

Table des matières

Remerciement

Liste des figures

Liste des tableaux

Résumé

Introduction Général..... 01

Chapitre 01 : l'ordonnancement de la production

Introduction

I.	Production.....	04
I.1.	Définition de production.....	04
I.2.	La gestion de production.....	04
I.2.1.	Le rôle de la gestion de production.....	04
I.2.2 .	Les systèmes flexibles de production.....	05
I.3.	Rôle de l'ordonnancement en gestion de production.....	07
I.4.	Système de production non-linéaire Job-Shop.....	07
II.	L'ORDONNANCEMENT.....	07
II.1.	Définition d'ordonnancement.....	07
II.2.	Les types (classes) d'ordonnancement.....	08
II.3.	Les éléments d'un problème d'ordonnancement.....	08
II.3.1.	Les tâches.....	08
II.3.2.	Les ressources.....	09
II.3.3.	Les contraintes.....	10

II.3.4. Les critères d'optimisation.....	11
II.4. Notation et définition des problèmes d'ordonnancement.....	11
II.5. Les ateliers de production.....	12
II.5.1. Une seule machine.....	13
II.5.2. Machines identiques en parallèle.....	13
II.5.3. Les ateliers de type flow-shop.....	13
II.5.4. Les ateliers de type job-shop.....	13
II.5.5. Les ateliers de type open-shop.....	13
II.6. Représentation des problèmes d'ordonnancement.....	14
II.6.1. Le diagramme de Gantt.....	14
II.6.2. Graphe potentiel-tache.....	14
II.6.3. Méthodes PERT.....	15
II.7. Méthodes de résolution.....	16
Conclusion.....	18

Chapitr02 : Les Problèmes D'ordonnancement De Job Shop Avec Transport

Introduction

I. Le problème d'ordonnancement Job-shop.....	20
I.1. Atelier à cheminement multiple (job shop).....	20
I.2. Types de job shop.....	21
I.2.1. Job shop classique.....	21
I.2.2. Job shop flexible.....	22
II. Système de transport dans les ateliers.....	23

II.1.1. Les convoyeurs.....	23
II.1.2. Les robots industriel.....	23
II.1.3. Le Véhicule Auto-Guidé (AGV).....	24
III. Ordonnancement job shop avec contrainte de transport.....	24
III.1. Définition de problème.....	24
III.2. Etat de l’art.....	25

Conclusion

Chapitr03 : Résolution De Problème D’ordonnancement De Job Shop Avec Contrainte De Transport

Introduction

I. Description du système étudié.....	29
II. Contraintes et notations spécifiques au problème	30
II.1. Notation.....	31
III. Démarche de résolution.....	32
III.1. Description de la solution proposée.....	32
III.2. Simulation et résultats.....	37
Conclusion	41

Conclusion Générale

Liste des Figures

Figure N° 01 : Objectif de la gestion de production.....	05
Figure N° 02 : Les sous-systèmes constituant le système de production.....	06
Figure N° 03 : Système de production non-linéaire Job-shop.....	07
Figure N° 04 :caractéristique d’une tâche faisant référence à l’exécution d’une opération	09
Figure N° 05 : diagramme de Gant.....	14
Figure N° 06 : Graphe potentiel-taches d’un ordonnancement.....	15
Chapitre 02	
Figure N° 07 : Exemple de job shop simple et hybride.....	21
Figure N° 08 : Job shop classique à 3 jobs et 4 machines	22
Figure N° 09 : Représentation d’un système de type Job shop flexible à deux étage.....	22
Chapitre 03	
Figure N° 10 : le schéma d’atelier.....	30
Figure N°11 : diagramme de gant du problème de $J3 \beta_1 C_{max}$	35
Figure N°12 : disponibilité de job et AGV.....	35
Figure N°13 : diagramme de gant de $J3(1) \beta_1 C_{max}$	35
Figure N°14 : l’algorithme proposé.....	36
Figure N°15 : Exemple de 4 jobs avec SPT.....	39
Figure N°16 : Exemple de 4 jobs avec LPT.....	39

Liste des tableaux

Tableau N° 01 : Tableau(1) : les différentes valeurs de paramètres α β γ	12
Tableau N° 02 : les temps de transport entre les machines.....	30
Tableau N° 03 : la notation.....	32
Tableau N° 04 : les temps opératoires des jobs.....	34
Tableau N° 05 : routage de jobs.....	34
Tableau N° 06 : le temps de transport d'AGV.....	37
Tableau N° 07 : le temps d'opérateur.....	37
Tableau N° 08 : Temps opératoires sur les machines pour 4 jobs.....	38
Tableau N° 09 : Temps opératoires sur les machines pour 10 jobs.....	40
Tableau N° 10 : Temps opératoires des produits.....	40
Tableau N° 11 : les résultats de Cmax obtenu par MATLAB.....	41

ملخص

في هذه المذكرة قمنا بدراسة واحدة من مشاكل الجدولة الأكثر تعقيدا. وهي جدولة نوع العمل-جوب شوب مع قيود النقل. دراستنا لنظام إنتاجي يتكون من أربع آلات تعالج عدة أنواع من القطع. تنتقل هذه الأجزاء من قبل عربة ذات تحكم ذاتي (AGV). و لحل هذه المشكلة اقترحنا نهجا مجازية على أساس قواعد الأولوية (LPT و SPT) حيث نسعى إلى حلول أقرب إلى الحل الأمثل. المعيار المستخدم هنا هو التقليل من المدة الإجمالية لتنفيذ المهام، وهي تحسين Cmax (Makespan.)

كلمات البحث: الجدولة، ورشة العمل جوب شوب، قواعد (SPT , LPT)، جوب شوب مع قيودا لنقل.

Résumé

Dans ce mémoire nous avons étudié l'un des problèmes d'ordonnancement les plus complexe ; c'est l'ordonnancement d'atelier du type job-shop avec contraintes de transport.

Notre étude a été faite sur un système de production de type job shop, Ce système est composé de quatre machines traitant plusieurs types de pièces. Le déplacement de ces pièces est assuré par un seul AGV. Pour la résolution de ce problème nous avons proposé des approches heuristiques basant sur des règles de priorité (SPT et LPT) par lesquelles nous cherchons des solutions plus proches à la solution optimale. Le critère retenu ici est la minimisation de la durée totale de l'exécution des tâches, soit l'optimisation du Cmax (Makespan).

Mots clés: Ordonnancement, Job-shop, Règle de priorité, Ordonnancement avec transport.

Abstract:

In this paper we have studied one of the most complex scheduling problems; this is the job-shop type scheduling with transport constraints. Our study was based on a job shop type production system. This system consists of four machines dealing a several types of parts. These parts are moved by a single AGV. To solve this problem we proposed the heuristic approaches based on priority rules (SPT and LPT) where we seek solutions nearer to the optimal solution. The criterion used here is the minimization of the total duration of the execution of the tasks, namely the optimization of Cmax (Makespan).

Keywords: Scheduling, Job-shop, Priority rule, Scheduling with transport.

Introduction générale

L'industrie actuelle se caractérise par une forte demande de produits personnalisés de bonne qualité et à bas prix, dans des délais de plus en plus raccourcis. En effet, l'ouverture des marchés internationaux, ainsi que l'évolution et la mondialisation ont poussé les industriels à se diriger vers des systèmes de fabrication de plus en plus flexibles, ce qui a imposé la mise en cause de plusieurs habitudes de production et en particulier la gestion des ateliers de production qui joue un rôle crucial dans la productivité et la raccourcissement des délais de production. Un système de production est dit flexible s'il peut assurer la production simultanée de plusieurs types de pièces avec des quantités variables et s'il est capable de s'adapter à la production de nouveaux produits pour lesquels le système n'a pas été étudié.[29][13]

La productivité peut être affectée directement par la qualité de l'ordonnancement des opérations sur les machines. car un atelier de production peut réaliser une grande variété de produit avec des coûts réduits, grâce à une meilleure utilisation des ressources. Le domaine d'application de l'ordonnancement est vaste : par exemple, la gestion de la charge des processus en informatique, la gestion de la production dans l'industrie, la gestion de projets, etc.

Le problème d'ordonnancement est classé parmi les problèmes fortement combinatoires, et il est toujours renouvelable, car jusqu'à maintenant, il n'existe aucune méthode de résolution générale et de faible complexité algorithmique.

On trouve une grande variété de problèmes d'ordonnancement qui sont liés à plusieurs paramètres : Les tâches ou opérations (préemptives et non préemptives, indépendantes ou non), les paramètres relatifs aux ressources (renouvelables consommables), types de contraintes sur les tâches (précédence, disjonctions), critères d'optimalité (minimisation de la durée totale d'achèvement « makespan », minimisation du retard total des tâches, minimisation de la charge des machines...) etc.

Dans les années passées les problèmes d'ordonnancement de job shop sont considérés comme des problèmes simples dont les déplacements des pièces d'une machine à une autre n'étaient pas pris en compte dans le calcul de l'ordonnancement ou bien sont considérés intégrés dans la tâche elle-même. Or, cette hypothèse est souvent non justifiée en pratique. En effet, le transport des tâches doit être intégré aux données du problème.

Dans ce contexte, le travail présenté dans ce mémoire traite l'un de ces problèmes d'ordonnancement dont le temps de déplacement des produits est considéré indépendant de la durée des tâches. Ce problème concerne un système de production de quatre machines où les pièces sont transportées par un seul AGV. Pour résoudre ce problème nous proposons une démarche de deux étapes ; la première consiste à trouver un ordonnancement des jobs sans tenir compte de la contrainte de transport dans ce cas nous avons proposé d'utiliser les règles SPT et LPT. La deuxième partie consiste à gérer le déplacement de l'AGV selon la base de l'ordonnancement défini dans la première partie. Le critère retenu ici est la minimisation de la durée totale de l'exécution des tâches, soit l'optimisation du C_{max} (Makespan).

Ce mémoire est organisé en 3 chapitres :

Dans le premier chapitre, on commence par une présentation concise des systèmes de production et de la gestion de production où apparaît le rôle de l'ordonnancement. La suite du chapitre est consacrée pour les différents types d'ateliers. Nous terminons notre chapitre par la présentation des problèmes d'ordonnancement ainsi que les différentes méthodes de résolution qui existent dans la littérature.

Le deuxième chapitre aborde le problème d'ordonnancement du job-shop avec transport où nous allons présenter : le problème d'ordonnancement job shop, le système de transport dans les ateliers et on complètera ce chapitre par un état de l'art sur le problème d'ordonnancement du job-shop avec transport.

Dans le troisième et le dernier chapitre nous allons présenter notre problème d'ordonnancement d'atelier de type Job-Shop avec contrainte de transport puis nous allons présenter la démarche de résolution proposée pour notre problématique ainsi que les résultats obtenus après l'application sur plusieurs exemples.

CHAPITRE 1

L'ORDONNANCEMENT DE LA PRODUCTION

CHAPITRE 01 : l'ordonnancement de la production

Introduction

La gestion des systèmes de production, de biens ou de services pose de très nombreux problèmes touchant la gestion de production, le marketing, la gestion des ressources humaines.

La résolution de ces problèmes nécessite l'utilisation de techniques d'optimisation et/ou d'évaluation de performances issues de domaines très variés. Nous nous intéressons dans ce chapitre à l'ordonnancement de la production et plus particulièrement aux problèmes d'atelier, où les ressources sont des machines et les tâches à ordonnancer sont des opérations à réaliser sur ces machines.

I. La production

I.1-Définition de la production

La production est le processus conduisant à la création de produits par l'utilisation et la transformation de ressources [03]. Le processus de production est alors, constitué d'un ensemble d'opération qui est les activités conduisant à la création de biens et de services. [04]

I.2-La gestion de production

I.2.1-Le rôle de la gestion de production

La gestion de production s'occupe d'un ensemble de problèmes liés à la production tels que la gestion des données, la planification, le contrôle de la production, la gestion des stocks, la prévision, l'ordonnancement etc.[23] La figure suivante représente un schéma globale montrant le rôle de gestion de la production.

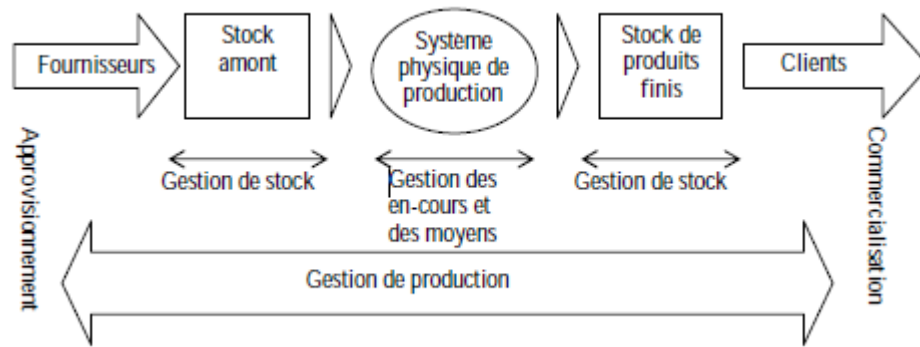


Figure (1) : Objectifs de la gestion de production [04]

La gestion de production est une fonction complexe. Cette complexité conduit généralement à la hiérarchiser afin de la simplifier et de rendre possible la gestion du système. Les niveaux hiérarchiques de la gestion de production couramment retenus sont nombre de trois: stratégique, tactique et opérationnel.

- **Le niveau stratégique:** Il s'agit de la formulation de la politique à long terme de l'entreprise (à un horizon de plus de deux ans). Elle porte essentiellement sur la gestion des ressources durables, afin que celles-ci soient en mesure d'assurer la pérennité de l'entreprise.
- **Le niveau tactique:** Il s'agit des décisions à moyen terme. Elles assurent la liaison entre le niveau stratégique et le niveau opérationnel. L'objectif est de produire au moindre coût pour satisfaire la demande prévisible, en s'inscrivant le cadre fixé par le plan stratégique de l'entreprise.
- **Le niveau opérationnel:** Il s'agit des décisions à court et à très court terme. C'est une gestion quotidienne pour faire face à la demande au jour le jour, dans le respect des décisions tactiques.[34]

I.2.2. Les systèmes flexibles de production

a. Définition des Systèmes Flexibles de Production (SFP)

[16]donne une définition du SFP en intégrant les deux aspects fonctionnel et structurel ;« *Un SFP est un système de production capable de produire différents types de pièces et qui est composé d'un ensemble de machines à commande numérique interalliées par un système automatisé de T/M. La gestion et le contrôle de ce système sont informatisés*».

b. Décomposition du Système Flexibles de Production

Comme tous systèmes de production SFP peut se décomposer en trois sous-systèmes :

le système physique de production, le système de décision et le système d'information.[26][04] Cette décomposition est structurée en fonction de la nature des flux qui traversent chaque système.

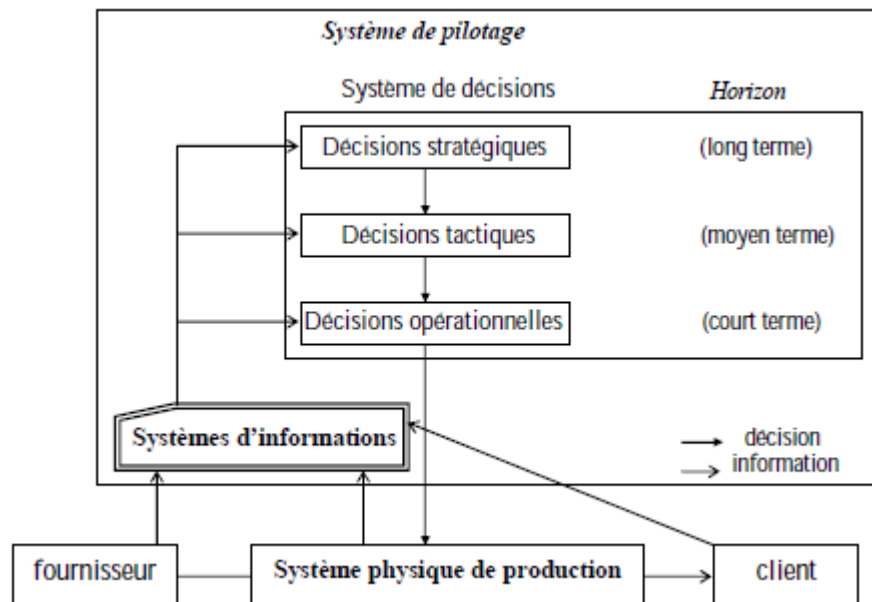


Figure (2): Les sous- systèmes constituant le système de production.[26][25]

- ✓ Le système physique de production (le système de fabrication): Transforme les matières premières ou composantes en produits finis. Ce système est composé d'un ensemble des machines à commande numérique capables de faire des changements autonomes d'outils, un système automatisé de transport/manutention/stockage (chariots filoguidés, tapis roulants, robots dédiés au déplacement des pièces...)
- ✓ Le système de décision (le système de contrôle): Contrôle le système physique de production. Il en coordonne et organise les activités en prenant des décisions basées sur les données transmises par le système d'information.
- ✓ Le système d'information: intervient à plusieurs niveaux : à l'interface entre les systèmes de décision et de production; à l'intérieur du système de décision, pour la gestion des informations utilisées lors de prises de décisions; et à l'intérieur du système physique de production. Son rôle est de collecter, stocker et transmettre des informations de différents types. [25]

I.3. Rôle de l'ordonnancement en gestion de production

La place de l'ordonnancement varie entre le niveau tactique et le niveau opérationnel. IL s'occupe de la réalisation des décisions venant de niveaux supérieurs.

Il couvre un ensemble d'action qui transforment les décisions de fabrication définies par le programme directeur de production en instructions d'exécution détaillées destinées à contrôler et piloter à court terme l'activité des postes de travail.

En sortie de la fonction ordonnancement, on obtient un planning ou ordonnancement, qui restitue l'affectation des tâches fournies en entrée à des dates précises pour des durées déterminées sur les différentes ressources. Ce planning cherche à satisfaire des objectifs, en respectant le plus possible les contraintes imposées.

1.4. Système de production non-linéaire Job-Shop

Un job shop est composé d'un ensemble de machines reliées entre elle par des ressource de transport (AGV, chariots,...).un système job shop permet de garantir la production simultanée de plusieurs types de produits. Chaque produit passe par un sous ensemble de l'ensemble des machine, l'ordre de passage des différents types de produits sur ces machines n'est pas identique.

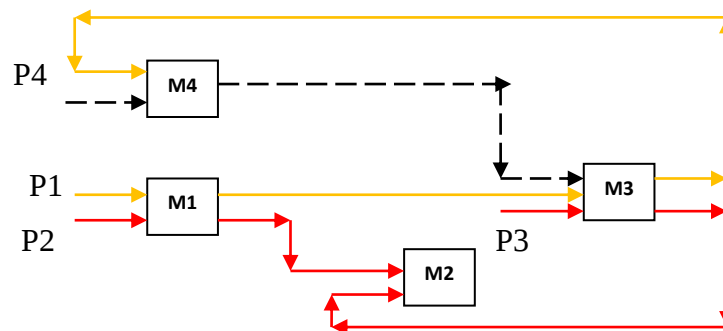


Figure (3) : système de production non linéaire job-shop

II. l'ordonnancement

II.1. Définition d'ordonnancement

« Ordonnancer, c'est programmer l'exécution d'une réalisation en attribuant des ressources aux tâches et en fixant leurs dates d'exécution »[19].

Soit un problème d'ordonnancement. Les notations suivantes sont introduites pour définir l'énoncé du problème. Soit T l'ensemble de n tâches, $T=\{T1, T2, ..., Tn\}$, P l'ensemble de m

processeurs (machines) $P=\{P_1, P_2, \dots, P_m\}$ et R l'ensemble des ressources additionnelles $R=\{R_1, R_2, \dots, R_s\}$. On peut définir le problème d'ordonnancement comme suit :

« *L'ordonnancement, de manière générale, est l'affectation des processeurs de P et (éventuellement) des ressources de R aux tâches de T dans le but de réaliser toutes les tâches en respectant les contraintes* »[09] .

II.2. Les types (classes) d'ordonnancement

Plusieurs classes d'ordonnancement sont distinguées

- ❖ **Ordonnancement sans délai** : un ordonnancement O est dit sans délai lorsqu'aucune machine n'est restée inoccupée alors que dans le stock d'entrée se trouve une tâche en attente ;
- ❖ **Ordonnancement actif** : un ordonnancement O est dit actif si tout décalage à gauche oblige à retarder l'exécution d'une autre opération ou à violer une contrainte.
- ❖ **Ordonnancement semi actif** : un ordonnancement O est dit semi-actif si on ne peut pas décaler à gauche aucune opération sans modifier l'ordre de traitement des jobs sur les ressources (machines). En d'autre terme un ordonnancement O est dit semi-actif si on ne peut décaler à gauche aucune opération sans modifier l'ordre de traitement des jobs sur les ressources (machines) ce qui oblige à retarder l'exécution d'une autre opération ou à violer une contrainte.

II.3. Les éléments d'un problème d'ordonnancement

Un *ordonnancement* est décrit, par les éléments qui le constituent. Quatre éléments sont retenus pour décrire d'une manière explicite un ordonnancement, ces éléments sont : les tâches, les ressources, les contraintes et les critères à prendre en considération lors du processus d'optimisation. Dans ce qui suit, on donnera une définition détaillée de chacun de ces éléments.

II.3.1. Les tâches :

Une tâche est un travail élémentaire dont la réalisation nécessite un certain nombre d'unité de temps (sa durée) et d'unités de chaque ressource.

Une tâche (i) est localisée dans le temps par une date de début (t_i) et/ou de fin (C_i) , et par une durée d'exécution (P_i) : telle que $P_i = t_i - C_i$ (figure 4)

Deux autres éléments définissant la tâche sont le r_i ; la date de disponibilité de la tâche et le F_i ; *représente* la durée de séjour de l'opération i sur la machine avant qu'elle redevienne disponible (flow time en anglais) ;

Si les tâches sont, liées entre elles par des conditions d'origines diverses on dit alors que les tâches sont interdépendantes, dans le cas contraire, elles sont dite indépendantes.

De même, les tâches sont dites interruptibles si elles peuvent être exécutées par morceaux par une ou plusieurs ressources ; dans le cas contraires, si on ne peut interrompre une tâche une fois celle-ci commencée, on parle alors de tâches non préemptif. [24]

La Figure (4) montre les relations entre les différentes variables représentant une tâche.

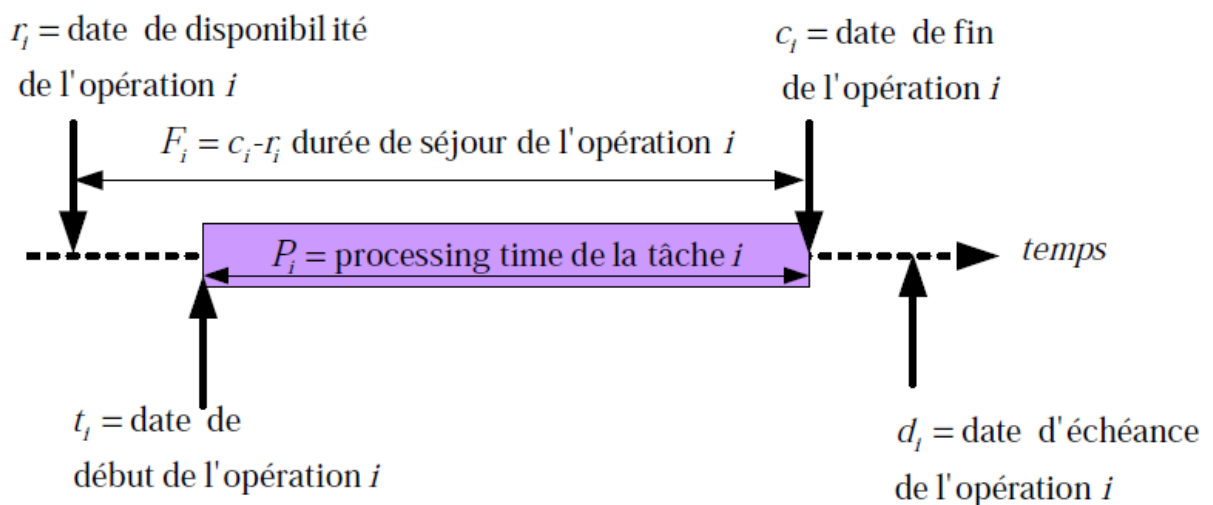


Figure (4) : Caractéristiques d'une tâche faisant référence à l'exécution d'une opération. [15]

II.3.2. Les ressources :

Les ressources sont les moyens ou les outils nécessaires à la réalisation des tâches. On distingue deux grande familles de ressources ;

- Les ressources renouvelables sont les ressources réutilisables après la fin d'exécution des opérations. Cette famille est constituée des machines, d'hommes et des outils de productions. Souvent des ressources additionnelles sont considérées comme par exemple l'utilisation des robots, ou des moyens de transport, ou des ressources d'un autre genre comme les buffers d'entrée/sortie. Parmi ces ressources on distingue aussi, des ressources disjonctives qui ne peuvent exécuter qu'une seule opération à la fois et des ressources cumulatives qui peuvent exécuter plusieurs opérations à la fois c'est le cas par exemple des machines parallèles ; notons qu'une machine fait partie de l'ensemble des ressources

renouvelable Une machine de capacité 1 est une ressource disponible en un unique exemplaire.

- *La deuxième famille* est dite consommable et regroupe toutes les ressources qui deviennent indisponibles après une utilisation, c'est le cas des matières premières de l'argent, de l'énergie etc. [15]

II.3.3. Les contraintes

Selon [13] les contraintes expriment les restrictions que peuvent prendre conjointement les variables de décision. Donc les contraintes nous renseignent sur les limites imposées par l'environnement. On distingue plusieurs types de contraintes, qui sont [12], [13]:

- **les contraintes de localisation temporelles** : Pour la réussite d'un projet ou d'un plan de production, on doit se soumettre aux impératifs de production (réalisation) traduits par le respect d'un échéancier fixé auparavant. Au niveau global on parlera d'une date de lancement d'un projet et d'une date de livraison. A un autre niveau de détail plus fin, pour une tâche ou une opération. Ce dernier niveau se traduit par la définition des dates suivante: (i) date de disponibilité : r_i ; (ii) date d'échéance : d_i .
- **les contraintes de précédence** : une contrainte qui lie le début d'une activité à la fin d'un autre est appelée contrainte de succession ou de précédence. Les gammes opératoires sont un exemple de ces contraintes, d'autres contraintes sont imposées dans certain cas telles que les contraintes de synchronisation, de simultanéité, ou de recouvrements etc. [13]
- **Les contraintes de ressources** : Les contraintes de ressources représentent le fait que les activités requièrent tout au long de leur exécution une certaine quantité de ressources [12]. Les contraintes de ressources concernent les deux points suivants [13] :
 - (i) l'utilisation de ces ressources, (leur nature, la quantité nécessaire, et les caractéristiques d'utilisation) ;
 - (ii) la disponibilité et la quantité de ces ressources.

Ces contraintes nous donnent une information précise sur la nature de l'atelier, et influencent le choix des méthodes d'optimisation ; Il est important de noter qu'en fonction des objectifs de l'étude, ces contraintes peuvent être strictes ou non. Selon [11], lorsqu'elles sont strictes, elles constituent des obligations à respecter. Lorsqu'elles ne sont pas strictes, on peut ne pas les satisfaire et on les appelle alors des contraintes de préférence.

II.3.4. Les critères d'optimisation

Les critères constituent les objectifs qu'on souhaite optimiser. En fonction du nombre d'objectifs on parle d'optimisation mono-objective (on s'intéresse à l'optimisation d'un seul critère) et d'optimisation multi-objectifs (on s'intéresse à l'optimisation de plusieurs critères).

L'optimisation s'exprime par une fonction de minimisation ou maximisation de ces critères.

Dans [10] on trouve la classification suivante :

- ❖ Les critères liés aux dates de fin de livraison ; constituent la catégorie des critères les plus étudiés en optimisation : on retrouve le makespan $C_{max} = \max(c_i)$ qui représente la fin d'achèvement de toutes les tâches du problème, la somme des dates de fin $C_\Sigma = \sum(c_i)$, la somme pondérée des dates de fin $\sum w_i C_i \dots$
- ❖ Les critères liés aux volumes des encours ; dans cette catégorie on s'intéresse au nombre de tâches en cours d'exécution à chaque instant t et on cherche généralement à minimiser le nombre de tâches en attente ou à maximiser le nombre de tâches terminées sur la dernière machine.
- ❖ Les critères liés à l'utilisation des ressources permettent de satisfaire un objectif donné et d'autre part d'évaluer des critères de performance, tels que, l'utilisation moyenne des ressources, le temps d'inactivité de l'atelier [10]

La combinaison entre les différents critères est possible : maximisation des profits et minimisation de tous les coûts inhérents à la production. On note aussi que des similitudes et des équivalences entre critères et que des réductions permettent de déduire certains critères de certains d'autres.

II.4. Notation et définition des problèmes d'ordonnancement

La notation $\alpha|\beta|\gamma$ a été proposée pour la première fois par [17]. Cette notation constituée de 3 paramètres ayant chacun une signification différente : le paramètre α nous renseigne sur l'environnement d'usinage, le type d'atelier de production et le nombre de machines présentes dans le système, tandis que le champ β concerne les caractéristiques des problèmes d'ordonnancement et regroupe les contraintes imposées par l'environnement de production et par les ressources. Le champ γ concerne les critères de performance qui vont représenter les objectifs à atteindre pour le problème en question. Le Tableau 1, donne les différentes valeurs possibles pour les 3 paramètres $\alpha|\beta|\gamma$ de reprendre et d'étendre les propositions faites dans [08] .

Exemple des notations : : $\alpha_1=J$, $\alpha_2=4$ | $\beta_2=prec$, $\beta_7=J$ $\gamma=C_{max}$ est un problème de type job-shop où on a besoin de traiter 5 jobs sur 4 machines, avec les contraintes de précédence entre les opérations.

Classe	factor	Description	Description en langue française	Valeur
α	α_1	Manufacturing system type	Type d'atelier	$P, Q, R, X, G, O, J, F, PMPM$
	α_2	Number of machines	nombre de machine	O, K
	α_3		Extension du champ α Pour l'adapter à chaque problème	
	α_4		Extension du champ α Pour l'adapter à chaque problème	
β	β_1	Job/operation preemption	Job/opération avec préemption	$O, pmtn$
	β_2	Precedence constraints	Contraintes de précédence	$prec, chain, tree, sp- graph$
	β_3	Ready times		O, r_j
	β_4	Restrictions on processing times	restriction sur les temps opératoires des jobs	$P_j = 1, P_{ij} = 1, P_j = p, P_{inf} \leq P_j \leq P_{sup}$
	β_5	Due dates (deadlines)	date au plus tôt et date au plus tard	O, d_j
	β_6	Batches/families processing	traitement par lot ou par catégories	$O, batch$
	β_7	Number of jobs or tasks in a job	nombre de jobs ou nombre de tâches dans un job	$O, n_j O$

	β_8	Job/task priorities	prioritésdes Job/tâches	O, w_j
	β_9	Dynamic machine availability	Disponibilité de machine dynamique	$O, avail$
	β_{10}	Additional/auxiliary resources	ressourcesadditionnelleouauxiliaires	O, aux
	β_{11}	Buffers	stocks	$O, no\ wait$
	β_{12}	Setup (changeover)	temps de reconfiguration	$O, setup^*$
γ	γ	Performance Measure	indice de mesure ou de performance	$C_{max}, \sum C_j, \sum w_j C_j, L_{max}, \sum T_j$

Tableau(1) : les différentes valeurs de paramètres α, β, γ

II.5. Les ateliers de production

Les systèmes de production sont divers et variés et ont différentes organisations au niveau des ateliers de production. Ces organisations déterminent les problèmes d'ordonnancements.

Un atelier est caractérisé par le nombre de machines qu'il contient et par son type, ainsi on trouve :

II.5.1. Une seule machine

C'est le plus simple, chaque tâche J_j de durée de traitement P_j s'exécute sur une machine qui ne peut traiter plus qu'une tâche à la fois.

II.5.2. Machines identiques en parallèle

Il y a m machines identiques en parallèle ; chaque tâche peut être exécutée indifféremment sur une des m machines, Celles-ci peuvent cependant avoir des vitesses différentes (à l'opposé des vitesses uniformes).

II.5.3. Les ateliers de type flow-shop

Chaque ordre de fabrication doit être traité par chacune des m machines en série et dans le même ordre. Toutes les tâches ont donc le même routage. En temps identiques, Dans les ateliers de type flow-shop hybride, une machine peut exister en plusieurs exemplaires identiques fonctionnant en parallèle.

II.5.4. Les ateliers de type job shop

Chaque tâche possède son propre routage, Une tâche peut revenir une seconde fois sur la même machine .C'est le phénomène de recirculation.

II.5.5. Les ateliers de type open-shop

C'est un type d'atelier moins contraint que le type Flow shop et le type job shop, puisque l'ordre des opérations n'est pas fixe à priori, Le problème d'ordonnancement consiste d'une part à déterminer le cheminement de chaque travail et d'autre part à ordonnancer les travaux en tenant compte des gammes trouvées, les opérations peuvent être effectuées dans n'importe quel ordre.

II.6. Représentation des problèmes d'ordonnancement

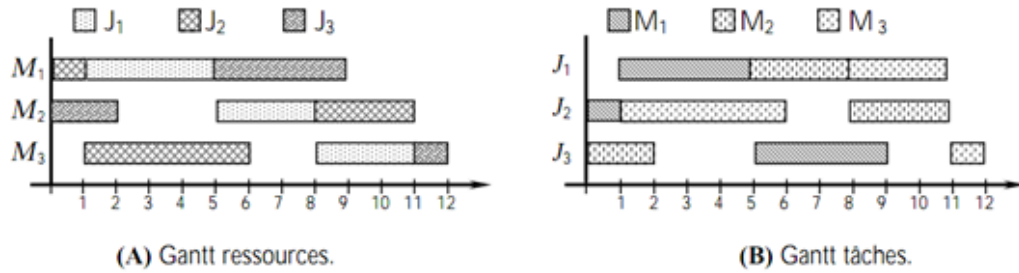
Il existe des sortes de représentations possibles d'un problème d'ordonnancement : le diagramme de Gantt, le graphe Potentiel-Tâches et la méthode PERT.

II.6.1. Le diagramme de Gantt :

Le but de ce diagramme est de représenter de manière claire et rapide la solution à un problème d'ordonnancement. Deux types de diagramme de Gantt sont utilisés : Gantt ressources et Gantt tâches (figure 5)

Le diagramme de Gantt ressource, est composé d'une ligne horizontale pour chaque ressource (machine). Sur cette ligne, sont visualisées les périodes d'exécution des différentes opérations en séquence et les périodes de l'oisiveté de la ressource. [13]

Le diagramme de Gantt tâches permet de visualiser les séquences des opérations des tâches, en représentant chaque tâche par une ligne sur laquelle sont visibles, les périodes d'exécution des opérations et les périodes où la tâche est en attente des ressources.



Figure(5) :diagramme de Gantt[5]

II.6.2. Graphe Potentiel-Tâches

Cette outil graphique a été développé grâce à la théorie des réseaux de Pétri qui ont surtout servi à modéliser les systèmes dynamiques à événements discrets [18].

Dans ce genre de modélisation, les tâches sont représentées par des nœuds et les contraintes par des arcs [19], comme le montre la figure 6. Ainsi, les arcs peuvent être de deux types :

- les arcs conjonctifs illustrant les contraintes de précédence et indiquant les durées des tâches,
- les arcs disjonctifs indiquant les contraintes de ressources [20], [21].

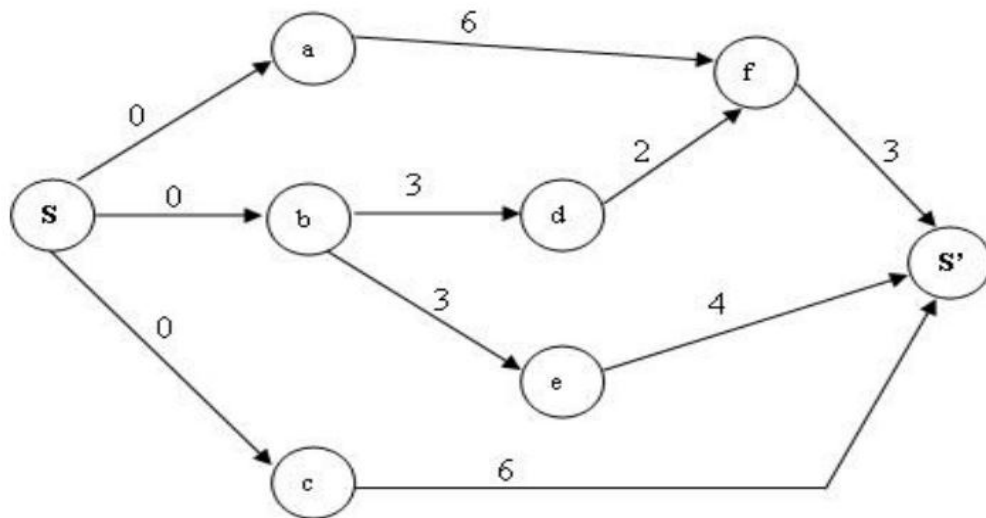


Figure (6)- Graphe Potentiel-Tâches d'un ordonnancement

La durée du projet, qui correspond au plus long chemin entre S (tâche de début du projet) et S' (tâche de fin du projet).

II.6.3. Method PERT (Program Evaluation and Research Task)

Cette représentation, semblable à la précédente, permet de représenter une tâche par un arc, auquel est associé un chiffre qui représente la durée de la tâche. Entre les arcs, figurent des

cercles, appelés sommets ou événements, qui marquent l'aboutissement d'une ou de plusieurs tâches. Ces cercles sont numérotés afin de suivre l'ordre de succession des divers événements.

Les méthodes graphiques ont connu une très importante évolution surtout avec l'apparition des Réseaux de Pétri (RdP) [22], [23], qui permettent de traduire plusieurs notions fondamentales ayant un lien avec les problèmes d'ordonnancement, telles que : les conflits sur les ressources, les durées opératoires certaines, les durées opératoires aléatoires, les gammes, les disponibilités, les multiplicités et les capacités de ressources la répétitivité, etc ...

II.7. Méthodes de résolution

Au fil des années, de nombreuses méthodes de résolution de problèmes ont été proposées. Ainsi, une grande variété de concept et principe, de la stratégie et des performances ont été discernées. Cette variété et ces différences ont permis de regrouper les différentes méthodes de résolution de problèmes d'ordonnancement en deux classes principales : la classe de méthodes exactes et la classe des méthodes approchées.

II.7.1 Les Méthodes Exactes

Elles recherchent un ordonnancement optimal qui minimise ou maximise un des critères présentés ou une combinaison de plusieurs critères. Les techniques utilisées sont les méthodes par séparation et évaluation, la programmation dynamique, la déduction mathématique, la théorie des jeux, la théorie des graphes etc. Cette technique est coûteuse en temps de calcul.

II.7.2 Les Heuristiques

Les méthodes dites heuristiques sont des méthodes spécifiques à un problème particulier. Elles nécessitent des connaissances du domaine du problème traité. En fait, ce sont des règles empiriques qui se basent sur l'expérience et les résultats acquis afin d'améliorer les recherches futures

II.7.3 Les Méta-heuristiques

Les méthodes dites méta heuristiques sont des méthodes générales, des heuristiques polyvalentes applicables sur une grande gamme de problèmes. Elles peuvent construire une alternative aux méthodes heuristiques lorsqu'on ne connaît pas l'heuristique spécifique à un problème donné. [07].

Plusieurs types de méta heuristique sont proposés :

- Le recuit simulé : Cette technique provient de l'observation de la formation d'une structure cristalline quand un métal refroidit. Le recuit simulé est une technique permettant de trouver, au bout d'un temps raisonnable, une solution pour des problèmes de complexité élevée. Elle est utilisée pour la détermination de la répartition statique des tâches. Elle est considérée comme technique stochastique et ne garantit pas de solution optimale.

-Les algorithmes génétiques : Ils sont fondés sur les principes de l'évolution biologique des espèces. Comme le recuit simulé, ils permettent de trouver au bout d'un temps raisonnable des solutions pour des problèmes de complexité élevée. Ils sont facilement parallélisation. Les algorithmes génétiques sont basés sur une notion étendue de voisinage. Plusieurs solutions sont maintenues simultanément et évoluent par des techniques de recombinaison et de mutation. des difficultés rencontrée est le codage des solutions.

- Les réseaux de neurones : Ils ont été appliqués à la résolution approchée des problèmes d'optimisation en général et à d'ordonnancement en particulier. Leur avantage est la possibilité d'utiliser des circuits analogiques dont l'évolution converge vers des états stables correspondant à une solution approchée du problème d'ordonnancement. Grâce à leur vitesse élevée de convergence, leur utilisation en ligne peut être envisageable. Leur inconvénient est le blocage à la rencontre d'un minimum local.
- La logique floue : Elle apporte des solutions pour les problèmes qu'on ne sait pas spécifier complètement et quantitativement. On associe un degré de satisfaction à la terminaison de chaque tâche si elle se termine avant sa date critique, la satisfaction est totale. Le degré d'insatisfaction augmente au fur et à mesure que l'on s'éloigne de cette date critique.

Conclusion :

La préparation d'un bon ordonnancement est un processus très important qui influence directement la productivité du système, quel que soit le type de système de production, En effet un mauvais ordonnancement signifie une utilisation non adéquate de ressources de production et donc, une consommation physique (ressource) et temporelle (temps de production) élevée, qui se traduit par une performance basse du système de production.

Les problèmes d'ordonnancement sont des problèmes combinatoire qui sont classé parmi les problèmes d'optimisation les plus difficiles. Le job shop avec contrainte de transport est l'un des problèmes d'ordonnancement les plus complexe. Alors le deuxième chapitre permettra de présenter une vue globale sur ce problème et sa résolution.

Chapitre 02 : Les Problèmes D'ordonnancement De Job Shop Avec Transport

Chapitre 02 : Les Problèmes D'ordonnancement De Job Sop Avec Transport

Introduction :

Avant les problèmes de job shop sont considérés comme des problèmes simples dont les déplacements des pièces d'une machine à une autre n'étaient pas pris en compte dans le calcul de l'ordonnancement ou bien sont considérés intégrés dans la tâche elle-même. Or, cette hypothèse est souvent non justifiée en pratique. En effet, le transport des tâches doit être intégré aux données du problème. Ce présent chapitre est consacré la présentation de ce type de problème

I. Le Problème d'ordonnancement Job-Shop

I.1. Atelier à cheminement multiple (job shop)

Dans un atelier de type job shop chaque tâche emprunte un chemin qui lui est propre, En effet les tâches ne s'exécutent pas sur toutes les machines dans le même ordre.

Ce type d'organisation correspond généralement à une production par lot, notamment dans une unité de production disposant de moyens polyvalent utilisés suivent des séquences différentes afin de réaliser des produit divers [14].

Si, pour chaque machine, on dispose d'un et un seul exemplaire, on dit que l'organisation est un Job Shop simple. Si, au contraire, pour un au moins des machines (postes de travail), on dispose de plus d'un exemplaire, on l'appelle Job Shop Hybride. (Comme il est montré dans la figure 7)

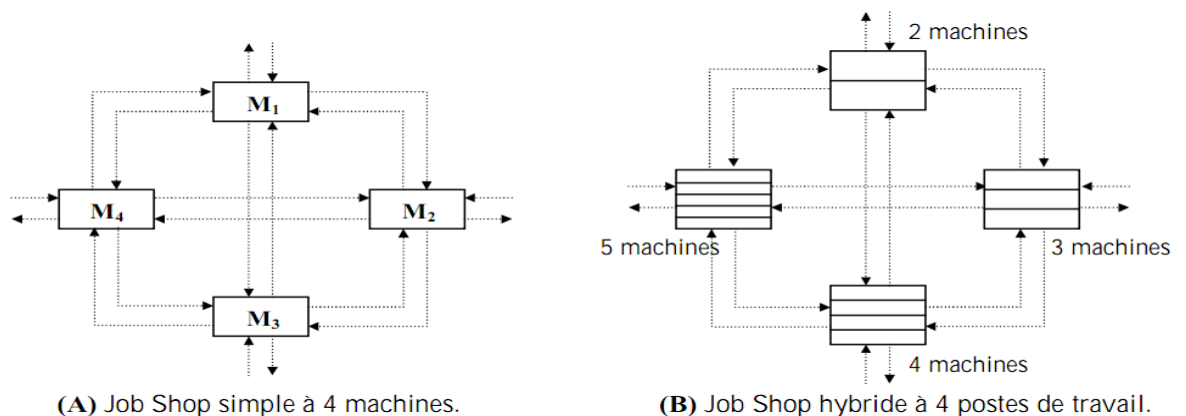


Figure (07) : Exemple de job shop simple et hybride.

I.2. Types de job shop

I.2.1. Job shop classique

Le problème d'ordonnancement de job shop constitue l'un des problèmes d'ordonnancement d'atelier les plus étudiés dans la littérature, les variations autour du problème de job shop sont nombreuses et il n'est pas possible de trouver dans la littérature une formulation unique de celui-ci. Nous présentons ici la formulation la plus générale possible du problème de job shop simple.

Le problème d'ordonnancement de type job shop consiste à réaliser un ensemble de n jobs sur un ensemble de m machines. En cherchant d'atteindre certains objectifs. Chaque job J_i est composé d'une suite de n_i opérations devant être exécutées sur les différentes machines selon un ordre préalablement défini. Par ailleurs, un ensemble de contraintes concernant les machines et les tâches souvent existantes, comme :

- Les machines sont indépendantes les unes des autres.
- Une machine ne peut exécuter qu'une seule opération à la fois à un instant donné.
- Les machines sont disponibles pendant toute la durée de l'ordonnancement.
- Les jobs sont indépendants les uns des autres. En particulier, il n'existe aucun ordre de priorité attaché aux jobs.

La figure (8) représente un problème type de job shop classique composé de 3 jobs et 4 machines, les gammes opératoires sont les suivantes:

J1 :M1-M2-M4.J2 :M2-M3-M1.J3 :M4-M3-M2.

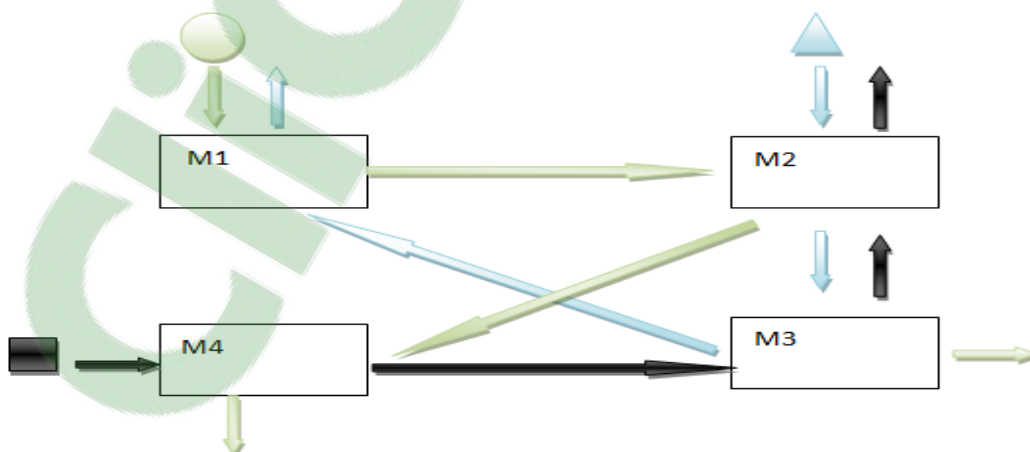


Figure (8): job shop classique à 3 jobs et 4 machines.

I.2.2. Job shop flexible

Le job shop flexible est une extension du modèle job shop classique. Sa particularité essentielle réside dans le fait que chaque opération peut être exécutée sur plusieurs machines. Dans ce modèle, les machines qui effectuent la même opération sont groupées dans un même étage. Il s'ensuit qu'il offre plus de flexibilité par rapport au job shop classique grâce à la polyvalence de ces machines. Toutefois, cela induit une complexité supplémentaire due à la nécessité de la détermination des affectations adéquates avant d'établir l'ordre de passage des différentes opérations sur les machines. Un exemple de ce problème à m étages et trois machines maximum par étage, est donné dans la Figure(9). [24]

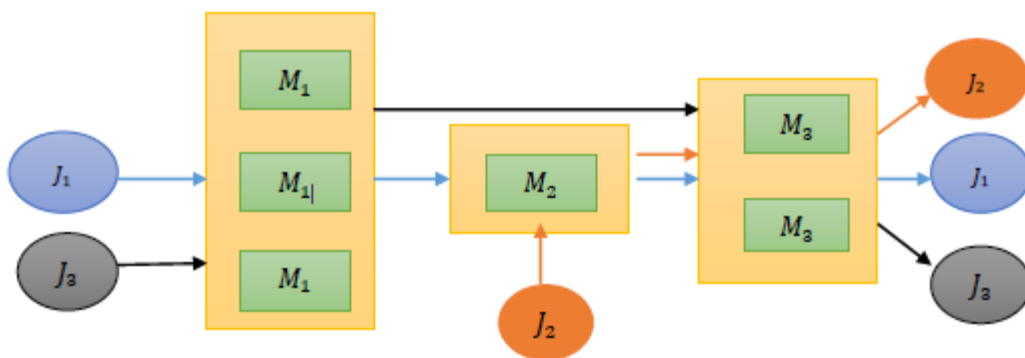


Figure (9): Représentation d'un système de type Job-shop flexible à deux étages.

Une remarque importante dans les problèmes de type job-shop hybride est que le nombre de machines peut varier d'un étage à l'autre, ainsi que les performances des machines qui ne sont pas forcément les mêmes pour toutes les machines d'un même étage. Ces performances nous permettent de classer les systèmes avec machines parallèles en trois groupes, à savoir :

- Machines identiques : La durée d'exécution d'une tâche est la même sur toutes les machines.
- Machines uniformes : La durée d'exécution d'une tâche est similaire sur toutes les machines.
- Machines indépendantes : La durée d'exécution d'une opération dépend de la machine sur laquelle elle est exécutée.

II. Système de transport dans les ateliers

Dans les ateliers de production un système de transport peut être défini comme un système intégré qui sert à déplacer, contrôler, et stocker/déstocker les matières. Cette action de transport est assurée par l'effet de gravité, des efforts manuels ou machines utilisant des énergies.

Plusieurs catégories d'équipements sont utilisées dans la manutention dans les installations de production et dans les entrepôts

II.1.1. Les convoyeurs :

Les convoyeurs sont utilisés lorsque des quantités importantes de produits doivent être transportées entre les localisations à travers une trajectoire fixe. Ils peuvent être motorisés ou non-motorisés, il existe plusieurs types de convoyeurs : Transporteur à rouleaux et à galets, Transporteur à courroie, ...etc.

Les systèmes de convoyeurs motorisés sont classés selon leurs fonctionnements et caractéristiques en : convoyeur à sens unique, convoyeur en boucle continue.

II.1.2. Robot industriel :

Un robot est une machine capable d'affecter des tâches et de manipuler des objets selon un programme de façon automatique. Ils sont généralement utilisés pour remplacer les humains dans des situations où ces derniers sont incapables d'effectuer le travail, des situations plus dangereuses, des hautes précisions ou répétitives.

Les robots industriels se distinguent souvent par un système articulé semblable à un bras humain. Ce bras peut se déplacer, dans certains cas, sur plusieurs axes et est muni d'un outil propre à l'opération à effectuer : l'organe effecteur. Les applications typiques de ces robots en industrie sont le soudage, la peinture, la manipulation, l'assemblage et l'inspection.

Les robots sont de plus en plus utilisés en industrie, car ils sont très rapides, précis et ne se fatiguent jamais.

II.1.3. Le Véhicule Auto-Guidé (AGV) est un véhicule automatisé qui se déplace tout seul dans l'entreprise grâce à des points de marquage, des rails ou des lasers. Il peut notamment soulever, pivoter, empiler et transporter des charges sur une distance plus longue. Il s'agit d'un engin

autonome intelligent, capable de réaliser des tâches complexes, de travailler dans des entrepôts automatisés, d'opérer de manière indépendante dans des environnements dangereux. Plusieurs avantages des AGV sont distingués:

- Toujours le trajet le plus optimal : grâce à l'architecture multi-agent révolutionnaire, les AGV sont autopilotés et intelligents.
- Disponibilité maximale de flotte : Grâce à la 'powercharge' de la technologie de batterie au lithium, la batterie peut être chargée dans un délai de 8 minutes.
- Large gamme de chariots de magasinage AGV.
- Utilisation hybride possible grâce à un pilotage manuel très ergonomique.

III. Ordonnancement job shop avec contrainte de transport

III.1. Définition du problème

Dans les années passées les problèmes de job shop sont considérés comme des problèmes simples dont les déplacements des pièces d'une machine à une autre n'étaient pas pris en compte dans le calcul de l'ordonnancement ou bien sont considérés intégrés dans la tâche elle-même. Or, cette hypothèse est souvent non justifiée en pratique. En effet, le transport des tâches doit être intégré aux données du problème.

Dans de tels systèmes, le transport d'une tâche peut être décrit comme le chargement sur le matériel de transport, du transport entre deux machines et du déchargement de la tâche. Il est donc possible de, soit considérer l'opération de transport comme englobant toutes ses actions, soit de les considérer séparément, chacune devenant une opération distincte

La prise en compte de la manipulation matérielle amène à étendre la notation $\alpha/\beta/\gamma$ de Graham *et al.* [17] vu dans le Chapitre 1. La notation est étendue à $\alpha(\kappa)|\beta/\gamma$ ou K dénote le nombre de moyen de transport dans le système (soit par exemple dans notre cas les AGV). [28].

III.2. Etat de l'art

Dans [06] on trouve, un bon état de l'art concernant le job-shop de son début jusqu'à la fin des années 90. Il liste les articles en les classant par méthode de résolution utilisée pour le JS.

Durant les vingt dernières années furent créés des algorithmes plus performants utilisant entre autre la relaxation lagrangienne qui consiste en l'abandon de certaines contraintes pour résoudre des problèmes plus proches de la réalité avec plus d'une centaine de jobs et d'une cinquantaine de machines. Cette technique de relaxation fut travaillée afin d'associer à chaque contrainte relâchée une pénalité appelée technique de relaxation lagrangienne augmenté.

Les premières méthodes sont des heuristiques de construction apparues à la fin des années 50 et début des années 60 comme celles proposées par [27] et [05].

Les années 80, de l'apparition des Méta-heuristiques telles que les algorithmes génétiques, la recherche Tabou et le recuit simulé.

L'année 2010 Mohand Larabi dans sa thèse [15]: a traité le problème du job-shop auquel il ajoute des contraintes de transport. Son apport consiste en l'ajout de contraintes de deux types

- la prise en compte de plusieurs moyens de transport de capacité unitaire ;
- la prise en compte de plusieurs moyens de transport de capacité non unitaire.

Pour la résolution de ces problèmes, il généralisa démarche classique appliquée au job-shop et qui repose :

- sur la notion de graphe disjonctif non orienté pour modéliser un problème ;
- sur la notion de graphe disjonctif orienté pour modéliser une solution ;
- sur la notion d'algorithme de plus long chemin pour le calcul des dates au plus tôt des opérations.

Dans [02] un couplage méta-heuristique/simulation appliqué au problème de job shop avec transport. Cette solution consiste à simuler le problème de job shop (4 machines avec 2 AGV). Il utilise les méta-heuristiques (algorithme génétique et recherche local de recuit simulée) pour minimiser le Makespan.

La thèse [01] présente la résolution du problème job shop à deux machines et un seul convoyeur. Le critère d'évaluation de la qualité d'une solution, est celui du Makespan, deux approches de résolution approchée à été proposé, Le premier est constructif : trois algorithmes, basés sur des règles de priorité, La seconde approche de résolution est itérative: l'algorithme de recherche avec tabous, L'algorithme de recherche avec tabous donne, en moyenne, de meilleures solutions que les algorithmes basés sur les règles de priorité.

Conclusion

Plusieurs travaux de recherche ont été faits pour trouver une solution au problème job shop avec contrainte de transport dont ils ont appliqué des méthodes exactes et des méthodes approchées leur objectif est toujours de rapprocher plus à la réalité.

Dans le chapitre suivant nous allons, de notre part, de donner notre proposition sur la résolution de ce problème où le système de transport est constitué d'un seul AGV.

Chapitre 03 :

**Résolution De Problème D'ordonnancement De
Job Shop Avec Contrainte De Transport**

Chapitre 03 : Résolution de Problème D'ordonnancement De Job Shop avec Contrainte de Transport

Introduction

Dans ce chapitre nous allons présenter notre problème d'ordonnancement ainsi que la démarche de résolution proposée.

L'objectif du travail est de trouver, en utilisant des méthodes heuristiques, les meilleures solutions possibles pour un ordonnancement sous la contrainte de transport, tout en minimisant le temps total d'exécution (Makespan). La résolution de ce problème se heurte à des tailles mémoire et de temps de calcul trop importants. L'objectif n'est plus alors d'obtenir systématiquement l'optimum mais plutôt d'obtenir une solution proche de l'optimum ou de « bonne qualité » en un temps minimal.

I. Description du système étudié

Le système étudié dans ce mémoire est un système de production de type job shop.

Son principe de fonctionnement est inspiré d'un système réel qui existe dans le laboratoire de productique MELT (université de Tlemcen)

Ce système est constitué de quatre machines (M1, M2, M3 et M4) positionnées sous forme d'un carré (Figure 10).

Les produits (pièces) sont stockés dans un stock S1 que l'on suppose d'une capacité illimité.

Ces produits sont transportés, devant les machines par un seul AGV. Ce dernier a une capacité de transporter une pièce à la fois.

Un bras manipulateur est installé à coté de chaque machine qui sert à déplacer les pièces d'AGV vers les machines et l'inverse.

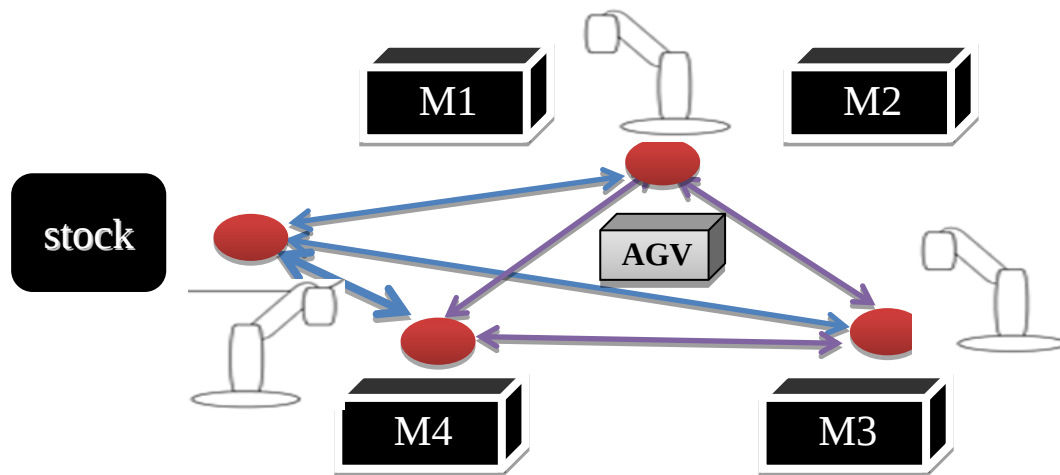


Figure 10 : le schéma d'atelier .

Le tableau suivant regroupe l'ensemble des temps de transport des produits entre les différents postes de travail :

	M1	M2	M3	M4
M1	0	17	14	14
M2	17	0	14	14
M3	14	14	0	0
M4	14	14	0	0

Tableau(02) : les temps de transport entre les machines.

II. Contraintes et notations spécifiques au problème

La problématique que nous traitons, elle est dénotée par $J4(1)/\beta1/C_{max}$ selon la notation la plus utilisée dans la littérature.[28]

- Fonction objectif : Minimisation de Makespan (C_{max})
- Les contraintes
 1. Une machine ne peut traiter qu'un seul produit à la fois
 2. Un produit ne peut s'exécuter que sur une seule machine à la fois
 3. $\beta1 = \emptyset$: Pas de préemption (le produit doit terminer toute l'opération avant de quitter la machine)
 4. Chaque machine possède une file d'attente avec une capacité illimitée.
 5. Tous les produits sont disponibles à $t=0$.
 6. L'AGV ne transporte qu'un seul produit à la fois.

II.1. Notations utilisées

Dans notre travail nous avons proposé une notation (le tableau 03) qui sera utilisée par la suite dans la résolution du problème.

Notation	Description
N	Nombre total de jobs
M	Nombre total de machine
DM(k)	Représente la disponibilité pour la machine k
DJ(j)	Représente la disponibilité pour le job j.
Agv	Temps de transport entre les deux machines k et k'
emj(j)	Ancienne machine pour le job (j)
pmj(j)	Prochaine machine pour le job(j)
Dagv	Représente la disponibilité de l'AGV.
NT	Nombre d'opérations
Emagy	Ancienne machine pour l'AGV.
$t_{i,j}$	Le temps de départ de l'opération de job j dans la machine i
$p_{i,j}$	Le temps d'opérateur de job j dans machine i
$tf_{i,j}$	temps de fin d'opération j dans machine i

Tableau (03) : la notation

III. Démarche de résolution

Dans cette partie nous allons présenter la démarche de résolution ainsi que l'algorithme proposé pour résoudre notre problème, en essayant d'expliquer en détail comment nous avons implémenté les deux règles SPT et LPT .

III.1. Description de la solution proposée

Pour encadrer notre problématique nous avons proposé un ensemble d'hypothèses concernant le fonctionnement de notre système

- Les produits sont toujours disponibles à l'instant $t=0$.
- Chaque machine possède une file d'attente avec une capacité illimitée.
- Pas de panne de machines.
- Pas de panne de l'AGV.
- Le temps de changement de production n'est pas pris en compte.
- L'AGV se déplace entre les machines selon un programme qui sera défini par la résolution du problème.
- Pas de recirculation (un produit ne peut être exécuté sur une machine qu'une seule fois)
- En cas de conflit dans la séquence, le choix de produit à déplacer se fait selon les règles proposées.
- On ne prend pas en considération le temps de déplacement du ou vers le stock.

Notre objectif est de trouver le C_{max} minimum.

Calcul de C_{max} :

Pour déterminer le C_{max} nous avons utilisé les méthodes heuristiques (SPT/LPT). Elles sont représentées par la méthode de diagramme de Gantt .

SPT (Shortest Processing Time) : la tâche ayant le temps opératoire le plus court est traitée en premier lieu.

LPT (Longest Processing Time) : la tâche ayant le temps opératoire le plus important est ordonnée en premier lieu.

C_{max} : représente le temps de sortie du dernier produit exécuté dans le système.

Ce problème peut être divisé en deux parties :

- 1- Partie d'ordonnancement
- 2- Partie de transport d'AGV
- Partie d'ordonnancement :
 - a- Les jobs sont ordonnancés dans chaque machine selon l'heuristique choisie.
 - b- Le routage des jobs dans le système suit l'heuristique choisie.

L'exemple suivant explique notre algorithme :

Nous avons le problème de $J3|S1|C_{max}$.

Pour résoudre ce problème, on choisit l'heuristique SPT.

Jobs / Machines	M1	M2	M3
J1	2	3	1
J2	1	2	3

Tableau(04) : les temps opératoires des jobs.

- a- La séquence des jobs dans les machines selon SPT :

M1 : J2-J1.

M2 : J2-J1.

M3 : J1-J2.

- b- Le routage des jobs sur les machines selon SPT :

Job / opération	Opération 1	Opération 2	Opération 3
J1	M3	M1	M2
J2	M1	M2	M3

Tableau (05) : routage de jobs

la figure (11) présente le diagramme de gant de cet exemple.

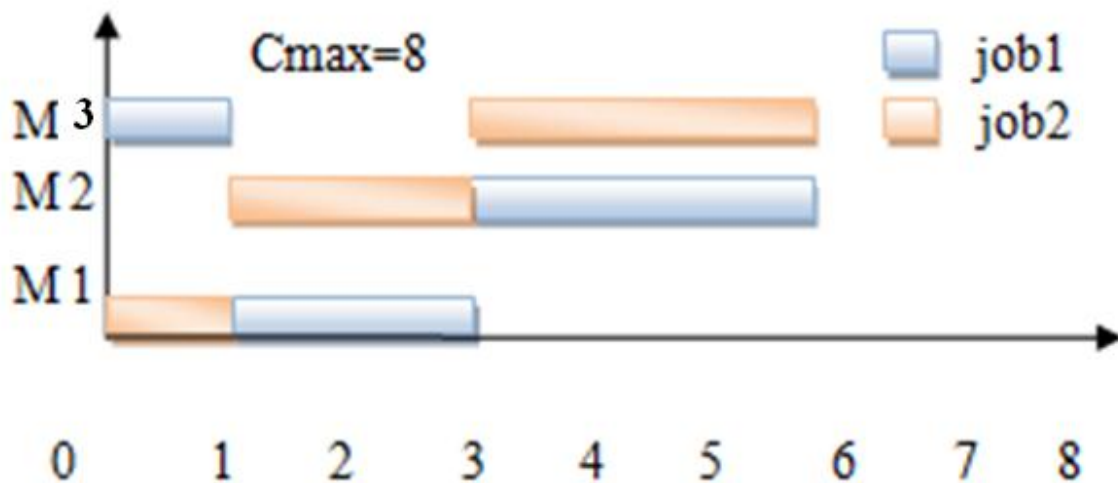


Figure (11) : diagramme de Gantt du problème de $J3|\beta_1|C_{max}$.

- **Partie de transport d'AGV :**

La figure (12) montre la démarche de cette partie.

- I- $dt = \max(D_{agv}, DJ[j])$: pour transporter le job j sur l' AGV , il faut que les deux (job et AGV) sont disponibles.

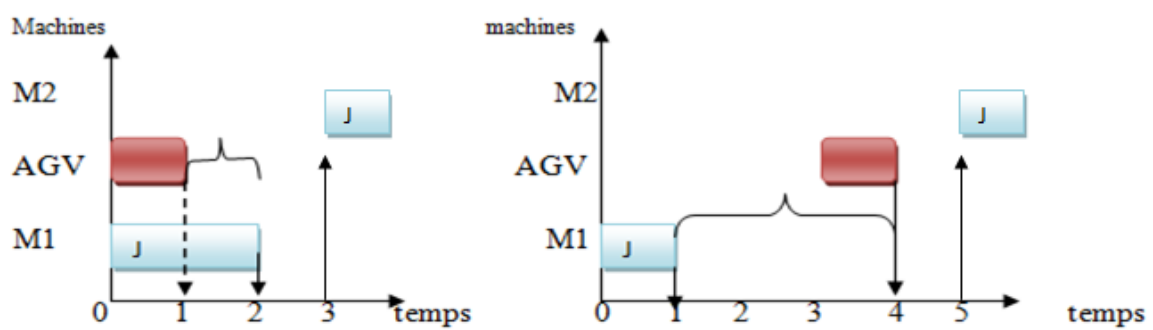


Figure (12) : disponibilité de job et AGV.

a- $dt = \max(1, 2)$

$dt=2$. Le temps de départ de transport de job par l'AGV . En ce cas, AGV attend la disponibilité de job (j).

b- $dt = \max(4, 1)$

$dt=4$. Le temps de départ de transport de job par l'AGV . En ce cas, le job (j) attend la disponibilité l'AGV.

II- $dt=dt+agv(emagv,emj(j))+ agv(emj(j),pmj(j))$:le temps de transport de job par l'AGV.

a- $dt=2+1, dt=3$.

b- $dt=4+1, dt=5$

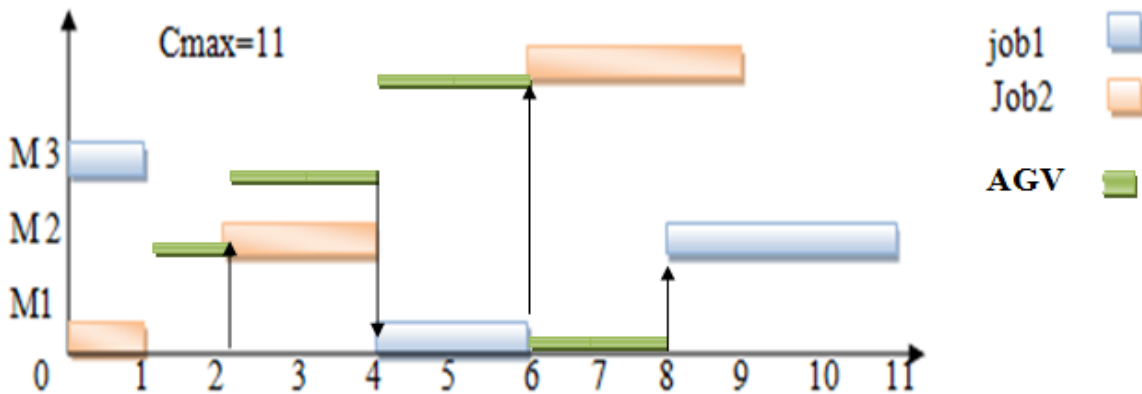
III- $t_{2,j} = \max(DM[2],dt)$: le temps de depart de traitement de job j sur la machine 2

a- $t_{2,j} = \max(3,3)=3$

b- $t_{2,j} = \max(5,5)=5$

Dans ce cas la machine 2 est disponible. En cas la machine 2 n'est pas disponible le job J entre dans la file d'attente.

- On a appliqué notre algorithme sur le problème de $J3(1)|\beta1|C_{max}$, on obtient la solution suivante figure (13) :



Figure(13) : diagramme de gant de $J3(1)|\beta1|C_{max}$

Dans ce problème, on a un conflit, les deux jobs font un appel de transport par l'AGV à l'instant $t=1$. L'AGV répond à l'appel des jobs selon l'ordre de machine.

(i.e : AGV transporte job 2 de machine 1 vers la machine2, puis il transport le job1 de machine 3 vers la machine 1).

On présente l'algorithme proposé :

- $DM[k]=0$ for $k \in \{1, \dots, M\}$
 - $DJ[j]=r_j$ for $j \in \{1, \dots, N\}$
 - $Agv=[k,k']$ for $k \in \{1, \dots, M\}, k' \in \{1, \dots, M\}$
 - $emj(j)$ for $j \in \{1, \dots, N\}$
 - $Dagv=0$
 - POUR $z \in \{1, \dots, NT\}$:
 - $dt = \max(Dagv, DJ[j])$
 - $dt = dt + agv(emagv, emj(j)) + agv(emj(j), pmj(j))$
 - $t_{i,j} = \max(DM[k], dt), t_{f_{i,j}} = t_{i,j} + p_{i,j}$,
 - Mettre a jour $Dagv : Dagv = dt$,
 - Mettre a jour $DM : DM[k] = t_{i,j} + p_{i,j}$,
 - Mettre a jour $DJ : DJ[j] = t_{i,j} + p_{i,j}$,
- FIN POUR

Figure (14) : l'algorithme proposé.

III.2.Simulation et résultats

Pour valider l'algorithme proposé dans la partie précédente nous avons développé un programme sur le logiciel Matlab qui est ensuite appliqué sur plusieurs exemples. Pour vérifier notre algorithme, nous avons calculé le Cmax de quelques exemples manuellement qui sont représenté par le diagramme de Gantt.

- **Exemple de deux machines et deux jobs**

Pour calculer le Makespan (Cmax) nous avons utilisé la méthode de diagramme de Gantt qui représente l'ordonnancement de deux jobs à exécuter dans les machines M1,M2 en respectant l'ordre d'exécution des produits selon les règles proposées.

	M1	M2
M1	0	1
M2	1	0

Tableau (06) : le temps de transport d'AGV

	M1	M2
J1	2	3
J2	1	2

Tableau (07) : le temps d'opérateur

La simulation a été faite sur un ordinateur hp CORE i7 dont nous avons obtenu les résultats suivants :

Pour le problème $J2(1)|\beta_1|C_{max}$ on obtient les résultats suivantes :

Selon la règle SPT $C_{max} = 7$

Selon la règle LPT $C_{max} = 8$

Remarque

Dans le problème 2 machines avec 2 jobs, la règle SPT donne le Cmax minimum en les deux cas (avec \ sans) AGV.

- **Exemple de 3 machines et 2 jobs :**

Les temps opératoire de problème J3(1)|β1|Cmax pour 2 jobs tableau(04)

Pour le problème J3(1)|β1|Cmax on obtient les résultat suivantes :

Selon la règle SPT Cmax= 11

Selon la règle LPT Cmax= 10

Remarque :

Dans le problème J3(1)|β1|Cmax, la règle LPT donne le Cmax minimum.

- **Exemple de petite taille (problème de 4 produits) :**

Le tableau suivant représente le temps opératoire de chaque Job sur les machines :

JOB/MACHEN	M1(temps: s)	M2(temps: s)	M3(temps: s)	M4(temps: s)
J1	40	35	80	90
J2	1	45	50	25
J3	60	1	35	20
J4	30	20	55	28

Tableau(08) : Temps opératoires sur les machines pour 4 jobs.

- Résultat obtenu avec Matlab pour les deux heuristiques avec l'AGV :

-Selon l'heuristique SPT avec AGV :

```

1 -   clc
2 -   clear all
3 -   J=4; %NOMBRE total de jobbs
4 -   M=4;
5
6
7 -   NOM=[4;4;4;4];           %NOMBRE d'opération par machine
8
9 -   NOJ=[4 4 4 4];          %NOMBRE d'opération par jobb
10
11   %   m1   m2
12 -   TO=[40  35  80  90
13         1  45  50  25
14         60 1 35 20
15         30 20 55 28
16         1.];

```

Command Window

```

TDM =

    131    232    397    472

C j job [ 1 ] = 472
C j job [ 2 ] = 307
C j job [ 3 ] = 249
C j job [ 4 ] = 452
Cmax=
    472
fx >>

```

Figure (15) : Exemple de 4 jobs avec SPT

-Selon l'heuristique LPT avec AGV :

```

1 -   clc
2 -   clear all
3 -   J=4; %NOMBRE total de jobbs
4 -   M=4;
5
6
7 -   NOM=[4;4;4;4];           %NOMBRE d'opération par machine
8
9 -   NOJ=[4 4 4 4];          %NOMBRE d'opération par jobb
10
11   %   m1   m2
12 -   TO=[40  35  80  90
13         1  45  50  25
14         60 1 35 20
15         30 20 55 28
16         1.];

```

Command Window

```

TDM =

    131    232    431    456

C j job [ 1 ] = 381
C j job [ 2 ] = 429
C j job [ 3 ] = 456
C j job [ 4 ] = 401
Cmax=
    456
fx >>

```

Figure (16) : Exemple de 4 jobs avec LPT

Nous avons remarquée, à travers cet exemple, que l’heuristique LPT donne le Cmax minimum par apportà l’heuristique SPT.

- **Exemple de problème de 10 produits:**

Dans cet exemple nous avons essayé d’appliquer notre approche de résolution sur dix produits supposés d’être exécutés dans notre système. Le tableau (03) représente le temps d’exécution des produits sont inspirées de la littérature.

Produit/Machine	M1(s)	M2(s)	M3(s)	M4(s)
1	40	35	80	90
2	0	45	50	25
3	60	0	35	20
4	30	20	55	28
5	20	0	0	10
6	40	35	80	90
7	46	35	81	92
8	44	35	87	90
9	40	35	80	90
10	46	35	81	92

Tableau (09) : Temps opératoires sur les machines pour 10 jobs.

- **Exemple de problème de 15 produits :**

Le tableau suivant représente les temps d’opératoires des 15 jobs :

Produit/Machine	M1(s)	M2(s)	M3(s)	M4(s)
1	40	35	80	90
2	0	45	50	25
3	60	0	35	20
4	30	20	55	28

5	20	0	0	10
6	40	35	80	90
7	46	35	81	92
8	44	35	87	90
9	40	35	80	90
10	46	35	81	92
11	44	35	87	90
12	44	35	87	90
13	40	35	80	90
14	46	35	81	92
15	44	35	87	90

Tableau (10) : Temps opératoires des produits.

Après l'adaptation de notre algorithme sur des problèmes de différentes tailles des jobs (4, 10, 15), et 4 machines avec l'AGV qui déplace selon le schéma (figure 10) nous avons trouvé les résultats suivantes :

	L'heuristique SPT	L'heuristique LPT
4 produits	472	456
10 produits	1871	1714
15 produits	2837	2673

Tableau (11) : les résultats de Cmax obtenu par MATLAB.

Remarque :

Dans cette partie ; après l'adaptation de notre algorithme sur les différentes tailles des jobs (4, 10, 15), nous avons remarqué que l'heuristique LPT avec contrainte de transport d'AGV donne le Cmax minimum.

Conclusion

Dans ce chapitre nous avons présenté et expliqué l'implémentation du programme que nous avons proposé pour la résolution du problème de job shop avec contrainte de transport où les produits sont transportés par l'AGV entre les machines. Ce programme est basé sur les deux heuristiques (LPT/SPT) qui nous donnent des résultats proches de l'optimum dans des temps de traitement très faibles. La simulation a été faite, en utilisant le logiciel Matlab, sur plusieurs exemples de différentes tailles de ressource et des jobs à travers lesquels nous avons essayé de tester l'efficacité de notre algorithme.

A partir des différents résultats obtenus nous avons remarqué que notre algorithme est très efficace et donne des meilleures solutions surtout pour les problèmes de petites tailles (2 job et 2 machine / 2 job et 3 machine) où il donne des solutions presque 100% optimums.

Conclusion générale

Le problème d'ordonnancement des ateliers de type Job Shop est l'un des problèmes d'ordonnancement intensivement étudiés. C'est un problème extrêmement complexe. Il est classé parmi les problèmes combinatoires difficiles au sens fort. Cette complexité est due à l'explosion combinatoire du nombre de solutions qui croît exponentiellement avec la taille du problème. L'utilisation de méthodes exactes en vue de l'obtention de solutions optimales semble non réaliste. Le recours à des méthodes approchées comme les heuristiques est devenu incontournable.

Dans ce mémoire, nous nous sommes intéressés aux problèmes d'ordonnancement des ateliers de production. Plus particulièrement, les problèmes de job-shop avec contrainte de transport (Auto-Guide Véhicule) dont nous avons fait le recours aux approches(heuristiques) pour la résolution de cette problématique.

La démarche de résolution proposée était basé essentiellement sur les deux règles de priorité SPT et LPT dont nous l'avon dévisé en deux parties, la première consiste à trouver un ordonnancement selon les règles SPT et LPT, tandis que la deuxième sert à donner le meilleur déplacement de l'AGV. Une simulation sous le logiciel Matlab du programme proposé nous a permet de traiter plusieurs exemples par lesquels nous avons fait une comparaison entre l'utilisation des deux règles.

D'après les résultats obtenus nous pouvons dire que nous avons pu participé à la résolution du problème considérée malgré que ces résultats n'ont pas été comparés avec des résultats données par une méthode exacte.

Pour enrichir ce travail, nous pouvons finalement ajouter des perspectives telles que :

- La proposition de méthodes exactes pour la résolution de ce type de problème.
- La résolution du problème d'ordonnancement de job shop avec contrainte de transport en considérant plusieurs AGV.
- La définition de nouvelle modélisation et développée des heuristiques basées sur la recherche taboue et l'algorithme génétique avec la contrainte de panne d'AGV.

Références

- [01]: Ruhlmann Carine «étude du problème de job shop avec un convoyeur ». Comme Exigence Partielle, Université du Québec à Chicoutimi, 2007.
- [02]: Laurent Deroussi, Michel Gourgand «Un couplage métaheuristique / simulation appliqué au problème du job shop avec transport» LIMOS CNRS UMR 6158 IUT de Montluçon, Avenue Aristide Briand B.P. 2235, 03101 Montluçon ISIMA, Campus des Cézeaux B.P. 10125, 63173 Aubière .
- [03]: Giard V. Gestion de la production, 2ème édition, Economica, Paris, 1988.
- [04]: Javel G, Organisation et Gestion de production, 3ème édition, DUNOD, Paris, 2004
- [05]: Fisher H. e Thompson G.L. (1963), Probabilistic learning combination of local jobshop Scheduling rules. Industrial Scheduling, Prentice Hall; 225-251.
- [06]: Jain A.S. and Meeran S. (1999), Deterministic job-shop scheduling: Past, present and future. European Journal of Operational Research; 113 (2): 390-434.
- [07] : I.H. Osman, G. Laporte .Metaheuristics : A bibliography. Operations Research. 63(1996) 513-623.
- [08] : Leonilde Rocha Varela, Joaquim Nunes Aparicio, and Silvio Carmo Silva, I. Dans H. Unger, T. Böhme, and A. Mikler (Eds.) (2002): I2CS 2002, LNCS 2346, pp. 63–74, 2002. Springer- Verlag Berlin Heidelberg.
- [09]: Blazewicz et al., (1996) J. Blazewicz, W. Domschke E. Pesch (1996), The job Shop Scheduling Problem : Conventional and new solution techniques. European Journal of Operational Research; 93: 1-33.
- [10] : Hentous A. (1999), Contribution au pilotage des systèmes de production de type jobshop. Thèse de doctorat, INSA Lyon, N° 99 ISAL 0028.
- [11] : Letouzey A. (2001), ordonnancement interactif basé sur des indicateurs : Application à La gestion de commandes incertaines et à l'affectation des opérateurs. Thèse de doctorat, Institut National Polytechnique de Toulouse.
- [12] : Baptiste P. (1998), une étude théorique et expérimentale de la propagation des contraintes de ressources. Thèse de doctorat, université Technologique de Compiègne No : 98 COMP 1141.
- [13] : Lopez P. & Esquirol P (1999), L'ordonnancement. Economica, Paris.

- [14]:Duvivier D. « Etude de l'hybridation des méta-heuristique, Application à un problème d'ordonnancement de type Job-shop », thèse de doctorat, Université de littoral cote d'opale, LIL, calais 2000.
- [15] : Mohand Larabi. «Le problème de job-shop avec transport : modélisation et optimisation», Thèse de Doctorat, Université Blaise Pascal - Clermont Ferrand II.
- [16] : MacCarthy B.L. et Liu J. (1993), A new classification Scheme For Flexible Manufacturing Systems. International Journal of Production; 31(2): 299-309.
- [17] : Graham R.L., Lawler E.L., Lenstra J.K., Rinnooy Kan., A.H.G. (1979), Optimisation and approximation in deterministic sequencing and scheduling: a survey, Annals of Discrete Mathematics; 5: 236-287.
- [18]: J. Carlier, P. Chrétienne et C. Girault, « Modelling scheduling problems with Petri nets. Advanced studies in Petri nets ». Lecture notes in Computer Science, Springer Verlag, Paris, 1984.
- [19]: Carlier, J. et Chrétienne, P. (1988), Problèmes d'ordonnancement : modélisation / complexité /algorithmes. Edition Masson.
- [20]: Gotha, « Les problèmes d'ordonnancement ». RAIRO-Recherche Opérationnelle, vol. 27, n° 1, pp. 77-150, 1993.
- [21]: A.S. Jain et S. Meeran, « Deterministic job-shop scheduling : past, present and future ». European Journal of Operational Research, vol. 113, pp. 390-434, 1999.
- [22] : Blondel F,Gestion de la production,3^{ème} édition,DUNOD,Paris,2004.
- [23]: Kebabla Mebarek , Utilisation des stratégies Métaheuristiques pour l'rdonnancement des ateliers de type Job Shop ,Thèse de magister, Université de Batna,2008.
- [24] : Driss Imen « Analyse D'un Système Job Shop Aspect Ordonnancement»,Thèse de Doctorat Université de Batna 2, 2016.
- [25] : Fontanili F, Intégration d'outil de simulation et d'optimisation pour le pilotage d'une ligne d'assemblage multiproduit à transfert asynchrone, Thèse de Doctorat, Université Paris XIII,1999.
- [26] : Letouzey A, Ordonnancement interactif basé sur des indicateurs :Application à la gestion de commandes incertaine et à l'affectation des opérateurs, Thèse de doctorat, Institut National Polytechnique de Toulouse,2001.
- [27]:Giffler, B. et Thompson, G.L. (1960), Algorithms for solving production scheduling problems. Operations Research; 8(4): 487-503.

[28]:RUHLMANN CARINE « étude du problème de job shop avec un convoyeur »
Universite Du Quebec 2007.

[29]: Bourdeaud'huy.T,Korbaa.O, « Un modèle mathématique pour la résolution du problème d'ordonnancement cyclique avec minimisation de l'en-cours » 6^e Conférence de Modélisation et Simulation, MOSIM'06, Rabat, Maroc,2006.

[30] : Fatma Tangour Toumi, « Ordonnancement Dynamique dans les industries agroalimentaires », Thèse de doctorat en Automatique et informatique Industrielle,2007.