explorer.

CHAPITRE 6 :

PROBLÈME D'ORDONNANCEMENT INTÉGRÉ DE LA PRODUCTION ET DE LA MANUTENTION INCLUANT LE ROUTAGE SANS CONFLITS ET UNE CAPACITÉ LIMITÉE DES FILES D'ATTENTE

Le troisième problème, présenté dans ce chapitre, utilise la structure du réseau de l'atelier pour fabriquer les routes pour les tâches de manutention et les voyages à vide sans conflits. Des contraintes de capacité limitée pour les files d'attente sont également ajoutées. De plus, nous ajoutons à la gamme de chaque travail des tâches de préparation et de palettisation qui se déroulent à un poste de capacité suffisante. Un travail débute son séjour en atelier à partir de son départ de l'entrepôt des matières premières. Son séjour finit au moment de son entreposage à l'entrepôt des produits finis. Les modèles présentés plus loin sont très riches et détaillés. C'est la première fois dans la littérature qu'on propose des modèles aussi riches pour un problème d'ordonnancement en production.

6.1. Problème

Une implantation représentative de l'atelier à l'étude, est présentée à la figure 6.1. Elle inclut six machines dotées d'une file d'attente de capacité limitée et quelques segments de route. Elle inclut également un poste de préparation et de palettisation, et un entrepôt pour les matières premières et les produits finis. Les postes de préparation et de palettisation sont représentés par un seul lieu physique qui possède une capacité suffisante pour accueillir tous les travaux dans l'atelier. C'est également le cas pour les entrepôts de matières premières et de produits finis.

Dans ce problème, nous considérons que les équipements de manutention, les machines, les files d'attente et les segments du réseau de manutention ainsi que les nœuds représentant les extrémités de ces segments, qui peuvent représenter des intersections, sont des ressources contraignantes. Cette définition est nécessaire à l'élimination des conflits sur les nœuds.

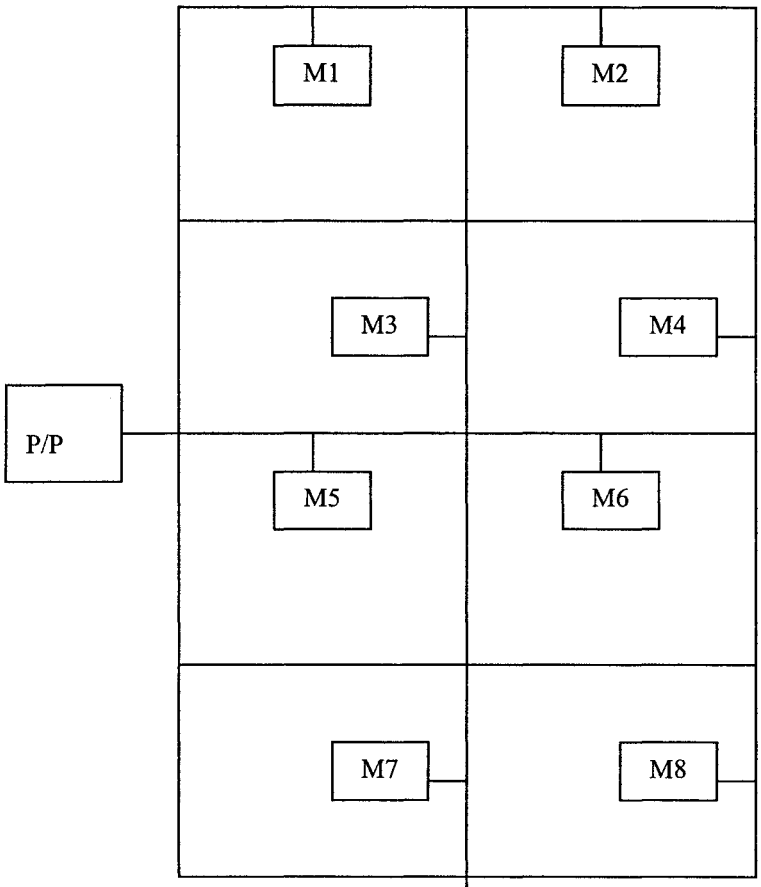
La tâche de manipulation est exécutée par une ressource locale qu'on ne considère pas contraignante. Cette ressource est disponible au niveau de la machine. La tâche de manipulation peut être consommatrice de temps nécessaire pour effectuer le mouvement. Dans ce problème, on considère qu'elle se fait en temps masqué.

Au début du processus de production, un délai de préparation est prévu pour chaque produit. À la fin du processus, un délai de palettisation est prévu pour les différents produits avant de les entreposer dans l'entrepôt de produits finis. Les ressources nécessaires pour effectuer les tâches de préparation et de palettisation ne sont pas contraignantes. Des missions de manutention de l'entrepôt des matières premières et à l'entrepôt des produits finis sont prévus pour chaque travail. Au niveau du traitement, une liste de tâches est à effectuer pour la fabrication du nombre de pièces constituant le travail, la première tâche étant la préparation et la dernière étant la palettisation.

Les files d'attente sont une seule entité physique servant à la fois comme file d'attente d'entrée et file d'attente de sortie. Une capacité totale pour cette file sera donc considérée indépendamment du fait que les pièces entrent ou sortent de la machine. Nous considérons que les arcs sont bidirectionnels. Les chariots dans l'atelier peuvent donc circuler dans les deux sens.

C'est la première fois qu'on présente des modèles riches et complets pour le problème d'ordonnancement dans un atelier multigamme. Nous proposons deux formulations, l'une en programmation linéaire mixte et l'autre en programmation par contraintes. Il

s'agit d'un problème de très grande taille quand on considère des données industrielles. L'amélioration de la performance des modèles demeure un sujet d'étude intéressant.



Légende

E	Entrepôt des matières premières et de produits finis
M*x*	Machine no. *x*
P/P	Poste de préparation et de palettisation

Figure 6.1. Exemple d'implantation

6.2. Formulation en programmation mathématique

Le réseau représentant le problème inclut un ensemble de nœuds N et un ensemble d'arêtes A. Les nœuds représentent les machines, les entrepôts et les postes de préparation et de palettisation ainsi que les extrémités des segments de routes. Les nœuds adjacents *n n'*, dans le réseau, sont reliés par des arêtes de l'ensemble A. Des nœuds virtuels existent aussi dans le réseau. Un nœud virtuel est créé au niveau de chaque poste. Il est relié au poste par un arc virtuel permettant l'accès au poste. Ce même nœud est relié au premier vrai nœud dans le réseau relié au poste, et qui permet la sortie d'un flux du poste. L'utilité de créer ces nœuds est illustrée à la figure 6.2. Ils sont utilisés pour représenter le flux d'un chariot vide dans le cas où la destination d'une tâche de manutention est, en même temps, l'origine de la tâche de manutention suivante sur un chariot. Les arêtes sont orientées tel que présenté dans la figure 6.2. Le nœud *m* représente une machine dans l'atelier, le nœud *n* est le seul nœud adjacent au nœud *m*. Il représente l'entrée à la machine dans le réseau physique. Il est par la suite relié à d'autres nœuds dans le réseau. Le nœud *v* est un nœud virtuel relié par deux arcs virtuels aux nœuds *n* et *m*. Un chariot vide sortant de la machine *m* pour effectuer une tâche de manutention dont l'origine est cette même machine, passe par le nœud *n*, suivi par le nœud *v* pour revenir au nœud *m*.

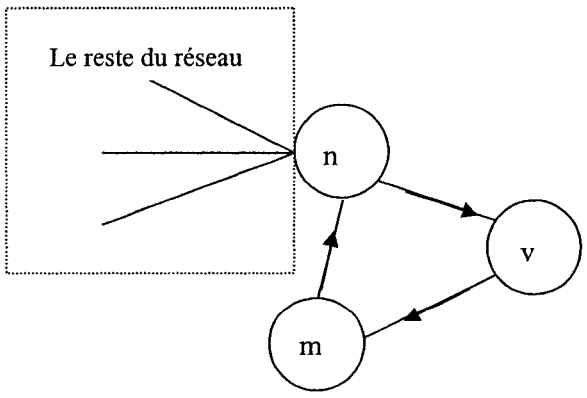


Figure 6.2. Représentation des nœuds virtuels nécessaires au flux des voyages à vide

Des arcs virtuels sont aussi nécessaires pour modéliser les conflits sur les nœuds. Pour ce faire, chaque nœud est représenté par un nombre de nœuds équivalent au degré du noeud. Cela est illustré à la figure 6.3. Des arcs virtuels, de longueur variable selon la tâche de manutention ou le voyage à vide qui va l'occuper, relient ces nœuds. Des contraintes de disjonction seront considérées pour ces arcs. De plus des relations d'antériorité existent entre la visite de ces arcs et les vrais arcs du réseau. Toutefois, dans ce cas on aura besoin d'utiliser des contraintes d'égalité pour les dates de fin et début des tâches successeurs sur les arcs et les nœuds. Cela n'est pas facilement modélisable avec les contraintes formulées par la grande valeur (H). Pour contourner cette difficulté des variables *libres* d'attente sont utilisées. Elles peuvent avoir une valeur positive ou négative. Ces variables auront une longueur égale à *la date de début de l'occupation d'un vrai arc suivant un nœud (arc virtuel représenté dans la figure 6.3)*, **moins** *la date de fin d'occupation d'un vrai arc précédant ce même nœud* pour un travail donné. Les contraintes de disjonction sont écrites pour tous les arcs fictifs représentant un nœud. Si l'un est occupé, tous les autres sont mobilisés jusqu'à la fin de l'attente du travail qui l'occupe. Une tâche peut éventuellement passer par un seul arc fictif parmi ceux représentant un seul nœud. Cette partie n'est pas présentée dans la formulation pour en alléger la lecture. Elle montre aussi la difficulté d'un modèle de programmation linéaire mixte pour ce problème.

Les contraintes de disjonction sont telles qu'au *moment* de la fin de l'occupation d'un nœud par une tâche, une autre tâche peut commencer à occuper le nœud. Nous considérons que l'évitement de ce conflit instantané relève du contrôle optimal et qu'il n'est pas traité par le niveau de l'ordonnancement que représente le modèle. Par contre, au niveau de l'ordonnancement, nous nous assurons qu'aucun nœud ne peut être occupé par une tâche pendant *l'intervalle* d'occupation de ce même nœud par une autre tâche.

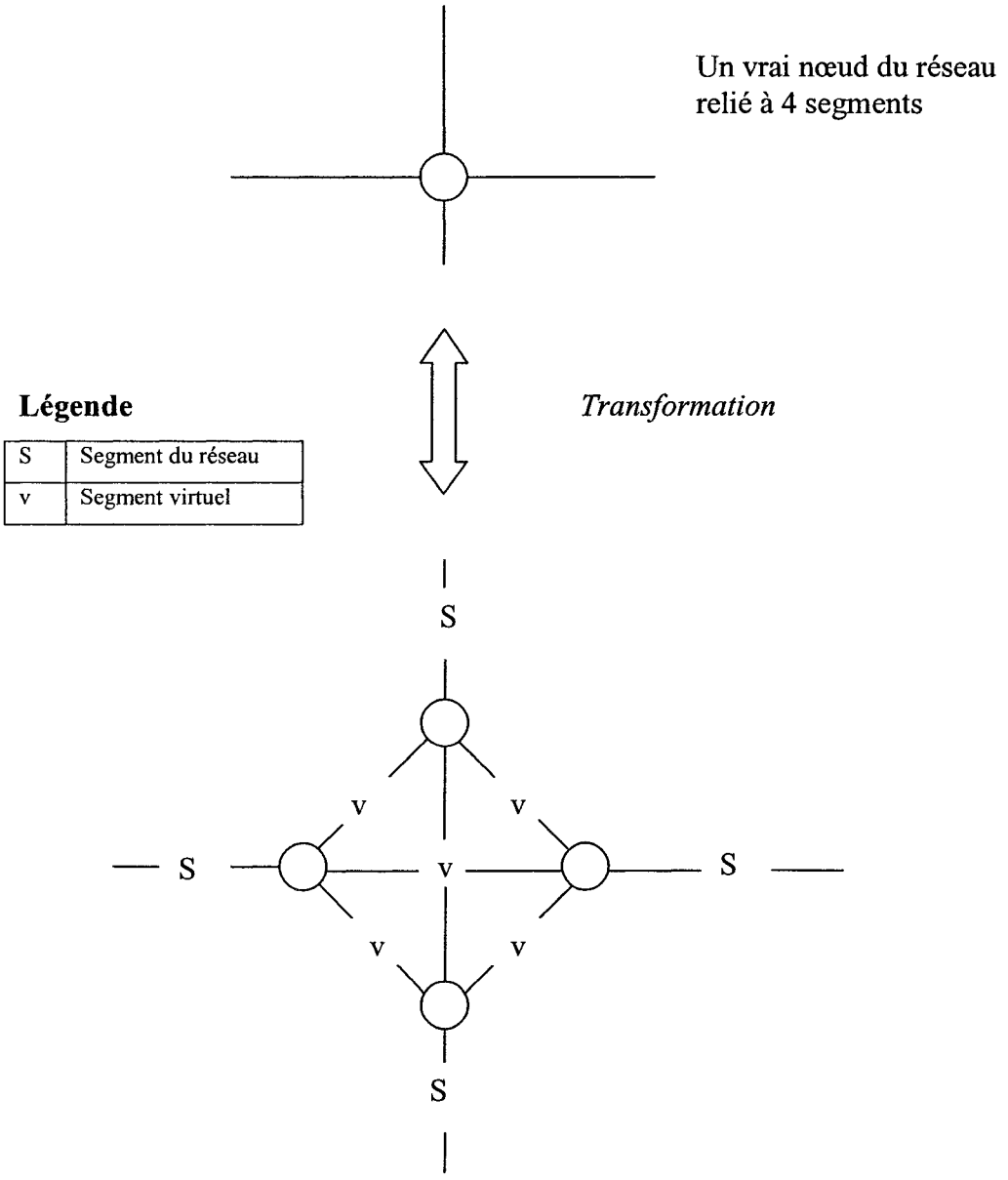


Figure 6.3. Transformation des nœuds en arcs virtuels pour modéliser les conflits

Les files d'attente sont discrétisées en *f* files d'attente d'entrée ou de sortie pour chacune des machines de façon à ce qu'on puisse en modéliser la capacité.

Dans l'ensemble, il existe quatre types de contraintes dans le modèle présenté plus bas :

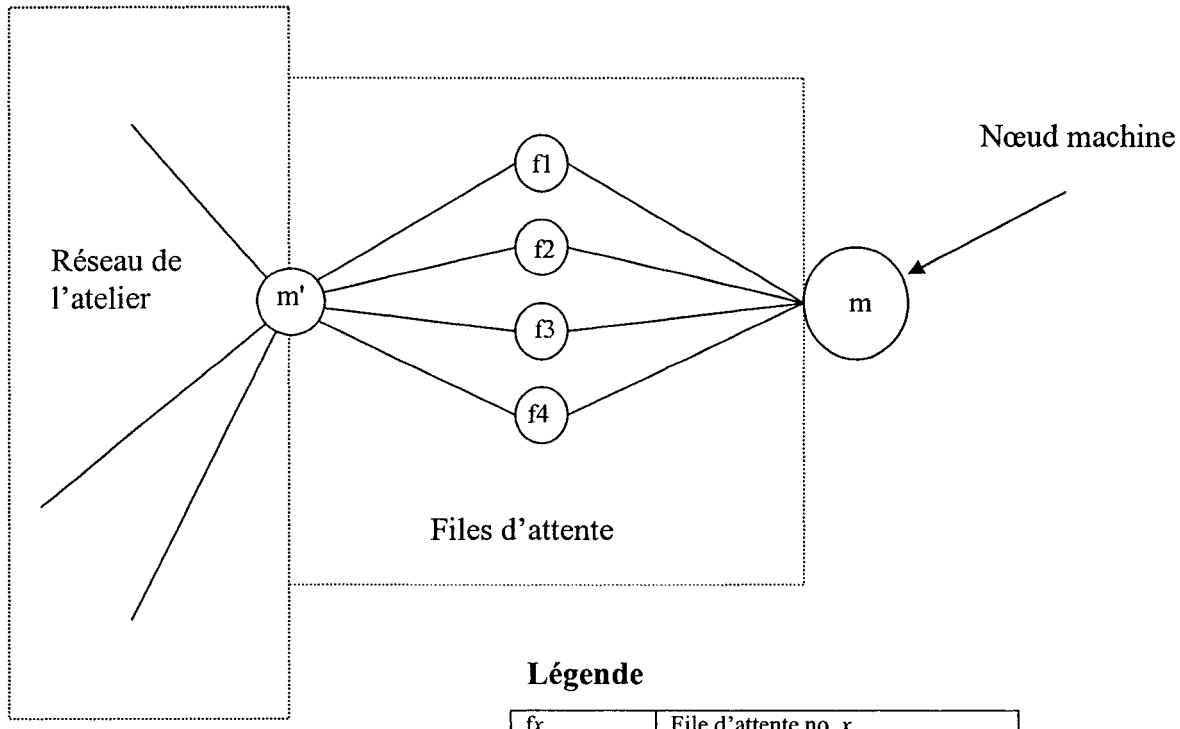
1- ***l'antériorité technologique entre les tâches*** : les tâches sont préparation et palettisation, entreposage, production sur machines, manutention, voyages à vide et séjour dans les files d'attente.
2- ***la capacité des ressources*** : ce sont les contraintes de disjonction écrites pour les machines, les segments et les nœuds du réseau, les files d'attente et les chariots. Ces contraintes nécessitent la déclaration des variables de détermination d'ordre de visite des différentes ressources.
3- ***conservation de flux des travaux et des chariots*** : le flux du chariot se conserve en passant par tous les nœuds sur son chemin jusqu'à ce qu'il ait accompli toutes les tâches qui lui sont affectées et les voyages à vide résultant de cette affectation. De même, les travaux manutentionnés par un chariot sont conservés en passant par les différents nœuds et ce à partir des machines sources jusqu'aux machines destinations des différentes tâches de manutention de tous les travaux.
4- ***affectation aux ressources*** : ces contraintes sont écrites pour les chariots, les files d'attente et les segments qui ne sont pas déterminés a priori pour les différentes tâches.

Dans l'écriture du modèle, ces quatre types de contraintes sont représentés par plusieurs familles de contraintes à cause des différents types de tâches pour lesquels on ne connaît pas nécessairement les décisions d'affectation aux ressources et la durée. Par conséquent, on ne peut pas représenter toutes les tâches de la même façon dans la modélisation.

Il existe un nombre de voyages à vide égal au nombre des tâches de manutention à effectuer dans l'atelier. Chaque voyage à vide se produit juste après l'accomplissement d'une tâche de manutention. Il correspond au déplacement du chariot vide jusqu'à la source de la prochaine tâche de manutention qui lui est affectée. Il est possible que la source de cette tâche soit la même que la destination de la tâche précédente.

Pour les fins d'une meilleure compréhension du modèle, nous illustrons le flux dans le réseau au niveau des postes. Chaque machine possède, selon le modèle, une file d'attente de capacité limitée. Or, autour d'une machine, il se trouve de la place suffisante pour accueillir, par exemple, quatre palettes. Ceci est représenté à la figure 6.4. Le premier nœud relié à la machine *m*, dans le réseau physique, est le nœud *m'*.

Les travaux qui sont à traiter sur une machine passent par une file d'attente avant le traitement. De même, une fois le traitement fini sur une machine, le travail passe par une file d'attente avant que la tâche de manutention suivante ne commence. À partir du nœud *m'*, on peut passer dans le reste du réseau de l'atelier.

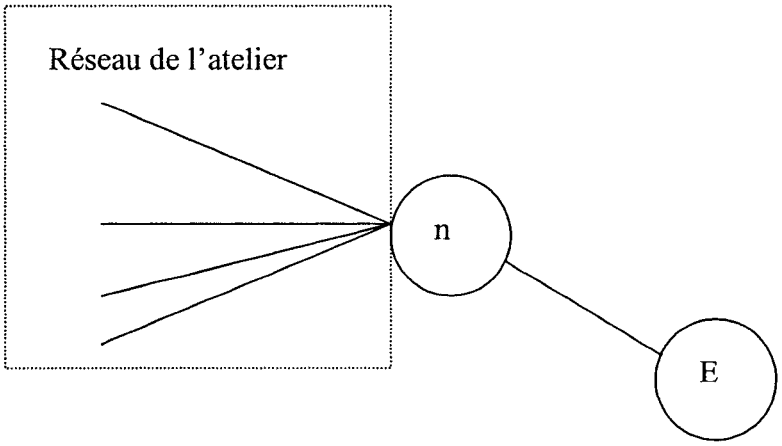


Légende

f*x*	File d'attente no. *x*
m	Machine
m'	Premier nœud du réseau relié à la machine

Figure 6.4. Représentation de flux dans les files d'attente

Les entrepôts possèdent une capacité suffisante et donc ils ne posent pas de problèmes de conflits. Dans l'atelier, les deux entrepôts occupent le même lieu physique. Ils sont modélisés comme des noeuds offrant ou absorbant un flux. Un chariot livrant une palette à l'entrepôt de produits finis peut sortir de l'entrepôt pour effectuer d'autres tâches de manutention. Si toutes les tâches sont accomplies, le chariot reste à l'entrepôt. Le nœud représentant l'entrepôt de produits finis est relié à un premier nœud *n* qui représente physiquement l'entrée à l'entrepôt. Par la suite, ce même nœud *n* est relié au reste du réseau. Le nœud *n* fait partie du réseau de manutention et, donc, est utilisé par les chariots dans la manutention des palettes. On considère par une simple approximation que l'arc « n,E » est de longueur zéro. Les entrepôts sont représentés sur la figure suivante.



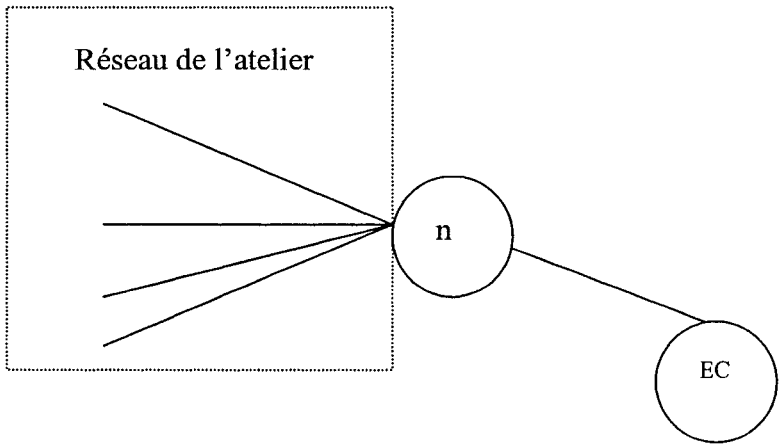
Légende

E	Entrepôt des matières premières et des produits finis
n	Premier nœud dans le réseau relié à l'entrepôt

Figure 6.5. Représentation de l'entrepôt dans le réseau de l'atelier

L'entrepôt de produits encours n'est pas représenté dans le modèle. Il est facilement représentable par un nœud *EC* dans le réseau relié à l'atelier par un arc « *EC,n* » de

longueur zéro à un nœud *n* dans le réseau comme présenté à la figure 6.6. Le nœud *EC* ne pose pas de problèmes de conflits. Un chariot peut alors trouver son chemin à cet entrepôt, y rester pendant un certain temps avant de revenir à l'atelier. Comme politique d'opération, on considère que le chariot emprunte l'arc relié au nœud *EC* en gardant sa charge. Il peut passer un certain temps avec sa charge dans l'entrepôt des produits en cours. Il ne se décharge pas pour aller prendre d'autres charges, les livrer et par la suite retourner prendre la charge déposée. En d'autres termes, la tâche de manutention (identifiée par le chariot auquel elle est affectée) n'est pas préemptive. Cependant, à cause de l'immobilisation possible du chariot le délai total de la tâche peut être prolongé.



Légende

EC	Entrepôt des produits en cours
n	Premier nœud dans le réseau relié à l'entrepôt

Figure 6.6. Représentation d'un entrepôt de produits en cours

Parmi les ressources du problème, il existe les segments de route qui représentent deux arcs opposés formés à partir des nœuds constituant les extrémités des segments. Ces

ressources accueillent un flux qui doit circuler sans conflits. Aucune attente n'est permise sur les arcs. L'attente se fait sur les nœuds représentés par les arcs virtuels.

Les machines sont des ressources disjonctives qui peuvent poser des problèmes de conflits. Elles sont représentées comme des nœuds sources et destinations de flux, le flux étant les tâches de manutention. Les chariots sont des ressources mobiles. Ils ne sont pas représentés sur le réseau. Ils sont plutôt affectés aux travaux, et ils circulent avec ou sans ces derniers dans le réseau. Initialement, les chariots se trouvent à l'entrepôt des matières premières, là où une première mission de manutention des matières premières est à accomplir.

Pour bien représenter les contraintes de conservation de flux, et de disjonction des segments de route, il faut introduire l'ensemble de voyages à vide. Nous n'avons pas besoin d'énumérer tous les couples de tâches de manutention qui peuvent se succéder sur un chariot. Les contraintes de conservation de flux assurant la connectivité, il suffit juste de définir le voyage à vide en fonction de la tâche de manutention précédente effectuée par le chariot. Nous avons défini des tâches de type *A***acro** où *A* signifie *après* et **acro** est l'**acronyme** déjà choisi pour une tâche de manutention.

Dans la modélisation, pour bien tracer l'occupation des ressources, on considère que l'utilisation d'une ressource par un travail constitue une tâche. Ceci correspond aux étapes figurant au diagramme de déroulement matières présenté au chapitre 3. Les différentes tâches, pour chaque travail, sont listées ci-après :

- départ de l'entrepôt des matières premières (tâche de durée zéro)

Tâche de manutention (visite d'un ensemble d'arcs et d'un ensemble de nœuds)

- séjour à une file d'attente de la première machine dans la gamme
- visite de la première machine dans la gamme
- séjour à une file d'attente de la première machine dans la gamme

Tâche de manutention (visite d'un ensemble d'arcs et d'un ensemble de nœuds)

- séjour à une file d'attente de la deuxième machine dans la gamme
- visite de la deuxième machine dans la gamme
- séjour à une file d'attente de la deuxième machine dans la gamme

Tâche de manutention (visite d'un ensemble d'arcs et d'un ensemble de nœuds)

Ainsi de suite

- séjour à une file d'attente de la dernière machine dans la gamme
- visite de la dernière machine dans la gamme
- séjour à une file d'attente de la dernière machine dans la gamme

Tâche de manutention (visite d'un ensemble d'arcs et d'un ensemble de nœuds)

- arrivée à l'entrepôt des produits finis (tâche de durée zéro)

Ces différentes tâches sont utilisées pour établir des relations d'antériorité technologique présentées dans le modèle.

6.2.1. Notation

O	Ensemble de toutes les tâches, y compris les voyages à vide, les tâches correspondant au séjour dans les files d'attente, les tâches de préparation et de palettisation et les tâches fictives de début et fin de trajet des chariots
O^{FA}	Ensemble des tâches correspondant au séjour dans une file d'attente $O^{FA} \subset O$
O^{M}	Ensemble des tâches de manutention $O^{M} \subset O$
O^{P}	Ensemble des tâches de production $O^{P} \subset O$
O^{F}	Ensemble des premières tâches de tous les travaux $O^{F} \subset O$
O^{L}	Ensemble des dernières tâches de tous les travaux $O^{L} \subset O$
O^{V}	Ensemble des voyages à vide $O^{V} \subset O$
O^{MV}	Ensemble des tâches de manutention et des voyages à vide $O^{MV} \subset O$

O^{PR}	Ensemble des tâches de préparation $O^{PR} \subset O$
O^{PL}	Ensemble des tâches de palettisation $O^{PL} \subset O$
O^{DF}	Ensemble des tâches fictives de début $O^{DF} \subset O$
O^{FF}	Ensemble des tâches fictives de fin $O^{FF} \subset O$
N	Ensemble de tous les noeuds du réseau $n \in N$
A	Ensemble des arcs du réseau
E	Ensemble des nœuds représentant les extrémités des arcs $E \subset N$
SG	Ensemble des segments du réseau, chaque segment étant l'entité physique liant deux nœuds adjacents par deux arcs dans deux directions opposés $s \in SG$
R(s)	Ensembles des paires des arcs qui sont représentés par un segment du réseau, $R \subset A, s \in SG$
M	Ensemble des nœuds machines $m \in M$
V	Ensemble des chariots $v \in V$
FA	Ensemble des files d'attente $f \in FA$
W	Ensemble des nœuds entrepôts $w \in N$
PP	Ensemble des nœuds postes de préparation et palettisation $p \in N$
T	Ensemble des nœuds postes $t \in N$ $\quad T=M \cup PP \cup E$
O^m	Ensemble des tâches sollicitant la machine m
nn(j)	Tâche de manutention ne pouvant pas succéder au voyage à vide *j*
O^m	Ensemble des tâches sollicitant la machine m
OF(p)	Ensemble des tâches offertes par le poste p, $OF \subset O$, $p \in P$
OD(p)	Ensemble des tâches demandées par le poste p, $OD \subset O$, $p \in P$
F(j)	Ensemble des files d'attente pouvant accueillir la tâche *j* , $F \subset FA$, $j \in O^{FA}$
Φ	Ensemble des paires de tâches possédant une relation d'antériorité
H	Une grande valeur
p_j	Temps de traitement de la tâche *j*, $j \in O^P \cup O^{PR} \cup O^{PL}$
d_s	Temps de parcours du segment $s \in SG$

r_j Paramètre représentant le temps requis pour régler une machine pour traiter une tâche *j*

Ω_{ijv} Variable ayant la valeur 1 si le voyage à vide *i* est suivi d'une tâche de manutention *j* sur un chariot $v \in V$, $i \in O^V \cup O^{DF}$, $j \in O^M \cup O^{FF}$, zéro sinon

$X_{nn'jc}$ Variable ayant la valeur 1 si le chariot *c* assure la manutention de la tâche de manutention *j* sur un arc formé par les nœuds nn'

Y^s_{ji} Variable ayant la valeur 1 si la tâche de manutention ou le voyage à vide *j* précède la tâche de manutention ou le voyage à vide *i* sur le segment *s*, zéro sinon

S_{ji} Variable ayant la valeur 1 si la tâche de production *j* précède la tâche de production *i* sur leur machine sollicitée, zéro sinon

D_j Variable indiquant la date du début du traitement de la tâche *j* de production, de préparation ou de palettisation ou de sortie/arrivée à l'entrepôt

C_{max} Variable indiquant la date de fin de l'horaire

D^s_j Variable indiquant la date de début de l'occupation du segment *s* par la tâche de manutention ou le voyage à vide *j*

At_j Attente de la tâche de production *j* sur une machine

DF_j Variable indiquant la date de début de séjour de la tâche *j* correspondant au séjour dans une file d'attente

DS_j Variable indiquant la durée de séjour de la tâche *j* correspondant au séjour dans une file d'attente admissible *f*

Bf_{jf} Variable ayant la valeur 1 si la tâche *j* correspondant au séjour dans une file d'attente occupe la file d'attente admissible *f*

B_{ji} Variable ayant la valeur 1 si la tâche *j* correspondant au séjour dans une file d'attente précède la tâche *i* correspondant au séjour dans une file d'attente, zéro sinon

6.2.2. Modèle

La formulation du problème en programme linéaire mixte est présentée ci-après. L'objectif du modèle est de minimiser la date de fin de l'horaire. Les contraintes sont suivies d'explications.

Objectif

$\text{Min } C_{max}$ (1)

Minimiser la date de fin de l'horaire.

Relations d'antériorité

$C_{max} \geq D_j \qquad \forall j \in O^L$ (2)

La plus grande date de fin est plus grande ou égale à la date de début des tâches de fin de durée zéro qui correspondent à l'arrivée à l'entrepôt de produits finis.

Conservation de flux pour les chariots et offre pour les voyages à vide

$\sum_{nn' \in A} X_{nn'tc} = \sum_{nn' \in A} X_{n'nvc} \qquad \forall v \in O^V, \forall c \in V, \; t = nn(v) | \; t \in OD(n')$ (3)

Le voyage à vide suivant une tâche de manutention *t* a comme origine le nœud destination de la tâche de manutention *t*, et est aussi manutentionné par le même chariot.

Affectation des tâches de manutention aux chariots

$\Omega_{jtc} <= \sum_{nn' \in A} X_{nn'tc} \qquad \forall j \in O^V, \forall c \in V, \forall t \in O^M \cap OF(n)$ (4)

Si une tâche de manutention *t* suit le voyage à vide *j,* sur un chariot *c,* ce même chariot est utilisé pour effectuer la tâche de manutention.

Affectation des voyages à vide aux chariots et détermination de leur destination

$\Omega_{jtv} <= \sum_{nn' \in A} X_{nn'jv} \qquad \forall j \in O^V, \forall v \in V, \forall t \in O^M \cap OF(n')$ (5)

Si une tâche de manutention *t* suit le voyage à vide *j* sur un chariot *v*, ce même chariot est utilisé pour effectuer le voyage à vide. En même temps, cette contrainte détermine que le nœud destination du voyage à vide *j* est le nœud source de la tâche de manutention suivante *t*.

Détermination d'ordre d'affectation des tâches de manutention aux chariots

$$\sum_{v \in V} \sum_{i \in O^V \cup O^{DF}} \Omega_{ijv} = 1 \qquad \forall j \in O^M \cup O^{FF} \tag{6}$$

Un seul voyage à vide ou une tâche fictive de début, peut précéder une tâche de manutention ou une tâche fictive de fin sur un chariot.

Détermination d'ordre d'affectation des tâches de manutention aux chariots

$$\sum_{v \in V} \sum_{j \in O^M \cup O^{FF}} \Omega_{ijv} = 1 \qquad \forall i \in O^V \cup O^{DF} \tag{7}$$

Une seule tâche de manutention ou une tâche de fictive de fin peut suivre un voyage à vide ou une tâche fictive de début sur un chariot.

Détermination d'ordre d'affectation des tâches de manutention aux chariots

$$\Omega_{ijv} = 0 \qquad \forall i \in O^V \cup O^{DF}, j = nn(i), \forall v \in V \tag{8}$$

Un voyage à vide *v* ne peut pas être suivi d'une tâche de manutention inadmissible *t*. Une tâche de manutention inadmissible est celle qui doit être faite juste avant le voyage à vide. Un voyage à vide *v* est créé après l'accomplissement d'une tâche de manutention *t*.

Détermination d'ordre d'affectation des tâches de manutention aux chariots

$$\Omega_{ijv} = 0 \qquad \forall i \in O^{DF}, j \in O^{FF}, \forall v \in V \tag{9}$$

Une tâche de début fictive ne peut pas être suivie d'une tâche de fin fictive quel que soit le chariot.

Satisfaction de la demande pour les tâches de manutention

$$\sum_{nn' \in A} X_{nn'jv} = 1 \qquad \forall j \in O^M \cap OD(n'), \ v \in V \tag{10}$$

Une tâche de manutention *j* doit arriver à sa machine destination *n'* par un chariot. Dans ce cas, la variable $X_{nn'jv}$ correspondant à l'occupation de l'arc *nn'* par le chariot *v* manutentionnant *j* est égale à 1.

Offre pour les tâches de manutention

$$\sum_{nn' \in A} X_{nn'jv} = 1 \qquad \forall j \in O^{M} \mid j \in OF(n'),\ v \in V \tag{11}$$

Une tâche de manutention *j* doit partir de sa machine source *n* par un chariot *v*. Dans ce cas, la variable $X_{nn'jv}$ correspondant à l'occupation de l'arc *nn'* par le chariot *v* manutentionnant *j* est égale à 1.

Conservation de flux pour les tâches de manutention

$$\sum_{nn' \in A} X_{nn'jv} - \sum_{n'n \in A} X_{n'njv} = 0 \qquad \forall n' \in E, j \in O^{MV}, \forall v \in V \tag{12}$$

Un chariot *v* effectuant une tâche de manutention ou un voyage à vide *j* en passant par un nœud dans le réseau qui n'est pas sa source ni sa destination ne doit pas être absorbé dans ce nœud.

Affectation aux files d'attente

$$\sum_{f \in F(j) A} Bf_{jf} = 1 \qquad \forall j \in O^{FA} \tag{13}$$

Un travail peut occuper seulement une des files d'attente admissibles *f* à l'entrée ou à la sortie des différentes machines.

Détermination d'ordre de visite des segments par les tâches de manutention et les voyages à vide

$$Y^{s}_{ji} + Y^{s}_{ij} = 1 \qquad \forall j \in O^{MV}, \forall i \in O^{MV}, \forall s \in SG \mid i \neq j \tag{14}$$

Contraintes définissant l'ordre de visite des tâches de manutention et des voyages à vide aux segments du réseau de manutention. Y^{s}_{ji} est égale à 1 si la tâche de manutention ou le voyage à vide *i* suit la tâche de manutention ou le voyage à vide *j* sur le segment *s*.

Détermination d'ordre de visite des files d'attente

$B_{ji} + B_{ij} = 1 \qquad \forall F(j), \forall F(i), \forall j \in O^{FA}, \forall i \in O^{FA} \mid i \neq j \qquad (15)$

Contrainte définissant l'ordre de visite des travaux aux files d'attente à l'entrée ou à la sortie d'une machine. B_{ji} est égale à 1 si la tâche *i* succède à la tâche *j* dans une file d'attente.

Détermination d'ordre de visite des machines

$S_{ji} + S_{ij} = 1 \qquad \forall m \in M, \forall j \in O^{m}, \forall i \in O^{m} \mid i \neq j \qquad (16)$

Contrainte définissant l'ordre de visite de travaux aux machines. S_{ji} est égale à 1 si la tâche *i* succède à la tâche *j* sur une machine.

Relations d'antériorité entre les tâches de séjour dans une file d'attente et les tâches de production

$D_i = DF_j + DS_j \qquad \forall i \in O^{P}, \forall j \in O^{FA} \mid (j,i) \in \Phi \qquad (17)$

La date de début de traitement d'une tâche *i* sur une machine est égale à la date de début de séjour plus la durée de séjour dans la file d'attente à l'entrée d'une machine.

Relations d'antériorité entre les tâches de séjour dans une file d'attente et les tâches de production

$DF_j = D_i + p_i + At_i \qquad \forall i \in O^{P}, \forall j \in O^{FA} \mid (i,j) \in \Phi \qquad (18)$

La date de début de séjour d'une tâche *j* dans la file d'attente à la sortie d'une machine est égale à la date de fin de traitement de la tâche *i* sur la machine plus une attente éventuelle sur la même machine.

Relations d'antériorité déterminant la date de fin d'un travail

$D_j = D^{s}_{i} + \sum_{v \in V} \sum_{n'n \in A} d_s X_{n'niv}$

$\forall i \in O^{M}, \forall j \in O^{L}, s \in SG \mid (i,j) \in \Phi, n'n \in R(s), n \in W \qquad (19)$

La date de fin d'un travail est égale à la date de début d'occupation d'un arc menant à l'entrepôt des produits finis plus la durée de cet arc.

Relations d'antériorité entre les tâches de séjour dans une file d'attente et les tâches de manutention

$$Df_j = D^s_i + \sum_{v \in V} \sum_{n'n \in A} d_s X_{n'niv}$$

$$\forall j \in O^{FA}, \forall i \in O^M, s \in SG \mid (i,j) \in \Phi, n'n \in R(s), i \in OD(n) \qquad (20)$$

La date de début de séjour de la tâche *j* dans la file d'attente à l'entrée de la machine correspondante est égale à la date de début d'occupation d'un arc précédent la file d'attente plus le délai requis pour parcourir cet arc par un chariot *v* effectuant la tâche de manutention prédécesseur *i*.

Relations d'antériorité entre les tâches de séjour dans une file d'attente et les tâches de manutention

$$D^s_j = Df_i + DS_i$$

$$\forall i \in O^{FA}, \forall j \in O^M, \forall s \in SG \mid (i,j) \in \Phi, n,n' \in R(s), i \in OF(n) \qquad (21)$$

La date de début d'occupation du premier arc sur le trajet d'une tâche de manutention *j*, est égale à la date de fin de la tâche précédente *i* correspondant au séjour dans une file d'attente à la sortie d'une machine.

Disjonction des files d'attente

$$Df_j \geq Df_i + DS_i + (B_{ij} + Bf_{if} + Bf_{jf} - 3) * H$$

$$\forall i \in O^{FA}, \forall j \in O^{FA}, \forall f \in FA \mid f \in F(i), f \in F(j), i \neq j \qquad (22)$$

Contraintes d'évitement des conflits dans les files d'attente. Si deux travaux *i* et *j* sont affectés successivement à une file d'attente, c'est-à-dire $B_{ij} = 1$, $Bf_{if} = 1$ et $Bf_{jf} = 1$, il faut que le premier travail soit déplacé avant que le deuxième occupe la file.

Respect du temps de réglage et disjonction des machines

$$D_j \geq D_i - H * (1 - S_{ij}) + At_i + p_i + r_j \qquad \forall m \in M, \forall i \in O^m, \forall j \in O^m \mid i \neq j \qquad (23)$$

La date de début de traitement d'une tâche de production *j* sur une machine est plus grande ou égale à la date de fin de la tâche *i* qui la précède sur la machine plus l'attente

éventuelle de cette tâche *i* sur la machine plus le temps de réglage requis pour préparer la machine pour traiter la tâche *j*

Relations d'antériorité entre la visite des arcs consécutifs pour les tâches de manutention et les voyages à vide

$$D^d_j \geq D^s_j + d_s - H * (2-\sum_{v\in V} X_{n'njv} - \sum_{v\in V} X_{nn''jv})$$

$$\forall j \in O^{MV}, \forall s \in SG, \forall d \in SG \mid s \neq d, \forall n'n \in R(s), nn'' \in R(d) \qquad (24)$$

La date de début d'occupation d'un arc sur le trajet d'une tâche de manutention ou un voyage à vide est plus grande ou égale à la date de début de l'occupation de l'arc précédent plus la durée de ce dernier.

Relations d'antériorité entre les tâches de préparation/palettisation et les tâches de manutention

$$D_i = D^s_j + d_s$$

$$\forall i \in O^{PL} \cup O^{PR}, \forall j \in O^M, \forall s \in SG, \forall nn' \in R(s) \mid (j,i) \in \Phi, n' \in PP \qquad (25)$$

La date de début d'une tâche de palettisation ou et de préparation est égale à la date de début de l'occupation du dernier arc de la tâche de manutention précédente plus le délai requis pour parcourir cet arc.

$$D^s_j = D_i + p_i$$

$$\forall i \in O^{PL} \cup O^{PR}, \forall j \in O^M, \forall s \in SG, \forall nn' \in R(s) \mid (i,j) \in \Phi, n \in PP \qquad (26)$$

La date de début de l'occupation du premier arc sur le trajet de la tâche de manutention *j* suivant une tâche de préparation/palettisation *i* est égale à la date de début de cette dernière plus sa durée.

Disjonction des chariots et des segments

$$D^s_j \geq D^s_i + d_s - H * (3-\sum_{v\in V} \sum_{n'n\in R(s)} X_{n'njv} - \sum_{v\in V} \sum_{n'n \in R(s)} X_{n'niv} - Y^s_{ij})$$

$$\forall j \in O^{MV}, \forall s \in SG \qquad (27)$$

Si une tâche de manutention ou un voyage à vide *j* suit une tâche de manutention ou un voyage à vide *i* sur un segment *s*, et que les deux tâches ont à occuper ce même segment, il faut que la tâche *i* finisse avant que *j* ne commence.

Relations d'antériorité pour les voyages à vide les reliant avec les missions de manutention prédécesseurs

$$D^d_i \geq D^s_j + \sum_{v \in V} \sum_{n'n \in R(s)} d_s X_{n'njv}$$

$$\forall i \in O^v, \forall j \in nn(i), s \in SG, d \in SG \mid j \in OD(n), nn' \in R(d), n'n \in R(s) \qquad (28)$$

La date de début d'occupation du premier arc, sortant d'une machine ou un poste, sur le trajet d'un voyage à vide *i* suivant une tâche de manutention *j* est plus grande ou égale à la date de fin de l'occupation du dernier arc sur le trajet de la tâche de manutention *j*.

Relations d'antériorité pour les voyages à vide les reliant avec les missions de manutention successeurs

$$D^s_j \geq D^d_i + d_s - H^* (3 - \sum_{v \in V} \Omega_{ijv} - \sum_{v \in V} X_{n'njv} - \sum_{v \in V} X_{nn'iv})$$

$$\forall i \in O^v, \forall j \in O^M, s \in SG, d \in SG \mid j \in OD(n), nn' \in R(d), n'n \in R(s) \qquad (29)$$

Si une tâche de manutention *j* suit un voyage à vide *i* sur un chariot, cela implique que la date de début de l'occupation du premier arc sur le trajet de la tâche de manutention soit plus grande ou égale à la date de début de l'occupation du dernier arc sur le trajet du voyage à vide *i*.

Détermination des chemins pour les chariots

$$\Omega_{ijv} = 0 \qquad \forall i \in O^v, j = nn(i), \forall v \in V \qquad (30)$$

Un voyage à vide *i* ne peut pas être suivi par une tâche de manutention inadmissible.

Détermination des chemins pour les chariots

$$\Omega_{ijv} = 0$$

$$\forall i \in O^{DF}, \forall j \in O^{FF}, \forall v \in V \qquad (31)$$

Une tâche fictive de fin ne peut pas suivre directement une tâche fictive de début.

Connectivité des chemins des chariots

$$\sum_{j \in O^M \cup O^{FF}} \Omega_{ijv} - \sum_{k \in O^V \cup O^{DF}} \Omega_{klv} = 0$$

$$\forall i \in O^v, \forall k \in O^v \forall l \in O^M, \forall v \in V | l = nn(i) \quad (32)$$

Il faut que la tâche de manutention *l*, qui ne peut pas suivre le voyage à vide *i*, soit affectée au même chariot *v*.

Le modèle présenté est très riche et détaillé, ce qui augmente considérablement sa taille. La programmation mathématique oblige à créer beaucoup de variables et à écrire beaucoup de contraintes pour modéliser la détermination de l'ordre de visite des ressources et la disjonction de ces dernières. La politique de pilotage de l'atelier détermine aussi d'autres subtilités qui affectent la modélisation de l'attente des chariots voyageant à vide. Dans le cas où le chariot voyage à vide après l'accomplissement d'une tâche de manutention dont la destination est un poste à capacité suffisante, c'est-à-dire préparation/palettisation ou entrepôts, l'attente ne pose pas de problème. C'est également le cas si la destination d'un voyage à vide est un poste à capacité suffisante.

Dans le cas où le voyage à vide arrive à ou part d'un poste à capacité limité, c'est-à-dire une machine, il faut déterminer si oui ou non le chariot a le droit de rester au niveau de la machine. Deuxièmement, dans le cas où l'attente est permise, il faut préciser la relation d'antériorité reliant cette attente avec une tâche. Troisièmement, il faut affecter le chariot vide à une file d'attente en respectant les contraintes de disjonction.

L'expérimentation de ce modèle est discutée à la section 6.4. Le modèle présenté dans la section suivante est beaucoup plus compact grâce à la possibilité d'utilisation des relations d'implication en programmation par contraintes. Moins de variables et de contraintes sont utilisées pour formuler la même réalité.

6.3. Formulation en programmation par contraintes

Le même problème, présenté dans la section précédente, a été modélisé en utilisant la programmation par contraintes. Les ressources déclarées dans le modèle sont les chariots, les machines, les segments, les nœuds et les files d'attente.

Les relations d'antériorité impliquent qu'à la date de fin du traitement sur une ressource commence le traitement sur la ressource suivante, et ce pour ne pas générer des conflits sur les ressources contraignantes. Pour les ressources à capacité suffisante tel que le poste de préparation et de palettisation cela n'est pas nécessaire.

Pour représenter les contraintes de conservation de flux, nous avons choisi de définir un ensemble de 7 nœuds/8 segments à visiter pour chaque tâche de manutention ou voyage à vide. Le choix de ce nombre de segments et de nœuds peut sembler limitatif, mais on garde cette définition pour les fins de la présentation du modèle. Le nombre de nœuds/segments doit être déterminé en fonction des données de sorte à ne pas éliminer des solutions potentiellement optimales. Une tâche commence par la visite d'un arc sortant de son nœud source et finit par la visite d'un arc entrant dans son nœud destination. La visite de chacune de ces ressources constitue une sous-tâche. Les contraintes logiques utilisées consistent à satisfaire une demande en ressources par des tâches qui représentent soit un séjour à un nœud soit un parcours d'un arc. La condition suivante doit s'appliquer : deux nœuds choisis pour satisfaire la demande de deux tâches consécutives sur les nœuds doivent former un arc. Dans ce cas là, les deux nœuds et l'arc les reliant sont mobilisés par les trois tâches correspondantes. On impose que le premier nœud dans un trajet soit la source de la tâche de manutention ou le voyage à vide et que le dernier nœud dans un trajet soit la destination de la tâche de manutention ou du voyage à vide tel que déterminé par l'affectation aux chariots.

Plusieurs sous-ensembles existent dans l'ensemble des tâches. Chaque tâche de manutention est décomposée en quinze tâches (correspondant à la visite des 7 nœuds/ 8 segments) qui forment un sous-ensemble ordonné. Ces quinze tâches correspondant au trajet qu'effectue une tâche de manutention allant de son nœud source à son nœud destination, et passant par plusieurs segments et nœuds du réseau. Pour représenter ce passage, nous modélisons la conservation de flux par des relations d'implication de façon très concise et élégante. Nous avons, aussi, des contraintes d'antériorité entre chaque deux éléments successeurs de cet ensemble ordonné. Il ne suffit pas d'écrire des contraintes du type « $\geq$ », car il faut en tout temps repérer les chariots dans le réseau pour éviter les conflits. Des relations d'antériorité de ce type sont écrites.

Pour deux tâches successeurs t, q , dans l'ensemble ordonné :
a [t].end = a [q].start

Dans le cas où le passage sur un arc (a[t]) est prédécesseur au passage sur un nœud (a[q]), nous avons
a [t].start + longueur de l'arc = a [q].start

Dans le cas où le passage sur un nœud (a[t]) est prédécesseur au passage sur un arc (a[q]), nous avons
a [t].start + Attente [t] = a [q].start

Attente [t] est une variable car on ne sait pas a priori combien de temps passera le chariot sur un nœud.

Un autre ensemble de tâches est aussi créé incluant toutes les tâches distinctes successeurs. Par exemple, si une tâche de manutention est successeur d'une autre tâche, il faut juste considérer une relation d'antériorité entre la tâche en question et la première tâche dans l'ensemble incluant les tâches correspondantes au trajet de la tâche de

manutention. Les autres contraintes d'antériorité inter-tâche sont assurées par des familles de contraintes écrites pour chaque ensemble ordonné de sous-tâches représentant un trajet. D'autres relations d'antériorité sont imposées dans l'éventualité où deux tâches de manutention se suivent sur un chariot. Elles sont entre la première tâche de manutention et le voyage à vide résultant de l'affectation, et entre ce dernier et la tâche de manutention suivante. Au même titre que les tâches de manutention, chaque voyage à vide est composé de plusieurs sous-tâches pour lesquelles les relations d'antériorité doivent être respectées.

6.3.1 Notation

Noeuds	Ensemble des nœuds du réseau
Tasks	Ensemble de toutes les tâches à accomplir, les tâches fictives de détermination de chemins de chariots « Debut*x* et Fin*x* »et une tâche fictive de fin « makespan »
Chariots	Ensemble des chariots
Files	Ensemble des files d'attente
TasksInFile	Ensemble de toutes les tâches correspondant à un séjour dans une file d'attente
TasksLast	Ensemble des dernières tâches à traiter
FILES [t]	Ensemble des files d'attente pouvant accueillir la tâche $t \in TasksInFile$
Postes	Ensemble des postes de travail $Postes \subset Nœuds$
Tasksman	Ensemble des tâches de manutention $Tasksman \subset Tasks$
VAV	Ensemble des voyages à vide $VAV \subset Tasks$, définis en fonction de la tâche de manutention précédente
AllTasks [t]	Ensemble ordonné des quinze tâches (t_i précédent t_j) constituant une seule tâche de manutention ou un voyage à vide $t \in Tasksman \cup VAV$

Dernier [t]	Dernier élément de l'ensemble de quinze tâches constituant une seule tâche de manutention ou un voyage à vide $t \subset Tasksman \cup VAV$
Premier [t]	Dernier élément de l'ensemble de quinze tâches constituant une seule tâche de manutention ou un voyage à vide $t \subset Tasksman \cup VAV$
ArcTask	Ensemble de toutes les sous-tâches de manutention ou voyages à vide qui empruntent un arc dans le réseau
NodeTask	Ensemble de toutes les sous-tâches de manutention ou voyages à vide qui passent par un nœud dans le réseau
Machines	Ensemble des machines $Machines \subset Nœuds$
Tasksmach	Ensemble des tâches à traiter sur les machines, les postes de préparation et de palettisation et la sortie/arrivée à l'entrepôt $Tasksmach \subset Tasks$
MANorigine	Ensemble des tâches fictives de début $MANorigine \subset Tasks$ nécessaires pour tracer le chemin d'un chariot
MANfictif	Ensemble des tâches fictives de fin $MANfictif \subset Tasks$ nécessaires pour tracer le chemin d'un chariot Tasks
TasksmanFIC	Ensemble de tâches qui peuvent suivre une autre sur un chariot $TasksmanFIC = Tasksman \cup MANfictif$
TasksmanSour	Ensemble de tâches qui peuvent précéder une autre sur un chariot $TasksmanSour = Tasksman \cup MANorigine$
TasksmanSourFIC	Ensemble de tâches de manutention et les tâches fictives de début et de fin $TasksmanSourFIC = Tasksman \cup MANfictif \cup MANorigine$
MANvav	Ensemble de paires de tâches de manutention et de voyages à vide successeurs

Arcs	Ensemble des arcs du réseau, chacun reliant deux nœuds du réseau plus un arc fictif $a \in Arcs$
SEG [a]	Segment du réseau formé de deux arcs entre deux nœuds dans deux directions différentes dont l'un est l'arc a
ARCS [n]	Ensemble des arcs dont l'origine est le nœud *n*
Prec	Ensemble des tâches distinctes possédant une relation d'antériorité
destination [t]	Donné de type nœud indiquant le nœud destination pour une tâche de manutention $t \in Tasksman$
ORIGIN [t]	Donnée de type nœud indiquant le nœud origine pour une tâche de manutention $t \in Tasksman$
resourcem [t]	Donnée de type machine indiquant la machine sur laquelle une tâche de production sera traitée $t \in Tasksmach$
Duree [t]	Temps de traitement de la tâche de production, de préparation, de palettisation et d'arrivée/départ de l'entrepôt $t \in Tasksmach$
Longueur[a]	Temps requis pour parcourir un arc $a \in Arcs$
ARCchoisi[t]	Variable de type arc indiquant quel arc sera utilisé pour accomplir une partie d'une tâche de manutention ou un voyage à vide $t \in ArcsTask$
followsOnChariot[t]	Variable de type « Tasks » indiquant quelle tâche suit une tâche $t \in TasksmanSour$ sur un chariot
carries [t]	Variable de type « Chariots » indiquant à quel chariot la tâche $t \in Tasksman$ sera affectée
FileChoisie[t]	Variable de type Files indiquant à quelle file d'attente la tâche t sera affectée $t \in TasksInFile$
DSejour [t]	Variable entière indiquant le délai de séjour d'une tâche t dans une file d'attente $t \in TasksInFile$

DArcTask	Variable entière indiquant la durée de la sous tâche sur un arc $t \in ArcTask$
DNodeTask	Variable entière indiquant la durée de la sous tâche sur un noeud $t \in NodeTask$
S[t].start	Date de début de la tâche $t \in Tasks$
S[t].end	Date de fin de la tâche $t \in Tasks$
Destination [t]	Variable de type « nœud » indiquant le nœud destination du voyage à vide $t \in VAV$

6.3.2. Modèle

Dans le modèle les machines, les nœuds, les chariots, les segments et les files d'attente sont définies comme étant des ressources **unaires**. Cette définition assure la prise en compte des contraintes de disjonction. Les contraintes de disjonction ne sont pas incluses dans le modèle pour en alléger la lecture. L'objectif est de minimiser la date de fin de l'horaire. Les contraintes sont suivies d'explications.

Objectif

Minimiser

S [makespan]. end (1)

L'objectif est de minimiser la date de fin de l'horaire en respectant les contraintes suivantes :

Relations d'antériorité entre les tâches de manutention et les voyages à vide

$a[t_d].end = a[v_p].start$

$\forall (t,v) \in MANvav \mid t_d =$ Dernier[t] & $v_p =$ Premier [v]

Une tâche de manutention est suivie d'un voyage à vide. Le voyage à vide commence au moment où la tâche de manutention finit. (2)

Une tâche de manutention est suivie d'un voyage à vide. Le voyage à vide commence au moment où la tâche de manutention finit. (2)

Relations d'antériorité entre les tâches de manutention et les voyages à vide

followsOnChariot[t] = q $\Rightarrow$ a [v_d].end=a[q_p].start (3)

$\forall$ (t,v) $\in$ MANvav, q $\in$ Tasksman

Une tâche de manutention suivant une autre tâche de manutention sur un chariot est en réalité successeur d'un voyage à vide suivant la première tâche de manutention.

Détermination des chemins pour les chariots

carries [Debut*x*]= carries[Fin*x*] (4)

x=1…..nombre de chariots.

Deux tâches fictives de début et de fin sont affectées à chaque chariot.

alldifferent carries [Debut*x*] (5)

x=1…..nombre de chariots.

Détermination d'un début de chemin pour chaque chariot.

alldifferent carries [Fin*x*] (6)

x=1…..nombre de chariots.

Détermination d'une fin de chemin pour chaque chariot.

Relations d'antériorité entre les dernières tâches et la tâche fictive de fin

a[makespan].start $\geq$ a[t].start + duree [t] (7)

$\forall$ t $\in$ TasksLast

La date de fin de l'horaire est plus grande ou égale à la date de fin des dernières tâches.

Détermination d'ordre de visite des chariots

alldifferent (followsOnChariot) (8)

Chaque tâche de manutention occupe une seule place sur un seul chariot.

followsOnChariot [p] ∈ TasksmanFIC (9)

∀ t ∈ TasksmanSour

Une tâche fictive de début ou une tâche de manutention peut être suivie, sur un chariot, par une autre tâche de manutention ou une tâche fictive de fin.

Détermination d'ordre de visite des chariots

followsOnChariot [p] <> p (10)

∀ p ∈ TasksmanSour

Une tâche de manutention ne peut pas être consécutive à elle-même.

Relations d'antériorité pour les sous-tâches

$a[t_i].end = a[t_j].start$ (11)

$\forall t_i,t_j \in AllTasks [t], t \in Tasksman \cup VAV$

Pour deux tâches successeurs dans l'ensemble de sous-tâches constituant une tâche *t*, la fin de la première détermine le début de la deuxième.

Affectation aux chariots

followsOnChariot [t] ⇒ carries[t]=carries [q] (12)

∀ t ∈ Tasksman , q ∈ Tasksman

Deux tâches de manutention qui sont des successeurs directs sont affectées à un seul chariot.

carries[t] = carries [v] ∀ (t,v) ∈ MANvav (13)

Une tâche de manutention et le voyage à vide successeur sont affectés au même chariot.

Définition des durées des tâches sur les nœuds

DNodeTask [t]= a[t_a].start - [t_b].end (14)

$\forall$ t $\in$ NodeTask | t_b et t_a sont les tâches prédécesseur et successeur de *t* respectivement

La durée d'attente sur un nœud est déterminée par la date de fin et de début des tâches prédécesseur et successeur respectivement.

Relations d'antériorité entre les tâches de séjour dans une file d'attente et les tâches de manutention

a[f].end = a[t_p].start (15)

$\forall$(f,t)$\in$ Prec| f $\in$ TasksInFile & t $\in$ Tasksman & t_p = Premier [t]

Une tâche de séjour dans une file d'attente à la sortie d'une machine finit au moment où débute la tâche de manutention successeur.

a[t_d]. end = a[f]. start (16)

$\forall$(t,f)$\in$ Prec| f $\in$ TasksInFile & t $\in$ Tasksman & t_d = Dernier [t]

Une tâche de séjour dans une file d'attente à l'entrée d'une machine débute au moment où finit la tâche de manutention prédécesseur.

Relations d'antériorité entre les tâches de séjour dans une file d'attente et les tâches de production

a[t]. end = a[f]. start (17)

$\forall$(t,f)$\in$ Prec| f $\in$ TasksInFile & t $\in$ Tasksmach

Une tâche de séjour dans une file d'attente à la sortie d'une machine débute au moment où finit le traitement de la tâche prédécesseur sur la machine.

a[f]. end = a[t]. start (18)

$\forall$(f,t)$\in$ Prec| f $\in$ TasksInFile & t $\in$ Tasksmach

Une tâche de séjour dans une file d'attente à l'entrée d'une machine finit au moment où débute le traitement de la tâche successeur sur la machine.

Relations d'antériorité entre les tâches de manutention et les tâches de préparation, palettisation ou entreposage

$a[q].end = a[t_p].start$ (19)

$\forall(q,t) \in Prec | q \in Tasksmach \ \& \ t \in Tasksman \ \& \ t_p = Premier[t]$

Une tâche de manutention successeur d'une tâche de production débute au moment où finit cette dernière. Ici les tâches de production sont les tâches de préparation et de palettisation ou un séjour dans un entrepôt.

$a[t_d].end = a[q].start$ (20)

$\forall(t,q) \in Prec | q \in Tasksmach \ \& \ t \in Tasksman \ \& \ t_d = Dernier[t]$

Une tâche de manutention prédécesseur d'une tâche de production finit au moment où débute cette dernière. Ici les tâches de production sont les tâches de préparation et de palettisation ou un séjour dans un entrepôt.

Relations d'antériorité entre les tâches de manutention et les dernières tâches

$a[t_d].end = a[q].start$ (21)

$\forall(t,q) \in Prec | q \in TasksLast \ \& \ t \in Tasksman \ \& \ t_d = Dernier[t]$

Une tâche de manutention prédécesseur d'une dernière tâche finit au moment où débute cette dernière.

Conservation de flux pour les tâches de manutention et les voyages à vide

$ARCchoisi[t_p] \in ARCS[ORIGIN[t]]$ (22)

$\forall t \in Tasksman \cup VAV, t_p = Premier[t]$

La première sous-tâche de manutention ou d'un voyage à vide consiste à visiter un arc dont le premier nœud est l'origine de la tâche de manutention ou le voyage à vide.

$NodeTask[t_i]$ **requires** $n \ \& \ n <> Destination[t] \Rightarrow ARCchoisi[t_j] \in ARCS[n]$ (23)

$\forall t \in Tasksman \cup VAV, \forall t_i \in NodeTask | t_i \in AllTask[t], t_j$ suivant t_i

Si un nœud *n* est mobilisé par une sous-tâche t_i et qu'il n'est pas la destination de t, un arc dont l'origine est *n* est choisi par la sous-tâche suivante t_j .

a[t_i] **requires** n & n = Destination [t] => ARCchoisi [t_j] ∈ ArcFictif (24)

∀ t ∈ Tasksman ∪ VAV, ∀ t_i ∈ NodeTask| t_i ∈ AllTask [t], t_j suivant t_i

Si un nœud *n* est mobilisé par une sous-tâche t_i et qu'il est la destination de t, un arc dont l'origine est *n* est choisi par la sous-tâche suivante t_j .

Conservation de flux et disjonction des noeuds

ARCchoisi [t_i] ∈ ArcFictif => a [t_j] **requires** Destination [t] (25)

∀ t ∈ Tasksman ∪ VAV, ∀ t_j ∈ NodeTask| t_j ∈ AllTask [t], t_j suivant t_i

Si un arc fictif est choisi par une sous-tâche t_i , le nœud destination de la tâche t est mobilisé par la sous-tâche suivante t_j.

Disjonction des noeuds

ARCchoisi [t_i] = a => a[t_j] **requires** SecondNode[a] (26)

∀ t ∈ Tasksman ∪ VAV, ∀ t_j ∈ NodeTask| t_j ∈ AllTask [t], t_j suivant t_i

Si un arc *a* est choisi par une sous-tâche t_i , son deuxième nœud est mobilisé par la sous-tâche suivante t_j sur un nœud.

a [t_d] **requires** Destination [t] (27)

∀ t ∈ Tasksman ∪ VAV, ∀ t_d = Dernier [t]

La dernière sous-tâche t_d mobilise le nœud destination d'une tâche de manutention ou d'un voyage à vide.

Détermination d'une destination d'une tâche de manutention

Destination [t] = destination [t] (28)

∀ t ∈ Tasksman

La destination d'une tâche de manutention *t* est le poste où le prochain traitement aura lieu (fourni dans les données du problème).

Détermination d'une destination d'un voyage à vide

followsOnChariot [t]= q => Destination [v] = ORIGIN [q] (29)

$\forall$ (t,v)$\in$ MANvav, $\forall$ q $\in$ Tasksman

La destination d'un voyage à vide est l'origine de la tâche de manutention suivante sur le trajet d'un chariot.

Détermination d'une origine d'un voyage à vide

ORIGIN [v] = Destination [t] (30)

$\forall$ (t,v)$\in$ MANvav

L'origine d'un voyage à vide est la destination de la tâche de manutention prédécesseur.

Disjonction des arcs

ARCchoisi [t] = r $\Rightarrow$ a [t] **requires** SEG [r] (31)

$\forall$ t $\in$ ArcTask

Un arc choisi par une tâche sur les arcs mobilise le segment représentant cet arc.

Définition des durées des tâches sur les arcs

DArcTask [t] = Longueur [ARCchoisi [t]] (32)

$\forall$ t $\in$ ArcTask

La durée d'une tâche sur un arc est égale à sa longueur

6.4. Expérimentation et conclusion

Nous avons effectué quelques tests préliminaires pour valider les deux modèles avec des jeux de taille réduite par rapport aux modèles présentés dans les chapitres 4 et 5. Nous avons utilisé les mêmes jeux de données en considérant juste les trois premiers travaux

de chaque jeu de données. Cependant, la taille des deux modèles demeure très grande à cause de la décomposition des tâches, notamment les tâches de manutention et les voyages à vide. Cela multiplie considérablement le nombre de variables et de contraintes pour un même problème où l'on ne considère pas le détail du réseau comme dans les deux chapitres précédents.

Le modèle de programmation mathématique est résolu par CplexMIP et le modèle de programmation par contraintes est résolu par « Scheduler ». Les deux solveurs sont intégrés dans le logiciel « OPLStudio ». La performance du modèle linéaire mixte est prédictible et dépend surtout du nombre des variables. Quant au modèle en programmation par contraintes, nous pensons qu'il a beaucoup de potentiel mais il nécessite un effort d'optimisation de la structure de données et également le développement d'une bonne stratégie de recherche. Cette stratégie doit être taillée sur mesure et doit prendre en compte les spécifités du problème et ne se réduit pas à quelques règles simples. Cet effort pourra rendre le modèle utilisable pour des problèmes de taille industrielle.

La résolution des modèles présentés dans ce chapitre permet l'obtention d'une solution heuristique, si une solution optimale est difficile ou longue à trouver. Il suffit de faire tourner les modèles et d'arrêter la résolution après un certain temps. Nous pensons que les résultats sont encourageants vu la grande taille des formulations. Dans le cas de la formulation linéaire mixte, on atteint un nombre de variables de 24898 et de contraintes de 6821 pour des jeux de données de deux travaux dans l'atelier. Les deux travaux comptent 54 tâches incluant les voyages à vide, les tâches de séjour dans les files d'attente et les tâches fictives de début et de fin de trajet de chariots. Les réseaux d'ateliers utilisés sont présentés en Annexe A. Une première solution réalisable est trouvée en 3 minutes et la preuve d'optimalité prend plusieurs heures. La formulation en programmation par contraintes est d'une taille moindre et d'une performance semblable.

Dans la formulation linéaire mixte, les contraintes de flux combinées avec les contraintes d'antériorité technologique compliquent significativement la résolution. Pour accélérer la résolution, le flux doit être *guidé* en ajoutant plus de contraintes qui évidemment réduisent l'espace de solution réalisables. En plus des contraintes, présentées, une série d'inégalités valides est nécessaire au bon fonctionnement du modèle. Il faut imposer qu'un voyage à vide ne peut pas être suivi par un ensemble de tâches inadmissibles. Les tâches inadmissibles sont toutes les tâches de manutention précédant la tâche prédécesseur direct du voyage à vide. Ces contraintes sont normalement assurées par les relations d'antériorité. D'autre part, nous imposons que la première tâche de manutention sur le trajet d'un chariot soit une première tâche de manutention pour un travail, et que la dernière tâche sur le trajet d'un chariot soit une dernière tâche de manutention pour un travail. Pour formuler ces contraintes, nous utilisons des inégalités à la place des égalités quand cela est possible. Cela aide à la résolution. De même, nous imposons qu'une tâche de manutention suivant un voyage à vide est, soit la tâche de manutention suivante dans la gamme du travail, soit une tâche de manutention d'un autre travail.

Comme illustré au début du chapitre, pour représenter le flux à vide dans le cas où il sort et entre au même noeud, nous créons un nœud fictif et un segment fictif composé d'un seul arc. Cela implique que le flux sorte du nœud par un segment et rentre par un autre au même nœud. Cette hypothèse est nécessaire car dans le modèle on interdit le retour sur un côté déjà visité. Dans le cas contraire, on peut rencontrer des problèmes avec des contraintes de conservation de flux combinées avec les contraintes d'antériorité. Un flux peut revenir sur un côté pour ne pas respecter les contraintes d'antériorité et donc donner un faux résultat.

Ces difficultés ne se posent pas dans la formulation en programmation par contraintes. La façon dont les variables sont définies et les contraintes correspondantes impliquent

nécessairement que le flux soit guidé. Des variables de type « arc » sont définies et un arc suivant est nécessairement accessible à partir d'un arc précédent.

Les deux modèles présentés dans ce chapitre mettent en évidence la différence d'approche de modélisation dans le cas de programmation mathématique et de programmation par contraintes. À titre d'exemple, les contraintes de conservation de flux, que nous avons proposées dans le modèle en programmation par contraintes, sont très concises. À travers les tests nous avons remarqué qu'il est facile, avec la formulation en programmation par contraintes, de trouver une première solution représentant le flux de chariots. Cela est plus difficile avec la modélisation en programmation mathématique. De plus, la modélisation en programmation par contraintes évite la définition de tous les nœuds fictifs nécessaires à la conservation de flux des chariots vides. Une analyse complète des deux modèles peut permettre l'identification des possibilités d'intégration des deux méthodologies de résolution pour combiner leurs avantages respectifs.

Il serait intéressant d'effectuer des expérimentations pour tester la performance des modèles dans certains contextes spécifiques. Il est fort possible qu'elle soit meilleure avec des données plus simples représentant des cas spéciaux en production. Les deux modèles, intégrant une bonne partie, de la réalité industrielle méritent aussi l'essai dans des contextes ou quelques contraintes de ressources sont relaxées, par exemple, les files d'attente ou les segments. Il existe des réalités industrielles ou l'on a peu de contraintes d'espace.

Dans un cas où les segments du réseau de manutention sont très longs, nous pouvons davantage raffiner les formulations de façon à éviter la mobilisation, par un seul chariot, de tout le segment. Ce dernier peut être discrétisés en plusieurs petits segments, à considérer séparément. Il peut alors exister des nœuds fictifs qui ne correspondent pas à de vrais nœuds dans le réseau physique. Cette discrétisation augmentera d'une part la

taille du modèle, mais d'autre part, elle permettra une meilleure utilisation des ressources.

La formulation des trois problèmes présentés dans cette thèse n'a pas été faite dans le but de comparer les deux méthodologies proposées ni de comparer les approches hiérarchique et intégré. Nous nous sommes intéressés en premier lieu à l'étude de différents problèmes d'ordonnancement intégré et au développement des outils utiles pour le praticien. Toutefois, les différents développements nous permettent de tirer de conclusions. Il est clair qu'une approche hiérarchique est sous-optimale par rapport à une approche intégrée. Mais, l'intégration a un coût.

Comme précisé au chapitre 4, nous avons considéré un réseau bidirectionnel pour tester le modèle présenté au chapitre 5 où l'on considère des alternatives en routage. Cela veut dire qu'on considère des plus courts chemins entre une origine et une destination contrairement aux données de la littérature qui considèrent un réseau unidirectionnel, ce qui augmente la durée de manutention dans une direction donnée. De ce fait, les horaires du modèle du chapitre 5 sont parfois plus longs car les conflits sont considérés et plusieurs routes sont mobilisées durant l'accomplissement des tâches de manutention. D'autres fois, ils sont plus courts car il a été possible en effectuant l'ordonnancement intégré de trouver une solution sans conflits avec les durées de plus courts chemins du réseau bidirectionnel. Les résultats sont obtenus dans des temps très courts et justifient une intégration.

Dans le dernier modèle, nous n'avons pas testé des problèmes de la même taille. De plus, nous avons inclus d'autres ressources (files d'attente), et nous avons ajouté des tâches de préparation et de palettisation, ce qui rend la comparaison difficile. Dans l'état actuel, le temps de résolution est long pour ce problème et le coût de l'intégration est élevé. Enfin, la décision d'intégration doit, aussi, être prise en fonction de la réalité industrielle et de ses ressources critiques.

CHAPITRE 7 :

PROBLÈMES D'ORDONNANCEMENT EN PRODUCTION: PERTINENCE DES OBJECTIFS D'OPTIMISATION

Dans ce chapitre, nous discutons de la pertinence des formulations dans la littérature sur les problèmes d'ordonnancement, et nous présentons une décomposition des différents coûts de production pertinents. Cette décomposition des coûts sert à formuler des objectifs d'optimisation plus réalistes et intéressants pour les industriels.

7.1. Objectifs recherchés dans la résolution d'un problème d'ordonnancement

La vaste littérature sur l'ordonnancement traite souvent des problèmes classiques définis depuis les années 50. Peu de réflexions quant à la pertinence des objectifs et des contraintes ont été présentées. On trouve alors des formulations des problèmes peu riches qui ne représentent pas des cas réels dans le milieu industriel. Cette affirmation a été appuyée par Browne et al. (1981).

La formulation des problèmes comprend les contraintes et les objectifs. De nombreuses réalités présentent des contraintes très différentes. Souvent toutes les contraintes ne sont pas formellement prises en compte. Or, quelques-unes peuvent être traitées de façon approximative en un deuxième temps après qu'une première solution soit calculée sans la prise en compte de toutes les contraintes du problème.

Quelques auteurs ont étudié des problèmes d'ordonnancement tirés du milieu industriel où des contraintes spécifiques aux différents cas ont été prises en compte (Geiger 1997

et Solomon 1991). Les objectifs classiques sont utilisés dans la formulation de ces problèmes. Pourtant, une hiérarchisation intéressante de plusieurs objectifs a été présentée dans l'article de Solomon (1991). Cette hiérarchie a été déterminée en fonction de la capacité disponible des différentes ressources.

Par objectifs classiques nous faisons référence aux deux types d'objectifs qui sont utilisés dans les formulations analytiques décrivant les problèmes d'ordonnancement depuis les premiers travaux dans le domaine. Il s'agit des objectifs systèmes et des objectifs de respect des dates d'échéance (Graves et al. 1981). Le tableau 7.1 donne la liste des objectifs souvent rencontrés dans la littérature pour les problèmes d'ordonnancement de la production. Les objectifs usuels considérés dans les problèmes d'ordonnancement de la manutention ont été présentés dans le chapitre 2, au tableau 2.1. Les problèmes traités sont généralement mono-objectifs. Quelques problèmes d'ordonnancement d'une seule machine ont été traités pour des objectifs multi-critères (Bagchi 1989), et ce sans la prise en compte des contraintes relatives au système de manutention.

Cette situation propose que les objectifs formulés dans le cadre des premiers problèmes d'ordonnancement correspondaient à une certaine réalité rencontrée en pratique et reflétaient donc le point de vue des gestionnaires ou encore celui du chercheur qui s'est intéressé à l'étude du problème. Depuis cette époque le choix des objectifs n'a pas été remis en cause.

Comme de nombreuses réalités correspondent à des contraintes différentes, on retrouve également des objectifs différents qui découlent des objectifs des gestionnaires concernés.

Nous pensons que la complexité des problèmes d'ordonnancement, même ceux de petite taille, a amené les chercheurs vers le développement de meilleures méthodes de résolution sans beaucoup s'attarder sur la compatibilité et la pertinence des objectifs.

La littérature ne contient pas de travaux traitant du problème de compatibilité de ces objectifs avec les objectifs souhaités par les gestionnaires dans le milieu industriel. Un autre problème relié aux objectifs d'ordonnancement est la place qu'occupent les objectifs par rapport aux contraintes et celle qu'occupent les contraintes par rapport aux objectifs.

En (1973), Holloway et Nelson présentent un argument qui suppose que les problèmes formulés dans la littérature sont abordés de façon très différente de celle des gestionnaires. Or, la formulation des objectifs et des contraintes est en quelque sorte mélangée selon les deux points de vue. L'exemple rapporté dans l'article est celui d'un problème classique d'atelier multigamme avec l'objectif de minimisation des retards sous les contraintes de respect de capacité fixe des machines et de respect de l'antériorité technologique entre les tâches. Les auteurs proposent deux formulations alternatives décrivant le même problème selon des points de vue différents. La première formulation présente un point de vue pratique :

- minimiser les ressources nécessaires ou encore le temps supplémentaire pour pouvoir rencontrer les commandes et ceci sous les contraintes de respect des dates d'échéance et de respect des relations d'antériorité technologique.

La deuxième formulation est intéressante sur le plan de la résolution :

- minimiser les violations des relations d'antériorité technologique sous les contraintes de respect des dates d'échéance et de respect de capacité fixe des machines. Trouver une solution pour cette formulation sans violer des contraintes d'antériorité donnerait éventuellement une solution optimale pour la formulation initiale du problème. Cette deuxième formulation a permis

également le développement d'une heuristique permettant la résolution du problème et donnant de bonnes solutions.

La taille des problèmes tests a été limitée (jusqu'à 7 machines et 14 travaux). À notre connaissance, cette remise en cause de la formulation des problèmes d'ordonnancement n'a pas été reprise dans la littérature.

La première formulation proposée reflète un point de vue important. En milieu industriel, il faut respecter les dates d'échéance selon un coût à déterminer. Avoir recours au temps supplémentaire est parfois inévitable. Par contre, même le temps supplémentaire en équipements et en opérateurs possède une capacité limitée imposant une contrainte au problème. Si on doit dépasser la capacité supplémentaire disponible, on fait alors appel à la sous-traitance.

Dans un sens, l'idée avancée par les auteurs propose de résoudre un problème de satisfaction de contraintes. Dans cette thèse, nous proposons la résolution du problème d'ordonnancement intégré par la programmation par contraintes. Cette technique est très efficace pour la formulation et la résolution des problèmes comme problèmes de satisfaction de contraintes. Cela permet alors de résoudre de façon efficace et élégante des formulations pratiques comme celle décrite plus haut.

Tableau 7.1. Objectifs usuels utilisés dans la formulation des problèmes d'ordonnancement de la production

Objectif	**Parmi les contributions**
Minimiser la date de fin de l'ordonnancement (C_{max})	Gilmore et Gomory (1964)
Minimiser le retard algébrique maximal (L_{max})	Lawler et moore (1969)
Minimiser le retard positif maximal (T_{max})	Lawler (1973)
Minimiser la somme des retards (ΣT_j)	Schrage et Baker (1978)
Minimiser la somme des dates de fin pondérées ($\Sigma w_j C_j$)	Lenstra et al. (1977)
Minimiser la durée de séjour (ΣC_j)	Lenstra et al. (1978 a)
Minimiser la somme actualisée des dates de fin pondérées $\Sigma w_j(1-re^{-rcj}\ dt)$	Lenstra et al. (1978 a)
Minimiser la somme des retards pondérés ($\Sigma w_j T_j$)	Lawler et moore (1969)
Minimiser le nombre des travaux en retard (ΣU_j)	Moore (1968)
Minimiser le nombre pondéré des travaux en retard ($\Sigma w_j U_j$)	Karp (1972)

Ces objectifs usuels, présentés dans le tableau 7.1., sont encore repris dans le cadre de la formulation des problèmes d'ordonnancement intégré.

Parmi, les objectifs intéressants considérés pour les problèmes d'ordonnancement, nous retenons ceux appelés " juste à temps " qui visent à minimiser les retards de production

ainsi que les avances (Biskurp, D. et Cheng, T.C.E., 1999). Ceci se fait dans le but d'épargner les coûts de stockage de même que d'épargner la pénalité de retard. Cette vision du problème propose la prise en compte des coûts importants dans le processus de production. Pourtant, le vrai problème serait de respecter les dates d'échéance tout en minimisant les coûts de stockage et des ressources supplémentaires si nécessaire. Or, il faut faire un compromis au niveau des différents coûts pertinents. En effet, l'objectif de minimiser l'ensemble de coûts reliés au fonctionnement du système de production, rarement étudié (Lasserre, 1992), serait le plus pratique et pertinent. Cette formulation tient compte d'un coût unitaire de production, un coût de possession du stock, un coût de pénurie de stock et un coût de réglage. Le problème est formulé pour un ensemble de périodes.

Les objectifs reliés à l'optimisation des coûts sont généralement utilisés dans les modèles de planification pour le calcul des lots de production. Habituellement, ils ne sont pas utilisés dans le cadre des problèmes d'ordonnancement. McNaughton (1959) présente un objectif de minimisation de la somme des coûts linéaires de retard pour un problème d'une seule machine, ce qui revient à minimiser la somme des retards.

En fait, la définition d'un objectif d'optimisation pour un problème d'ordonnancement reflète un certain coût qu'on suppose être le plus important. Par exemple, en minimisant la date de fin de l'horaire, on minimise un temps mort pour les équipements et les travailleurs et donc on minimise un coût à l'entreprise. La minimisation de la somme des retards ou le retard maximal reflète aussi un coût qui serait lié à, par exemple,

- la perte d'un client
- un supplément pour choisir un choix d'expédition plus cher pour respecter les échéances.

Il serait donc intéressant de prendre en compte les coûts directs, indirects et les coûts de pénalité et d'opportunité ce qui n'a pas été présenté de façon complète dans la littérature

dans la formulation du problème. De même, il faut accorder les bons coefficients aux coûts pour que la somme des différents coûts soit significative. Cela impliquera donc de calculer/estimer les différents coûts.

Les coûts encourus par les entreprises manufacturières ont été identifiés par Lovett, JR., (1995) :

- coûts d'ingénierie, design et développement
- main-d'œuvre de fabrication
- coûts d'équipements et d'outils
- coûts des matières
- supervision
- coûts de contrôle de la qualité, fiabilité, et tests
- coûts de réception et d'expédition
- coûts d'emballage
- coûts de manutention et de stockage
- coûts de distribution et de marketing
- financement
- taxes et assurances
- frais généraux
- frais administratifs.

Parmi les coûts listés ci-dessus, quelques-uns sont directement liés au problème d'ordonnancement tandis que le reste sort du cadre du problème et présente des dépenses encourues par l'entreprise manufacturière quel que soit l'horaire de production.

7.1.1. Coûts pertinents

Les coûts pertinents sont listés ci-après avec une proposition de définition et de notation :

- **main-d'œuvre de fabrication** (un seul coût total ayant des composants directs et indirects comme la formation et les avantages sociaux est considéré). De plus, on considère un seul taux pour tous les opérateurs d'un seul type d'équipement. Les raffinements selon les compétences et l'ancienneté ne sont pas pris en compte.

Le coût de la main-d'œuvre de fabrication = MO (r) + MO (sr) +MO (sf)
MO (r) = main-d'œuvre régulière
MO (ss) = main-d'œuvre supplémentaire pendant les jours de semaine
MO (sf) = main-d'œuvre supplémentaire pendant les jours fériés

Les coûts reliés aux opérateurs doivent être calculés en fonction de l'équipe du travail en question pour permettre de considérer le coût des heures supplémentaires ou encore du personnel supplémentaire.

Si l'horaire de production établi est d'une longueur donnée, z unités de temps par exemple, on considère que les premières x unités de temps font partie du temps régulier et que les y unités de temps suivantes font partie du temps supplémentaire.

Les taux horaires de la main-d'œuvre de fabrication diffèrent selon la spécialité de l'opérateur (leurs postes de travail respectifs : emballage, test ou autre), et leurs fonctions. Un coût de supervision peut donc être considéré.

- **coût d'équipements et d'outils** (coût d'utilisation/unité de temps). On considère que le coût d'acquisition, l'amortissement et l'inflation sont déjà inclus dans ce coût. Le temps mort des équipements n'est pas à estimer et il fait partie des décisions à prendre à d'autres niveaux.

Un coût supplémentaire d'utilisation d'équipements soit de manutention soit de production est reflété par des frais des maintenances plus rapprochées, suite à un certain

nombre d'heures d'utilisation. Alors, pour un horaire qui comporte y unités de temps supplémentaire, on considère un coût encouru de

$$(y/nbHM)^* CM$$

où nbHM = nombre d'heures de travail de l'équipement avant d'effectuer la maintenance.

CM = coût de la maintenance pour l'équipement.

De même, la maintenance impose une contrainte aux ressources supplémentaires en équipements.

Même s'il s'agit de produire la même commande qui demande un nombre fixe d'heures d'opération, un horaire plus étalé fait varier des choses. Les heures supplémentaires restent des heures pendant lesquelles l'équipement est en marche et qui influencent alors des décisions de maintenance.

Les coûts de contrôle de la qualité, fiabilité, et tests (coût d'un poste de travail qui peut posséder des équipements spécialisés) peuvent sensiblement différer par rapport aux coûts d'opération normale.

- **coûts de manutention** : en plus du coût correspondant aux temps supplémentaires d'opération, la maintenance, et les superviseurs du système et éventuellement les opérateurs, il existe un coût correspondant à la distance parcourue.

Pour la commande il faut alors minimiser $D_t^* C_p$

où D_t = distance totale parcourue dans l'atelier.

C_p = coût de parcourir une unité de distance.

- **coût de stockage** : les commandes en cours de production constituent un stock en cours qui représente un coût à l'entreprise correspondant au délai de séjour en atelier. Il est à noter que les matières premières ayant moins de valeur ajoutée

coûtent moins par rapport aux produits presque finis. De même, la sortie de la marchandise va générer de l'argent et est considérée comme une source de financement. Ainsi, la possession des produits correspond à un capital immobilisé et donc à un coût d'opportunité.

Pour simplifier le calcul de ce coût, on peut considérer trois coûts de stockage, même si on atteint plusieurs niveaux de valeur ajoutée pendant la durée de séjour du produit en atelier.

CsMP= coût de stockage des matières premières.
CsPC= coût de stockage des produits en cours de production.
CsPF= coût de stockage des produits finis.

Par ailleurs, d'autres coûts sont à inclure :

- **les pénalités de retard**. Les pénalités de retard sont évaluées selon les termes des contrats et ils peuvent atteindre le double de la valeur de la commande. Ce coût est relié à un certain niveau de service promis et éventuellement il peut correspondre à une perte d'un client.

- **coûts de mise en course/réglage des équipements**. Ce coût est à prendre en compte dans le cas où la production peut être interrompue ou encore dans le cas où l'on utilise des lots de transfert et on fait plusieurs réglages pour un même lot de production. Il correspond à un coût d'arrêt de production et le coût d'opérateurs spécialisés sollicités plus souvent pour effectuer plusieurs réglages. Une stratégie de production qui coûte moins que l'arrêt de l'équipement est donc encouragée.

- **coût supplémentaire** encouru dans le choix **d'option d'expédition** pour satisfaire les délais.

- **coût des palettes** : ce coût est à considérer dans le cas où on considère plusieurs lots de transfert. On peut également considérer le coût d'utilisation des palettes en fonction du temps.

Dernièrement, un mauvais ordonnancement peut résulter en un nombre élevé d'unités de temps mort qui représente aussi bien comme les capitaux immobilisées un coût d'opportunité à l'entreprise manufacturière.

Dans cette réflexion nous avons essayé de limiter les coûts liés au problème d'ordonnancement. À première vue on peut penser que les coûts pertinents concernent uniquement les opérations effectuées et les ressources requises dans l'atelier, mais selon la liste présentée ci haut, on remarque que d'autres coûts peuvent être également liés.

Il convenait de cerner les coûts liés au problème d'ordonnancement de sorte qu'ils soient complets et représentatifs et en même temps liés à l'échelle correspondant au problème. Bien que la majorité des coûts de l'entreprise manufacturière soient liés, il ne faut pas confondre les différents niveaux de prise de décision.

7.1.2. Estimation des différents composants de coût

Plusieurs techniques comptables sont à la base de l'estimation et l'allocation des coûts. L'allocation des coûts aux différentes activités de l'entreprise peut être problématique si les ressources de l'atelier sont utilisées pour d'autres fins. Nous ne considérons pas ce scénario. Les ressources qu'on considère appartiennent à l'atelier et sont utilisées dans toutes les activités reliées à l'horaire de production.

Souvent les techniques d'estimation de coûts sont utilisées pour les fins de fixation de prix pour les produits. Ceci se fait de façon très agrégée en considérant des données historiques sur les délais de fabrication d'un produit en particulier. Or la composante

temps (horaire) est souvent une donnée fixe en ce qui concerne le calcul de coûts de fabrication. Il est vrai qu'un certain taux de production retenu des données historiques peut être une information intéressante dans la détermination des temps opératoires pour une unité de produit. Par contre quand plusieurs opérations appartenant à plusieurs produits sont à effectuer en atelier, l'ordre établi peut largement influencer les coûts encourus. Les coûts méritent donc une analyse selon le point de vue de prix de revient et pas seulement selon un point de vue comptable.

Il est à noter aussi que les données sur les coûts sont souvent attribuées d'après une certaine structure de coûts « cost break-down structure » qui n'est pas celle utilisé pour l'établissement de l'horaire et donc les activités génératrices de coût peuvent être difficilement repérables. Tout dépend aussi du niveau de raffinement considéré dans le découpage des tâches. À titre d'exemple, le tableau 7.2 donne des taux horaires du personnel de production.

Tableau 7.2. Taux horaires du personnel de production

Classement	**Taux horaire moyen en $US**
Débutant	8.50
Opérateur d'outil pneumatique	13.90
Soudeur	14.50
Soudeur par résistance par points	14.10
Opérateur de machines	16.50

Source : Lovett, J. N., Jr., Cost estimating in manufacturing (1995), in Cost Estimator's Reference Manual, John Wiley & Sons, inc.

7.2. Objectifs correspondants à plusieurs niveaux hiérarchiques

Une question importante relative au choix des objectifs est la place qu'occupe le problème d'ordonnancement traité dans la hiérarchie des problèmes relatifs au système

de production. La hiérarchie de décisions dans les problèmes de **planification manufacturière** (dans le sens le plus générique) correspond à un ensemble des problèmes traités généralement l'un après l'autre et donc chaque niveau impose des *contraintes* au niveau successeur dans la hiérarchie.

La **planification manufacturière et le pilotage**, ou encore plus brièvement, la **planification manufacturière** se veut une activité générique qui inclut trois phases distinctes (Vollmann et al. 1992, p.5) :

- la création d'un plan global de fabrication qui donne les produits finis et leurs différentes caractéristiques;
- la planification détaillée des besoins en matières et en capacité pour soutenir le plan global de fabrication;
- l'exécution des plans au niveau de l'atelier et au niveau des achats.

La figure 7.1 présente un schéma simplifié du système de planification manufacturière et de pilotage.

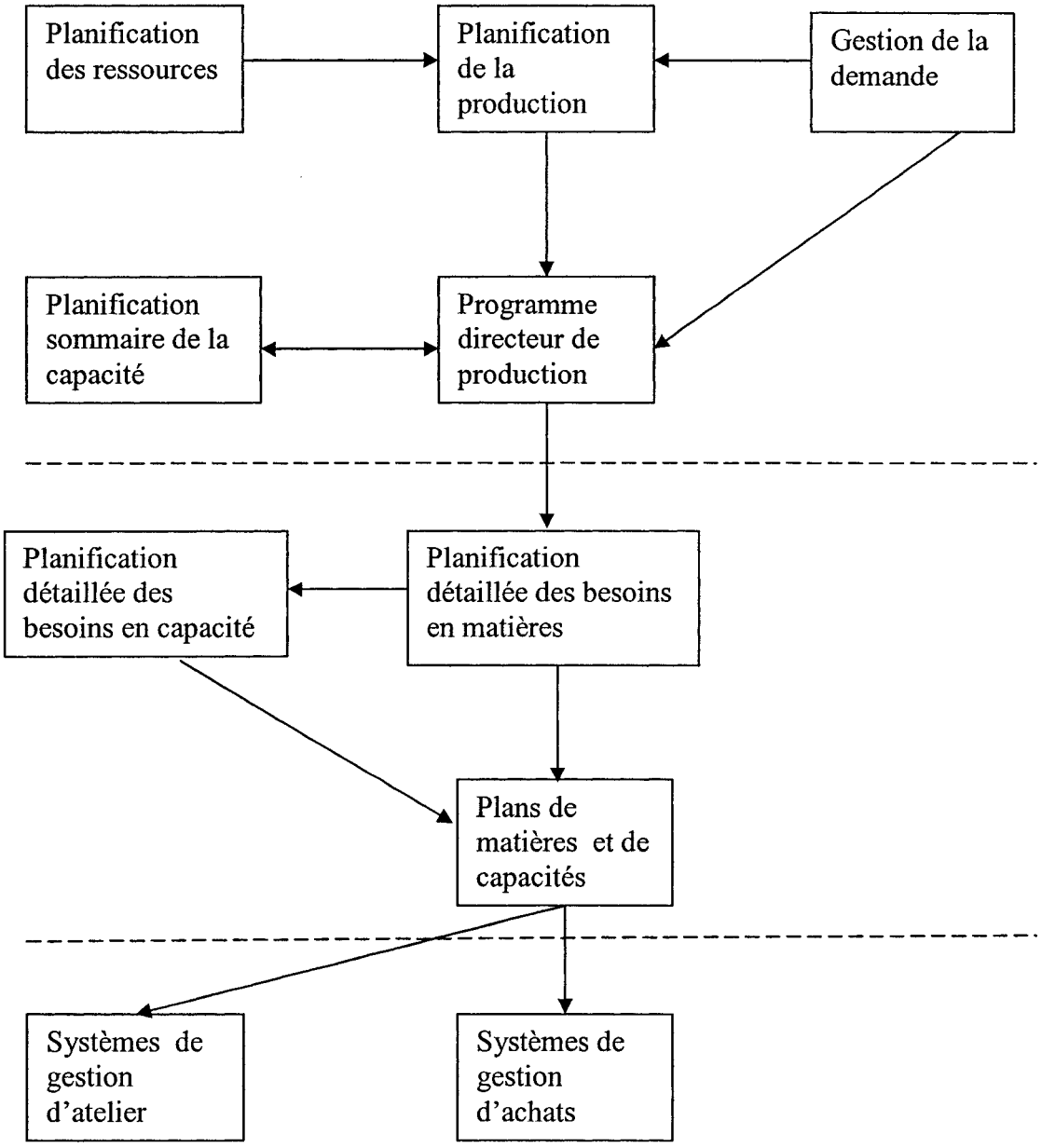


Figure 7.1. Système de planification manufacturière et de pilotage

Tiré de Vollmann et al. (1992). *Manufacturing planning and control systems.* IRWIN, P. 211

La question de la non faisabilité de quelques problèmes à certains niveaux hiérarchiques a été soulevée par certains auteurs qui présentaient des contributions intégrant plusieurs niveaux de décisions dans une formulation d'un seul problème.

Un exemple est celui de Lasserre (1992) qui résout un problème intégré de planification et d'ordonnancement. Au niveau de la **planification,** les quantités sont générées ainsi que les dates de lancement des produits finis, des sous-assemblages et des composants et on décide également des dates d'échéance. Les gammes sont donc déterminées ainsi que les différentes tâches à effectuer. Au niveau **d'ordonnancement,** on affecte les tâches de fabrication dans le temps.

Il faut donc garder une cohérence au niveau des contraintes affectant les différentes décisions hiérarchiques. On peut intégrer les décisions et on peut se servir également d'une approche de planification qui contournerait les désavantages des approches allant de haut en bas en mieux considérerait les *contraintes* aux niveaux inférieurs dans la hiérarchie de décisions. Cette question est abordée différemment avec une analyse axée sur les *objectifs* plutôt que sur les *contraintes*. Les contraintes ainsi que les objectifs à de différents niveaux doivent être bien reliés pour ne pas créer des incohérences à différents niveaux de décision.

Roy (1998) propose que les méthodes d'ordonnancement à capacité finie deviennent efficaces (signifiantes) si elles sont intégrées dans un cadre hiérarchique définissant les objectifs de l'entreprise et ce à plusieurs niveaux.

En effet, les objectifs d'optimisation dans les problèmes d'ordonnancement doivent non seulement traduire les objectifs souhaités par les gestionnaires responsables des systèmes de production au niveau de l'atelier, mais ils doivent s'insérer également dans un cadre global pour la réalisation des objectifs des organisations.

Les différents niveaux de pilotage ont été identifiés par l'auteur :

- budgétaire (design du système)
- stratégique (optimisation du processus)
- système manufacturier (pilotage du processus)
- atelier (contrôle de gestion).

À différents niveaux, quelques mesures de performance (objectifs) seront surveillées pour s'assurer que l'objectif fixé par le niveau le plus élevé dans la hiérarchie sera rencontré. Dans, ce contexte les objectifs sont définis en terme de cibles et non pas en terme de minimisation et de maximisation.

Si nous prenons comme hypothèse que les gains et les pertes en industrie manufacturière peuvent être traduits en terme de coûts et de profits, la représentation des objectifs de coûts sera donc justifiée à différents niveaux et le transfert d'un niveau à un autre se fera de façon assez souple sans avoir besoin de « traduire » les objectifs *dans la langue* d'un autre niveau.

Comme présenté plus haut, pour le problème d'ordonnancement, nous avons intérêt à diminuer les coûts de production encourus au niveau de l'atelier de production et les coûts qui peuvent être éventuellement engendrés à cause de la performance de l'horaire, par exemple, les coûts de stockage.

7.3. Planification et ordonnancement : les échelles de temps

Dans la littérature francophone, les mots planification « planning » et ordonnancement « scheduling » sont parfois utilisés pour désigner l'affectation des activités dans le temps, souvent appelées tâches. Selon la discussion présentée plus haut, ces deux activités font partie du système de planification manufacturière et du contrôle.

Dans la littérature anglophone la distinction est mieux faite, pour l'ordonnancement de la production « scheduling », nous retenons la définition suivante :
C'est la définition des dates de début et de fin et l'affectation des ressources à chacune des tâches des travaux donnés, tout en prenant en compte les différentes contraintes qui peuvent être liées aux travaux ou aux ressources (Portmann, M.-C. 1997).

Pourtant, pour la planification de la production on ne trouve pas de définition si claire et standard. La planification de la production est probablement l'aspect le moins compris de la planification manufacturière et du pilotage (Vollman 1992, P. 275). Il est à noter également que l'existence de plusieurs types de représentations des systèmes de production rend difficile la standardisation d'une définition.

Il existe, dans la littérature, trois types de modèles pour représenter les systèmes de production : les modèles **agrégés**, les modèles **fonctionnels** et les modèles **hiérarchiques** (Saad 1990). Les paragraphes suivants discutent des différents modèles.

Les modèles **agrégés** traitent de la détermination des quantités à produire sur un horizon de temps donné, typiquement une année. Les aspects d'ordonnancement ne sont pas traités. Les modèles **fonctionnels** reconnaissent les relations d'interdépendance qui existent entre la fonction de production et les autres fonctions de l'entreprise. Les modèles **hiérarchiques** reconnaissent la nature hiérarchique du processus, généralement défini en terme des niveaux stratégique, tactique et opérationnel. Ce sont ces derniers modèles qui situent la fonction d'ordonnancement dans une hiérarchie de prise de décisions. On trouve plusieurs modèles hiérarchiques dans la littérature tels que les modèles de Hax et Meal (1975) et de Pinedo présentés au chapitre 2.

Beaucoup de représentations ont également été développées dans le cadre de l'étude des SMFs. Stecke (1983) a présenté une hiérarchie de deux niveaux d'un SMF, un niveau agrégé et un autre détaillé. Le niveau agrégé optimise les taux de production moyens et

la longueur des files d'attente en se basant sur des données agrégées. Le niveau détaillé considère les contraintes actuelles du SMF telles que la capacité du magasin d'outils, le type des machines et la disponibilité du temps d'opération. En 1984, l'auteur présente les différents problèmes des SMFs relatifs aux différents niveaux de prise de décision. Un modèle de trois niveaux a été présenté par Jaikumar et Van Wassenhove (1989). Le premier niveau est un problème de planification agrégée. Le deuxième niveau est un problème de chargement d'outils et le niveau le plus bas dans la hiérarchie représente un contrôle en temps réel du SMF.

Sodhi et al. (1994) présentent un modèle hiérarchique à 4 niveaux pour le pilotage d'un SMF. Au premier niveau, les pièces à produire par le SMF sur un horizon de planification à moyen terme sont sélectionnées. On fait l'hypothèse que la demande à long terme est connue. Au deuxième niveau, la flexibilité de la configuration du SMF est utilisée pour produire des lots complémentaires de pièces durant chaque période de travail. Le niveau trois représente les décisions de sélection des routes qui tiennent compte de la capacité du système de manutention, ainsi que les décisions sur l'allocation d'outils supplémentaires pour faciliter le routage. Le quatrième niveau concerne le pilotage du système à court terme. Des heuristiques sont utilisées à cette fin (routage actuel, séquence, manutention, contrôle des allées).

La motivation pour développer les modèles hiérarchiques est la suivante. Un modèle représentant tous les niveaux serait difficile à résoudre à cause de la complexité du problème résultant. Dans une approche hiérarchique, chacun des niveaux de prise de décision correspond à la résolution d'un problème. Ceci permet de formuler des problèmes moins complexes. De même, cette approche permet la révision des décisions aux niveaux les plus proches de l'origine des perturbations. Chaque problème est traité à son niveau correspondant. Par exemple, dans le modèle de Sodhi et al. (1994) un problème d'une machine durant une période de travail est traité au niveau le plus bas dans la hiérarchie et les écarts du calendrier optimal sont reportés à un niveau supérieur

pour permettre de faire des révisions. Dans quelques cas, la hiérarchie de production est la même que la hiérarchie organisationnelle du milieu manufacturier.

Cette représentation hiérarchique correspond également à des périodes définies dans le temps et donc elle correspond à des échelles de temps allant des plus longues aux niveaux élevés dans la hiérarchie aux plus courtes aux niveaux inférieurs dans la hiérarchie. Les durées varieront d'un cas à un autre. L'intervalle de temps entre la sortie de deux unités du produit (taux de production) varie donc selon l'industrie et détermine les durées types pour les activités de planification et celles d'ordonnancement dans différents domaines de production.

Tableau 7.3. Exemples de l'ordre de grandeur des intervalles de temps entre la sortie de deux unités de produit

Intervalle de temps se mesurant en	**Exemples d'industrie**
Secondes	Raffinerie d'huile Industries alimentaires
Minutes	Montres
Jours	Camions
Semaines	Avions
Mois	Maisons Bateaux

Il est à noter que des niveaux de décision qui sont des successeurs directs dans une certaine représentation du système de production peuvent être divisés plus finement comme ils peuvent être combinés dans un seul niveau dans d'autres représentations.

Pour conclure avec une définition pour la planification de la production, nous proposons la définition suivante :

La planification de la production fait le lien entre la haute direction d'une entreprise manufacturière et la fabrication. Elle détermine la mission que la fabrication doit accomplir pour réaliser les objectifs de l'entreprise. On peut donc décider des produits (gammes), taux de production, ou inversement des quantités à produire, leurs dates d'échéance ainsi que les tailles des lots selon les données sur la demande et les ressources éventuellement disponibles. Le programme directeur de production présente de façon plus explicite et étalée dans le temps les quantités à produire. La planification correspond normalement à un horizon de temps représentant des multiples de la période pour laquelle un horaire détaillé de production (ordonnancement) est à calculer.

7.3.1. Critères influençant les échelles du temps pour les activités de planification et d'ordonnancement

Selon le type d'industrie et le mode de fonctionnement dans le marché, les durées de temps correspondant à effectuer une activité de planification ou d'ordonnancement peuvent largement changer. Notons par exemple les industries automobiles et les industries alimentaires : les temps nécessaires pour planifier et pour ordonner les activités de production peuvent largement différer. De plus, produire de façon proactive et avoir une certaine stratégie marketing est différent de produire de façon réactive suite à des commandes des clients. La notion de planification change beaucoup.

Nous proposons ces trois critères, qui influencent les échelles du temps pour les activités de planification et d'ordonnancement :

- le type d'industrie (cycle de vie du produit/taux de production)
- le mode de fonctionnement dans le marché (réponse à la demande)
- la répétitivité au niveau des opérations (contexte projet ou pas).

L'ordonnancement est l'affectation des tâches dans le temps. L'horizon de temps est une composante très importante dans les problèmes d'ordonnancement. Dans la résolution des problèmes, cet horizon n'est pas défini a priori. C'est plutôt l'ensemble de travaux qui est donné pour produire un horaire. L'horaire se déroule par la suite sur un nombre d'unités de temps correspondant à la longueur de celui-ci qui débute à une certaine heure qui serait précisée plus tard. Cette heure peut correspondre, par exemple, au début d'une équipe de travail.

Dans le contexte d'ordonnancement de projets l'horaire se mesure normalement en un nombre de jours, de mois ou même d'années. La même situation peut être rencontrée dans quelques domaines de production tel que dans la construction navale. Le mode de fonctionnement dans ces milieux donne un sens différent aux notions de planification et d'ordonnancement et les frontières entre ces deux notions sont parfois éliminées.

L'échelle de temps dans les problèmes d'ordonnancement pose une difficulté supplémentaire. Comme précisée plus haut, la durée sur laquelle se déroulent les activités n'est pas connue a priori et elle est déterminée par l'horaire à produire. Notons que dans les problèmes d'ordonnancement, on ne formule pas explicitement les contraintes temporelles relatives au début et à la fin d'un horizon de temps et donc dans la résolution de ces problèmes, on peut s'attendre à avoir une solution qui s'étend sur une période représentant une équipe de travail plus une fraction d'une équipe de travail et donc l'horaire pourra être divisé n'importe comment d'une façon qui ne correspondrait pas à l'organisation du travail au sein de l'entreprise.

Une autre éventualité serait de découper l'horaire sur les équipes de travail de façon réfléchie en considérant les capacités des machines à travailler sans ou avec surveillance. En effet, la prise en compte des détails des équipes de travail pourrait complexifier le problème, mais au moins dans un deuxième temps, un traitement peut se faire pour s'assurer de produire un horaire qui peut être adopté par le personnel de l'entreprise.

D'autre part, la divisibilité de l'horaire peut provoquer la question du temps supplémentaire. Deux scénarios peuvent se poser :

- continuer le travail commencé aujourd'hui avec un coût de temps supplémentaire, ou
- continuer le travail commencé demain avec la prochaine équipe de travail.

Ces deux scénarios nous ramènent de nouveau à la discussion sur les objectifs et attire l'attention sur la considération des objectifs très importants, ceux des coûts. Finir aujourd'hui ou demain évoque l'évaluation des gains et des pertes. Il existe un coût pour utiliser des ressources supplémentaires et il existe également un coût relatif à une pénalité de retard. L'évaluation de ces deux coûts est importante pour la prise d'une telle décision.

Nous supposons qu'une certaine évaluation des coûts est derrière la formulation des objectifs qui sont devenus classiques dans la littérature sur l'ordonnancement. Ceci se faisait de façon intuitive et presque automatique en pensant que ces objectifs reflètent vraiment ce que ciblent les gestionnaires et qu'ils contiennent les coûts importants.

7.3.2. Échelles de temps et le lien avec les problèmes d'ordonnancement intégré

Han et McGinnis (1989) citent que les travaux passent 5% de leur durée de séjour en atelier sur les machines et le restant du temps est souvent attribué aux activités de manutention et aux attentes.

L'intérêt pour étudier les problèmes d'ordonnancement intégré a été justifié chez quelques auteurs (Sabuncuoglu 1992) par le fait que les temps de traitement et les temps de manutention sont devenus comparables suite à l'avancement technologique dans les machines qui sont devenues très rapides et donc, ces deux données peuvent largement influencer les horaires de production. En effet, cet argument est vrai dans la mesure où

le lot de production reste relativement petit. Ce ratio peut largement changer avec différentes tailles de lots. C'est surtout le fait que les ressources en équipements de manutention soient contraignantes qui justifie le choix de l'ordonnancement intégré. Dans cet ordre d'idée, nous avons opté pour un choix ajusté des données de manutention dans la validation de nos modèles. Nous avons donc considéré que les temps de manutention sont inférieurs aux temps de production.

D'autre part, la vitesse des chariots détermine le temps requis pour parcourir les segments et donc elle guide la décision sur le niveau de raffinement qu'il faut considérer au niveau de la discrétisation du réseau de manutention. Les niveaux de précision sont à déterminer selon l'environnement de production.

7.4. Conclusion

Tel que présenté dans les chapitres précédents, tous les modèles développés ont l'objectif classique de minimiser la date de fin de l'horaire. Vu la complexité du problème, il a été important de franchir un premier pas dans la direction de résoudre un problème plus complet. Nous avons donc choisi d'étudier ces problèmes en considérant cet objectif. Cette étape était nécessaire pour une meilleure connaissance du problème et pour tester de façon pratique le potentiel des méthodologies disponibles. La contribution présentée dans cette thèse, présente un début de travaux dans un domaine peu étudié. Les objectifs de coûts discutés dans ce chapitre pourront faire l'objet des recherches futures et aideront à proposer des formulations encore plus réalistes pour le problème d'ordonnancement.

CHAPITRE 8 :

CONCLUSION

Les problèmes d'ordonnancement en production sont très complexes. Beaucoup de méthodes ont été développées pour les résoudre. Pourtant, ces problèmes tels qu'ils sont posés dans la littérature ne correspondent pas à ce que l'on trouve dans les vraies usines (Browne et al. 1981). Dans l'ensemble, il existe trois paradigmes pour aborder les problèmes d'ordonnancement : le paradigme de l'optimisation dont des variations sont la simulation et l'intelligence artificielle, le paradigme de traitement de données et le paradigme de contrôle (Dugan et Browne 1991). Nos travaux s'insèrent dans le paradigme d'optimisation. L'objectif était d'en faire une utilisation plus signifiante pour les industriels.

Le but de cette thèse était de proposer des formulations plus réalistes pour le problème d'ordonnancement en production et de les résoudre. Cela implique alors l'intégration des ressources habituellement négligées dans le traitement du problème. Nous avons considéré dans notre recherche, en plus des machines, le réseau de manutention et les ressources qui y sont rattachées : les chariots, les segments de routes, les intersections et les files d'attente comme étant contraignantes.

Pour arriver à proposer une formulation réaliste, nous avons procédé à l'analyse des différentes formulations approximatives dans la littérature pour en évaluer le potentiel et l'applicabilité à notre problème. Nous avons proposé, par la suite, des formulations détaillées passant par trois étapes. Trois problèmes d'ordonnancement intégré de la production et de la manutention, où la définition des ressources contraignantes est différente, ont été formulés et résolus. Chacun des trois problèmes est une contribution importante où nous résolvons pour la première fois à l'optimalité la problématique d'ordonnancement intégré de la production et de la manutention.

Dans le premier modèle, nous considérons un plus court chemin pour les missions de manutention. Seuls, les chariots dans le système de manutention sont considérés comme ressources contraignantes. Dans le deuxième problème, nous avons introduit un ensemble de routes prédéterminées entre chaque paire de machines. Ces routes sont utilisées par les tâches de manutention ou les voyages à vide. Évidemment, il existe des routes conflictuelles qui partagent des segments. Dans ce cas, les conflits sont évités en mobilisant toute la route pendant qu'elle accueille une tâche de manutention ou un voyage à vide.

Le troisième problème, présenté dans cette thèse, est le plus détaillé et complet. C'est la première fois dans la littérature qu'on présente un modèle aussi riche et représentatif de l'environnement de production dans un atelier multigamme. Il considère les segments de routes comme étant des ressources distinctes. Les chemins pour effectuer les tâches de manutention ou les voyages à vide sont construits successivement et déterminés par la solution du problème. Chaque machine est dotée d'une file d'attente pouvant accueillir quatre palettes. Les entrepôts de matières premières et de produits finis sont également pris en compte. La formulation du problème peut également être adaptée pour représenter un entrepôt de produits en cours de production. Il s'agira d'un poste dans l'atelier pour lequel la contrainte de disjonction ne sera pas écrite. Ce poste ne présente aucune offre ni demande de travaux. Les chariots peuvent y passer une durée d'attente indéterminée.

Deux formulations ont été présentées pour chaque problème : une en programmation linéaire mixte et l'autre en programmation par contraintes. Les deux premiers problèmes ont été résolus à l'optimalité dans des temps très courts. Les tests ont été effectués avec des données de la littérature. La performance des deux formulations proposées pour chaque problème était bonne et parfois complémentaire. Pour quelques jeux, la formulation linéaire mixte a mieux fonctionné et pour d'autres jeux l'inverse était vrai.

Le troisième modèle est plus intéressant au niveau de la représentation complète du problème. Mais il est beaucoup moins performant au niveau de la résolution. Les développements décrits au chapitre 6 constituent un début de travaux vers le formalisme et l'étude des problèmes plus complets en ordonnancement dans un contexte manufacturier. En même temps, ils témoignent de la difficulté du problème résultant. Cela dit, il ne faut pas simplement affirmer la difficulté des problèmes en proposant de les attaquer de façon moins rigoureuse et approximative.

Nous pensons qu'il serait intéressant de proposer aux industriels ce qu'on peut appeler une méthodologie de diagnostic d'un problème d'ordonnancement. Cette méthodologie doit comprendre une étape d'analyse et d'évaluation de la criticité des ressources pour mieux identifier les éléments nécessaires à inclure dans un premier traitement du problème. Vu les limites des technologies de résolution disponibles, l'intégration de toute la réalité dans une formulation semble intéressante pour le traitement de quelques cas spéciaux. Nous pensons notamment aux temps de traitement égaux et à des relations d'antériorité simples. Cela reste à confirmer à travers les tests.

Ce diagnostic doit se faire avec un moyen simple et efficace d'aide à la décision. Cette dernière précisant de quel problème **formel** il s'agit. Pour illustrer cette méthodologie, nous présentons un schéma (Figure 8.1). Il s'agit de répondre à trois questions mais avec beaucoup de précision.

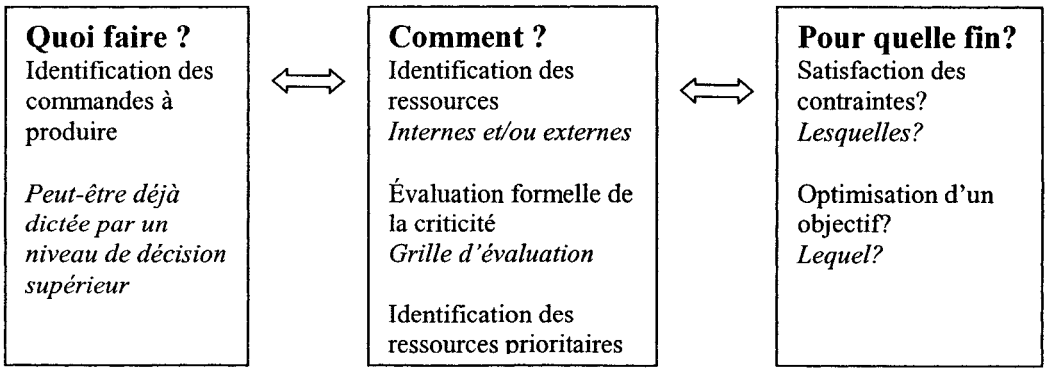


Figure 8.1. Méthodologie de diagnostic d'un problème d'ordonnancement

Cette méthodologie propose une simplification/décomposition du problème et éventuellement d'en traiter une partie en un deuxième temps, alors que le but de cette thèse était d'intégrer les décisions. Mais, cette décomposition se veut différente et réfléchie. En fait, une approche classique serait de formuler le problème dans son intégralité pour proposer par la suite une décomposition au niveau de la méthodologie de résolution. Dans ce cas, on cible plutôt les structures des modèles sans trop tenir compte des données telles que les temps, les ressources et les relations d'antériorité déterminant la criticité d'une ressource ou encore des phénomènes de criticité plus ponctuels. Une décomposition basée sur la définition du problème et l'analyse des données semble prometteuse et permet de ne pas sous-estimer ni sur-estimer le choix d'une méthode de résolution. En d'autres termes, cela permet de ne pas simplifier les modèles quand c'est pénalisant et de ne pas les complexifier pour rien quand ça ne semble pas porter fruit.

Cependant, la proposition d'une **grille d'évaluation de la criticité des ressources** pour un problème d'ordonnancement ne constitue pas une tâche facile et peut en soi faire le sujet d'une thèse. Cette évaluation doit donner une information rapide et pertinente sur la partie importante à considérer en premier lieu dans la résolution d'un problème difficile. Il ne s'agira pas de résoudre tout le problème pour le savoir. Il faut pouvoir mesurer cette criticité par des indicateurs quantifiables. Cette information aidera à proposer la formulation la plus appropriée pour un problème d'ordonnancement.

Nous pensons que partir d'une formulation intégrant les ressources les plus critiques est le premier déterminant dans la résolution efficace et satisfaisante d'un problème d'ordonnancement. Les ressources critiques sont évidemment différentes d'une réalité à une autre. Cela peut aussi offrir des pistes intéressantes sur le plan méthodologique. Parfois, la difficulté se trouve loin de là où l'on pense qu'elle existe.

Dans l'industrie, il se trouve des contextes où les machines sont les ressources les plus importantes, dans d'autres les opérateurs humains spécialisés peuvent être les plus

importants et dans d'autres les tâches de manutention peuvent occuper cette place. Les tâches de manutention peuvent encore se multiplier et devenir plus importantes dans le cas où l'on considère les **lots de transfert**. Ceci est un aspect complémentaire qui mérite l'étude dans les problèmes d'ordonnancement. Les contributions de la littérature font toujours l'hypothèse que les tailles des lots de production égalent les tailles des lots de transfert, ce qui n'est pas vrai dans plusieurs cas. Mais, avoir des lots de transfert différents des lots de production demeure un élément difficile qui n'a pas été intégré dans nos formulations et qui sera un élément à considérer dans des recherches futures.

Un autre aspect complémentaire dans l'étude d'un problème d'ordonnancement est l'utilisation des chariots pour manutentionner des ressources nécessaires au processus de production en plus des travaux en traitement. Le même système de manutention peut être utilisé pour la manutention des outils nécessaires pour le traitement des pièces ainsi que des montages. Des relations d'antériorité peuvent donc exister entre le traitement d'un travail sur une machine et la mission de manutention livrant des outils nécessaires pour le traitement. Ce scénario peut constituer le contexte d'un problème d'ordonnancement intéressant à l'étude. D'autres enrichissements comprennent les dates de disponibilité pour les travaux et les dates de disponibilité pour les machines.

L'étude des problèmes plus complets implique non seulement la bonne définition de la réalité, souvent faite à travers les contraintes, mais aussi la bonne formulation des objectifs, l'étape 3 dans la méthodologie de diagnostic proposée. Selon les modèles présentés dans cette thèse, un travail correspond à une commande et, donc, les objectifs peuvent être directement formulés en fonction des travaux. Dans une situation réelle, une commande provenant d'un client peut correspondre à plusieurs travaux dans l'atelier et, pour la livraison, le traitement de tous ces travaux doit être terminé. La formulation de ce type d'objectifs pourrait enrichir les modèles présentés dans la thèse. D'autres objectifs intéressants comprennent les objectifs juste-à-temps et les objectifs de coûts. Une décomposition et une définition des coûts pertinents en production sont présentées

dans cette thèse. Ces coûts servent à formuler des objectifs d'optimisation. De plus, nous discutons de la place qu'occupe l'objectif par rapport aux contraintes dans un problème d'ordonnancement et vice versa. Cette remise en cause aide à mieux comprendre et à mieux formaliser un problème d'ordonnancement auquel on fait face.

Bien que parfois pénalisant, les conditions particulières d'opération d'un système imposant des décisions d'affectation ou de routage a priori sont une réalité dans quelques contextes. Les chariots automatiques peuvent être dédiés à des machines ou bien à des travaux. Pour ce cas simplifié, nos modèles ont une meilleure performance.

Un aspect intéressant pour la communauté industrielle est l'utilisation de l'outil de résolution des problèmes d'ordonnancement intégré comme moyen d'analyse fiable dans la fixation des dates d'échéance réalistes. Les méthodes de fixation des dates d'échéance présentées dans la littérature considèrent uniquement les durées de traitement des travaux sur les machines. L'ordonnancement intégré pourrait aussi être utilisé comme un outil d'évaluation des besoins en machines ou en équipements de manutention (Jaikumar et Solomon 1990). Nos développements peuvent servir à cette fin.

Sur le plan méthodologique, les développements présentés dans cette thèse jettent la lumière sur des idées intéressantes au point de vue d'une hybridation éventuelle des méthodologies de résolution. Une même famille de contraintes a une performance différente selon la méthodologie utilisée : programmation mathématique ou programmation par contraintes. Ils offrent aussi des idées de modélisation intéressantes notamment pour les modèles de programmation par contraintes. Par exemple, pour la fabrication des tournées, les approches de modélisation sont très différentes d'une méthode à l'autre. On note aussi qu'en programmation par contraintes, les stratégies de recherche sont une piste très importante qui peut transformer radicalement la performance d'un modèle donné et qui mérite davantage d'études.

Les modèles de programmation par contraintes présentés dans cette thèse reposent beaucoup sur la nomenclature d'OPLStudio car au départ on voulait utiliser le solveur « Scheduler » qui fait partie du logiciel et qui est conçu pour résoudre les problèmes d'ordonnancement. Il serait intéressant de tester la performance des modèles plus standard du point de vue de la programmation par contraintes. Beaucoup d'emphase est à mettre aussi sur les structures de données. La nomenclature d'OPL permet des aspects intéressants tels qu'indicer des variables par des variables et donc de diminuer la taille des modèles mais cela est conditionné par le choix de la structure de données.

Les modèles en nombres entiers bénéficieront évidement des développements méthodologiques dans le domaine. Le dernier modèle restera très difficile pour des problèmes de taille réelle. Il comporte dans sa structure un problème d'ordonnancement de la production, de tournées de véhicules, un problème multiflux et un problème d'affectation qui ne sont pas facilement résolubles. Dans ce sens, on peut penser que l'utilisation des métaheuristiques peut être une voie intéressante pour la résolution de ce problème très détaillé.

Une piste de recherche très intéressante dans ce contexte est l'étude des séquences et des affectations possibles ainsi que le développement des critères de dominance pour quelques-unes de celles-ci. Cela peut réduire l'éventail de solutions et ainsi aider dans l'élaboration d'une stratégie de recherche.

Les problèmes présentés en nombre entiers ont une bonne performance pour le premier et le deuxième problème. Ils n'ont pas présenté des problèmes de précision numérique à cause de la grande valeur (H) utilisée dans la modélisation. Cela devient plus difficile dans le troisième modèle où beaucoup de conditions sont à respecter et où les contraintes de disjonction sont nombreuses pour plusieurs ressources. Au début des travaux, nous avons opté pour ce type de formulation car nous voulions éviter la non-linéarité et la génération des horaires a priori. D'autre part, nous avons évité d'utiliser un horizon

discrétisé pour diminuer le nombre de variables binaires dans la formulation. Un effort de modélisation est à fournir pour améliorer la performance du modèle.

Dans les tests effectués dans la thèse, nous avons considéré le cas statique où tous les travaux sont disponibles au début de l'horizon. Le cas dynamique est aussi très important et mérite d'être étudié. De nos jours, les ateliers de production flexibles sont conçus pour faire face à un environnement de production où les commandes sont de plus en plus variées. Ces ateliers proposent de traiter des lots de production moyens et petits. La tendance est vers une production plus personnalisée et une fabrication sur commande de clients. Même au niveau des grands ateliers d'assemblage, les commandes d'assemblage sont reçues de la direction de façon dynamique (Agnetis et al. 1997). Nos modèles pour le premier et le deuxième problèmes sont facilement utilisables dans un contexte dynamique vu la rapidité de l'outil développé. De plus, nous sommes en mesure de trouver une solution rapide en résolvant un problème de satisfaction de contraintes, même quand une solution optimale s'avère difficile à atteindre.

Beaucoup d'auteurs se sont intéressés à l'étude des problèmes d'ordonnancement dans les environnements dynamiques où la production se fait sur commande de clients. Parmi plusieurs, nous citons Hendry et Kingsman (1991), Fuméro et Vercellis (1994), Valério de Carravillas et Guimaraes Rodrigues (1995). Ces contributions accordent peu de poids à la fonction d'ordonnancement. Nos développements présentent une contribution qui fournit à l'industriel un moyen plus fiable et plus exact en intégrant plusieurs ressources pour répondre aux besoins des clients, spécialement dans un environnement dynamique où les activités de planification et d'ordonnancement se chevauchent tel que discuté au chapitre 7.

Nous avons remarqué que l'horizon de temps considéré dans les différentes contributions de la littérature n'est pas défini de façon précise. Les périodes durant lesquelles se déroulent les activités de production sont vaguement rapportées. Tel que

présenté au chapitre 7, dans des recherches futures, il faudra proposer des horaires tels que leurs longueurs soient signifiantes pour une implantation en atelier avec toutes les contraintes imposées par le personnel, la sécurité, les jours fériés ou autres. Cela constitue une piste très intéressante surtout que dans les problèmes d'horaires on ne connaît pas a priori la longueur éventuelle qu'on va avoir une fois qu'un problème d'ordonnancement est résolu.

Nous avons essayé, dans cette thèse, de joindre la théorie à la pratique. Malgré le fait que ces deux pôles soient assez distants, nous pensons que cet exercice est primordial pour le raffinement des définitions des problèmes et pour mieux orienter les développements méthodologiques. Il est clair qu'il restera toujours beaucoup de pas à faire entre la pratique et la théorie. Mais, c'est dans les essais de rapprochement qu'on peut avoir des découvertes scientifiques intéressantes.

RÉFÉRENCES

ADAMS, J., BALAS, E. et ZAWAK, D. (1988). The shifting bottleneck procedure for job shop scheduling. *Management Science,* 34/3, 391-401.

AGNETIS, A., PACCIARELLI, D., et ROSSI, F. (1994). Lot scheduling in a two-machine cell with swapping devices. *IIE Transactions 28*

AGNETIS, A., PACIFICI, A., ROSSI, F., LUCERTINI, M., NICOLETTI, S., NICOLÒ, F., ORIOLO, G., PACCIARELLI, D. et PESARO, E. (1997). Scheduling of flexible flow lines in an automobile assembly plant. *European Journal of Operational Research,* 97/2,348-362.

AGRAWAL, A. HARHALAKIS, G. MINIS, I. et NAGI, R. (1996), Just in time production in large assemblies. *IIE Transactions, 28*, 653-667.

AKTURK, M. S. et YILMAZ, H. (1996). Scheduling of automated guided vehicles in a decision making hierarchy. *International Journal of Production Research,* 34, 577-591.

ANEJA, Y. P. et KAMOUN, H. (1999). Scheduling of parts and robot activities in a two machine robotic cell. *Computers and Operations Research*, 26/4,297-312.

ANWAR, M. F. et Nagi, R. (1997). Integrated conflict free routing of AGVs and workcenter scheduling in a just in time production environment. Industrial Engineering Research – Conference Proceedings. Proceedings of the 1997 6th annual Industrial Engineering Research Conference, IERC May 17-18 1997. 1997 Miami Beach, FL, USA, IIE Norcross GA USA p 216-221 IERCE9.

ANWAR, M. F. et Nagi, R. (1998). Integrated Scheduling of material handling and manufacturing activities for just in time production of complex assemblies. *International Journal of Production Research,* 36/3, 653-681.

AZIZOGLU, M. et KIRCA, O. (1999). On the minimization of the total weighted flow time with identical and uniform parallel machines. *European Journal of Operational Research,* 113/1, 91-100.

AZRI,Y. (1995). On line scheduling in a multi-cell flexible manufacturing system. *International Journal of Production Research,* 33/12, 3267-3300.

BAGCHI, U. (1989). Simultaneous minimization of mean and variation of flow time and waiting time in single machine systems. *Operations Research,* 37, 118-125.

BAGCHI, U., JULIEN, F. M. et MAGAZINE, M.J. (1994). Note : Due-date assignment to multi-job customer orders. *Management science,* 40/10, 1389-1392.

BAKER, K.R. (1974). *Introduction to sequencing and scheduling*, John Wiley & Sons, New York.

BAKER, K.R. et MERTEN, A. G. (1973). Scheduling with parallel processors and linear delay costs. *Naval Research Logistic Quarterly*, 20, 193-204.

BAKER, K.R. et SCHRAGE, L.E. (1978). Finding an optimal sequence by dynamic programming: An extension to precedence-related tasks. *Operations Research*, 26/1, 111-121.

BALAKRISHNAN, N., KANET, J. et SRIDHARAN, S. V. (1999). Early/Tardy scheduling with sequence dependent setups on uniform parallel machines. *Computers and Operations Research*, 26/2, 127-141.

BALAKRISHNAN, N., SRIDHARAN, S.V. et PATTERSON, J.W. (1996). Rationing capacity between two product classes, *Decision Sciences*, 27/2, 185-214.

BALAS, E., LENSTRA, J.K. et VAZACOPOULOS, A. (1995). The one machine problem with delayed precedence constraints and its use in job-shop scheduling, *Management Science,* 41/1, 94-109.

BAPTISTE, P. LE PAPE, C. et NUIJTEN, W. (2001). *Constraint-Based Scheduling - Applying Constraint Programming to Scheduling Problems.* Kluwer Academic Publishers.

BARNES, J.W. (1993). A tabu search experience in production scheduling. *Annals of Operations Research,* 41, 141-156.

BARTHOLDI, J. J., et PLATZMAN, L. K. (1989). Decentralized control of automatic guided vehicles on a simple loop. *IIE Transactions*, 21, 76-81.

BEASLEY, J. E., OR-Library test problems, http://mscmga.ms.ic.ac.uk/info.html

BELL, C. E. et HAN, J. (1991). A new heuristic solution method in resource constrained project scheduling. *Naval Research Logistics*, 38/3, 315-331.

BELL, C. E. et PARK, K. (1990). Solving resource constrained project scheduling problems by A* search, *Naval Research Logistics*, 37, 61-84.

BELSLEY, D. A. (1969). *Industry Production Behavior : The Order-Stock Distinction*, North-Holland Publishing Company, Amsterdam.

BILGE, U. et ULUSOY, G. (1995). A time window approach to simultaneous scheduling of machines and material handling system in an FMS. *Operations Research*, 43/6, 1058-1070.

BISKURP, D. et CHENG, T.C.E. (1999). Multiple machine scheduling with earliness, tardiness and completion time penalties. *Computers and Operations Research,* 26/1, 45-57.

BLAIR, E. L., CHARNSETHIKUL, P. et VASQUES, A. (1987). Optimal routing of driverless vehicles in a flexible material handling system. *Material Flow*, 4, 73-83.

BLAZEWICZ, J., BURKARD, R. E., FINKE, G. et WOEGINGER, G. J. (1994). Vehicle scheduling in two-cycle flexible manufacturing systems. *Mathematical and Computer Modelling*. 20, 19-31.

BLAZEWICZ, J., DROR, M., et WEGLARZ, J. (1991). Mathematical programming formulations for machine scheduling. A Survey. *European Journal of Operational Research*, 51/3, 283-300.

BLAZEWICZ, J., EISELT, H. A., FINKE, G., LAPORTE, G. et WEGLARZ, J. (1991). Scheduling tasks and vehicles in a flexible manufacturing system. *International Journal of Flexible Manufacturing Systems*, 4, 5-16.

BLAZEWICZ, J., SLOWINSCKI, R., WEGLARZ, J. et CELLARY, W. (1986). Scheduling under resource constraints- Deterministic Models. *Annals of Operations Research*, 7.

BODIN, L. et GOLDEN, B. (1981). Classification in vehicle routing and scheduling. *Networks*, 11/2, 97-108.

BOWMAN, E.H. (1959). The schedule-sequencing problem. *Operations Research*, 7, 621-624.

BOZER, Y. A. et SRINIVASAN, M. M. (1991). Tandem configurations for automated guided vehicle systems and analysis of single vehicle loops. *IIE Transactions*, 23, 72-82.

BOZER, Y. A. et SRINIVASAN, M. M. (1992). Tandem AGV systems: a partitioning algorithm and performance comparison with conventional AGV systems. *European Journal of Operational Research*, 63, 173-191.

BOZER, Y. et YEN, C.-K. (1996). Intelligent dispatching rules for trip-based material handling systems. *Journal of Manufacturing Systems*, 15/4, 226-239.

BRAH, S. A. et LOO, L. L. (1999). Heuristics for scheduling in a flow shop with multiple process. European Journal of Operational Research, 113/1, 113-122.

BROWNE, J., BOON, J. E. et DAVIES, B. J. (1981). Job shop control. *International Journal of Production Research*, 19/6, 633-643.

BRUCKER, P., DREXEL, A., MOEHRING, R., NEUMANN, K., et PESCH, E. (1999). Resource-constrained project scheduling: Notation, classification, models and methods. European Journal of Operational Research, 112/1, 3-41.

BUGNON, B., STOFFEL, K. et WIDMER, M. (1995). FUN : A dynamic method for scheduling problems. *European Journal of Operations Research,* 83, 271-282.

BU-HULAIGA, M. I., et CHAKRAVARTY, A. K. (1988). An object-oriented knowledge representation for hierarchical real-time control of flexible manufacturing. *International Journal of Production Research,* 26/5, 777-793.

CARLIER, J. (1982). The one-machine sequencing problem. *European Journal of Operational Research*, 11, 42-47

CARLIER, J. et CHRÉTIENNE, Ph. (1988). *Les Problèmes d'ordonnancement,* Masson, Paris.

CARRAVILLA, M. A. et DE SOUSA, J.P. (1995). Hierarchical production planning in a make-to- order company : A case study. *European Journal of Operational Research*, 86/1, 43-56

CATTRYSSE, D.; SALOMON, M.; KUIK; R.; et VAN WASSENHOVE, L.N. (1993). A dual ascent and column generation heuristic for the discrete lotsizing and scheduling problem with setup times. *Management Science,* 39/4, 477-486.

CHANG, S.-H. et EGBELU, P.J. (1995). Dynamic relative positioning of AGVs in a loop layout to minimize the mean system response time. Working paper, Department of Industrial and Manufacturing Engineering, Pennsylvania State University.

CHARLTON, J.M. et Death, C.C. (1970). A generalized machine scheduling algorithm. *Operational Research Quarterly*, 21/1, 127-134.

CHEN, C.-L. et BULFIN, R. L. (1993). Complexity of single machine, multi-criteria scheduling problems. *European Journal of Operational Research*, 70/1, 115-125.

CHENG, T. C. E. (1987). A simulation study of automated guided vehicle dispatching. *Robotics & Computer-Integrated Manufacturing*, 3, 335-338.

CHENG, T. C. E. et DING, Q. (1998). Complexity of single machine scheduling with two distinct deadlines and identical decreasing rates of processing times. *Computers & amp; Mathematics with Applications*, 35/12, 95-100.

CHIU, C. et YIH, Y. (1995). A learning-based methodology for dynamic scheduling in distributed manufacturing systems. *International Journal of Production Research,* 33/11, 3217-3232.

CHOI, H.-G., KWON, H.-J. et LEE, J. (1994). Traditional and tandem AGV system layouts: a simulation study. *Simulation*, 63/2, 85-93.

CHUNG, Y. et FISCHER, G. W. (1995). A neural algorithm for finding the shortest flow path for an automated guided vehicle system. *IIE Transactions*, 27, 773-783.

CO, C. G. et TANCHOCO, J. M. A.. (1991). A review of research on AGVS vehicle management. *Engineering Costs and Production Economics*, 21,35-42.

CONWAY, R.W., MAXWELL, W.L. et MILLER, L.W. (1967). *Theory of Scheduling*. Addison-Wesley, Reading, Mass.

CORNER, J.L., FOULDS, L.R., et NEUMANN, K. (1997). Heuristics and applications for resource-constrained project scheduling with minimal and maximal time lags. *INES'97: 1997 IEEE International conference on Intelligent Engineering Systems. Proceedings*, 403-408.

CRAMA, Y. (1997). Combinatorial optimization models for production scheduling in automated manufacturing systems. *European Journal of operational research*, 99/1, 136-153.

CRAMA, Y. et VAN DE KLUNDERT, J. (1997). Cyclic scheduling of identical parts in a robotic cell. *Operations Research*, 45/6, 952-965.

DANNENBRING, D. G. (1977). An evaluation of flowshop sequencing heuristics. *Management Science*, 23, 1174-1182.

DANNENBRING, D. G. (1977). An evaluation of flowshop sequencing heuristics. *Management Science*, 23, 1174-1182.

DAVIS, E.W. (1973). Project scheduling under resource constraints- Historical review and categorization of procedures. *AIIE Transactions*, 5/4, 297-313.

DAVIS, E.W. et PATTERSON, J.H. (1975). A comparison of heuristic and optimum solutions in resource constrained project scheduling. *Management Science*, 21/8, 944-955.

DAY, J.E. et HOTTENSTEIN, M.P. (1970). Review of sequencing research. *Naval Research Logistic Quarterly*, 17,11-40.

DELL'AMICO, M. et TRUBIAN, M. (1993). Applying tabu search to the job shop scheduling problem. *Annals of Operations Research*, 41, 231-252.

DELLAERT, N. P. et MELO, M. T. (1996). Production strategies for a stochastic lot-sizing problem with constant capacity. *European Journal of Operational Research*, 92/2, 281-301.

DESROCHERS, M., LENSTRA, J. K., SAVELSBERGH, M. W. P. et SOUMIS, F. (1988). Vehicle Routing with time windows. In *Vehicle Routing: Methods and Studies,* B. L. Golden and A. A. Assad (Eds), North-Holland, Amsterdam, 65-84.

DHOUIB, K. et KADI, D. A. (1994). Expert system for AGV managing in bi-directional networks: KADS methodology based approach. *International Journal of Production Economics*, 33, 31-43.

DIJKSTRA, E. W. (1959). A note on two problems in connexion with graphs. *Numerische Mathematik*, 1, 269-271.

DONATH, M. et GRAVES, R. J. (1988). Flexible assembly systems : an approach for near real time scheduling and routing of multiple products. *International Journal of Production Research,* 26/12, 1903-1919.

DOWSLAND, K. A. et GREAVES, A. M. (1994). Collision avoidance in bi-directional AGV systems. *Journal of the Operational Research Society*, 45, 817-826.

DUGGAN, J. et BROWNE, J. (1991). Production activity control: a practical approach to scheduling. *The International Journal of Flexible Manufacturing Systems,* 4, 79-103.

EGBELU, P. J. (1987). Pull versus Push strategy for automated guided vehicle load movement in a batch manufacturing system. *Journal of Manufacturing Systems*, 6, 209-221.

EGBELU, P. J. (1993). Positioning of automated guided vehicles in a loop layout to improve response time. *European Journal of Operational Research*, 71, 32-44.

EGBELU, P. J. et ROY, N. (1988). Material flow control in AGV/unit load based production lines. *International Journal of Production Research*, 26, 81-94.

EGBELU, P. J. et TANCHOCO, J. M. A. (1984). Characterization of automatic guided vehicle dispatching rules. *International Journal of Production Research*, 22, 359-374.

EGBELU, P. J. et TANCHOCO, J. M. A. (1986). Potentials for bi-directional guide-path for automated guided vehicle based systems. *International Journal of Production Research*, 24, 1075-1097.

EGLESE, R. (1990). Simulated annealing: A tool for operational research. *European Journal of Operational Research.* 46/2, 271-281.

ELMAGHRABY, S. E. (1978). The economic lot scheduling problem (ELSP): review and extensions. *Management Science,* 24/6, 587-598.

ESAIGNANI, S., HALL, N. G. et SRISKANDARAJAH, C. (1999). Scheduling and lot streaming in two-machine no-wait openshops. Journées de l'Optimisation 1999, Montréal, 10-12 Mai.

FEO, T. A., BARD, J.F. et HOLLAND, S. D. (1995). Facility-wide planning and scheduling of printed wiring board assembly. *Operations Research*, 43/2, 219-230.

FERLAND, J. A. et FORTIN, L. (1989). Vehicles scheduling with sliding time windows. *European Journal of Operational Research*, 38, 213-226.

FRANCK, B., NEUMANN, K. et SCHWINDT, C. (1997). A capacity- oriented hierarchical approach to single-item and small-batch production planning using project scheduling methods. *OR Spektrum*, 19/2, 77-85.

FRENCH, S. (1982). Sequencing and scheduling. Ellis Horwood Limited, England.

FUJII, S., et SANDOH, H. (1987). A routing algorithm for automated guided vehicles in FMS. In *Proceedings of IXth International Conference on Production Research*, II, 2261-2267.

FUJII, S., SANDOH, H. et HOHZAKI, R. (1988). Routing control of automated guided vehicles in FMS. In *Proceedings of the USA-Japan Symposium on Flexible Automation, Minneapolis, MN, Juillet 1988, 629-636.*

FUMERO, F. et VERCELLIS, C. (1994). Capacity analysis in repetitive assemble-to-order manufacturing systems. *European Journal of Operational Research*, 78, 204-215.

GANESHARAJAH, T., HALL, N. G. et SRISKANDARAJAH, C. (1996a). AGV-served manufacturing systems: scheduling and design in loop layouts. Working paper # 96-03, Department of Industrial Engineering, University of Toronto.

GANESHARAJAH, T., HALL, N. G. et SRISKANDARAJAH, C. *(1998). Design and operational issues in AGV-served manufacturing systems.* Annals of Operations Research, *76, 109-154.*

GAREY, M. R., JOHNSON, D. S. et SETHI, R. (1976). The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, 1, 117-129.

GEIGER, C.D., KEMPF, K. G. et UZSOY, R. (1997). A Tabu search approach to scheduling an automated wet etch station. *Journal of Manufacturing Systems*, 16/2, 102-116.

GÉLINAS, S. (1997). *Problèlmes d'ordonnancement.* Thèse de doctorat, École Polytechnique de Montréal, Canada.

GERE, W.S. (1966). Heuristics in job shop scheduling. *Management Science*, 13, 167-190.

GIFFLER, B. et Thompson, G. L. (1960). Algorithms for solving production scheduling problems. *Operations Research*, 8, 478-503.

GILMORE, P. C. et GOMORY, R. E. (1964). Sequencing a one-state variable machine: a solvable case of the travelling salesman problem. *Operations Research*, 12, 665-679.

GLASS, C. A., GUPTA, J. T. N. et POTTS, C. N. (1994). Lot streaming in three-stage production processes *European Journal of Operational Research,* 75/3, 378-394.

GLASS, C.A. (1992). Feasibility of scheduling lot sizes of three products on one machine. *Management Science,* 38/10, 1482-1494.

GLOVER, F. (1987). *Tabu Search,* Technical Report, Center for Applied Artificial Intelligence, University of Colorado, Boulder, CO.

GOLDEN, B. L., et ASSAD, A. (1986). Perspectives on vehicle routing: exciting new developments. *Operations Research*, 34, 803-810.

GRAHAM, R.L.; LAWLER, E.L., LENSTRA, J.K. et RINNOOY KAN, A.H.G. (1979). Optimization and approximation in deterministic sequencing and scheduling : A survey. *Annals of Discrete Mathematics*, 5, 287-326.

GRASSO, V. (1990). AGV-served assembly lines : influence of sequencing and launch policies on system performances. *International Journal of Production Research*, 28, 1385-1399.

GRAVES, S.C. (1981). A review of production scheduling. *Operations Research,* 29, 646-675.

GUPTA, J.N.D. et HARIRI, A. M. A. (1997). Two-machine flowshop scheduling to minimize the number of tardy jobs. *Journal of the Operational Research Society*, 48, 212-220.

HAINES, C. L. (1985). An algorithm for carrier routing in a flexible material handling system. *IBM Journal of Research and Development*, 29/4, 356-362.

HALL, N. G. et SRISKANDARAJAH, C. (1996). A survey of machine scheduling problems with blocking and no-wait in process. *Operations Research,* 44/3, 510-525.

HALL, N. G., KAMOUN, H. et SRISKANDARAJAH, C. (1995). Scheduling in robotic cells: complexity and steady state analysis. Working paper, Fisher college of Business, The Ohio state University.

HALL, N. G., KAMOUN, H. et SRISKANDARAJAH, C. (1997). Scheduling in robotics cells: classification, two and three machine cells. *Operations Research*, 45/3, 421-439.

HALL, N. G., SETHI, S. P., et SRISKANDARAJAH, C. (1991). On the complexity of generalized due date scheduling problems. *European Journal of Operational Research*, 51/1, 100-109.

HAN, M.-H., et McGINNIS, L. F. (1989). Control of material handling transporter in automated manufacturing. *IIE Transactions*, 21, 184-190.

HAO, G. et LAI, K.K. (1996). Solving the AGV problem via a self-organizing neural network. *Journal of the Operational Research Society*, 47/12, 1477-1493.

HAO, G., SHANG, J.S., et VARGAS, L.G. (1995). A neural network model for on-line control of flexible manufacturing systems. *International Journal of Production Research,* 33/10, 2835-2854.

HAX, A. C. et Meal, H. (1975). Hierarchical Integration of production planning and scheduling. In *Logistics* (M. A. GEISLER, Ed.), 53-69. North Holland, Amsterdam.

HENDRY, L.C. et KINGSMAN, B.G. (1991). Job release: Part of a hierarchical system to manage manufacturing lead times in make-to-order companies. *Journal of the Operational Research Society,* 42/10, 871-883.

HENDRY, L.C.; KINGSMAN, B.G. et CHEUNG, P. (1998). The effect of workload control (WLC) on performance in make-to-order companies. *Journal of Operations Management*, 16/1, 63-75.

HERTZ, A. et WIDMER, M. (1996). An improved tabu search approach for solving the job shop scheduling problem with tooling constraints. *Discrete Applied Mathematics.* 65, 319-345.

HO, J.C. et CHANG, Y-L. (1991). Heuristics for minimizing mean tardiness for m parallel machines. *Naval Research Logistics*, 38, 367-381.

HOCHBAUM, D.S. et LANDY, D. (1997). Scheduling semi-conductor Burn-in operations to minimize total flowtime. *Operations Research*, 45/6, 874-885.

HODGSON, T. J., KING, R. E., MONTEITH, S. K. et SCHULTZ, S. R. (1987). Developing control rules for an AGVS using Marcov decision processes. *Material Flow*, 4, 85-96.

HOFF, E. B. et SARKER, B. R. (1998). An overview of path design and dispatching methods for automated guided vehicles. *Integrated Manufacturing Systems*, 9/5, 296-307.

HOLLAND, J. H. (1975). *Adaptation in natural and artificial systems*, University of Michigan Press, Ann Arbor, MI.

HOLLOWAY, C. A. et NELSON, R. T. (1973). Alternative formulation of the job shop problem with due dates. *Management Science,* 20/1, 65-75.

HOLTSCLAW, H.H. et UZOY, R. (1996). Machine criticality measures and subproblem solution procedures in shifting bottleneck methods: A computational study. *Journal of the Operational Research Society*, 47, 666-677.

HUANG, J., PALEKAR, U. S. et KAPOOR, S. G. (1993). A labeling algorithm for the navigation of automated guides vehicles. *ASME Transactions Journal of Engineering for Industry*, 115, 315-321.

HUSSAIN, M. F. et JOSHI, S. B. (1998). Genetic algorithm for job shop scheduling problems with alternate routing. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics. Proceedings of the 1998 IEEE International Conference on Systems, Man and Cybernetics. Part 3, Oct 11-14 ,1998*, 3, San Diego, CA, USA, p 2225-2230.

HWANG, H. et KIM, S. H. (1998). Development of dispatching rules for automated guided vehicle systems. *Journal of Manufacturing Systems*, 17/2, 137-143.

ICMELI, O. et ERENGUC, S. S. (1994). A tabu search procedure for the resource constrained project scheduling problem with discounted cash flows. *Computers and Operations Research*, 21, 841-853.

IOACHIM, I. et SANLAVILLE, E. (1996). The basic cyclic scheduling model for a robotic production cell. *Les Cahiers du Gérad*, G-94-15.

IOACHIM, I. et SOUMIS, F. (1995). Schedule efficiency in a robotic production cell. *International Journal of Flexible Manufacturing Systems*, 7/1, 5-26.

ISHIBUCHI, H., MISAKI, S. et TANAKA, H. (1995). Modified simulated annealing algorithms for the flow shop sequencing problem. *European Journal of Operational Research,* 81/2, 388-398.

ISLAM, A. et EKSIOGLU, M. (1997). A tabu search approach for single machine mean tardiness problem. *Journal of the Operational Research Society,* 48, 751-755.

JACKSON, J. R. (1956). An extension of Johnson's results on job lot scheduling. *Naval Research Logistics Quarterly,*3, 201-203.

JAIKUMAR, R. et SOLOMON, M.M. (1990). Dynamic scheduling of automated guided vehicles for a certain class of systems. *Journal of Manufacturing Systems.* 9/4, 315-323.

JAIKUMAR, R. et VAN WASSENHOVE, L. N. (1989). Production planning in flexible manufacturing systems. *Journal of Manufacturing and Operations Management,* 2, 52-78.

JAIN, A. S. et MEERAN, S. (1999). Deterministic job-shop scheduling: Past, present and future. *European Journal of Operational Research,* 113/2, 390-434.

JAIN, V. and GROSSMANN, I.E. (2001). Algorithms for Hybrid MILP/CP Models for a Class of Optimization Problems. *INFORMS Journal of Computing.* 13/4.

JAW, J., ODONI, A. R., PSARAFTIS, H. N., et WATSON, N. H. M. (1986). A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research,* 20/series B, 243-257.

JAW, J., ODONI, A. R., PSARAFTIS, H. N., et WATSON, N. H. M. (1986). A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research,* 20/series B, 243-257.

JAWAHAR, N., ARAVINDAN, P., PONNAMBALAM, S. G., SURESH, R. K. (1998). AGV schedule integrated with production in flexible manufacturing systems. *International Journal of Advanced Manufacturing Technology*, 14/6, 428-440.

JOHNSON, S.M. (1954). Optimal two and three stage production schedules with setup times included. *Naval Research Logistics Quarterly*, 1, 61-67.

KAMOUN, H. et SRISKANDARAJAH, C. (1993). Complexity of scheduling jobs in repetitive manufacturing systems. *European Journal of Operational Research*, 70/3, 350-364.

KAMOUN, H., HALL, N. G. et SRISKANDARAJAH, C. (1993). Scheduling in robotic cells: heuristics and cell design. Working paper # 93-08, Department of Industrial Engineering, University of Toronto.

KAO, E.P.C. et QUEYRANNE, M. (1982). On dynamic programming methods for assembly line balancing. *Operations Research,* 30/2, 375-390.

KARP, R.M. (1972). Reducibility among combinatorial Problems, dans *Complexity of Computer Computations*. R. E. MILLER et J.W. THATCHER (eds.), 85-103. Plenum Press, New York.

KHEMELNITSKY, E., KOGAN, K.; et MAIMON, O. (1995). Optimal flow control for continuous-time scheduling in flexible manufacturing systems. *International Transactions in Operational Research*, 2/4, 331-9.

KIM, C. W. et TANCHOCO, J. M. A. (1991). Conflict-free shortest time bi-directional AGV routeing. *International Journal of Production Research*, 29, 2377-2391.

KIM, C. W. et TANCHOCO, J. M. A. (1993). Operational control of a bidirectional automated guided vehicle system. *International Journal of Production Research*, 31, 2123-2138.

KIM, K. H. et KIM, J. Y. (1997). Estimating mean response time and positioning idle vehicles of automated guided vehicle systems in loop layout. Computers and industrial engineering, 33/3-4, 669-672.

KING, R. E., HODGSON, T.J. et CHAFEE, F. W. (1993). Robot task scheduling in a flexible manufacturing cell. *IIE Transactions*, 25/2, 80-87.

KINGSMAN, B., HENDRY, L., MERCER, A., et DE SOUZA, A. (1996). Responding to customer enquiries in make -to-order companies: Problems and solutions. *International Journal of Production Economics,* 46/47, 219-31.

KINGSMAN, B.G., et MERCER, A. (1997). Strike rate matrices for integrating marketing and production during the tendering process in make-to-order subcontractors. *International Transactions in Operational Research*, 4/4, 251-257.

KINGSMAN, B.G., TATSIOPOULOS, I. P. et HENDRY, L.C. (1989). A structural methodology for managing lead times in make to order companies. *European Journal of Operational Research*, 40, 196-209

KIRCA,O. et KOKSALAN, M.M. (1996). An integrated production and financial planning model and an application. *IIE Transactions* , 28/8, 677-686.

KISE, H., SHIOYAMA, T. et IBARAKI, T. (1991). Automated two-machine flowshop scheduling: a solvable case. *IIE Transactions*, 23, 10-16.

KLEIN, C. M. et KIM, J. (1996). AGV dispatching. *International Journal of Production Research*, 34, 95-110.

KOGAN, K. (1997). Discrete event control of production flows : make-to-stock and make to order environments. *International journal of production research*, 35/6, 1729-41.

KOLEN, A. W. J., RINNOOY KAN, A. H. G., et TRIENEKENS, H. W. J. M. (1987). Vehicle routing with time windows. *Operations Research*, 30, 1327-1351.

KOLISCH, R. and DREXEL, A. (1997). Local search for nonpreemptive multi-mode resource-constrained project scheduling. *IIE Transactions*, 29, 987-999.

KOLONKO, M. (1999). Some new results on simulated annealing applied to the job shop scheduling problem. *European Journal of Operational Research,* 113/1, 123-136.

KOULMAS, C., ANTONY, S. et JAEN, R. (1994). A survey of simulated annealing applications to operations research problems. *Omega International. Journal of Management Science.* 22/1, 41-56.

KRISHNAMURTHY, N. N., BATTA, R. et KARWAN, M. H. (1993). Developing conflict-free routes for automated guided vehicles. *Operations Research*, 41/6, 1077-1090.

KUMAR, N. S. HEMANT et SRINIVASAN, G. (1996). A genetic algorithm for job-shop scheduling- A Case study. *Computers in industry*, 31/2, 155-160.

KUSIAK, A. (1983). Scheduling of vehicles and algorithms. *In Proceedings of the 5th International Conference on Automation Warehousing. IFS Publications.* Institute of Industrial Engineers, 1983 ICAW Proceedings, 227-232.

KUSIAK, A. et CYRUS, P. (1985). Routing and scheduling of automated guided vehicles. In Proceedings of the 8th International Conference on Production Research and, 5th conference of the Fraunhofer-Institute for Industrial Engineering, Springer Verlag, 247-251.

KUTANOGLU, E. et SABUNCUOGLU, I. (1999). Analysis of heuristics in a dynamic job-shop with weighted tardiness objectives. *International Journal of Production Research*, 37/1, 165-187.

LAARHOVEN, P.J.M., AARTS, E.H.L. et LENSTRA, J.K. (1992). Job shop scheduling by simulated annealing. *Operations Research*, 40/1, 113-125

LAGEWEG, B.J., LENSTRA, J.K. et RINNOOY KAN, A.H.G. (1977). Job shop scheduling by implicit enumeration. *Management Science*, 24/4, 441-450.

LAMBRECHT, M.R., CHEN, S. et VANDAELE, N. J. (1996). A lot sizing models with queuing delays : The issue of safety time. *European Journal of Operational Research*, 89/2, 269-276.

LANGEVIN, A., LAUZON, D. et RIOPEL, D. (1996). Dispatching, routing and scheduling of two automated guided vehicles in a flexible manufacturing system. *International Journal of Flexible Manufacturing Systems*, 8, 247-262.

LANGEVIN, A., RIOPEL, D. et STECKE, K. E.(1999). Transfer batch sizing in flexible manufacturing systems, *International Journal of Flexible Manufacturing Systems*,18/2 , 140-151.

LASSERRE, J. B. (1992). An integrated model for job-shop planning and scheduling, *Management Science*, 38/8, 1201-1211.

LAWLER, E. L. (1979). *Efficient implementation of dynamic programming algorithms for sequencing problems.* Report BW 106, Mathematical Centrum, Amsterdam.

LAWLER, E.L. et MOORE, J.M. (1969). A functional equation and its application to resource allocation and sequencing problems. *Management Science,* 16/1, 77-84.

LEE, C.-C., et LIN, J. T. (1995). Deadlock prediction and avoidance based on Petri nets for zone-control automated guided vehicle systems. *International Journal of Production Research*, 33/12, 3249-3265.

LEE, D.Y. et DiCESARE, F. (1994). Integrated scheduling of flexible manufacturing systems employing automated guided vehicles," *IEEE Transactions on Industrial Electronics*, 41/6, 602-610.

LEE, J. et MANEESAVET, R. (1999). Dispatching rail-guided vehicles and scheduling jobs in a flexible manufacturing system. *International Journal of Production Research*, 37/1, 111-123.

LEE, J. H., LEE, B. H. et CHOI, M. H. (1998). Real-time traffic control scheme of multiple AGV systems for collision free minimum time motion: A routing table approach. *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, 28/3, 347-358.

LEE, J. K. et KIM, Y.D. (1966). Search heuristics for resource constrained project scheduling. *Journal of the Operational Research Society*, 47, 678-689.

LENSTRA, J. K. et RINNOOY KAN, A. H. G. (1978). Complexity of scheduling under precedence constraints. *Operations Research*, 26, 22-35.

LEUNG, J.Y-T. (1982). On scheduling independent tasks with restricted execution times, *Operations Research*, 30/1, 163-171.

LEVNER, E., KOGAN, K. et LEVIN, I. (1995). Scheduling a two-machine robotic cell: a solvable case. *Annals of Operations Research*, 57, 217-232.

LEVNER, E., KOGAN, K. et MAIMON, O. (1995). Flowshop scheduling of robotic cells with job-dependent transportation and setup effects. *Journal of the Operational Research Society*, 46/12, 1447-1455.

LIN, J. T. et DGEN, P.-K. (1994). An algorithm for routeing control of a tandem automated guided vehicle system. *International Journal of Production Research,* 32, 2735-2750.

LIN, J. T., CHANG, C. C. K. et LIU, W.-C. (1994). A load-routeing problem in a tandem configuration automated guided-vehicle system. *International Journal of Production Research*, 32, 411-427.

LIU, J. et MacCARTHY, B. L. (1996). The classification of FMS scheduling problems. *International Journal of Production Research,* 34/3, 647-656.

LOGENDRAN, R. et SONTHINEN, A. (1997). A tabu search-based approach for scheduling job-shop type flexible manufacturing systems. *Journal of the Operational Research Society*, 48, 264-277.

LOGENDRAN, R., SRISKANDARAJAH, C. (1996). Sequencing of robot activities and parts in two machine robotic cells. *International Journal of Production Research*, 28, 1611-1622.

LOVETT, J. N., JR., (1995). Cost estimating in manufacturing. in *Cost Estimator's Reference Manual*, John Wiley & Sons, Inc, 407-443.

LUSTIG, I. et PUGET, J-F. (1999). Program!=Program : Constraint programming and its relationship to mathematical programming, ILOG.

MAGGU, P. L., DAS, G. et KUMAR, R. (1981). On equivalent-job for job-block in 2xn sequencing problem with transportation-times. *Journal of the Operations Research Society of Japan*, 24, 136-146.

MAHADEVAN, B. et NARENDRAN, T. T. (1994). A hybrid modeling approach to the design of an AGV-based material handling system for an FMS. *International Journal of Production Research,* 32, 2015-2030.

MANNE, A.S. (1960). On the job-shop scheduling problem. *Operations Research*, 8, 219-223.

MARRIOT, K. and STUCKEY, P.J. (1998). *Programming with constraints: an introduction.* The MIT Press.

MAXWELL, W. L. et MUCKSTADT, J. A. (1982). Design of automatic guided vehicle systems. *IIE Transactions*, 14/2,114-124.

MILTENBURG, J. et SPARLING, D. (1996). Managing and reducing total cycle time: models and analysis. *International Journal of Production Economics*, 46/47, 89-108.

MUTH, J.F. et THOMPSON, G. L. (1963). *Industrial Scheduling*, Englewood Cliffs, New Jersey, Prentice Hall.

NAKAMURA, N. et SALVENDY, G. (1988). An experimental study of human decision-making in computer-based scheduling of flexible manufacturing system. *International Journal of Production Research,* 26/4, 567-583.

NARASIMHAN, R., BATTA, R., et KARWAN, M. H. (1999). Routing automated guided vehicles in the presence of interruptions. *International Journal of Production Research,* 37/3, 653-681.

NEUMANN, K., et SCHWINDT, C. (1997). Activity-on-node networks with minimal and maximal time lags and their application to make-to-order production. *OR Spectrum*, 10/3, 205-18.

NORMAN, B. A. et BEAN, J. C. (1999). Genetic algorithm methodology for complex scheduling problems. *Naval Research Logistics,* 46/2, 199-211.

NUDTASOMBOON, N., RANDHAWA, S. U. (1997). Resource-constrained project scheduling with renewable and non-renewable resources and time-resource tradeoffs. *Computers & Industrial Engineering,* 32/1, 227-242.

OCCENA, L. G. et YOKOTA, T. (1991). Modeling of an automated guided vehicle system (AGVS) in a just-in-time (JIT) environment. *International Journal of Production Research,* 29, 495-511.

ORLOFF, C. S. (1976). Route constraint fleet scheduling. *Transportation Science,* 10, 149-168.

OZDAMAR, L. et YAZGAC, T. (1997). Capacity driven due date settings in make-to-order production systems. *International Journal of Production Economics,* 49/1, 29-44.

PANWALKER, S. (1991). Scheduling of a two-machine flowshop with travel time between machines. *Journal of the Operational Research Society,* 42, 609-613.

PANWALKER, S. S. et ISKANDER, W. (1977). A survey of scheduling rules. *Operations Research,* 25, 45-61.

PAPADIMITRIOU, C. H. et KANELLAKIS, P.C. (1980). Flowshop scheduling with limited temporary storage. *Journal of the Association for Computing Machinery,* 27, 533-549.

PARK, K.-S., POSNER, M.E. (1993). Single Crane Scheduling Problems. Working paper, Department of Industrial and Systems Engineering, The Ohio State University.

PATTERSON, J.H. (1973). Alternate Methods of project scheduling with limited resources. *Naval Research Logistics Quarterly,* 20/4, 767-784.

PATTERSON, J.W., BALAKRISHNAN, P.N. et SRIDHARAN, V. (1997). An experimental comparison of capacity rationing models. *International Journal of Production Research,* 35/6, 1639-49.

PAULLI, J. (1995). A hierarchical approach for the FMS scheduling problem. *European Journal of Operational Research,* 86, 32-42.

PESANT, G., GENDREAU, M., POTVIN, J-Y. and ROUSSEAU J-M. (1996). An optimal algorithm for the traveling salesman problem with time windows using constraint logic programming. Technical Report, Centre de recherche sur les transports, Montreal.

PINEDO, M. (1995). *Scheduling : Theory , algorithms and systems,* Prentice Hall, Englewood Cliffs, New Jersey.

PINEDO, M. et SINGER, M. (1999). Shifting bottleneck heuristic for minimizing the total weighted tardiness in a job shop. *Naval Research Logistics,* 46/1, 1-17.

PINTO, P.A. et RAO, B.M. (1992). Joint lot sizing and scheduling for multi-stage multi-product flow shops. *International Journal for Production Research;* 30/5, 1137-1152.

PORTMANN, M.-C. (1997). Scheduling methodology : optimization and compu-search approaches I. dans *The Planning and Scheduling of Production Systems, A. ARTIBA et S.E. ELMAGHRABY,* Chapman & Hall, London, UK, 271-300.

POTTS, C.N. et BAKER, K.R. (1989). Flow-shop scheduling with lot streaming. *Operations Research Letters,* 8, 297-303.

PRAKASH, A. et CHEN, M. (1995). A simulation study of flexible manufacturing systems. *Computers and Industrial Engineering,* 28/1, 191-199.

RAJOTIA, S., SHANKER, K. et BATRA, J. L. (1998). A semi-dynamic time window constrained routeing strategy in an AGV system. *International Journal of Production Research,* 36/1, 35-50.

RAJU, K. R. et CHETTY, O. V. K. (1993). Design and evaluation of automated guided vehicle systems for flexible manufacturing systems: an extended timed Petri net-based approach. *International Journal of Production Research,* 31, 1069-1096.

RAMAN, N. et TALBOT, F. B. (1993). The job shop tardiness problem: a decomposition approach. *European Journal of Operational Research*, 69, 187-199.

RAMAN, N., TALBOT, F. B. et RACHAMADUGU, R. V. (1986). Simultaneous scheduling of machines and material handling devices in automated manufacturing. . In K. E. Stecke and R. Suri (eds), *Proceedings of the second ORSA/TIMS conference on flexible manufacturing Systems: Operations Research Models and Applications,* 321-332. Elsevier Science Publishers B.V., Amsterdam.

RAMUDIN, A. et MARIER, P. (1996). The generalized shifting bottleneck procedure. *European Journal of Operational Research*, 93/1, 34-48.

RÖCK, H. (1984). The three machine no-wait flow-shop is NP-complete. *Journal of the Association for Computing Machinery*, 31, 336-345.

RODAMMER, F.A. et WHITE, K.P. Jr (1988). A recent Survey of production scheduling. *IEEE Transactions on Systems, Man and Cybernetics,* 18/6, 841-851.

ROTHKOFF, M.H. (1966). Scheduling independent tasks on parallel processors. *Management Science*, 12/6, 437-447.

ROY, B. (1964). Physionomie et traitement des problèmes d'ordonnancement, dans *Les Problèmes d'Ordonnancemnt : Applications et Méthodes,* CARRÉ (1964), Dunod, Paris.

ROY, R. (1998). Scheduling and control, performance measures and discrete event simulation. *Journal of the Operational Research Society*, 49, 151-156.

RUSSEL, R. S. et TANCHOCO, J. M. A. (1984). An evaluation of vehicle dispatching rules and their effect on shop performance. *Material Flow,* 1, 271-280.

SAAD, G. H. (1990). Hierarchical production-planning systems: extensions and modifications. *Journal of the Operational Research Society*, 41/7, 609-624.

SABUNCUOGLU, I. (1998). A study of scheduling rules of flexible manufacturing systems: a simulation approach. *International Journal of Production Research*, 36/2, 527-546.

SABUNCUOGLU, I. et HOMMERTZHEIM, D. (1989a). An Investigation of machine and AGV scheduling rules in an FMS. In K. E. Stecke and R. Suri (eds), *Proceedings of the third ORSA/TIMS conference on flexible manufacturing Systems: Operations Research Models and Applications,* 261-266. Elsevier Science Publishers B.V., Amsterdam.

SABUNCUOGLU, I. et HOMMERTZHEIM, D. (1992 a). Experimental investigation of FMS machine and AGV scheduling rules against the mean flow-time criterion. *International Journal of Production Research*, 30/7, 1617-1635.

SABUNCUOGLU, I. et HOMMERTZHEIM, D.L. (1992 b). Dynamic Dispatching Algorithm for Scheduling Machines and Automated Guided Vehicles in a Flexible Manufacturing System. *International Journal of Production Research*, 30/ 5, 1059-1079.

SABUNCUOGLU, I. et KARABUK, S. (1998). Beam search-based algorithm and evaluation of scheduling approaches for flexible manufacturing systems. *IIE Transactions*, 30/2, 179-191.

SABUNCUOGLU, I., et HOMMERTZHEIM, D.L. (1993). Experimental Investigation of an FMS due-date scheduling problem: Evaluation of machine and AGV scheduling rules. *The International Journal of Flexible Manufacturing Systems*, 5, 301-323.

SAHNI, S. et CHO, Y. (1979). Complexity of scheduling jobs with no wait in process. Mathematics of Operations Research, 4, 448-457.

SAHNI, S. K. (1976). Algorithms for scheduling independent tasks. *Journal of the Association for Computing Machinery*, 23/1, 116-127.

SALOMON, M., KROON, L.G., KUIK, R., et VAN WASSENHOVE, L. N. (1991). Some extensions of the discrete lotsizing and scheduling problem. *Management Science* , 37/7, 801-812.

SAYIN, S. et KARABATI, S. (1999). Bicriteria approach to the two-machine flow shop scheduling problem. *European Journal of Operational Research*, 113/2, 435-449.

SCHRAGE, L.E. et BAKER K.R. (1978). Dynamic programming solution of sequencing problems with precedence constraints, *Operations Research*, 26/3, 444-449.

SEIDMAN, A. et TENENBAUM, A. (1994). Throughput maximization in flexible manufacturing systems. *IIE Transactions,* 26/1, 90-100.

SEIFERT, R. W., KAY, M. G., WILSON, J. R. (1998). Evaluation of AGV routeing strategies using hierarchical simulation. *International Journal of Production Research*, 36/7, 1961-1976.

SETHI, S. P., SRISKANDARAJAH, C., SORGER, G., BLAZEWICZ, J. et KUBIAK, W. (1992). Sequencing of parts and robot moves in a robotic cell. *International Journal of flexible manufacturing systems*, 4, 331-358.

SHAH, M., LIN, L. et NAGI, R. (1997). Production order-driven AGV control model with object-oriented implementation. *Computer Integrated Manufacturing Systems,* 10/1, 35-48.

SHANG, J. S. (1995). Robust design and optimization of material handling in an FMS. *International Journal of Production Research*, 33, 2437-2454.

SHAPIRO, J. (1979). *Mathematical Programming: Structures and Algorithms,* John Wiley and Sons.

SIDNEY, J. B. et SRISKANDARAJAH, C. (1999). Heuristics for the two-machine no-wait openshop scheduling problem. *Naval Research Logistics,* 46/2, 129-145.

SINNAMON, G. et MILNER, S. (1995). Just- in time schedules for the small make-to-order shop. *Canadian Journal of Administrative Sciences*, 12/4 , 340-351.

SMITH, J.S., PETERS, B. A. et SRINIVASAN, A. (1999). Job shop scheduling considering material handling. *International Journal of Production Research.* 37/7, 1541-1560.

SMITH, W.E. (1956). Various optimizers for single stage production. *Naval Research Logistics Quarterly*, 3, 59-66.

SODHI, M. S. , ASKIN, R.G. et SEN, S. (1994). A hierarchical model for control of flexible manufacturing systems. *Journal of the Operational Research Society*, 45/10, 1185-1196.

SOLOMON, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35, 254-265.

SPACHIS, A.S. et KING, J. R. (1979). Job shop scheduling heuristics with local neighborhood search. *International Journal of Production Research,* 17/6, 507-526.

SRISKANDARAJAH, C. (1988). Production scheduling: complexity and approximate algorithms. *Les Cahiers du Gérad* , G-88-29.

SRISKANDARAJAH, C., et WAGNEUR E. (1998). Lot streaming and scheduling multiple products in two-machine no-wait flowshops. *Les Cahiers du Gérad*, G-98-25, Juin 1998

SRISKANDARAJAH, C., HALL, N. G., KAMOUN, H. (1996). Scheduling large robotic cells. Working paper, Department of Industrial Engineering, University of Toronto, Canada.

SRISKANDARAJAH, C., SIDNEY, J. B. et SETHI, S. P. (1999). Cycle-time comparison of a variety of robotic cells. Journées de l'Optimisation 1999, Montréal, 10-12 Mai.

STECKE, K. (1984). Design, planning, scheduling and control problems of flexible manufacturing systems. *Proceedings First ORSA /TIMS, Conference on Flexible Manufacturing Systems*, Ann Arbor, Michigan, USA.

STECKE, K. E. (1983). A hierarchical approach to solving machine grouping and loading problems of flexible manufacturing systems. *European Journal of Operational Research,* 24, 369-377.

STEIN, D. M.(1978). Scheduling dial-a-ride transportation systems. *Transportation Science*, 12, 232-249.

STERN, H. I., et VITNER, G. (1990). Scheduling parts in a combined production-transportation work cell. *Journal of the Operational Research Society*, 41/7, 625-632.

TADEI, R., DELLA CROCE, F. D. et MENGA, G. (1995). Advanced search techniques for the job shop problem: A comparison. *Operations Research,* 29/2, 179-194.

TAGHABONI, F. et TANCHOCO, J. M. A. (1988). A LISP-based controller for free ranging automated guided vehicle systems. *International Journal of Production Research,* 26/2, 173-188.

TAGHABONI-DUTTA, F. et TANCHOCO, J. M. A. (1995). Comparison of dynamic routing techniques for automated guided vehicle system. International Journal of Production Research, 33/10, 2653-2669.

TAWEGOUN, R., CASTELAIN, E., et GENTINA J.C. (1994). Real-time piloting of flexible manufacturing systems. *European Journal of Operations Research,* 78, 252-261.

TRIETSCH, D. et BAKER, K. R. (1993). Basic techniques for lot streaming. *Operations Research*, 41/6, 1065-1076.

ULUSOY, G. et BILGE, U. (1993). Simultaneous Scheduling of Machines and Automated Guided Vehicles. *International Journal of Production Research*, Vol. 31/12 , 2857-2873.

ULUSOY, G., FUNDA, S.-S. et BILGE, U. (1997). A genetic algorithm approach to the simultaneous scheduling of machines and automated guided vehicles. *Computers and Operations Research,* 24/4, 335-351.

VAIRAKTARAKIS, G. L. (1997). Analysis of scheduling algorithms for master-slave systems. *IIE Transactions*, 29, 939-949.

VALERIO DE CARVALHO, J.M. et GUIMARAES RODRIGUES, A.J. (1995). An LP-Based approach to a two stage cutting stock problem. *European Journal of Operational Research*, 84/3, 580-589.

VAN HENTENRYCK, P. (1989). *Constraint Satisfaction in Logic Programming.* The MIT Press.

VAN HENTENRYCK, P. (1999). *The OPL Optimization Programming Language.* The MIT Press.

VAN LAARHOVEN, P. J. M. et AARTS, E. H. L. (1987). *Simulated Annealing: Theory and Applications,* Reidel, Dordrecht, Netherlands.

VAN LAARHOVEN, P.J.M. et al. (1992). Job shop scheduling by simulated annealing. *Operations Research*, 40/1, 113-125.

VICKSON, R. G. et ALFREDSSON, B.E. (1992). Two and three-machine flow shop scheduling problems with equal sized transfer batches. *International Journal of Production Research*, 30/7, 1551-1574.

VILLENEUVE, B., DESAULNIERS, G., LANGEVIN, A. et RIOPEL, D. (1999). Scheduling and routing of automated guided vehicles: an exact approach. Journées de l'Optimisation 1999, Montréal, 10-12 Mai.

VOLGENANT, A. et TEERHUIS, E. (1999). Improved heuristics for the n-job single-machine weighted tardiness problem. *Computers and Operations Research,* 26/1, 35-44.

WAGNER, H.M. (1959). An integer programming model for machine scheduling. *Naval Research Logistics Quarterly*, 6, 131-140.

WAGNER, H.M. (1959). An integer programming model for machine scheduling. *Naval Research Logistics Quarterly*, 6, 131-140.

WANG, H.-P. B. et HAFEEZ, S. A. (1994). Performance evaluation of Tandem and conventional AGV systems using generalized stochastic Petri-nets. *International Journal of Production Research*, 32, 917-932.

WEBSTER, S. T. (1997). Complexity of scheduling job families about a common due date. *Operations Research Letters*, 20/2, 65-74.

WEIN, L. M. et CHEVALIER, P.B. (1992). A broader view of the job-shop scheduling problem. *Management Science*, 38/7, 1018-1033.

WEIN, L. M. et CHEVALIER, P.B. (1992). A broader view of the job-shop scheduling problem. *Management Science*, 38/7, 1018-1033.

WILKERSON, L.J. et IRWIN, J. D. (1971). An improved algorithm for scheduling independent tasks. *AIIE Transactions*, 3, 239-245.

WISNER, J. D., et SIFRED, S. P. (1995). A survey of US manufacturing practices in make-to-order machine shops. *Production and Inventory Management Journal*, first quarter, 1-7.

WU, X. B. (1998). The application of analytic process of resource in an AGV scheduling. *Computers and Industrial Engineering*, 35/1-2, 169-172.

YEH, M.-S. et YEH, W.-C. (1998). Deadlock prediction and avoidance for zone-control AGVS. *International Journal of Production Research*, 36/10, 2879-2889.

YEN, J. Y. (1971). Finding the k-shortest loopless paths in a network. *Management Science*, 17, 712-715.

YIM, D.-S., et LINN, R. J. (1993). Push and pull rules for dispatching automated guided vehicles in a flexible manufacturing system. *International Journal of Production Research*, 31, 43-57.

ZAREMBA, M. B., OUBUCHOWICZ, A., BANAZAK, Z. A. et JEDRZEJEK, K. J. (1997). Max-algebra approach to the robust distributed control of repetitive AGV systems. *International Journal of Production Research*, 35/10, 2667-2687.

ZEGORDI, S. H., ITOH, K. et ENKAWA, T. (1995). Minimizing makespan for flow shop scheduling by combining simulated annealing with sequencing knowledge. *European Journal of Operational Research,* 85, 515-531.

ZENG, L., WANG, H.-P. et JIN, S. (1991). Conflict detection of automated guides vehicles : a Petri net approach. *International Journal of Production Research*, 29, 865-879.

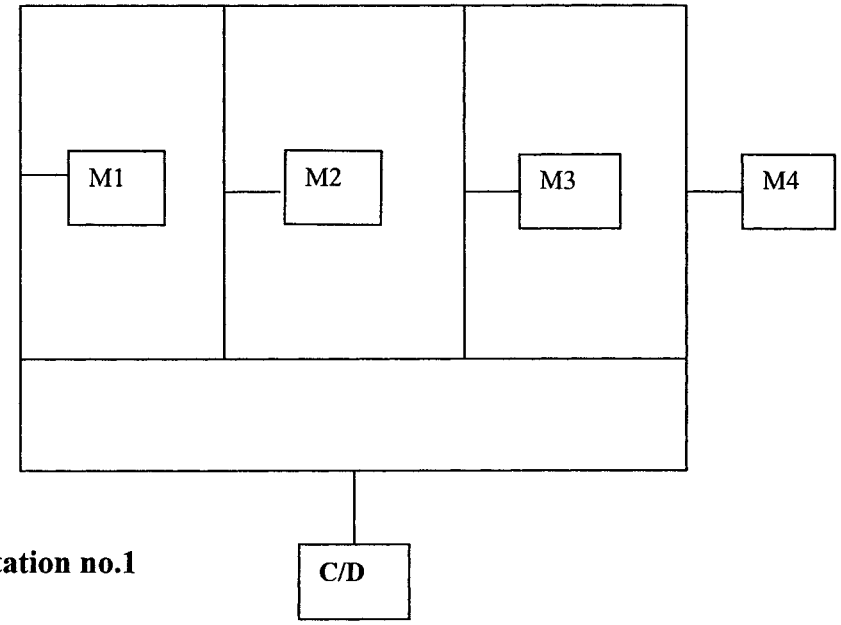
ZHAO, X., et LAM, K. (1997). Lot-sizing rules and freezing the master production schedule in material requirements planning systems. *International Journal of Production Economics,* 53/3, 281-305.

ANNEXE A : LES DONNÉES

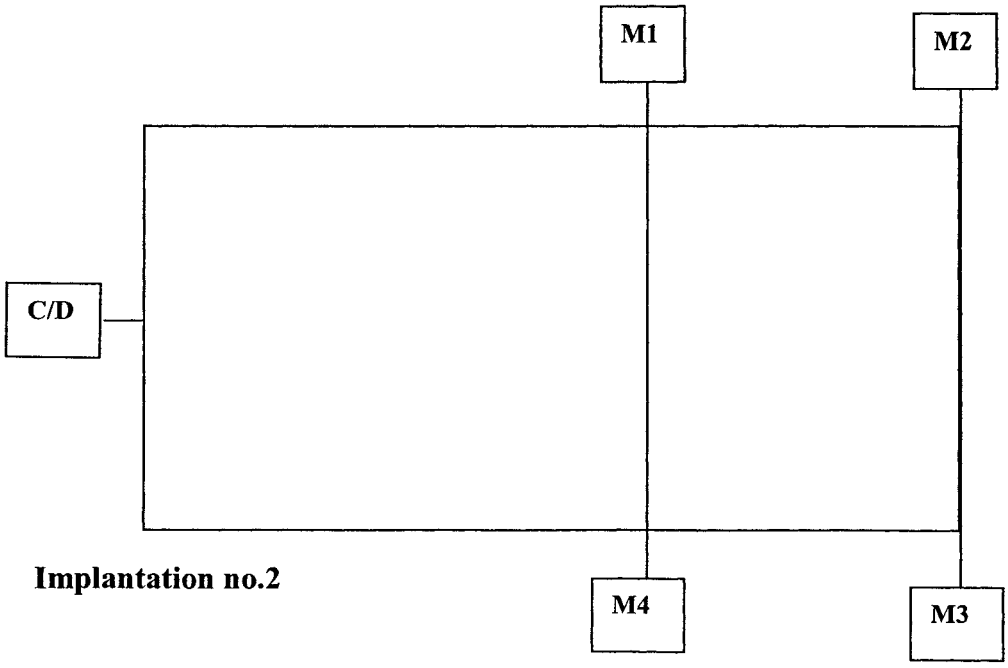
Légende

C/D Chargement/Déchargement

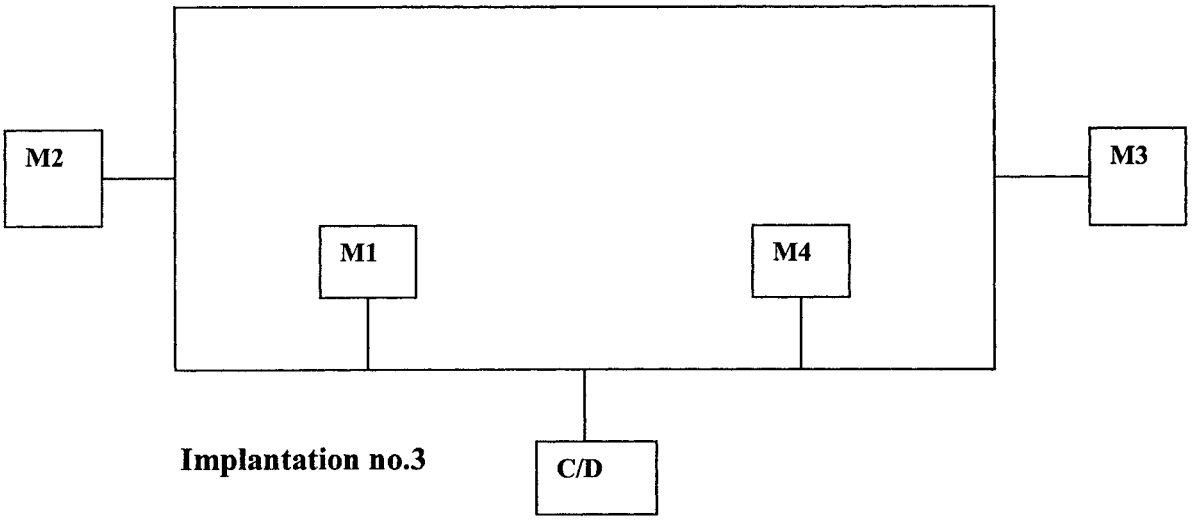
M Machine



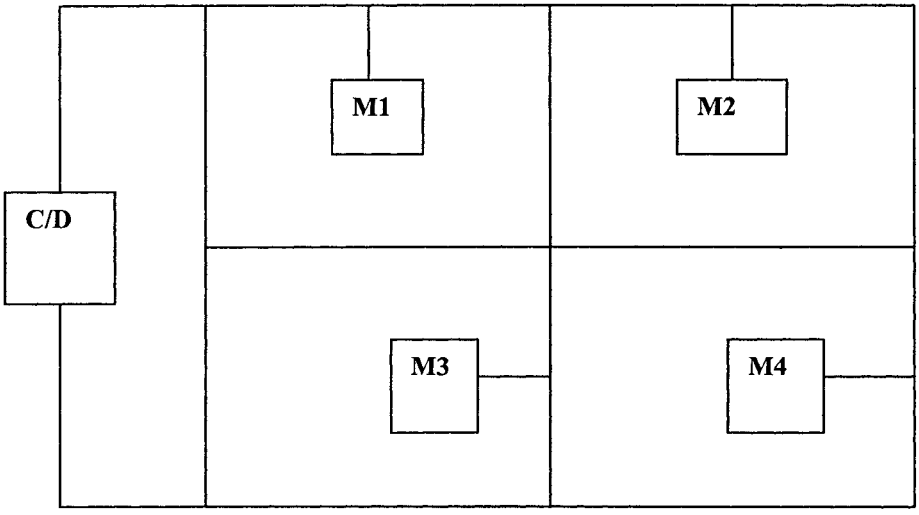
Implantation no.1



Implantation no.2



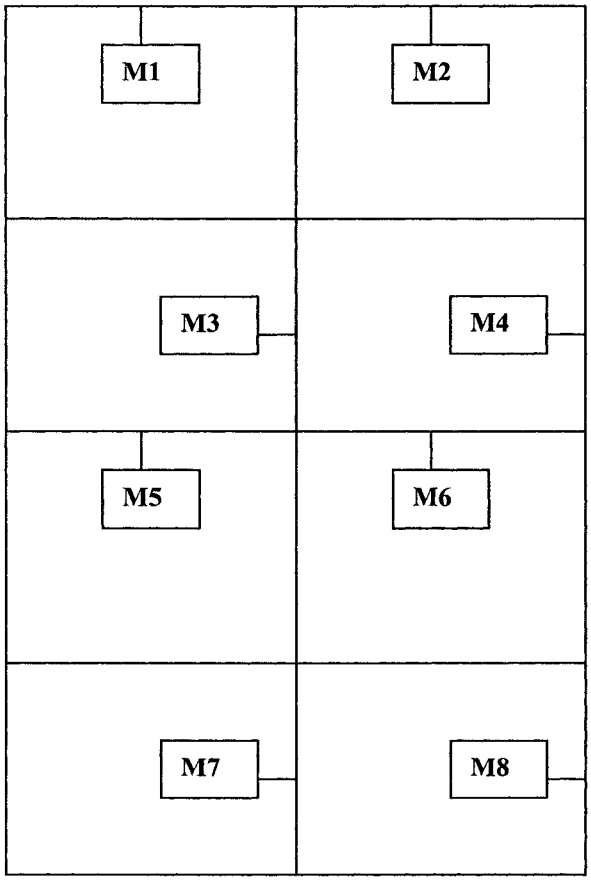
Implantation no.3



Implantation no.4

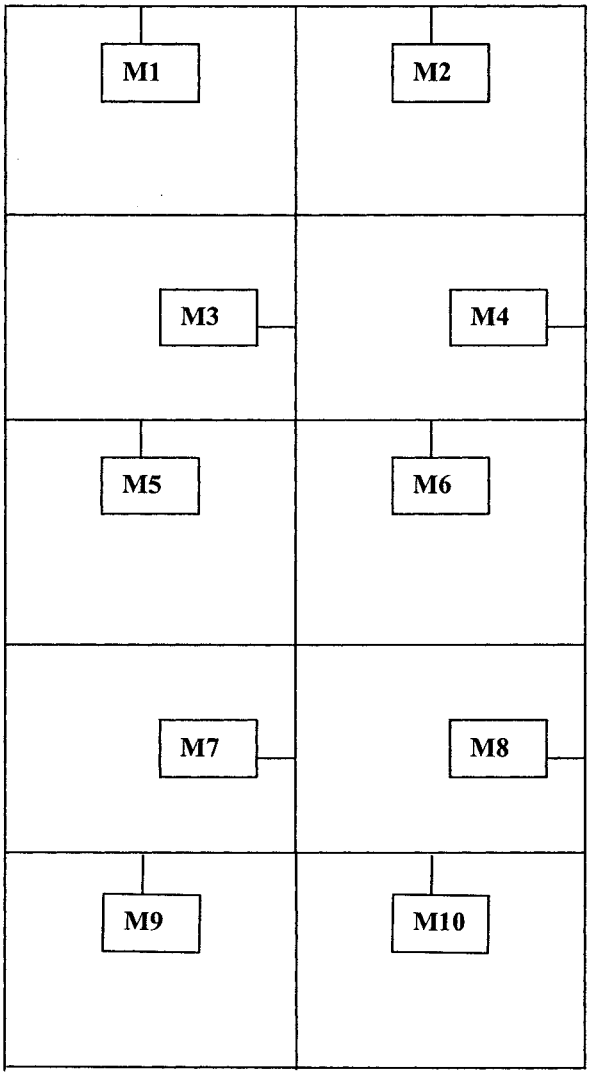
Implantation no. 5 (agrandissement de l'implantation 4)

Utilisé avec les jeux de données du tableau (4.7)



Implantation no. 6 (agrandissement de l'implantation 4 pour 10 machines)

*Utilisé avec le jeu de données 10*10, problème (*mt10) de Fisher et Thompson disponible dans la bibliothèque électronique de recherche opérationnelle http://mscmga.ms.ic.ac.uk/info.html



Matrice des distances entre les machines, implantation no. 1
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995)

	C/D	M1	M2	M3	M4
C/D	0	6	8	10	12
M1	12	0	6	8	10
M2	10	6	0	6	8
M3	8	8	6	0	6
M4	6	10	8	6	0

Matrice des distances entre les machines, implantation no. 2
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995)

	C/D	M1	M2	M3	M4
C/D	0	4	6	8	6
M1	6	0	2	4	2
M2	8	12	0	2	4
M3	6	10	12	0	2
M4	4	8	10	12	0

Matrice des distances entre les machines, implantation no.3
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995)

	C/D	M1	M2	M3	M4
C/D	0	2	4	10	12
M1	12	0	2	8	10
M2	10	12	0	6	8
M3	4	6	8	0	2
M4	2	4	6	12	0

Matrice des distances entre les machines, implantation no.4
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995)

	C/D	M1	M2	M3	M4
C/D	0	4	8	10	14
M1	18	0	4	6	10
M2	20	14	0	8	6
M3	12	8	6	0	6
M4	14	4	12	6	0

Matrice des distances entre les machines, implantation no. 1
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995) et temps modifiés à la baisse

	C/D	M1	M2	M3	M4
C/D	0	3	4	5	6
M1	6	0	3	4	5
M2	5	3	0	3	4
M3	4	4	3	0	3
M4	3	5	4	3	0

Matrice des distances entre les machines, implantation no. 2
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995) et temps modifiés à la baisse

	C/D	M1	M2	M3	M4
C/D	0	2	3	4	3
M1	3	0	1	2	1
M2	4	6	0	1	2
M3	3	5	6	0	1
M4	2	4	5	6	0

Matrice des distances entre les machines, implantation no. 3
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995) et temps modifiés à la baisse

	C/D	M1	M2	M3	M4
C/D	0	1	2	5	6
M1	6	0	1	4	5
M2	5	6	0	3	4
M3	2	3	4	0	1
M4	1	2	3	6	0

Matrice des distances entre les machines, implantation no. 4
Selon l'orientation des arcs proposée par Bilge et Ulusoy (1995) et temps modifiés à la baisse

	C/D	M1	M2	M3	M4
C/D	0	2	4	5	7
M1	9	0	2	3	5
M2	10	7	0	4	3
M3	6	4	3	0	3
M4	7	2	6	3	0

Matrice des distances entre les machines, implantation no.5
Arcs bidirectionnels et temps de manutention plus petits, 8 machines

	C/D	M1	M2	M3	M4	M5	M6	M7	M8
C/D	0								
M1		0	1	2	3	3	3	4	5
M2		1	0	2	2	3	3	4	4
M3		2	2	0	2	1	1	2	3
M4		3	2	2	0	2	1	3	2
M5		3	3	1	2	0	1	2	3
M6		3	3	1	1	1	0	2	2
M7		4	4	2	3	2	2	0	2
M8		5	4	3	2	3	2	2	0

Matrice des distances entre les machines, implantation no.6
Arcs bidirectionnels et temps de manutention plus petits, 10 machines

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10
M1	0	1	2	3	3	3	4	5	5	5
M2	1	0	2	2	3	3	4	4	5	5
M3	2	2	0	2	1	1	2	3	3	3
M4	3	2	2	0	2	1	3	2	4	4
M5	3	3	1	2	0	1	2	3	3	3
M6	3	3	1	1	1	0	2	2	3	3
M7	4	4	2	3	2	2	0	2	1	1
M8	5	4	3	2	3	2	2	0	2	1
M9	5	5	3	4	3	3	1	2	0	1
M10	5	5	3	4	3	3	1	1	1	0

Données sur les gammes

Jeu no. 1
Travail (1) : M1(8) ;M2(16);M4(12)

Travail (2) : M1(20) ;M3(10);M2(18)

Travail (3) : M3(12) ;M4(8);M1(15)

Travail (4) : M4(14) ;M2(18)

Travail (5) : M3(10) ;M1(15)

Jeu no. 2
Travail (1) : M1(10) ;M4(8)

Travail (2) : M2(10) ;M4(18)

Travail (3) : M1(10) ;M3(20)

Travail (4) : M2(10) ;M3(15);M4 (12)

Travail (5) : M1(10) ;M2(15);M4(12)

Travail (6) : M1(10) ;M2(15);M3(12)

Jeu no. 3
Travail (1) : M1(16) ;M3(15)

Travail (2) : M2(18) ;M4(15)

Travail (3) : M1(20) ;M2(10)

Travail (4) : M3(15) ;M4(10)

Travail (5) : M1(8) ;M2(10);M3(15);M4(17)

Travail (6) : M2(10) ;M3(15);M4(8);M1(15)

Jeu no. 4
Travail (1) : M4(11) ;M1(10);M2(7)

Travail (2) : M3(12) ;M2(10);M4(8)

Travail (3) : M2(7) ;M3(10);M1(9);M3(8)

Travail (4) : M2(7) ;M4(8);M1(12);M2(6)

Travail (5) : M1(9) ;M2(7);M4(8);M2(10);M3(8)

Jeu no. 5
Travail (1) : M1(6) ;M2(12);M4(9)

Travail (2) : M1(18) ;M3(6);M2(15)

Travail (3) : M3(9) ;M4(3);M1(12)

Travail (4) : M4(6) ;M2(15)

Travail (5) : M3(3) ;M1(9)

Jeu no. 6

Travail (1) : M1(9) ;M2(11);M4(7)

Travail (2) : M1(19) ;M2(20);M4(13)

Travail (3) : M2(14) ;M3(20);M4(9)

Travail (4) : M2(14) ;M3(20);M4(9)

Travail (5) : M1(11) ;M3(16);M4(8)

Travail (6) : M1(10) ;M3(12);M4(10)

Jeu no. 7

Travail (1) : M1(6) ;M4(6)

Travail (2) : M2(11) ;M4(9)

Travail (3) : M2(9) ;M4(7)

Travail (4) : M3(16) ;M4(7)

Travail (5) : M1(9) ;M3 (18)

Travail (6) : M2(13) ;M3(19);M4(6)

Travail (7) : M1(10) ;M2(9);M3(13)

Travail (8) : M1(11) ;M2(9);M4(8)

Jeu no. 8

Travail (1) : M2(12) ;M3(21);M4(11)

Travail (2) : M2(12) ;M3(21);M4(11)

Travail (3) : M2(12) ;M3(21);M4(11)

Travail (4) : M2(12) ;M3(21);M4(11)

Travail (5) : M1(10) ;M2(14);M3(18);M4(9)

Travail (6) : M1(10) ;M2(14);M3(18);M4(9)

Jeu no. 9
Travail (1) : M3(9) ;M1(12);M2(9);M4(6)

Travail (2) : M3(16) ;M2(11);M4(9)

Travail (3) : M1(21) ;M2(18);M4(7)

Travail (4) : M2(20) ;M3(22);M4(11)
Travail (5) : M3(14) ;M1(16);M2(13);M4(9)

Jeu no. 10
Travail (1) : M1(11) ;M3(19);M2(16);M4(13)

Travail (2) : M2(21) ;M3(16);M4(14)

Travail (3) : M3(8) ;M2(10);M1(14);M4(9)

Travail (4) : M2(13) ;M3(20);M4(10)

Travail (5) : M1(9) ;M3(16);M4(18)

Travail (6) : M2(19) ;M1(21);M3(11);M4(15)

Les jeux suivants sont augmentés et utilisés avec l'implantation bidirectionnelle no.5

Jeu no. 11
Travail (1) : M1(8) ;M2(16);M4(12)

Travail (2) : M1(20) ;M3(10);M2(18)

Travail (3) : M3(12) ;M4(8);M1(15)

Travail (4) : M4(14) ;M2(18)

Travail (5) : M3(10) ;M1(15)

Travail (6) : M5(8) ;M6(16);M8(12)

Travail (7) : M5(20) ;M7(10);M6(18)

Travail (8) : M3(12) ;M4(8);M7(15)

Travail (9) : M4(14) ;M6(18)

Travail (10) : M3(10) ;M6(15)

Jeu no. 12

Travail (1) : M1(10);M4 (18)

Travail (2) : M2(10) ;M4(18)

Travail (3) : M1(10) ;M3(20)

Travail (4) : M2(10) ;M3(15);M4(12)

Travail (5) : M1(10) ;M2(15);M5(15)

Travail (6) : M1(10) ;M2(15);M3(12)

Travail (7) : M5(10) ;M4(18)

Travail (8) : M2(10) ;M4(18)

Travail (9) : M1(10) ;M3(20)

Travail (10) : M2(10) ;M3(15);M4(12)

Travail (11) : M1(10) ;M2(15);M4(15)

Travail (12) : M1(10) ;M2(15);M3(12)

Jeu no. 13

Travail (1) : M1(16) ;M3(15)

Travail (2) : M2(18) ;M4(15)

Travail (3) : M1(20) ;M2(10)

Travail (4) : M3(15) ;M4(10)

Travail (5) : M1(8) ;M2(10);M3(15);M4(15)

Travail (6) : M2(10) ;M3(15);M4(8);M1(15)

Travail (7) : M1(10) ;M5(15)

Travail (8) : M2(15) ;M5(10)

Travail (9) : M7(15) ;M2(8)

Travail (10) : M4(8) ;M8(15)

Travail (11) : M1(20) ;M2(15);M3(7);M4(5)

Travail (12) : M6(20) ;M3(15);M4(7);M1(5)

Jeu no. 14
Travail (1) : M3(9) ;M1(18);M2(14);M4(10)

Travail (2) : M3(9) ;M2(18);M4(14)

Travail (3) : M1(10) ;M2(15);M4(7)

Travail (4) : M2(18) ;M3(12);M4(11)

Travail (5) : M3(24) ;M1(16);M2(13);M4(19)

Travail (6) : M4(15) ;M5(20);M2(5)

Travail (7) : M2(8) ;M3(10);M4(24)

Travail (8) : M3(13) ;M1(17);M5(18);M7(27)

Travail (9) : M8(25) ;M4(13);M1(16);M6(28)

Travail (10) : M5(30) ;M2(15);M8(26);M4(8);M3(10)

Jeu no. 15
Travail (1) : M1(6) ;M2(12);M4(9)

Travail (2) : M1(18) ;M3(6);M2(15)

Travail (3) : M3(9) ;M4(3);M1(12)

Travail (4) : M4(6) ;M2(15)

Travail (5) : M3(3) ;M1(9)

Travail (6) : M8(6) ;M2(12);M3(9)

Travail (7) : M1(18) ;M7(6);M4(15)

Travail (8) : M8(9) ;M4(3);M1(12)

Travail (9) : M7(6) ;M2(15)

Travail (10) : M5(3) ;M1(9)

Jeu no. 16
Travail (1) : M1(9) ;M2(11);M4(7)

Travail (2) : M1(19) ;M2(20);M4(13)

Travail (3) : M2(14) ;M3(20);M4(9)

Travail (4) : M2(14) ;M3(20);M4(9)

Travail (5) : M1(11) ;M3(16);M4(8)

Travail (6) : M1(10) ;M3(12);M4(10)

Travail (7) : M1(10) ;M4(18)

Travail (8) : M2(10) ;M4(18)

Travail (9) : M1(10) ;M3(20)

Travail (10) : M2(10) ;M3(15);M4(12)

Travail (11) : M1(10) ;M2(15);M4(15)

Travail (12) : M1(10) ;M2(15);M3(12)

Jeu no. 17

Travail (1) : M1(6) ;M4(6)

Travail (2) : M2(11) ;M4(9)

Travail (3) : M2(9) ;M4(7)

Travail (4) : M3(16) ;M4(7)

Travail (5) : M1(9) ;M3(18)

Travail (6) : M2(13) ;M3(19);M4(6)

Travail (7) : M1(10) ;M2(9);M3(13)

Travail (8) : M1(11) ;M2(9);M4(8)

Travail (9) : M8(6) ;M4(6)

Travail (10) : M8(11) ;M5(9)

Travail (11) : M6(9) ;M5(7)

Travail (12) : M3(16) ;M5(7)

Travail (13) : M8(9) ;M3(18)

Travail (14) : M5(13) ;M3(19);M1(6)

Travail (15) : M8(10) ;M4(9);M3(13)

Travail (16) : M8(11) ;M4(9);M5(8)

Jeu no. 18
Travail (1) : M2(12) ;M3(21);M4(11)

Travail (2) : M2(12) ;M3(21);M4(11)

Travail (3) : M2(12) ;M3(21);M4(11)

Travail (4) : M2(12) ;M3(21);M4(11)

Travail (5) : M1(10) ;M2(14);M3(18);M4(9)

Travail (6) : M1(10) ;M2(14);M3(18);M4(9)

Travail (7) : M2(12) ;M6(21);M4(11)

Travail (8) : M8(12) ;M3(21);M4(11)

Travail (9) : M2(12) ;M8(21);M5(11)

Travail (10) : M8(12) ;M5(21);M6(11)

Travail (11) : M1(10) ;M2(14);M3(18);M7(9)

Travail (12) : M1(10) ;M2(14);M3(18);M4(9)

Jeu no. 19
Travail (1) : M4(15) ;M1(11);M2(17)

Travail (2) : M3(12) ;M2(5);M4(18)

Travail (3) : M2(17) ;M3(5);M1(9);M3(8)

Travail (4) : M2(7) ;M4(18);M1(12);M2(6)

Travail (5) : M1(19) ;M2(3);M4(8);M2(10);M3(8)

Travail (6) : M4(9) ;M5(10);M2(17)

Travail (7) : M2(8) ;M3(5);M4(8)

Travail (8) : M3(5) ;M1(8);M5(20);M7(18)

Travail (9) : M8(20) ;M4(28);M1(10);M6(7)

Travail (10) : M5(15) ;M2(15);M8(20);M4(10)

Jeu no. 20
Travail (1) : M1(11) ;M3(19);M2(16);M4(13)

Travail (2) : M2(21) ;M3(16);M4(14)

Travail (3) : M3(8) ;M2(10);M1(14);M4(9)

Travail (4) : M2(13) ;M3(20);M4(10)

Travail (5) : M1(9) ;M3(16);M4(18)

Travail (6) : M2(19) ;M1(21);M3(11);M4(15)

Travail (7) : M6(10) ;M4(18)

Travail (8) : M7(10) ;M4(18)

Travail (9) : M6(10) ;M3(20)

Travail (10) : M2(10) ;M5(15);M7(12)

Travail (11) : M1(10) ;M2(15);M7(15)

Travail (12) : M1(10) ;M2(15);M5(12)

Les routes alternatives

Dans les problèmes du chapitre 5, deux choix de plus court chemin ont été effectués pour l'accomplissement des tâches de manutention ou les voyages à vide entre deux postes.

Matrice des 2 plus courts chemins entre les machines
Arcs bidirectionnels et temps modifiés à la baisse

	M1	M2	M3	M4
M1	0	r1,r2	r3,r4	r5,r6
M2	r1,r2	0	r7,r8	r9,r10
M3	r3,r4	r7,r8	0	r11,r12
M4	r5,r6	r9,r10	r11,r12	0

Longueur des plus courts chemins, implantation 1

r1	3
r2	3
r3	4
r4	4
r5	5
r6	5
r7	3
r8	3
r9	4
r10	4
r11	3
r12	3

Longueur des plus courts chemins, implantation 2

r1	1
r2	3
r3	2
r4	2
r5	1
r6	3
r7	3
r8	1
r9	2
r10	2
r11	3
r12	1

Longueur des plus courts chemins, implantation 3

r1	2
r2	10
r3	4
r4	8
r5	2
r6	10
r7	6
r8	6
r9	4
r10	8
r11	2
r12	10

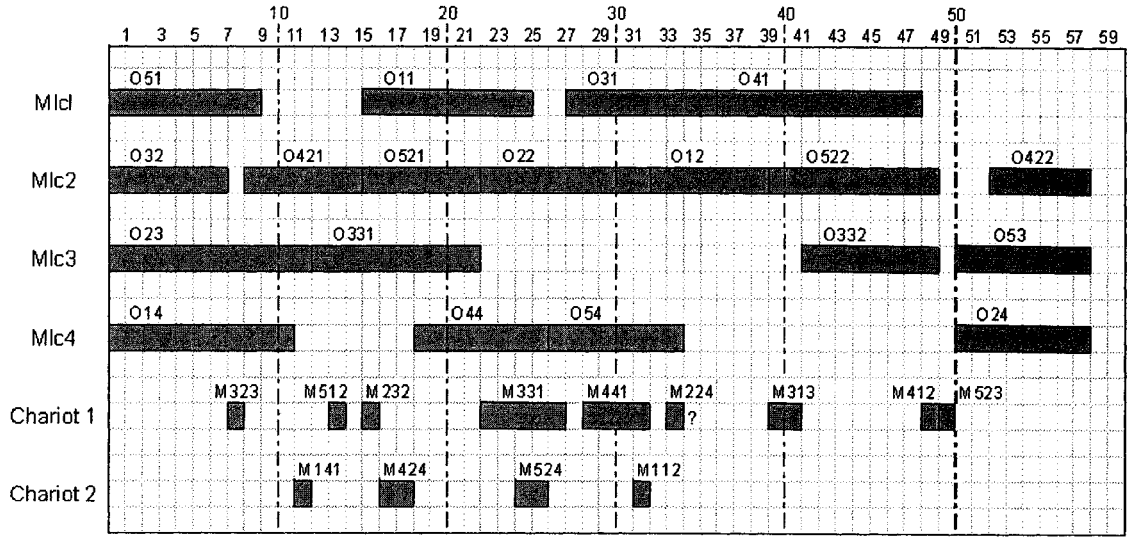
Longueur des plus courts chemins, implantation 4

r1	2
r2	6
r3	2
r4	7
r5	5
r6	7
r7	7
r8	3
r9	3
r10	7
r11	4
r12	6

ANNEXE B. EXEMPLES D'ORDONNANCEMENTS

Les ordonnancements présentés dans cette annexe présente un résultat intéressant. Le modèle du réseau bidirectionnel, sans conflits, présente un horaire plus court pour un même problème. Les données pour un réseau unidirectionnel où quelques chemins sont prolongés donnent un horaire plus long sans garantie d'éviter les conflits.

Resultat du modèle de plus court chemin, conflits pas considérés, objectif de 58
Problème PM-MA-4-2



Resultat du modèle des chemins prédéterminés sans conflits, réseau bidirectionnel, objectif de 56
Problème PM-CR-MA-4-2

