



N° d'ordre : .....

**UNIVERSITE DE M'SILA**  
**FACULTE DE MATHÉMATIQUES ET D'INFORMATIQUE**  
Département d'Informatique

**MEMOIRE**

Présenté pour l'obtention du diplôme de **MASTER**

**Domaine** : Mathématiques et Informatique

**Filière** : Informatique

**Spécialité** : système d'information avancée

**Par :**

**Laid MEKADER**

**SUJET**

La Méthode Tabou Pour La Résolution d'un  
Problème d'ordonnancement sur une Machine avec  
Périodes d'indisponibilité

Soutenu publiquement le : 27/06/2012 devant le jury composé de :

Mr. BENAHCEN Darradji Rachid

Université de M'sila

Président

Mr. LAMICHE Chaabane

Université de M'sila

Rapporteur

Mr. MENSOURI Kamal

Université de M'sila

Examinateur

**Promotion : 2011 /2012**

## Sommaire

Introduction générale.....	1
----------------------------	---

### *Chapitre I : Fondements théorique et problème d'ordonnancement*

1. Introduction .....	4
2. Définition d'ordonnancement .....	4
3. Les tâches .....	4
4. Les caractéristiques d'une tâche.....	5
5. Type de tâches .....	6
6. Les ressources .....	6
7. Les contraintes .....	7
7.1. Les contraintes potentielles.....	7
7.2. Les contraintes de ressources.....	8
7.3. Les contraintes disjonctives.....	8
7.4. Les contraintes cumulatives.....	8
8. Une séquence .....	8
9. Un ordonnancement.....	8
10. Dresser un calendrier d'exécution .....	8
11. Caractéristiques générales d'un ordonnancement.....	8
12. Critère .....	9
13. Critère d'optimisation régulier .....	10
14. Satisfaction de contraintes .....	10
15. Définition Problème d'ordonnancement .....	11
16. Quelque types de problèmes d'ordonnancement .....	11
17. Représentation des problèmes d'ordonnancement sur machine .....	11
18. Représentation des solutions avec Le diagramme de Gantt .....	13

### *Chapitre II : Les méthodes de résolution d'un problème d'ordonnancement*

1. Introduction.....	16
2. Complexité des algorithmes .....	16
2.1. Définition .....	17
2.2. Un algorithme polynomiale.....	17

2.3.	<i>Algorithme efficace</i> .....	17
3.	Complexité des problèmes.....	17
4.	Les Méthodes exactes.....	19
4.1.	<i>Principe</i> .....	19
4.2.	<i>Quelques méthodes exactes</i> .....	20
4.2.1.	<i>Enumération complète</i> .....	20
4.2.2.	<i>Analyse combinatoire</i> .....	20
4.2.3.	<i>Programmation dynamique (P.D.)</i> .....	21
4.2.4.	<i>Méthode par séparation et évaluation</i> .....	22
5.	Les méthodes approchées .....	23
5.1.	<i>Les Méthodes heuristiques</i> .....	23
5.2.	<i>Les Méta-heuristiques</i> .....	24
5.2.1.	<i>La recherche avec tabous</i> .....	24
5.2.2.	<i>Le recuit simulé (simulated annealing)</i> .....	24
5.2.3.	<i>Les algorithmes génétiques</i> .....	25
6.	Méthode exacte contre méthode approchée.....	26
	<i>Avantages d'une méthode approchée</i> .....	26
	<i>Inconvénients d'une méthode approchée</i> .....	26

### *Chapitre III : Ordonnancement sur une machine avec périodes d'indisponibilité*

1.	Introduction.....	28
2.	Description du problème .....	29
3.	Techniques de génération de voisinage .....	30
3.1.	<i>Le voisinage par permutation de deux tâches</i> .....	30
3.1.1.	<i>la permutation simple de deux tâches adjacentes</i> .....	30
3.1.2.	<i>la permutation simple de deux tâches.</i> .....	30
3.2.	<i>Le voisinage par insertion de tâches</i> .....	31
3.3.	<i>Voisinage par insertion en block</i> .....	32
3.4.	<i>Hybride1 et Hybride2</i> .....	32
4.	La méthode de recherche tabou .....	33
4.1.	<i>Description et principe de la méthode</i> .....	33
4.2.	<i>La liste tabou</i> .....	33
4.3.	<i>Sélection du voisin</i> .....	34
4.4.	<i>Critère d'aspiration</i> .....	34

4.5.	<i>Techniques d'amélioration</i> .....	34
4.5.1.	<i>L'intensification</i> .....	34
4.5.2.	<i>La diversification</i> .....	34
4.5.3.	<i>Algorithme générale de la méthode tabou</i> .....	35
5.	Application de la méthode de recherche tabou au problème P1 .....	35
5.1.	<i>Choix des paramètres de la méthode</i> .....	36
5.2.	<i>Exemple illustrative</i> .....	37
6.	Etude expérimentale .....	38
6.1.	<i>Le nombre d'itérations et la sequence initiale</i> .....	39
7.	Analyse des résultats .....	47
8.	Conclusion .....	48

#### *Chapitre IV : Réalisation du logiciel*

1.	Problématique .....	50
	<i>Mais c'est quoi Ordo ?</i> .....	50
2.	présentation de système et l'objectif de l'étude.....	51
	<i>Objectif de l'étude</i> .....	51
	<i>Hypothèse est contrainte de fonctionnement</i> .....	51
3.	conception du logiciel.....	51
4.	programmation .....	52
	<i>Choix de langage</i> .....	52
	<i>Qu'est-ce que Delphi ?</i> .....	52
5.	Test du logiciel.....	52
6.	Interface du logiciel.....	53
7.	Guide de l'utilisateur .....	53
	 Conclusion générale.....	 56
	Références bibliographiques.....	57

## Introduction générale

L'ordonnancement est la programmation dans le temps de l'exécution d'une série de tâche (ou activités, opérations) sur un ensemble de ressources physiques (humaines et techniques), en cherchant à optimiser certains critères (financiers ou technologiques), et en respectant le plus possible des contraintes sur les ressources et les tâches.

Il est clair que l'ordonnancement devient un outil stratégique, peut aider l'homme à prendre des décisions, influant sur le coût, les délais et la qualité d'un produit ou d'un projet.

Les problèmes d'ordonnancement apparaissent dans beaucoup de circonstances pratiques et sous une large variété d'aspects. Ils sont inhérents à toute organisation et revêtent une importance économique considérable.

En outre, ils sont des problèmes de recherche opérationnelle.

Cette dernière ne s'occupe pas des problèmes dans lesquels une solution de bon sens intervient tout naturellement. Son domaine réservé est celui des situations dans lesquelles, pour une raison quelconque, le sens commun se révèle faible ou impuissant.

La terminologie de la théorie de l'ordonnancement relève des applications industrielles, car celles-ci furent les premières à susciter des recherches sur les problèmes d'ordonnancement. Le premier développement théorique sur l'ordonnancement était fait par Johnson [JOH54], suivi de près par les résultats de Jackson [JAC55] et de Smith [SMI56]. Une façon naturelle d'aborder les problèmes d'ordonnancement est de les formuler comme des modèles de programmation mathématique. L'utilisation de la programmation en nombres entiers dans la résolution des problèmes d'ordonnancement peut remonter à 1959, lorsque Wagner [WAG59] était le premier à formuler une variété de problèmes d'ordonnancement comme des programmes en nombre entiers. Conway et al. [CON67] et Baker [BAK78] ont discuté les formulations de programmation en nombres entiers des problèmes d'ordonnancement [LIA92]. Un aperçu sur le premier développement de la programmation en nombres entiers pour l'ordonnancement peut être trouvé dans Rinnooy Kan [KAN76].

L'objet de ce mémoire est de trouver une solution approchée à un problème d'ordonnancement statique et déterministe sur une seule machine.

*WSP*

Il s'agit de la minimisation de la somme pondérée des retards de  $n$  tâches soumises aux contraintes  $(1, NC/\sum W_i T_i)$ .

Et puisqu'il est NP-difficile Nous avons choisi une métaheuristique dite la méthode recherche tabou pour trouver une solution approchée au problème considéré.

Notre mémoire est composé des chapitre suivants :

**Chapitre I :** traite les fondements théoriques des problèmes d'ordonnancement, les caractéristiques des contraintes, et les critères de performance.

On a donné la représentation des problèmes d'ordonnancement selon la notation à trois champs, ainsi qu'une façon de représenter la solution d'un problème.

**Chapitre II :** Afin de résoudre les problèmes d'ordonnancement, il faut tout d'abord les classer. Pour ce faire on a commencé ce chapitre par la définition de la notion de complexité des algorithmes et des problèmes. On a présenté ensuite, quelques méthodes de résolution exactes et approchées. Enfin, on a effectué une comparaison entre ces deux classes de méthodes de résolution.

**Chapitre III :** concerne la résolution du problème sujet étude. On a défini le problème à traiter. On a appliqué l'algorithme de la méthode recherche tabou pour résoudre notre problème, ensuite on a traité un exemple numérique et on a commenté les résultats obtenus.

**Chapitre IV :** réalisation du logiciel

On présente le logiciel, on décrit c'est quoi, pourquoi et comment on a fait le logiciel.

## Conclusion générale

L'objectif de ce mémoire concerne l'étude du problème d'ordonnancement des tâches sur une seule machine avec périodes d'indisponibilité où chaque tâche a une durée d'exécution et un poids lié à son importance et l'objectif est de déterminer une séquence de tâche sur la machine afin de minimiser la somme pondérée des retards. Le problème est connu d'être NP-difficile. Pour résoudre ce problème une méthode dite tabou est proposée pour déterminer une solution approchée et moins coûteuse de point de vue temps l'exécution.

Au cours de cette étude, deux technique de génération de voisinage ont été proposé, ces technique sont basées sur l'hybridation de certaines technique traditionnelles (permutation des tâches, insertion des tâches, insertion par bloc des tâches). L'objectif principale est de minimiser la somme pondérée des retards des tâches sur une machine en présence de certains périodes d'indisponibilité.

L'ordonnancement reste toujours un domaine de recherche très riche. Car ces problèmes sont variés d'un cas à l'autre.

Les résultats obtenus sur les problèmes à une machine fournissent des idées intuitives pour résoudre des problèmes généraux. De ce fait, ces résultats constituent la clé pour ouvrir la porte des méthodes de résolution des problèmes plus complexes. Alors il est possible d'utiliser l'heuristique proposée pour obtenir des résultats pour des problèmes complexes.

### *Références bibliographiques:*

- [ABD87] Abdul-Rasaq T.S.1987. Machine Scheduling problems Branch and Bound Approach.Ph.D.Thesis.Departement of Mathemetics University of Keele.
- [BAK78] Baker. K. R. and Schrage L. E. 1978. Finding an Optimal Sequence by Dynamique programming an Extention to Precedence-Related Tasks.
- [BEL 88] Belouadah. H. 1988. Scheduling and sequencing, branch and bound based on job splitting and Lagrangian Relaxation . Ph. D Thesis. University of Southampton, UK.
- [BLA96] J. Blazewicz, K.H. Ecker, E. Pesch, G. Schmidt & J. Weglarz. Scheduling in computer and manufacturing processes. Springer Verlag,1996.
- [CAR88] Carfier J. et Chrétienne P. 1988. Problèmes d'ordonnancement. Modélisation/Complexité/Algorithme.Mass.
- [COF76] E.G. Co\_man. Computer and job-shop scheduling theory. John Wiley and Sons, 1976.
- [CON67] Conway R. W. Maxwell W. L. and Miller L. W. 1967. Theory of scheduling Addition Wesley, Reading, MA.
- [EAS59] Eastman W.L.1959.A Solution to the Traveling Salesmen problem.Econometrica.27.28222.
- [E&L99] P. Esquirol & P. Lopez. L'ordonnancement. Economica, 1999.
- [HEL62] Held H. and Karps R.M.1962.A Dynamic programing Approach to Sequencing Problems.
- [IGN65] Ignall E.and Schrage L.1965.Application of the Branch and Bound Technique to Some Flow shop Scheduling Problems. Oper.Res.13,400-412.
- [JAK55] Jackson J. R. 1955. Scheduling a production Line to Minimize Maximum.
- [JOH54] Johnson S. M. 1954. Optimal two-and three-stage production Schedules with setup Times included.  
Naval Res. Logist. Quart. 1,61-68.
- [KAN76] Rinnooy Kan A. H.G. 1976. Machine problèms : Classification, Complexity and Computation.
- [LIA92] Liao C-J. and You C-T. 1992. An Improved Formulation for the Job-Shop Scheduling Problems.  
J. Oper. Res. 43. 1047-1054.
- [LA05] LA Hoang Trung : utilisation d'ordres partiels pour la caractérisation de solution robustes en ordonnancement. Thèse, 2005, Toulouse.
- [LAM05] Lamiche. C. Mémoire de magister : Le recuit simulé et la méthode tabou pour la Résolution d'un problème d'ordonnancement sur une machine avec périodes d'indisponibilité.

- [LAN60] Land A.H. and Doig A.G. 1960. An Automatic Method for Solving Discrete Programming Problems.
- [LOM65] Lomnicki Z.A.1965.ABranch and Bound Algorithm for the Exaaact Solution of the Three-Machine Scheduling Problems.
- [MAK97] Malek Khadidja. L'ordonnancement à une machine : Mémoire de fin d'étude pour obtenir le diplôme de magister, 1997, Université de Constantine.
- [PIN95] M. Pinedo. Scheduling : Theory, algorithms and systems. Prentice Hall, 1995.
- [SAK84] Sakarovitch. M. 1984. Optimisation Combinatoire .
- [SCH00] Schmidt G. 2000. Scheduling with limited machine availability. European Journal of Operational Research 121, 1 – 15.
- [SMI56] Smith W. E. 1956. Various optimizers for the Exact Solution of the Three-Machine Scheduling problèms.
- [WAG59] Wagner H. M. 1959. An Integer Linear-programming Model for Machine Scheduling.

### ملخص :

الهدف من هذه المذكرة هو دراسة مشكلة الترتيبات لمجموعة من الاشغال على آلة واحدة ذات ادوار زمنية غير متاحة بحيث يرتبط كل عمل بزمان تنفيذ ووزن يحدد أهميته بحيث نحصل على الترتيبية المناسبة لتقليص دالة المجموع الترجيحي لمدة تأخر الاشغال عن الوقت المحدد لهم وللعلم أنه لايمكن تحديد خوارزمي محدد يضبط حل نهائي للمشكل المطروح . لذلك لجأنا الى استعمال طريقة البحث مع العزل ( تابو) مع تحديد وسائط هذه الطريقة التقريبية من أجل إيجاد حل تقريبي للمشكل مع مراعاة تقليل زمن التنفيذ وذلك عن طريق تغيير بعض الإعدادات.

**كلمات مفتاحية :** ترتيبات , طريقة البحث مع العزل , أدوار زمنية غير متاحة .

### Résumé:

L'objectif de ce mémoire est l'étude du problème d'ordonnancement des tâches sur une seule machine avec périodes d'indisponibilité où chaque tâche a une durée d'exécution et un poids lié à son importance et l'objectif est de déterminer une séquence de tâche sur le machine afin de minimiser la somme pondérée des retards. Le problème est connu d'être NP- difficile. Pour résoudre ce type problème, une méthode dite tabou est proposée pour déterminer une solution approchée et moins coûteuse de point de vue temps l'exécution par la modification de certaines paramètres.

**Mots clés:** Ordonnancement – Méthode tabou– périodes d'indisponibilité

### Abstract :

The aim of this work is the study of a scheduling problem on a single machine with unavailability periods, where each task has an execution time and a weight to its importance and the objective is to determine a sequence task on the machine in order to minimize the weighted sum of retards. The problem is known to be NP-hard. To solve this problem a method called tabu search is proposed to determine an approximate solution and less costly in terms of time execution by changing some parameters.

**Key words:** Scheduling – Tabu search – unavailability periods.