



**HAL**  
open science

**Contribution à l'étude de la performance et de la  
robustesse des ordonnancements conjoints  
Production/Maintenance - Cas du Flowshop.**

Fatima Benbouzid Sitayeb

► **To cite this version:**

Fatima Benbouzid Sitayeb. Contribution à l'étude de la performance et de la robustesse des ordonnancements conjoints Production/Maintenance - Cas du Flowshop.. Automatique / Robotique. Université de Franche-Comté, 2005. Français. NNT: . tel-00257906

**HAL Id: tel-00257906**

**<https://theses.hal.science/tel-00257906>**

Submitted on 20 Feb 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# THESE

présentée à  
**L'UFR des Sciences et Techniques  
de l'Université de Franche-Comté**

pour obtenir le

## **GRADE DE DOCTEUR DE L'UNIVERSITE DE FRANCHE-COMTE**

**en Automatique et Informatique**

*(Ecole Doctorale Sciences Physiques pour l'Ingénieur et Microtechniques)*

### **CONTRIBUTION A L'ETUDE DE LA PERFORMANCE ET DE LA ROBUSTESSE DES ORDONNANCEMENTS CONJOINTS PRODUCTION/MAINTENANCE – CAS DU FLOWSHOP**

par

**Fatima BENBOUZID SITAYEB**

Soutenue le 29 Juin 2005 devant la Commission d'examen :

Rapporteurs  
Examineurs

GOURGAND M. Professeur, ISMA, Université Blaise Pascal, Aubière

TAHON C. Professeur, Université de Valenciennes

BOUMGHAR F. Professeur, USTHB, Alger

BOURJALOT A. Professeur, ENSMM, Besançon

ZERHOUNI N. Professeur, ENSMM, Besançon

Directeur de thèse  
Co-encadrant  
Invité

VARNIER Ch. Maître de Conférences, ENSMM, Besançon

DUDOGNON F. Responsable de site, CEGELEC, Belfort



*A mes parents,  
A Abderrezak, mon mari,  
A Majda et Farès, mes enfants,*

*Avec Tendresse,  
Qu'ils m'excusent pour tout le temps que j'ai consacré à ma thèse et que je leurs ai volé,  
J'espère qu'ils sont fiers de moi.*

# Remerciements

*Le travail présenté dans cette thèse a été initié au sein du laboratoire d'Automatique de Besançon (LAB). Puis il s'est poursuivi conjointement entre le LAB et le Laboratoire des Méthodes de Conception de Systèmes (LMCS) d'Alger. Je tiens donc à adresser mes vifs remerciements à Mrs Alain Bourjault, directeur du LAB et Abderrezak Henni, directeur du LMCS pour leurs accueils et leur souci constant à veiller à ce que cette thèse se passe dans les meilleures conditions.*

*Je souhaite remercier très vivement la personne sans laquelle tout ceci n'existerait pas. J'exprime en effet toute ma profonde gratitude à Mr Nouredine ZERHOUNI, professeur à l'Ecole Nationale Supérieure de Mécaniques et Microtechniques de Besançon (ENSM) pour m'avoir accueilli au sein de sa jeune équipe de maintenance et sûreté de fonctionnement. Je lui suis sincèrement reconnaissante pour la confiance qu'il m'a témoignée depuis le début et tout au long de la thèse, pour son encadrement, ses conseils et ses orientations. J'espère avoir été digne de cette confiance.*

*Je tiens à remercier tout particulièrement Mr Christophe VARNIER, maître de conférences à l'Université de Franche-Comté, non seulement pour son encadrement de très haut niveau, ses précieux conseils et orientations, mais également pour sa disponibilité et son dévouement. Qu'il sache combien ses conseils perfectionnistes, sa disponibilité et ses encouragements m'ont été d'une grande utilité.*

*Je remercie Monsieur Alain BOURJAULT, professeur à l'Ecole Nationale Supérieure de Mécaniques et Microtechniques de Besançon (ENSM), de me faire l'honneur de présider mon jury de thèse.*

*J'exprime toute ma gratitude à Monsieur Michel GOURGAND, Professeur à l'ISMA, Université Blaise Pascal (Aubière), et Monsieur Christian TAHON, Professeur à l'université de Valenciennes pour l'intérêt et l'attention qu'ils ont accordé à ce travail de thèse et d'avoir accepté d'en être les rapporteurs.*

*J'adresse mes sincères remerciements à Monsieur Alain BOURJAULT, professeur à l'ENSM de Besançon pour l'intérêt qu'il a porté à mes travaux depuis le début de ma thèse et pour sa participation à mon jury.*

*Je remercie Madame Fatima Boumgchar, Professeur à l'Université des Sciences et Technologies Houari Boumediene (USTHB) d'Alger et Monsieur Franck DUDOGNON, responsable de site à CEGELEC (BELFORT) pour avoir accepté d'examiner mon travail.*

*Que tous les membres du LMCS trouvent ici l'expression de ma profonde sympathie pour l'ambiance chaleureuse et conviviale qu'ils ont maintenue tout au long de ces cinq dernières années et pour l'aide qu'ils m'ont apportée dans le cadre de cette thèse. Je suis fière de faire partie de cette équipe.*

*Je remercie très sincèrement et sans exception, l'ensemble du personnel du Laboratoire d'Automatique de Besançon pour l'excellente ambiance qui y règne.*

*Un infini merci à ma petite famille. A **Farès**, ton sens de la répartie même si tu n'as que trois ans, a toujours égayé des fins de journées chargées. A mon adorable **Majda**, qui du haut de ses six ans, a toujours compris l'impératif de mes absences. J'espère qu'elle me pardonnera d'avoir été, pendant ces années de thèse, une maman en pointillé. A **Abderrezak** que je ne remercierai jamais assez pour le soutien moral qu'il m'a apporté et la patience dont il a fait preuve. Vous m'avez insufflé l'énergie nécessaire pour aller de l'avant et donner le meilleur de mes capacités.*

*Je remercie tout particulièrement Papa, Mama et Houneïda pour m'avoir encouragé à continuer même quand ce fût trop difficile et pour m'avoir remplacé dans mon rôle de maman lorsque ma thèse et mes déplacements ne me le permettaient pas.*

*A **Mohamed El-Hachemi**, **Aïcha EL-Khahla**, **Ali Bey** et **Houeïda** ainsi que leurs adorables petites familles, sans oublier ma grand-mère Youma et mes défunt grands-pères. Qu'ils sachent que leur simple présence m'est d'un grand réconfort.*

*Mes années d'enseignements à l'INI ont été pour moi l'occasion de connaître des personnes exceptionnelles qui m'ont tout simplement offert leur sincère amitié, et avec qui j'ai partagé d'agréables moments. Je remercie très chaleureusement **Karima**, t'avoir en face de moi chaque jour m'a aidé à remonter certaines pentes difficiles, **Mouloud**, pour ta gentillesse et ta noblesse d'esprit **Fahima**, on t'adore **Khaltou**, **Lotfi**, **Nacéra**, **Rachid**, **Karima**, **Hassina**, **Malika**, **Mr Souici**, **Mr***

*Dahmni et Mr Khelifati, sans oublier notre Kenza, pour leur amitié, leurs encouragements et leur présence indéfectible. Surtout merci de me compter parmi vos amis.*

*Merci à Nouredine et Christophe pour toute la confiance que vous avez porté en moi. Merci tout simplement pour votre sincère amitié et vos précieuses qualités humaines.*

*Ces années de thèse au LAB auraient été toutes autres sans le soutien et l'aide indéfectible de Karima, Yacine et tout particulièrement Denis et Annie. Merci pour votre amitié et votre gentillesse.*

*Je ne saurais finir sans remercier tout les PFE de l'INI qui ont travaillé sur la plate-forme informatique. Un merci tout particulier à Yacine Bessadi et Sid Ali Guebli puis à Samir Benkhalat et Ahcène Bendjoudi sans oublier Nawel Ikhlif*

*Je remercie, enfin, toute ma famille, ma belle famille, mes amis et tous ceux qui, de près ou de loin, ont contribué à la concrétisation de ce travail.*





---

# **SOMMAIRE**

---

---

## *Table des Matières*

---

<i>Table des Matières</i> .....	<i>ii</i>
<i>Listes des Figures</i> .....	<i>vi</i>
<i>Listes des Tableaux</i> .....	<i>ix</i>
<i>Listes des Algorithmes</i> .....	<i>x</i>
<i>Introduction Générale</i> .....	<i>3</i>
<b>CHAPITRE I.</b>	
<b><i>La Maintenance des Systèmes de Production</i></b> .....	<b><i>9</i></b>
<b>1. Introduction</b> .....	<b>10</b>
<b>2. Les systèmes de production - Généralités</b> .....	<b>11</b>
2.1. Typologie des systèmes de production .....	14
2.1.1. Typologie liée à la demande .....	15
2.1.2. Typologie liée aux ressources .....	15
2.2. Critères d'évaluation d'un système de production .....	16
2.2.1. Les performances .....	16
2.2.2. La sûreté de fonctionnement .....	17
2.3. Conclusion .....	17
<b>3. La maintenance dans les systèmes de production</b> .....	<b>18</b>
3.1. La maintenance – Généralités.....	19
3.1.1. Evolution de la maintenance .....	20
3.1.2. Les modes de gestion de la maintenance .....	21
3.2. Les politiques de maintenance .....	23
3.2.1. La Maintenance Corrective .....	24
3.2.2. La Maintenance Préventive.....	24
3.2.3. Commentaire.....	27
3.3. Le système de maintenance .....	28
3.3.1. Les actions de maintenance .....	28
3.3.2. Les fonctions et les tâches associées à la maintenance.....	30
3.3.3. Les niveaux de maintenance .....	31
3.4. La Gestion de la Maintenance Assistée par Ordinateur.....	32
3.5. Les nouvelles approches de maintenance .....	34
3.5.1. La télémaintenance .....	34
3.5.2. La maintenance productive totale .....	35
3.5.3. La maintenance basée sur la fiabilité .....	36
<b>4. Conclusion</b> .....	<b>37</b>

**CHAPITRE II.****Ordonnancement des Activités de Production et de Maintenance..... 41**

<b>1. Introduction .....</b>	<b>42</b>
<b>2. Introduction aux problèmes d'ordonnancement de production .....</b>	<b>43</b>
2.1. Modélisation des problèmes d'ordonnancement de production .....	44
2.1.1. Les tâches.....	44
2.1.2. Les Ressources.....	45
2.1.3. Les critères d'optimisation.....	46
2.1.4. Les contraintes .....	47
2.2. Les environnements machines.....	48
2.3. Les méthodes de résolution .....	50
2.3.1. Les méthodes exactes.....	50
2.3.2. Les métaheuristiques.....	51
<b>3. Ordonnancement de la maintenance .....</b>	<b>58</b>
3.1. Modélisation de l'ordonnancement de la maintenance .....	59
3.2. Les niveaux hiérarchiques .....	60
3.3. Les objectifs de l'ordonnancement de la maintenance .....	61
<b>4. Ordonnancement de la production en présence de maintenance .....</b>	<b>63</b>
4.1. La situation conflictuelle .....	64
4.2. Les stratégies d'ordonnancement conjoint .....	66
4.2.1. L'ordonnancement séparé.....	66
4.2.2. L'ordonnancement séquentiel.....	67
4.2.3. L'ordonnancement intégré.....	67
4.3. Enjeux d'un ordonnancement conjoint Production/Maintenance.....	68
4.4. Discussions et conclusion.....	69
<b>5. Etat de l'art : Ordonnancement conjoint production/maintenance .....</b>	<b>70</b>
5.1. Interaction entre politique de maintenance et gestion de production .....	71
5.2. Interaction entre la production et la maintenance.....	72
5.3. Synthèse.....	74
<b>6. Problématiques étudiées .....</b>	<b>78</b>
<b>7. Conclusion .....</b>	<b>82</b>

**CHAPITRE III.****Ordonnancement Conjoint Production/Maintenance : Cas du flowshop..... 87**

<b>1. Introduction .....</b>	<b>88</b>
<b>2. Contexte de l'étude .....</b>	<b>90</b>
2.1. Les données de la production .....	90
2.2. Les données de la maintenance.....	90
2.3. La fonction objectif.....	92
<b>3. La stratégie séquentielle.....</b>	<b>93</b>
3.1. Ordonnancement de la production.....	94
3.1.1. Représentation d'un ordonnancement.....	95
3.1.2. Approche dédiée .....	95
3.1.3. Approche itérative : La recherche Tabou.....	96

3.1.4. Approche évolutive : un algorithme génétique.....	98
3.2. Insertion des tâches de maintenance.....	102
3.2.1. Résolution du problème à une machine.....	102
3.2.2. Résolution du problème à plusieurs machines.....	108
3.3. Tests et Résultats.....	111
3.3.1. Ordonnancement de la production.....	112
3.3.2. Insertion de la maintenance.....	113
3.4. Synthèse.....	116
<b>4. La stratégie intégrée.....</b>	<b>118</b>
4.1. Représentation d'un ordonnancement.....	119
4.2. Approche dédiée.....	119
4.2.1. Heuristique de Palmer (resp. Gupta) intégrée.....	120
4.2.2. Heuristique RA intégrée.....	120
4.2.3. Heuristique CDS intégrée.....	121
4.2.4. Heuristique NEH intégrée.....	122
4.3. L'approche itérative : La recherche Tabou intégrée.....	123
4.3.1. Génération de la solution initiale.....	124
4.3.2. Génération du voisinage.....	124
4.3.3. Fonction d'aspiration.....	126
4.3.4. Critère d'arrêt.....	126
4.4. Approche évolutive : un algorithme génétique intégré.....	126
4.4.1. Croisement sur la maintenance.....	127
4.4.2. Mutation sur la maintenance.....	129
4.5. Tests et résultats.....	131
4.6. Synthèse.....	134
<b>5. La stratégie hybride.....</b>	<b>135</b>
5.1. Un algorithme génétique hybride.....	135
5.2. La recherche Tabou hybride.....	137
5.3. Tests et résultats.....	138
<b>6. Etude comparative.....</b>	<b>138</b>
<b>7. Conclusion.....</b>	<b>142</b>

## **CHAPITRE IV.**

### ***Robustesse des ordonnancements conjoints Production/Maintenance..... 147***

<b>1. Introduction.....</b>	<b>148</b>
<b>2. Sensibilité, Robustesse et Flexibilité.....</b>	<b>149</b>
<b>3. Ordonnancement en présence de données incertaines.....</b>	<b>155</b>
3.1. Classification des perturbations.....	155
3.1.1. Perturbation sur les ressources.....	156
3.1.2. Perturbation sur les travaux.....	157
3.2. Classification des méthodes d'ordonnancement en présence de perturbations.....	158
3.2.1. Les approches réactives.....	159
3.2.2. Les approches proactives ou robustes.....	162
3.2.3. Les approches proactives réactives.....	164
3.3. Synthèse.....	165
3.4. Discussions et conclusion.....	167

<b>4. Etude de la robustesse des ordonnancements conjoints .....</b>	<b>168</b>
4.1. Introduction.....	168
4.2. Modèle pour l'étude de la robustesse des ordonnancements conjoints production/maintenance.....	169
4.2.1. Contexte de l'étude .....	169
4.2.2. Présentation globale de l'approche .....	171
4.2.3. Méthodologie adoptée pour la conception du modèle .....	174
4.3. Définition du modèle .....	177
4.3.1. Définition du problème et de son environnement.....	177
4.3.2. Définitions des mesures de performances retenues .....	177
4.3.3. Définition du protocole de perturbation.....	179
4.3.4. Définition de la mention d'un ordonnancement .....	183
4.4. Implémentation du modèle .....	184
4.4.1. Le chevauchement entre tâches .....	184
4.4.2. Procédure de perturbation d'un ordonnancement de production.....	186
4.4.3. Procédure de perturbation d'un ordonnancement conjoint.....	187
4.5. Mise en œuvre du modèle.....	194
<b>5. Résultats et Synthèse .....</b>	<b>197</b>
5.1. La maintenance contribue-t-elle à la robustesse des ordonnancements Production/Maintenance? .....	199
5.2. Existe-t-il un critère dont l'optimisation rend l'ordonnancement robuste ?.....	201
5.3. Existe-t-il des méthodes d'ordonnancements robustes ?.....	204
<b>6. Conclusion .....</b>	<b>206</b>
 <i>Conclusion Générale.....</i>	 <i>211</i>
 <i>Références Bibliographiques .....</i>	 <i>217</i>
 <i>Annexes.....</i>	 <i>230</i>

---

## *Listes des Figures*

---

Figure 1.1: Modèle conceptuel d'un système de production .....	12
Figure 1.2: organisation hiérarchique d'un système de gestion de production. ....	14
Figure 1.3: le contenu de la fonction maintenance.....	19
Figure 1.4 : Phase d'évolution de la fonction maintenance.....	21
Figure 1.5: Typologie de la maintenance .....	23
Figure 1.6 : Effet d'une bonne maintenance préventive.....	25
Figure 1.7 : Impact de la maintenance sur la fiabilité des équipements [KAF01].....	27
Figure 1.8 : Les fonction et tâches associées à la maintenance.....	30
Figure 2.1: Caractéristique d'une tâche.....	45
Figure 2.2: Exemples d'ordonnancements semi-actifs et sans délai .....	47
Figure 2.3: Exemple d'ordonnement actif qui n'est pas sans délai .....	47
Figure 2.4 : Atelier de type <i>flowshop</i> pur.....	49
Figure 2.5: Exploration de X par une approche constructive.....	52
Figure 2.6 : Exploration de X par une approche séquentielle.....	53
Figure 2.7 : Exploration de X par une approche évolutive .....	55
Figure 2.8: Structure hiérarchique des objectifs de l'ordonnement de la maintenance.....	63
Figure 2.9: Ordonnement de la production.....	66
Figure 2.10: Planification de la maintenance.....	66
Figure 2.11: Conflits entre Maintenance et Production dans un ordonnancement séparé.....	66
Figure 2.12 : Ordonnement Séquentiel.....	67
Figure 2.13: Ordonnement Intégré .....	68
Figure 2.14: Problématiques étudiées .....	81
Figure 2.15: Ordonnement conjoint Production/Maintenance .....	82
Figure 3.1 : Stratégies et approches de résolution.....	89
Figure 3.2: Intervalle de tolérance d'une tâche de maintenance.....	91
Figure 3.3: Ordonnement conjoint Production/Maintenance par la stratégie séquentielle. ....	94
Figure 3.4 : Insertion de la maintenance sur une machine par l'heuristique HB* .....	104
Figure 3.5 : Insertion de la maintenance par HRP : Avances/Retards non tolérés .....	105
Figure 3.6: Insertion de la maintenance par HRP : Avances/Retards tolérés .....	107

Figure 3.7 : Principe d’insertion de la maintenance par de l'heuristique HA.....	109
Figure 3.8 : Principe d’insertion de la maintenance par de l'heuristique HD.....	110
Figure 3.9 : Déviation des heuristiques constructives par rapport à NEH.....	112
Figure 3.10: Ordonnancement conjoint P/M des heuristiques constructives.....	113
Figure 3.11 : Déviation par rapport à NEH après insertion de la maintenance.....	114
Figure 3.12: Ordonnancement conjoint P/M par la stratégie séquentielle.....	115
Figure 3.13: Insertion d'une tâche de maintenance.....	123
Figure 3.14 : Déviation par rapport à l'heuristique NEH.....	131
Figure 3.15 : Ordonnancement conjoint P/M par la stratégie intégrée.....	133
Figure 3.16: Principe de fonctionnement de l'AG hybride.....	136
Figure 3.17 : Hybridation de Tabou.....	137
Figure 3.18: Ordonnancement conjoint P/M par la stratégie hybride.....	138
Figure 3.19 : Perte de performance de l’heuristique NEH.....	139
Figure 3.20: Perte de performance de la recherche Tabou.....	139
Figure 3.21: Perte de performance de l'algorithme génétique.....	140
Figure 3.22: Comparaison entre toutes les approches.....	141
Figure 3.23 : Démarche de résolution.....	142
Figure 4.1 : Exemple d'un ordonnancement robuste [JEN01a].....	153
Figure 4.2 : Exemple d'ordonnancement flexible [JEN01a].....	154
Figure 4.3 : Graphe de la fonction d'optimisation continue.....	160
Figure 4.4: Situation du modèle par rapport au schéma global.....	171
Figure 4.5: Démarche Classification – Validation.....	176
Figure 4.6 : Exemple de perturbation d'une tâche.....	180
Figure 4.7: Chevauchement horizontal et vertical entre deux tâches.....	185
Figure 4.8 : Evolution de l'intensité des perturbations d'un niveau.....	187
Figure 4.9 : Intervalle de fiabilité.....	188
Figure 4.10 : Intervalle de Fiabilité et de Perturbation.....	189
Figure 4.11 : Chevauchement horizontal entre deux tâches de production.....	192
Figure 4.12 : Résolution du chevauchement P/M par décalage à droite (Right Shifting).....	193
Figure 4.13: Résolution du chevauchement P/M par permutation.....	193
Figure 4.14: Schéma de Classification.....	197
Figure 4.15: Distribution des cas acceptables par échantillons et intervalle de confiance ....	200
Figure 4.16: Distribution des cas acceptables par critère et IF pour l'AGS.....	202
Figure 4.17: Distribution des cas acceptables par critère et IF pour l'AGI.....	203

Figure 4.18: Distribution des cas acceptables par méthode et IF pour l'AGS.....	205
Figure 4.19: Distribution des cas acceptables par méthode et IF pour l'AGI. ....	206



---

## *Listes des Tableaux*

---

Tableau 1.1: les ressources nécessaires pour chaque niveau de maintenance. ....	32
Tableau 2.1: Critères d'optimisation .....	46
Tableau 2.2 : Classification des articles en fonction du type de maintenance et du problème d'ordonnement. ....	75
Tableau 2.3 : Classification des travaux en fonction de la stratégie de résolution .....	77
Tableau 3.1 : Résultats par heuristique d'insertion de la maintenance .....	116
Tableau 3.2 : Résultats par paramètre de la recherche Tabou intégrée.....	132
Tableau 3.3: Résultats par opérateurs des AG intégrés.....	132
Tableau 3.4: Tableau récapitulatif des meilleurs écarts par rapport à la production .....	141
Tableau 4.1: Les niveaux de perturbation .....	180
Tableau 4.2: Les classes de perturbation.....	180
Tableau 4.3 : Sous-Intervalles des mentions.....	184

---

## *Listes des Algorithmes*

---

Algorithme 2.1: Méthode générique de la recherche Tabou.....	54
Algorithme 2.2 : Principe de fonctionnement des AG.....	57
Algorithme 3.1: Génération aléatoire d'une solution.....	97
Algorithme 3.2: Méthode de recherche Tabou.....	98
Algorithme 3.3: Génération aléatoire de la population initiale.....	99
Algorithme 3.4 : Croisement à k points.....	100
Algorithme 3.5: Heuristique de Pamler(resp. Gupta) intégrée.....	120
Algorithme 3.6: Heuristique RA intégrée.....	121
Algorithme 3.7: Heuristique CDS intégrée.....	122
Algorithme 3.8: Génération aléatoire d'une solution conjointe production/maintenance.....	124
Algorithme 3.9: Génération du voisinage par décalage des tâches de maintenance.....	125
Algorithme 3.10: Exploration du voisinage.....	125
Algorithme 3.11: Croisement horizontal à k points pairs.....	127
Algorithme 3.12: Mutation aléatoire sur la maintenance.....	129
Algorithme 3.13: Mutation verticale à un point.....	130
Algorithme 3.14 : AG hybride.....	135
Algorithme 3.15: Stratégie aléatoire d'insertion de la maintenance.....	137
Algorithme 4.1: Principe de perturbation de l'ordonnancement de production.....	186
Algorithme 4.2: Principe de perturbation d'une ordonnancement conjoint.....	191

---

# **INTRODUCTION**

## **GENERALE**

---



---

## *Introduction Générale*

---

L'environnement actuel des entreprises est caractérisé par des marchés soumis à une forte concurrence et sur lesquels les exigences et les attentes des clients deviennent de plus en plus forte en termes de qualité, de coût et de délais de mise à disposition. Cette évolution se renforce par le développement rapide de nouvelles technologies de l'information et de la communication qui permettent une relation directe entre les entreprises (*Business to Business*) et entre les entreprises et leurs clients (*Business to Customer*) [LOP01].

Dans un tel contexte, rester toujours performant, passe obligatoirement par le maintien en état de fonctionnement de l'outil de production, qui reste toujours la préoccupation majeure de tous les gestionnaires dans un monde industriel où les notions de réactivité, de coûts et de qualité ont de plus en plus d'importance, et où il est vital de pouvoir s'appuyer sur un système de production performant à tout instant.

Ces multiples contraintes, marchande, productive et financière, ont des incidences majeures sur les fonctions qui ont des liens très étroits avec la production au niveau opérationnel. Ces liens intrinsèques de la production avec ces fonctions mitoyennes, et particulièrement la maintenance, qui est en charge d'assurer la disponibilité des moyens de production par des interventions curatives mais aussi préventives, font que leurs objectifs respectifs peuvent être partiellement en conflit.

En effet, la maintenance et la production sont deux fonctions qui agissent sur les mêmes ressources. Cependant l'ordonnancement de leurs activités respectives est indépendant, et ne tient pas compte de cette contrainte de taille. Les ressources (machines) sont toujours considérées comme disponibles à tout moment ou éventuellement durant certaines fenêtres de temps. Dès lors la planification de la maintenance n'est jamais prioritaire sur la production, pour effectuer des interventions préventives. Lorsque le responsable de la maintenance demande à immobiliser une ressource pour effectuer une opération de maintenance préventive, cela est généralement ressenti par les responsables de la production comme une perturbation supplémentaire et souvent malvenue. Cet arrêt de la machine a généralement été planifiée séparément des tâches de production et ne s'insère que très rarement dans un créneau horaire où elle ne nuirait pas à la productivité.

Cependant, les interventions de maintenance préventive ont un rôle prépondérant, puisqu'elles permettent au système de production de fonctionner de façon nominale. En effet, le coût engendré par une panne (impliquant une maintenance corrective, un arrêt non programmé de la production, des retards conséquents de livraison, etc.) peut être largement supérieur à celui d'un arrêt prévu de la production.

Nous pouvons considérer quatre facteurs de complexité du problème :

- ➔ La typologie des systèmes de production : atelier à flux de production unique, parallèle ou hybride ;
- ➔ Les tâches de maintenance : une seule maintenance périodique de chaque machine, des tâches de maintenance pouvant être préemptées par la production, ou encore un ensemble de tâches de maintenance préventive par machine ;
- ➔ L'horizon de planification : à long terme, à moyen terme (sur un horizon de 6 mois à 2 ans) ou à court terme (horizon de 1 jour à 1 mois);
- ➔ Les compétences des ressources humaines : un seul intervenant en maintenance, plusieurs intervenants en maintenance ou des équipes constituées en fonction de la compétence de chacun et des besoins du système de production.

Le travail présenté ici propose quelques éléments de réponse au problème de la planification commune et intégrée des tâches de maintenance et de production, avec comme objectif le respect des contraintes intrinsèques au problème. Les données du plan de production sont fournies par la gestion de production. Le plan de maintenance est caractérisé par un ensemble de tâches de maintenance préventive systématique. Chaque opération de maintenance est liée à une machine et nécessite la présence d'un intervenant en maintenance. L'objectif commun à atteindre serait la minimisation de la durée de l'ordonnancement de production d'une part, et la minimisation des retards de maintenance d'autre part.

Nous sommes donc face à un problème multicritères, d'une part ordonnancer la production sous les contraintes de respect des délais, coût et qualité des produits. Et d'autre part, planifier la maintenance, sous les contraintes de sûreté de fonctionnement des équipements, qui assurent la disponibilité et la pérennité de l'outil de production. La complexité du problème [CAR88] dû aux incertitudes, liées notamment aux données et des objectifs à atteindre, nous fait penser qu'une approche par les méthodes exactes n'est pas envisageable, et qu'une approche par les méthodes heuristiques permet de résoudre, même partiellement, ce type de problème.

L'objectif de notre travail est double :

- D'une part de démontrer la nécessité de développer des heuristiques d'ordonnancement conjoint production/maintenance pour atteindre l'objectif d'optimisation de la sûreté de fonctionnement du système de production et ceci grâce à une « bonne » maintenance et par voie de fait celui de la maximisation de la rentabilité. Dans ce contexte nous avons proposé l'adaptation d'un certain nombre d'heuristiques des approches constructive, par voisinage et évolutive pour le cas de l'ordonnancement conjoint production/maintenance dans un atelier de type *flowshop* de permutation ;
- D'autre part démontrer que la perte de performance, due à l'insertion des tâches de maintenance, se traduit par un gain en terme de robustesse et de stabilité pour les ordonnancements conjoints production/maintenance.

Ce mémoire est organisé en quatre chapitres comme suit :

Face à la bivalence structurelle et à l'importance primordiale de la production et de la maintenance puisque la longévité du premier dépend de la bonne mise en œuvre du second, le premier chapitre sera présenté en deux parties. La première partie abordera succinctement les systèmes de production à travers la définition de ce qu'est le système de production manufacturier au sein de l'entreprise. La deuxième partie sera consacrée à la maintenance des systèmes de production, nous y présenterons les concepts de la maintenance, son évolution ainsi que le positionnement et les interactions entre la fonction maintenance et les autres fonctions de l'entreprise.

Le second chapitre se présentera aussi en deux parties : la première relative à l'ordonnancement de production et la seconde à la planification des activités de maintenance. Nous présenterons, en première partie, les problèmes d'ordonnancement de la production auxquels nous nous intéressons dans cette thèse. En seconde partie le problème d'ordonnancement de la maintenance est exposé. La situation conflictuelle entre ordonnancement de production et planification de la maintenance et, plus particulièrement, la nécessité de coordination entre les deux, est présentée. Par la suite, nous présenterons la triptyque de notre travail à savoir : ordonnancement conjoint production/maintenance des activités de production et de maintenance préventive systématique, dans un atelier de type *flowshop* de permutation. Enfin, un état de l'art est présenté sur les travaux existants et relatifs aux problématiques étudiées, dans cette thèse.

Le troisième chapitre présente l'adaptation des différentes heuristiques, des approches constructive, par voisinage et évolutive, dans le cas de l'ordonnancement conjoint production/maintenance par les stratégies séquentielle et intégrée. Une étude comparative entre les différents résultats obtenus fait suite à cela. Il est organisé en trois parties. Nous présentons dans la première partie les données de notre problème concernant la production et la maintenance ainsi que la fonction objectif commune. Nous présentons dans la seconde partie l'adaptation des heuristiques d'ordonnancement de la production en y intégrant les activités de maintenance préventive par les trois stratégies: séquentielle, intégrée et hybride. La dernière partie est consacrée aux tests et résultats obtenus.

Enfin dans le dernier chapitre, nous étudierons les influence possibles, en terme de robustesse et de stabilité, après insertion de la maintenance, sur les ordonnancements conjoints Production/Maintenance pour démontrer que la perte de performance dû à l'insertion de la maintenance dans l'ordonnancement de production est au profit de la robustesse de ce dernier. Et d'autre part d'essayer d'identifier parmi les critères d'optimisation classiques de potentiels critères de robustesse, qui permettraient, s'ils sont optimisés lors de la génération de l'ordonnancement, d'assurer une certaine robustesse de ce dernier face à des incertitudes lors de son exécution. Il est organisé en trois parties : nous présenterons dans la première un état de l'art succinct sur la robustesse des ordonnancements. Nous présenterons par la suite la problématique de ce chapitre à savoir robustesse et maintenance à travers le modèle que nous proposons pour cette étude. Nous y définissons les perturbations, ainsi que les niveaux et classes associés, prises en compte. Puis les critères d'optimisations retenus. La dernière partie est consacrée à l'étude proprement dite.

En conclusion, nous présenterons les apports de notre travail ainsi que les perspectives pour chacune des deux problématiques étudiées dans cette thèse. Le premier concernant les heuristiques d'ordonnancement conjoint production /maintenance, et le second relatif à l'étude de la robustesse de ces mêmes ordonnancements.



---

# CHAPITRE I

## *La Maintenance Des Systèmes de Production*

---



---

# **CHAPITRE I.**

## ***La Maintenance des Systèmes de Production***

---

---

**Résumé :** Dans le nouveau contexte industriel, les entreprises subissent de grandes pressions de la part de leurs clients. Ces derniers deviennent de plus en plus exigeants et demandent, en outre, des produits et des services de bonne qualité, à moindre coût, livrés rapidement et au bon moment et un service après-vente défiant la compétition. Pour satisfaire la demande en qualité et en quantité tout en respectant les délais de livraison et les coûts, l'entreprise manufacturière doit disposer d'un outil de production fiable, donc bien entretenu.

Les entreprises sont de plus en plus sensibilisées à l'importance des coûts induits par les défaillances accidentelles des systèmes de production. Alors que la maintenance, jusqu'à très récemment, était considérée comme un centre de coûts, nous sommes de plus en plus conscients qu'elle peut contribuer d'une manière significative à la performance globale de l'entreprise. La complexité des mécanismes de dégradation des équipements a fait en sorte que la durée de vie de ces derniers a toujours été traitée comme une variable aléatoire. Cet état de fait a incité plusieurs entreprises à adopter des approches plutôt réactives, n'étant pas en mesure de justifier économiquement les avantages que peut procurer la mise en place d'une maintenance préventive.

Face à cette bivalence structurelle et d'importance primordiale de la production et de la maintenance puisque la longévité du premier dépend de la bonne mise en œuvre du second, ce chapitre est présenté en deux parties. Dans la première partie, on abordera succinctement **les systèmes de production**, quand à la deuxième partie elle sera consacrée à **la maintenance des systèmes de production**

---

## 1. Introduction

La principale conséquence du développement industriel est la complexité croissante des machines et des équipements de production. Ainsi, pour satisfaire une demande de produits avec une meilleure qualité et à des prix compétitifs tout en respectant les délais de livraison, le développement des ateliers manufacturiers doit intégrer à la fois automatisation et flexibilité. D'où l'augmentation du risque d'occurrence des pannes qui se traduit par un temps croissant de détection et de réparation des machines. Cela aurait, dans ce cas, pour effet la diminution de la disponibilité du système global. Mais comment assurer le fonctionnement des machines, maîtriser les délais de livraison et améliorer la disponibilité et la rentabilité des ateliers manufacturiers ?

Actuellement, la maintenance s'impose comme la meilleure solution permettant d'accroître les performances et d'améliorer le niveau de sûreté de fonctionnement, de tout système industriel. Elle permet d'assurer la pérennité des équipements et de veiller à ce que le système ne tombe pas en panne et donc de maintenir le fonctionnement de l'appareil de production au plus au niveau d'efficacité et garder ainsi le seuil de productivité à un niveau stable.

Les entreprises sont de plus en plus sensibilisées à l'importance des coûts induits par les défaillances accidentelles des systèmes de production manufacturiers. La maintenance, jusqu'à très récemment, était considérée comme un centre de coûts. Actuellement, les gestionnaires et décideurs, dans l'entreprise, sont de plus en plus conscients qu'elle peut contribuer d'une manière significative à la performance globale de l'entreprise. La complexité des mécanismes de dégradation des équipements a fait en sorte que la durée de vie de ces derniers a toujours été traitée comme une variable aléatoire. Cet état de fait a incité plusieurs entreprises à adopter des approches plutôt réactives, n'étant pas en mesure de justifier économiquement les avantages que peut procurer la mise en place d'une maintenance préventive.

Face à l'importance primordiale de la production et de la maintenance puisque la pérennité du premier dépend de la bonne mise en œuvre du second, ce chapitre sera présenté en deux parties. La première partie abordera succinctement **les systèmes de production** à travers la définition de ce qu'est le système de production manufacturier, ses constituants à savoir le système physique de production et le système de gestion de production, une présentation brève de la typologie des systèmes de production, et enfin les critères d'évaluation d'un système de production.

La deuxième partie sera consacrée à **la maintenance des systèmes de production** : nous commencerons par définir le concept de maintenance, son évolution ainsi que le positionnement et les interactions entre la fonction maintenance et les autres fonctions de l'entreprise. Nous passerons en revue les différents types de maintenance. Nous ferons également état dans ce chapitre, de ce qu'est le système de maintenance puis nous présenterons des outils permettant une gestion performante de la maintenance industrielle : la Gestion de la Maintenance Assistée par Ordinateur (G.M.A.O.). Nous concluons ce chapitre avec les principaux éléments qui caractérisent les nouvelles approches dans le domaine de la maintenance.

## **2. Les systèmes de production - Généralités**

Produire, c'est transformer [SAS98]. Le lieu et les moyens de ces transformations, c'est le système de production. Un processus de production est généralement composé d'un grand nombre d'opérations ou de transformations organisées en réseau. Ces opérations assurent des transformations de forme (modification des produits eux même), des transformations dans le temps (fonction de stockage) ou dans l'espace (fonction de transport). Les systèmes de production diffèrent par les objectifs que s'assigne le producteur, les attributs des objets transformés et par les caractéristiques des processus de production. Ils ne sont pas statiques et ils évoluent sous les effets conjugués de l'évolution du marché, des technologies et des sociétés dans lesquelles ils sont insérés. Ils se définissent par les attributs de sortie : coût des produits, qualité des produits, quantité des produits par unité de temps, délais moyen de livraison d'un produit (ou temps de service), etc.

La notion de système de production doit être vue au sens large de système industriel manufacturier et doit intégrer les principaux processus fonctionnels : définition et mise en œuvre des stratégies d'entreprise, conception des produits et des processus de production, conduite et gestion de ces systèmes.

De manière générale, nous pouvons classer les systèmes de production en trois grandes classes [SAS98, AMO99] :

- **Les processus continus** tels que la production électrique, la chimie ou la papeterie ;
  - **Les processus discrets** tels que l'usinage et toutes les activités d'assemblage, etc.
- Cette vision des systèmes est très fréquente dans l'industrie manufacturière ;
- **Les processus discontinus** qui se situent par définition à mi-chemin entre les processus continus et les processus discrets. Les deux types de processus sont couplés : la production est continue mais il y a un conditionnement discret des produits

Nous nous intéresserons dans ce qui suit aux systèmes de production manufacturiers qui se caractérisent par leur nature discrète.

La théorie des systèmes appliquée aux systèmes de production permet de décomposer ces derniers en trois sous-systèmes : **le système physique de production, le système d'information** et **le système de décision** [DOU84, ROB88] (Fig. 1.1). Le système couramment appelé **système de gestion de production** est constitué par la partie du système de décision et du système d'information traitant des fonctions rattachées directement à la production (par exemple, les achats, les approvisionnements, la planification, la gestion des ressources, la maintenance, etc.).

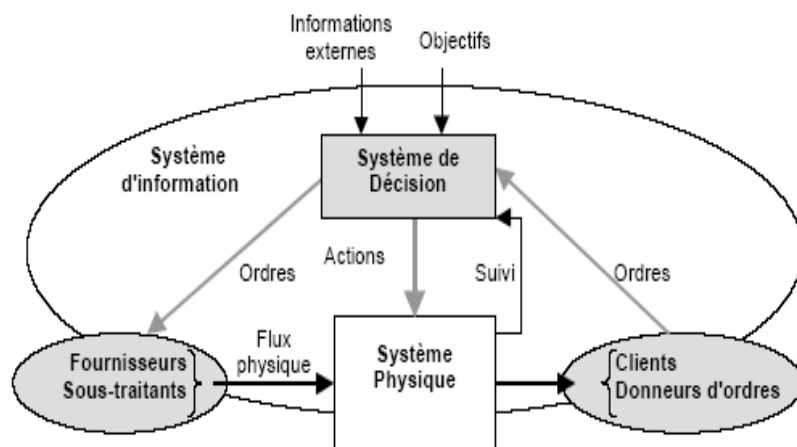


Figure 1.1: Modèle conceptuel d'un système de production

En terme de système, le **système physique** transforme les matières premières en produits finis. Pour effectuer cette transformation, il est commandé par le **système de gestion** qui transforme les informations à caractère commercial en ordres de fabrication et ordres d'approvisionnements. Le système est bouclé puisqu'en retour, il reçoit les informations de suivi du système physique pour pouvoir effectivement piloter ce dernier.

En terme de flux, un ensemble de flux régulés parcourent le système de production. Tout d'abord, le **flux physique ou de matière** qui transforme la matière première et les composants en produits finis, puis le **flux d'information ou de suivi** qui permet la circulation des informations nécessaires au contrôle et à la prise de décision . Enfin, le **flux de décision ou ordre** qui contrôle et pilote le système physique.

Le système de gestion est composé de différentes activités telles que :

- ❑ L'élaboration du Plan Directeur de Production (PDP);
- ❑ Le calcul des besoins bruts, nets et d'approvisionnements ;
- ❑ La gestion des stocks ;
- ❑ Les achats ;
- ❑ L'élaboration du plan de charge ;
- ❑ L'ordonnancement ;
- ❑ Et enfin le lancement.

Le système de gestion pilote le système physique pour qu'il atteigne les objectifs fixés. Cette gestion de la production s'effectue par un ensemble de décisions qui peuvent être hiérarchisées suivant un ordre temporel :

- ➔ **Le long terme** : les décisions prises sont **stratégiques** et influent sur l'évolution des produits et du système de production. Ces décisions définissent le Plan Directeur de Production ;
- ➔ **Le moyen terme** : il s'agit de décisions **tactiques** qui fixent la gestion des stocks, la planification de la production (quantités, positionnement sur le calendrier, ...) ;
- ➔ **Le court terme** : les décisions sont **opérationnelles**. Elles déterminent l'ordonnancement et le pilotage ;
- ➔ **Le très court terme** : c'est le niveau d'exécution du système. Il transmet les ordres de fabrication et d'affectation des ressources au système physique.

Le schéma de la figure 1.2 présente l'interaction qui existe entre les activités du système de gestion et les niveaux de planification de ces activités.

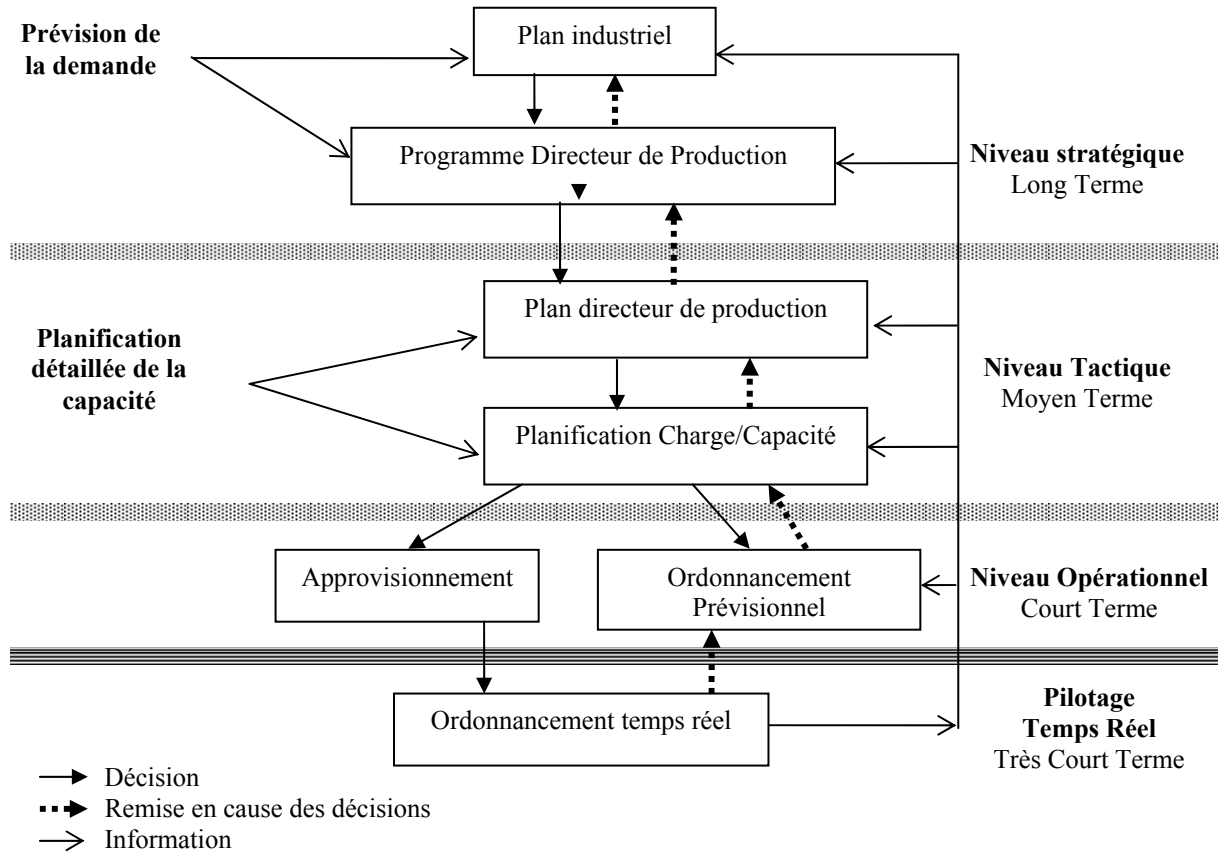


Figure 1.2: organisation hiérarchique d'un système de gestion de production.

Nous situons nos travaux au niveau opérationnel (Fig. 1.2 - Court terme) pour générer des ordonnancements prévisionnels. Ces ordonnancements sont de nature statique, vu qu'ils ne prennent pas en considération les aléas qui risquent de se produire, lors de leur exécution en atelier. La prise en charge des aléas s'effectue au niveau pilotage Temps réel (Fig. 1.2 - Très court terme), et les ordonnancements de ce niveau, sont de nature dynamique.

## 2.1. Typologie des systèmes de production

Un processus de fabrication est caractérisé par la séquence des opérations nécessaires pour la production d'un bien spécifique. Face à la diversité et à la complexité de ces processus de fabrication, les systèmes de production sont classés et gérés en fonction des critères de choix. Giard [GIA88] propose deux typologies : une première liée à l'origine de la demande et une seconde liée aux ressources.



### 2.1.1. Typologie liée à la demande

Elle est basée sur le fait que les ordres de fabrication émanent, soit directement des commandes, soit pour réapprovisionner le stock des produits finis. Deux types de systèmes de production en découlent :

- ***Systèmes basés sur la production à la commande*** : la production est déclenchée par les commandes fermes du client ou par les demandes aléatoires. Dans ce type de structure, le délai de fourniture se négocie avec le client, par contre l'état de stock en produit finis est quasiment nul.

- ***Systèmes basés sur la production pour stock*** : le client commande les articles selon les spécifications définies par l'entreprise. La production est alors déclenchée par anticipation d'une demande solvable. Nous remarquons dans ce cas que les délais de fourniture sont nuls, par contre le problème de gestion de stock se pose.

### 2.1.2. Typologie liée aux ressources

Elle est liée à la façon dont sont organisées les ressources pour traiter les flux de matières premières. Quatre types de systèmes de production en découlent :

- ***Système à production unitaire ou projet*** : ce type d'organisation concerne la réalisation de grands projets uniques (ou de petites séries) sur des périodes assez longues. La taille du projet étant importante relativement aux moyens mis en œuvre, ce sont les moyens de production qui viennent sur le site de construction pour effectuer les opérations sur le produit. L'objectif étant de réaliser le projet dans un délai optimal, le problème dans ce cas consiste à coordonner simultanément les tâches avec les moyens à mettre en œuvre en planifiant la succession des différentes opérations. Les méthodes de modélisation les plus répandues sont le diagramme de Gantt, les méthodes des potentiels [HAR95] et PERT [BAV02].

- ***Système de production en petite ou moyenne série ou atelier*** : il s'agit dans ce système de produire une grande variété de produits en faible quantité, tout en utilisant les mêmes moyens de production. Dans cette organisation, les équipements sont regroupés par fonctionnalité équivalente dans un ensemble d'ateliers spécialisés.

Une classification des ateliers peut s'opérer selon l'ordre d'utilisation des machines pour fabriquer un produit (gamme de fabrication). On rencontre des ateliers à cheminement unique (*flowshop*) où toutes les gammes sont identiques, des ateliers à cheminements multiples où chaque produit ou famille de produits possède (*jobshop*) ou ne possède pas (*openshop*) une gamme spécifique.

- ***Système de production en grande série ou masse*** : la production de masse s'appuie sur la fabrication de produits standards à grande consommation. Dans un tel système organisé en ligne de fabrication, les produits passent successivement et dans le même ordre par une séquence identique de postes de travail. Les équipements, qui sont dans ce cas spécialisés, sont agencés de manière à assurer un flux continu de produits sur toutes les lignes.
- ***Système de production continue ou processus*** : ce mode de production concerne les systèmes où la matière circule en flux continu. Peu de produits sont fabriqués mais en quantité très importante. Cela nécessite de forts investissements en équipements et peu de main d'œuvre. Il découle de ce type d'organisation une grande importance de la fonction maintenance et un niveau de stock d'en-cours quasiment nul.

## **2.2. Critères d'évaluation d'un système de production**

Les systèmes de production devenant de plus en plus complexes suite à une grande flexibilité et un fort degré de d'automatisation, leur conception et leur exploitation nécessite des techniques d'évaluation basés sur deux principaux critères : la performance et la sûreté de fonctionnement.

### **2.2.1. Les performances**

La performance d'un système peut être définie comme étant l'efficacité à fournir un service attendu à un instant donné et dans des conditions prédéterminées. Dans le domaine industriel, Tarondeau [TAR96] définit la performance comme étant un attribut mesurable et observable par lequel se définit la qualité d'un produit ou la rapidité d'un service. Elle est liée à tout le cycle de vie du système de production, à sa productivité, aux stocks et en-cours, aux coûts de production, aux délais de livraison, à la qualité du produit, etc.

Une façon de la mesurer consiste à analyser les objectifs à atteindre. Cette idée d'écart [CAR91] traduit la distance qui peut exister entre ce que l'entreprise obtient actuellement comme résultat (production, chiffre d'affaires, etc.) avec les moyens dont elle dispose, et ce qu'elle souhaite atteindre comme objectif. En théorie, c'est le fait d'atteindre les objectifs qui constitue le critère clé de la performance « réfléchie » de l'entreprise. En pratique, l'atteinte de l'objectif de l'entreprise est un indicateur de performance crédible dans la mesure où les objectifs sont eux-mêmes définis de manière volontariste.

Cette performance peut être qualitative (image de l'entreprise), mais souvent quantitative (Rentabilité, Délais, Coûts de production), la performance économique étant l'élément principal qui guide l'évaluation du SdP. Nous évaluons ainsi la performance par un ensemble d'indicateurs liés au suivi de la production tels que: la productivité, les délais, les coûts, la rentabilité, etc.

### **2.2.2. La sûreté de fonctionnement**

La Sûreté de Fonctionnement (SdF) permet d'établir le degré de fonctionnement que l'on peut attribuer à un système dans le cadre de la mission qu'il doit assurer. Dans le domaine de la production manufacturière, la sûreté de fonctionnement consiste à assurer le respect du cahier des charges en terme de productivité, en tenant compte des perturbations (défaillances, aléas, etc.) affectant un atelier, en limitant les coûts de conception et de fonctionnement, et en assurant une qualité et une disponibilité maximale. Elle est liée à tout le cycle de vie du système de production, à sa disponibilité, à la fiabilité et à la maintenabilité de ses machines, etc. elle consiste à connaître, détecter, évaluer, prévoir, mesurer et maîtriser les défaillances des machines [ZWI96].

Les moyens et les méthodes d'analyse de la SdF reposent sur la prévention, la tolérance, la prévision et l'élimination au mieux des défaillances. Ainsi, l'intégration d'un service de maintenance est indispensable pour l'amélioration de la SdF et l'augmentation des performances des systèmes de production.

### **2.3. Conclusion**

Dans cette section, nous avons présenté, de manière générale, les systèmes de production tant au niveau fonctionnel qu'au niveau structurel. Nous avons énoncé un certain nombre d'hypothèses concernant ce type de systèmes, afin d'introduire l'intérêt de la maintenance.

Le maintien en état de fonctionnement de l'outil de production a toujours été la préoccupation majeure de tous les gestionnaires dans un monde industriel où les notions de réactivité, de coûts et de qualité ont de plus en plus d'importance, et où il est vital de pouvoir s'appuyer sur un système de production performant à tout instant. Dans ce contexte de juste à temps, la disponibilité des équipements, au moment voulu, est une condition nécessaire au bon déroulement de la production. Cependant les arrêts du système de production, à des fins de maintenance, ont longtemps étaient considérés comme une source de perturbations et de perte de productivité. Cette activité autrefois un peu délaissée au profit de la production s'annonce

aujourd'hui comme une voie d'amélioration de la productivité et de la rentabilité du système de production.

Dans ce qui suit, nous allons présenter ce qu'est la maintenance dans les systèmes de production. Nous listerons une diversité de politiques de maintenance. Nous introduirons également les hypothèses considérées tout au long de ce travail, accompagnées de quelques définitions formelles.

### 3. La maintenance dans les systèmes de production

L'évolution et la complexité grandissante des systèmes de production ainsi que le besoin de produire vite et bien, ont obligé les industriels à créer de nouveaux concepts d'organisation et de nouvelles manières d'intervenir sur des structures de production concernant les produits manufacturés. Aujourd'hui, l'**entretien** a laissé la place à la **maintenance**. Ce changement ne réside pas uniquement dans un changement de dénomination, mais aussi dans un bouleversement complet de la manière de faire et de concevoir ce qui s'appelait « entretien » et que l'on appelle aujourd'hui « maintenance ». Il y a quelques dizaines d'années, les ateliers de production ne disposaient d'aucune structure de maintenance. L'entretien des machines ou des unités de production se faisait par des personnes, spécialisées ou non, sans logistique établie et surtout définie. La production intensive, la complexité des systèmes de production et surtout la rentabilité, ont poussé les industriels à créer un nouveau domaine et une structure accompagnatrice dans l'entreprise appelée « service maintenance ».

L'objectif de la maintenance est de tout mettre en œuvre pour maintenir le système de production en état de fonctionnement et ceci en évitant les pannes, sinon en agissant rapidement lorsqu'elles surviennent pour augmenter la disponibilité du matériel. La maintenance a beaucoup évolué depuis le début de l'ère industrielle. Cette évolution est liée à l'émergence successive de nouvelles technologies induisant la mise en place de différents types de maintenance vue la transformation des caractéristiques des équipements.

Les concepts de la maintenance définissent des façons de faire pour maximiser la performance globale de l'entreprise. La mise en œuvre de ces concepts exige des ressources humaines compétentes, des outils et du matériel adaptés aux équipements et aux installations à entretenir, un système de gestion de pièces de rechange adéquat et un système d'information bien pensé pour assurer un échange efficace entre les différents intervenants.

Nous allons développer dans cette partie les concepts liés à la notion de maintenance ainsi que son impact direct et indirect sur son environnement.

### 3.1. La maintenance – Généralités

La Norme AFNOR X 60-010 définit la maintenance comme étant « *l'ensemble des actions permettant de maintenir ou de rétablir un bien dans un état spécifié ou en mesure d'assurer un service déterminé* ».

RICHET [RIC96] définit la maintenance comme étant « *l'ensemble des activités destinées à maintenir ou rétablir un bien dans un état ou dans des conditions données de sûreté de fonctionnement, pour accomplir une fonction requise. Ces activités sont une combinaison d'activités techniques, administratives et de management* »

Maintenir c'est donc effectuer des opérations (dépannage, graissage, visite, réparation, amélioration, etc.) qui permettent de conserver le potentiel du matériel pour assurer la continuité et la qualité de la production. Bien maintenir, c'est assurer ces opérations au coût global optimum.

Retour & al [RET90] présentent la fonction maintenance comme un ensemble d'activités regroupées en deux sous-ensembles : les activités à dominante technique et les activités à dominante gestion comme le montre la figure 1.3.

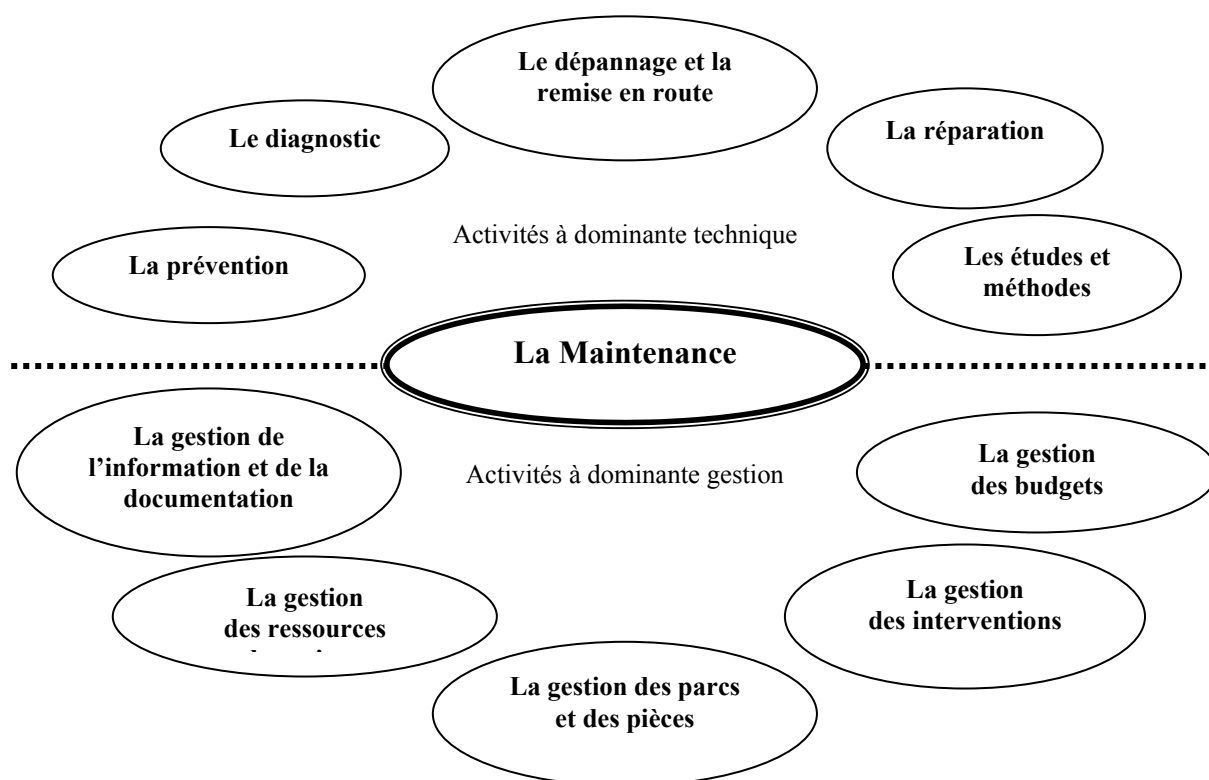


Figure 1.3: le contenu de la fonction maintenance.

### 3.1.1. Evolution de la maintenance

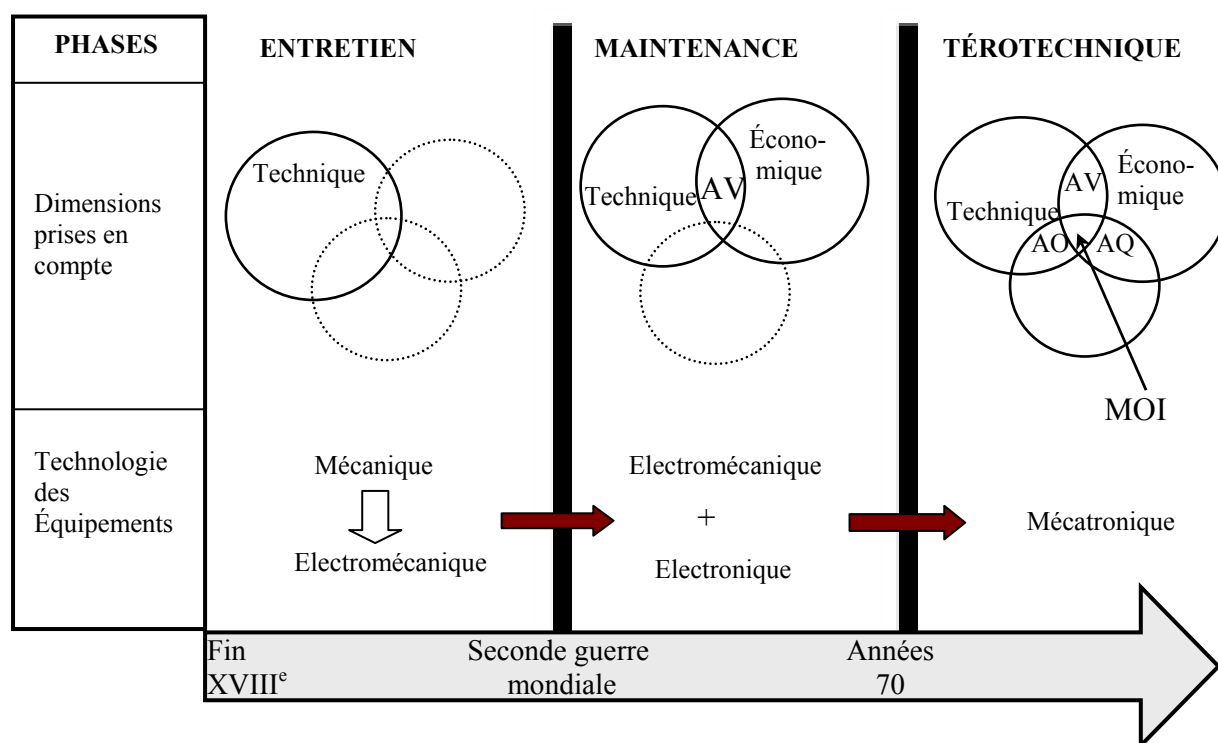
Les premières approches scientifiques de la gestion de maintenance datent des années cinquante et soixante [MCA65, PIE76]. A cette époque, la maintenance a été préconisée comme un moyen permettant de réduire les défaillances et les accidents imprévus. Dans plusieurs entreprises, de très gros programmes de maintenance basés sur le temps (préventive) ont été développés. Les premiers modèles de recherche opérationnelle pour la maintenance sont apparus dans les années soixante pour essayer d'optimiser ces programmes [VER88]. Dans les années soixante-dix, grâce aux contrôles des ateliers et à la surveillance, l'utilisation de l'information sur l'état actuel de l'équipement a permis de se concentrer sur des techniques pouvant prédire des défaillances. Cela semblait être plus efficace que les gros programmes de maintenance préventive. Des études détaillées de la part des fabricants, des défaillances de leurs produits ont abouti à de meilleures conceptions, avec moins de défaillances. Dans les années quatre-vingt, l'ordinateur apporte de l'aide à la fonction maintenance. Initialement, il a été utilisé pour faciliter les tâches administratives, ensuite pour la gestion de l'information disponible et, de nos jours, pour l'aide à la décision [BOU88].

Les progrès technologiques des équipements de production, fruits de la révolution industrielle, ont orchestré cette évolution de la fonction maintenance qui a connu trois phases (Fig. 1.4) :

1. Dans la première phase dite d'entretien, la priorité est accordée à la réalisation (dimension technique de l'activité). La fonction maintenance était purement technique. Elle était réduite aux dépannages et aux réparations, elle correspondait donc à ce qu'on appelle actuellement la maintenance corrective.
2. Dans la deuxième phase de maintenance proprement dite, l'importance est donnée à la dimension économique. La gestion du travail de maintenance et la gestion des coûts de maintenance sont le résultat de l'articulation des dimensions technique et économique, par l'analyse de la valeur (AV).
3. Dans la troisième dite térotechnique (de térotechnologie<sup>1</sup>), où on s'intéresse à la dimension sociale impliquant les utilisateurs des équipements de production, qui a conduit à un mode d'organisation intégré (MOI) des fonctions maintenance et production

---

<sup>1</sup> La térotechnologie est « une combinaison de principes de gestion, de finance, d'ingénierie de construction, etc., appliqués à des actifs physiques dans la recherche de coûts de cycles de vie les plus économiques » [VER88]



AV : Analyse de la Valeur  
AQ : Analyse de la Qualité

AO : Analyse de l'Organisation.  
MOI : Mode d'Organisation Intégrée.

Figure 1.4 : Phase d'évolution de la fonction maintenance

### 3.1.2. Les modes de gestion de la maintenance

La maintenance est une fonction technique de l'industrie, basée sur une organisation, qui doit toujours permettre une meilleure intégration dans l'entreprise dans le but d'assurer et de soutenir la production. Elle peut être assurée, de manière interne, par un service indépendant appelé service maintenance ou par un service technique intégrant les fonctions de fabrication et de maintenance. Comme elle peut être externalisée, pour être assurée par un service externe à l'entreprise

Les modes de gestion de la maintenance, au sein de l'entreprise, dépendent des décisions stratégiques prises par cette dernière. Ces décisions portent sur les ressources requises pour effectuer les activités de maintenance, sur les méthodes à utiliser et sur les instants auxquels les activités doivent être exécutées. Il peut y avoir plusieurs tendances [KAF01], l'entreprise peut décider pour chaque processus (activité) de maintenance de le réaliser en totalité ou en partie par des processeurs (les ressources humaines, matérielles et les pièces de rechange) internes ou externes. Le choix est dicté essentiellement par les décisions stratégiques de faire, faire-faire ou faire ensemble. La première option fait uniquement appel à des processeurs (ressources humaines, matérielles et pièces de rechanges) internes ; la seconde fait intervenir

des processeurs externes pour réaliser une partie ou la totalité des processus de maintenance ; la troisième propose de créer des alliances stratégiques avec d'autres partenaires pour réaliser les processus propres au système de maintenance.

**- Faire : Internaliser**

Ce choix de gestion implique que toutes les activités de maintenance soient assurées par les ressources internes de l'entreprise. Il va sans dire que, pour chaque processus concerné, les ressources nécessaires sont disponibles pour exécuter efficacement les différentes activités, tant techniques que de gestion, tout en apportant à l'entreprise un avantage concurrentiel.

**- Faire-faire : Externaliser**

La réalisation d'une partie ou de la totalité des processus ou activités de maintenance est confiée à des intervenants externes. Les organisations ayant opté pour ce type de gestion de la maintenance se sont fondamentalement basées sur des considérations économiques et technologiques. Les activités de chaque processus concerné sont sous-traitées ou imparties. L'externalisation de la maintenance permet à l'entreprise de se concentrer sur son cœur de métier. Mais surtout, elle lui offre la possibilité de maîtriser les coûts tout en bénéficiant des services d'un prestataire spécialisé, qui lui donnera accès aux dernières innovations technologiques.

**- Faire ensemble : Coopérer**

De nos jours, les accords de coopération entre les entreprises se multiplient. De tels accords sont, le plus souvent, conclus librement et misent généralement sur la participation active des partenaires. Ils reposent d'emblée sur une complémentarité des ressources, des technologies et du savoir-faire. Cependant, les partenaires peuvent aussi chercher, en s'associant, à combler un besoin commun comme celui de la maintenance.

Ces options amènent le décideur à se prononcer sur les questions suivantes :

- Comment choisir la meilleure option stratégique pour chaque processus de maintenance ?
- Quels critères de sélection retenir ?
- Comment choisir le partenaire externe ?
- Quels avantages l'entreprise peut-elle tirer en faisant affaire avec un partenaire externe ?
- Quelles sont les modalités d'opération avec les partenaires retenus pour une relation gagnant-gagnant ?



## 3.2. Les politiques de maintenance

Dans la définition de la maintenance, nous trouvons deux mots-clés : maintenir et rétablir. Le premier fait référence à une action préventive, soit avant la survenue de la panne. Le but de la maintenance préventive est d'exécuter des tâches (selon un programme établi) permettant d'anticiper et donc d'éviter une panne. Le deuxième fait référence à l'aspect correctif. Ce dernier permet de réagir (par des dépannages ou des réparations) juste après l'occurrence d'une panne (Fig. 1.5). Il en découle donc deux principales catégories [BAR60, VAL89]: la Maintenance Préventive (MP) et la Maintenance Corrective (MC). Nous présentons dans les paragraphes qui suivent les définitions de chacune.

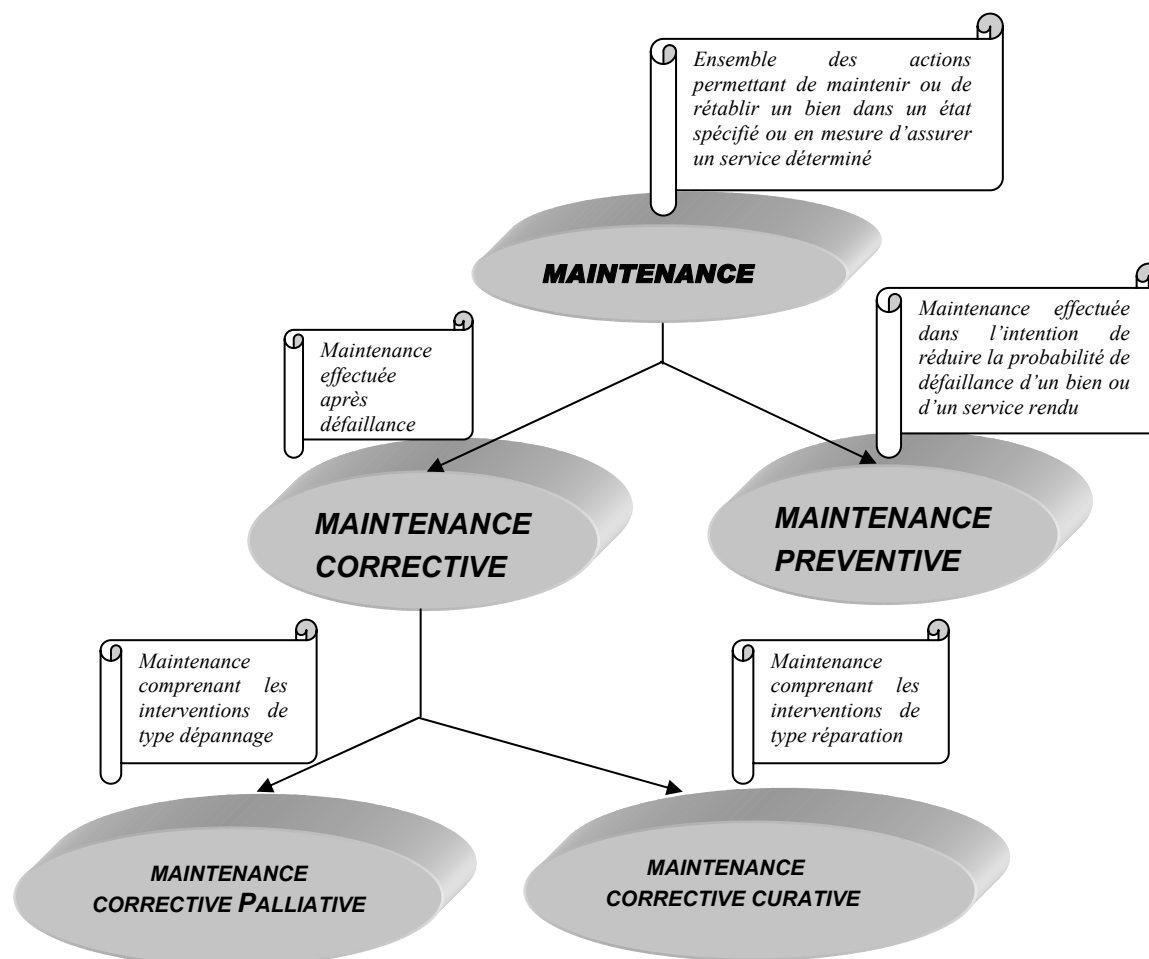


Figure 1.5: Typologie de la maintenance

### 3.2.1. La Maintenance Corrective

La Maintenance Corrective (MC) est «*l'ensemble des activités réalisées après la défaillance du bien ou la dégradation de sa fonction pour lui permettre d'accomplir une fonction requise, au moins provisoirement. Ces activités comportent notamment la localisation de la défaillance partielle ou complète et son diagnostic, la remise en état avec ou sans modification, et enfin le contrôle du bon fonctionnement*» [NFX60-010].

Elle intervient après la panne et devra s'appliquer automatiquement aux défaillances complètes et soudaines (défaillances catalectiques). Ce type de maintenance est réservé au matériel peu coûteux, non stratégique pour la production et dont la panne aurait peu d'influence sur la sécurité.

Trois politiques de maintenance corrective sont distinguées (maintenance corrective directe, différée et globale) et en tout, cinq types sont proposés :

- **Maintenance corrective palliative.** Elle regroupe les activités de maintenance corrective destinées à permettre à un équipement d'accomplir provisoirement une fonction requise. Appelée couramment dépannage, cette maintenance palliative est principalement constituée d'actions à caractère provisoire qui devront être suivies d'actions curatives.
- **Maintenance corrective curative.** Elle regroupe les actions de maintenance corrective ayant pour objet de rétablir un équipement dans un état spécifié ou de lui permettre d'accomplir une fonction requise. Le résultat des activités réalisées doit présenter un caractère permanent. Ces activités peuvent être des réparations, des modifications ou des aménagements ayant pour objet de supprimer la ou les défaillance(s).
- **Maintenance corrective directe.** C'est une opération de maintenance effectuée juste après la détection (localisation) de la défaillance, destinée à remettre la machine dans un état de fonctionnement normal. Elle peut avoir un caractère provisoire (palliative) ou définitif (curative).
- **Maintenance corrective différée.** C'est une opération de maintenance corrective (palliative ou curative) qui n'est pas déclenchée immédiatement après une détection de défaillance mais est retardée conformément à certaines règles définies préalablement. Par exemple, attendre que  $r$  équipements parmi  $m$  soient défaillants.
- **Maintenance corrective globale.** Elle est effectuée lorsque tout le système (toutes les machines) tombent en panne et sa remise en marche (provisoirement ou définitivement) ne peut être réalisée qu'après une intervention de durée importante.

### 3.2.2. La Maintenance Préventive

La Maintenance Préventive (MP) est définie comme une «*maintenance ayant pour objet de réduire la probabilité de défaillance ou de dégradation d'un bien ou d'un service rendu.*» [NFX60-010].

Elle permet d'éviter les défaillances des équipements en cours d'exploitation. Elle ne consiste pas à réparer les pannes mais à les anticiper. Elle est caractérisée par les actions suivantes :

- L'inspection qui consiste en une activité de surveillance s'exerçant dans le cadre d'une mission définie et qui n'est pas obligatoirement limitée à la comparaison avec des données préétablies. Cette activité peut s'exercer au moyen de rondes;
- La vérification de conformité avec des données préétablies, suivie d'un jugement. Elle peut comporter une activité d'information, inclure une décision (acceptation, rejet, ajournement) et déboucher sur les actions correctives;
- La visite (de maintenance) consistant en un examen détaillé et prédéterminé de tout (visite générale) ou d'une partie (visite limitée) des différents éléments de la machine et pouvant impliquer des opérations de maintenance de premier niveau. Certaines opérations de maintenance corrective peuvent être effectuées suite à des anomalies constatées lors de cette visite;

La maintenance préventive s'adresse aux éléments provoquant une perte de production ou des coûts d'arrêts imprévisibles jugés importants pour l'entreprise. L'analyse de ces coûts met en évidence un gain important dû aux arrêts qu'elle permet d'éviter. La figure 1.6 présente la réduction de l'accroissement du **taux de panne** de l'équipement par l'application d'une "bonne" maintenance préventive. Cependant, cet accroissement n'est pas réduit suffisamment par l'application d'une "mauvaise" maintenance préventive.

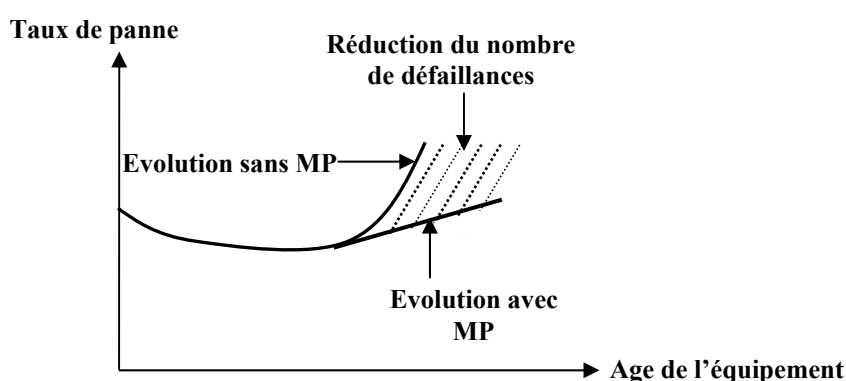


Figure 1.6 : Effet d'une bonne maintenance préventive

On peut distinguer quatre types de maintenance préventive : Maintenance Préventive Systématique, Maintenance Préventive Conditionnelle, Maintenance Prévisionnelle et Maintenance Proactive. Ces deux dernières représentent les nouvelles formes de MP.

- **Maintenance préventive systématique.** « *La maintenance préventive systématique comprend l'ensemble des actions destinées à restaurer, en totalité ou partiellement, la marge de résistance des matériels non défaillants, lorsque ces tâches sont décidées en fonction du temps ou de la production, sans considération de l'état des matériels à cet instant* » [NFX60-010].

Cette méthode nécessite de connaître : le comportement des équipements, les usures et les modes de dégradation. Elle intervient à intervalles fixés sur la base du minimum de vie des composants donnés par l'expérience ou par le constructeur. C'est pourquoi ce type de maintenance est aussi appelé maintenance préventive basée sur la durée de fonctionnement [BOU88].

La maintenance préventive systématique se traduit donc par deux types d'actions :

- des interventions planifiées qui consistent à nettoyer, réparer ou remplacer certains matériels tels que des composants ou sous-ensembles d'équipements ;
- des inspections périodiques qui consistent à contrôler ces mêmes composants et sous-ensembles, d'effectuer des révisions, mineures ou majeures, d'équipements, voire d'ateliers entiers lors d'arrêts généraux.

- **Maintenance préventive conditionnelle.** C'est une maintenance préventive subordonnée à un type d'événements prédéterminé (autodiagnostic, information donnée par un capteur, mesure d'une usure, etc.) révélateur de l'état de dégradation d'un bien [BOU88]. C'est donc une maintenance qui dépendant de l'expérience et fait intervenir des informations recueillies en temps réel [BOI90]. La pratique de la maintenance conditionnelle consiste à ne changer l'élément que lorsque celui-ci présente des signes de vieillissement ou d'usure mettant en cause, à brève échéance, ses performances.

- **Les Nouvelles formes de maintenance préventive.** La maintenance préventive conditionnelle a évolué vers la maintenance prévisionnelle (prédictive). Celle-ci est définie par la norme NF X 60-010 comme étant une « maintenance préventive subordonnée à l'analyse de l'évolution surveillée de paramètres significatifs de la dégradation d'un bien, permettant de retarder et de planifier les interventions ».

D'autre part, un nouveau type de maintenance a vu le jour aux Etats-Unis, il repose sur une démarche proactive. La maintenance proactive est une «*forme avancée de maintenance prévisionnelle consistant à déterminer les causes initiales de défaillances à partir de l'état de défaillance potentielle*».

La figure 1.7 présente la contribution des différents types de maintenance en ce qui concerne la fonction de fiabilité et la durée de vie utile de l'équipement. Il va sans dire qu'une réduction du taux de panne entraîne une amélioration de la fonction de fiabilité  $R(t)$ . C'est dans cette optique que la maintenance améliorative a été instaurée. La maintenance préventive, avec toutes ses variantes, va en revanche tenter de ramener le taux de panne à son niveau le plus bas en remplaçant la composante usée sans améliorer les caractéristiques de l'équipement.

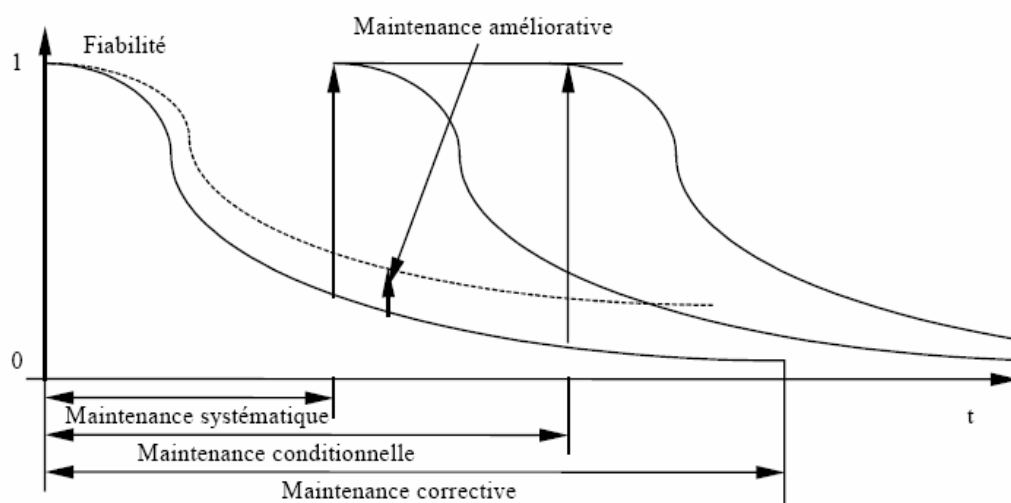


Figure 1.7 : Impact de la maintenance sur la fiabilité des équipements [KAF01]

### 3.2.3. Commentaire

Traditionnellement, on attend qu'un équipement tombe en panne avant de lancer des actions correctives pour le remettre en service. L'une des principales caractéristiques de la maintenance consiste à s'approcher des objectifs «*zéro panne*» et «*zéro défaut*». Pour parvenir au «*zéro*», dans n'importe quel domaine que ce soit, il faut éviter qu'il y ait une première fois. C'est pourquoi une action préventive doit être envisagée. Si on attend qu'un problème surgisse pour le résoudre, c'est trop tard.

En résumé de cette section nous dirons que la maintenance préventive s'adresse aux éléments stratégiques provoquant une perte de production ou de coûts d'arrêts imprévisibles jugés importants pour l'entreprise. L'analyse de ces coûts met en évidence un gain important dû aux arrêts qu'elle permet d'éviter. Elle se traduit par deux types d'actions : des interventions planifiées et des inspections périodiques. La maintenance corrective est utilisée soit seule, soit en complément de la maintenance préventive [MON96]. Elle est utilisée seule pour les équipements peu coûteux et non stratégiques de l'entreprise : leurs « défaillances n'ont pas d'impact significatif sur le fonctionnement du procédé » [ZWI96]. Elle est toujours utilisée en complément car, quel que soit le niveau de maintenance préventive mis en œuvre sur un équipement, des défaillances catalectiques subsisteront toujours, ce qui requiert l'application de la maintenance corrective [MON96, BOU88].

### **3.3. Le système de maintenance**

Après avoir présenté quelques définitions de la maintenance et de ses différents types, nous situons dans ce qui suit la maintenance par rapport au processus de production. Ainsi, nous présentons les actions de maintenance, puis les fonctions et les tâches associées à la maintenance.

#### **3.3.1. Les actions de maintenance**

La maintenance est définie comme un ensemble d'actions techniques et de gestion. Elles sont destinées à maintenir ou à remettre un équipement en état de fonctionnement. Seules, elles peuvent maintenir ou rétablir l'équipement dans des conditions normales d'opération.

Il nous a paru utile de définir quelques actions techniques essentielles qui vont être décrites ci-après, et que l'on retrouve aussi bien en maintenance corrective (C) qu'en maintenance préventive (P) [SAS98, ZWI96].

##### ***- Le diagnostic (C)***

C'est l'identification de la cause probable de la défaillance, qui se base sur un raisonnement logique fondé sur les informations issues du système.

##### ***- Le dépannage (C)***

Le dépannage est une opération de maintenance corrective donnant des résultats provisoires, et qui n'a pas de condition d'applications particulières.

**- La réparation (C)**

C'est une intervention définitive et limitée de maintenance corrective après panne ou défaillance. L'équipement réparé doit assurer les performances pour lesquelles il a été conçu. Elle peut être appliquée sur tous les équipements.

**- Le contrôle (C)**

Après avoir exécuté les opérations de maintenance corrective, il consiste à vérifier le bon fonctionnement du matériel.

**- L'inspection (P)**

Elle consiste à relever périodiquement des anomalies et exécuter des réglages simples ne nécessitant pas d'outillage spécifique, ni d'arrêt de l'outil de production, ou des équipements.

**- La visite (P)**

Ces interventions correspondent à une liste d'opérations définies au préalable qui s'opèrent selon une périodicité déterminée.

**- Le contrôle (P)**

En maintenance préventive, le contrôle consiste à faire des vérifications de conformité par rapport à des données fournies à l'avance et suivies d'un jugement. Le contrôle peut déboucher, comme la visite, sur des opérations de maintenance corrective.

**- La révision (P)**

C'est l'ensemble des actions d'examen, de contrôle et des interventions effectuées en vue d'assurer le bien contre toute défaillance majeure ou critique pendant un temps ou pour un nombre d'usage donné [NFX60-011].

**- Historique (P et C)**

Il consiste à structurer en mémoire toutes les actions réalisées au cours des interventions.

D'autres activités complètent les opérations de maintenance et participent à l'optimisation des coûts d'exploitation. Même si ces activités sortent du cadre direct de la maintenance, certains industriels trouvent des possibilités d'amélioration par l'intermédiaire de ces formes de maintenance. Ces activités sont : la rénovation, la modernisation, la reconstruction et la sécurité.

### 3.3.2. Les fonctions et les tâches associées à la maintenance

Nous identifions trois fonctions associées à la gestion de la maintenance (Fig. 1.8) : la fonction Etudes et méthodes, la fonction Exécution - mise en et la fonction documentation.

Les tâches associées à chacune de ces fonction, bien que différentes dans leurs descriptions, sont complémentaires dans leurs finalités [KAF01].

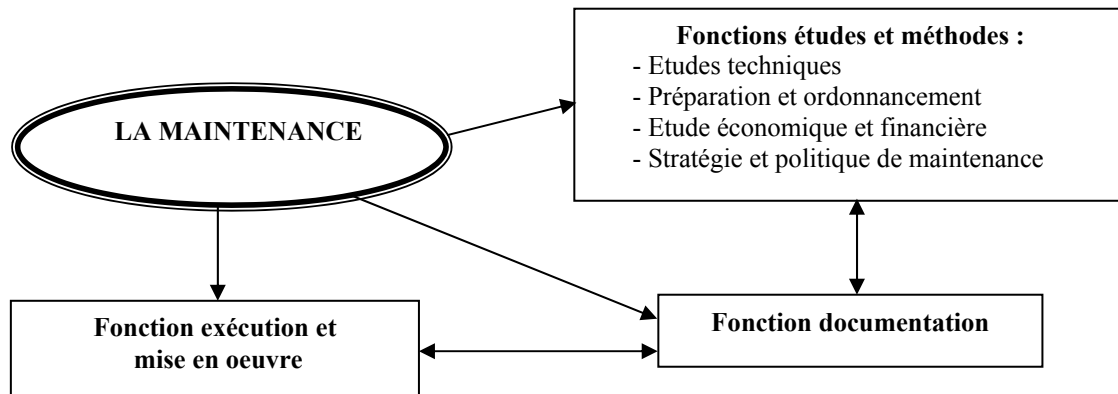


Figure 1.8 : Les fonction et tâches associées à la maintenance.

#### - *Etudes et méthodes*

Elle consiste à optimiser toutes les tâches en fonction des critères retenus dans le cadre de la formulation de la politique de maintenance. Cette partie regroupe quatre tâches principales : l'étude technique, la préparation et l'ordonnancement et l'étude économique et financière.

#### - *Exécution et mise en oeuvre*

Pour la fonction exécution - mise en oeuvre, une expérience considérable sur le matériel des entreprises modernes et une connaissance approfondie des différentes technologies sont nécessaires. Les principales tâches pour remplir cette fonction sont les suivantes :

- installer les machines et le matériel (réception, contrôle, etc.);
- informer le personnel sur la façon d'utiliser les équipements et faire la mise à niveau;
- appliquer les consignes d'hygiène, de sécurité et des conditions de travail ;
- gérer l'ordonnancement et l'intervention de la maintenance et établir le diagnostic de défaillance du matériel;
- coordonner les interventions de la maintenance et remettre en marche le matériel après intervention;
- gérer les ressources matérielles (les pièces de rechange, l'outillage, etc.).



### **- Documentation**

Le troisième type de fonction, à savoir la documentation, est complémentaire aux deux autres.

Ses principales tâches consistent à :

- ❑ établir et mettre à jour l'inventaire du matériel et des installations ;
- ❑ constituer et compléter les dossiers techniques, historiques et économiques ainsi que le dossier des fournisseurs;
- ❑ constituer et compléter une documentation générale (technique, scientifique, d'hygiène et de sécurité).

### **3.3.3. Les niveaux de maintenance**

Une autre condition pour réussir un système de maintenance est de spécifier les niveaux de maintenance dans l'entreprise. Monchy [MON96] et Lyonnais [LYO92] présentent cinq niveaux de maintenance. Dans ce qui suit, nous présentons les tâches associées à chaque niveau.

- Niveau I : Réglages simples prévus par le constructeur au moyen d'organes accessibles sans aucun démontage d'équipement ou d'échange d'éléments accessibles en toute sécurité.
- Niveau II. Dépannage par échange standard d'éléments prévus à cet effet ou d'opérations mineures de maintenance préventive.
- Niveau III. Identification et diagnostic de pannes, réparation par échange de composants fonctionnels, réparations mécanique mineures.
- Niveau IV. Travaux importants de maintenance corrective ou préventive.
- Niveau V. Travaux de rénovation, de reconstruction ou de réparations importantes confiées à un atelier central.

Ces niveaux de maintenance font référence à la complexité des tâches à effectuer et aux ressources humaines et matérielles nécessaires à la réalisation de chacune des tâches. Cette spécification est détaillée dans le tableau 1.1.

Le système de maintenance ainsi situé permet de préciser, de limiter et de dégager les responsabilités et les attentes envers ce système. Cependant, ceci constitue une condition nécessaire mais non suffisante pour réussir l'implantation d'un système de maintenance dans une entreprise.

Niveaux	Personnel d'intervention	Moyens
I	Exploitant sur place.	Outillage léger défini dans les instructions d'utilisation.
II	Technicien habilité sur place.	Outillage léger défini dans les instructions d'utilisation, plus pièces de rechange trouvées a proximité, sans délai.
III	Technicien spécialisé, sur place ou en local de maintenance	Outillage prévu plus appareils de mesure, banc d'essai, de contrôle, etc.
IV	Equipe encadrée par un technicien spécialisé, en atelier central	Outillage général plus spécialisé, matériel d'essai, de contrôle, etc.
V	Equipe complète, polyvalente en atelier central	Moyens proches de la fabrication par le constructeur.

Tableau 1.1: les ressources nécaissaires pour chaque niveau de maintenance.

### 3.4. La Gestion de la Maintenance Assistée par Ordinateur

Depuis quelques années, les entreprises sont confrontées à des variations de marchés très rapides. La concurrence est très vive. Pour résister, la qualité des produits fabriqués doit être sans reproche et au meilleur prix. A cet effet l'organisation de l'entreprise a été optimisée, les effectifs ont été réduits, et l'automatisation a gagné la fonction maintenance pour donner naissance au concept de la Gestion de la Maintenance Assistée par Ordinateur (G.M.A.O.).

La G.M.A.O. est une étape que les entreprises franchissent pour progresser vers plus d'**efficacité**, de **rentabilité** et de **productivité**. Une G.M.A.O. se doit de prendre en charge, d'une part, l'ensemble des données et des méthodes liées à une organisation de maintenance, puis d'en tirer les indicateurs stratégiques nécessaires à la prise des **bonnes décisions**.

D'autre part, elle se doit de répondre aux besoins quotidiens des hommes de la maintenance, c'est à dire permettre de réaliser des interventions sur des équipements avec le maximum d'efficacité. Pour cela, la G.M.A.O. va représenter la mémoire et le savoir faire du service de maintenance. Traditionnellement, cette mémoire et ce savoir-faire représentent une base de travail commune entre les techniciens et les ingénieurs, ce qui engendre de grosses difficultés pendant leurs absences. Bien sûr, dans beaucoup de cas, un historique papier des interventions est néanmoins enregistré dans un rapport journalier, mais l'information est longue à retrouver et le plus souvent ces rapports manquent de précision. La G.M.A.O., avec l'avantage de l'informatique, a permis de gérer la masse importante d'informations, de faciliter la recherche d'informations, de calculer des coûts et des ratios, et de proposer aux gestionnaires un planning de maintenance.

Une G.M.A.O. est constituée de différents modules dont les plus importants sont [GIR00] :

- Le module « **Gestion des équipements** » qui permet la codification et la description des équipements, machines et installations. Ces derniers seront décomposés en plusieurs niveaux d'ensembles et de sous-ensembles. Un dossier technique par équipement doit être établi. Ces dossiers techniques doivent comprendre les plans de l'équipement, l'historique des réparations ainsi que les procédures et gammes de maintenance de ce dernier;
- Le module « **Demande d'intervention** » permet de saisir et de référencer les demandes d'intervention provenant des utilisateurs de matériel extérieurs à la maintenance ou déclenchées par l'occurrence d'un événement planifié dans un module de maintenance préventive (date de visite, relevé de compteur, incident capté par un superviseur).
- Le module « **Gestion des travaux** » qui a pour fonction de traiter les demandes d'intervention qui arrivent dans un portefeuille de travaux à préparer. Il s'agit de réserver les pièces de rechange, les divers matériels et l'outillage, de coordonner et de planifier, d'affecter les personnels selon les compétences requises et d'adapter les plannings aux charges. Ce module permet l'édition d'Ordres de Travail (OT) ou de Bons de Travaux (BT). Ces modules ont la possibilité d'accéder à des outils de gestion de projet pour aider la planification des gros travaux.
- Le module « **Gestion de la maintenance préventive** » permet d'établir un planning de maintenance préventive par ligne de production et par équipement, dont le déclenchement se fera sur la base d'un mode de déclenchement particulier.
- Le module « **Retour d'expérience** » permet d'alimenter les historiques par la liste des causes et des modes de défaillances.
- Le module « **Gestion des stocks et achats** » permet de gérer les entrées/sorties du magasin. Le module doit être capable de faire le lien entre un OT, les pièces de rechange nécessaires et leur disponibilité. Il permet également le réapprovisionnement et les commandes de matériel ainsi que les commandes de service (sous-traitance).
- Le module « **Contrôle des budgets et des coûts** » permet d'établir des prévisions à partir des dépenses prévues pour les équipements ainsi que de connaître les coûts et suivre leur évolution par rapport au budget prévu.

En général, les raisons suivantes reviennent le plus souvent pour justifier la mise en place d'une G.M.A.O. :

- Mettre en place une informatisation de la Gestion de la Maintenance est **structurant** car cela impose une organisation minimale ;
- Le potentiel d'informations à gérer par un service maintenance est considérable et doit être accessible par de multiples clés d'accès. Or, l'informatique répond parfaitement à ce double besoin de **stockage important** et de gestion **d'accès multiples** ;
- Non seulement le volume d'informations est important, mais il est souvent dispersé physiquement dans l'entreprise et n'est pas toujours accessible au bon moment. Or l'informatique permet de mettre aisément à disposition toutes les informations souhaitées (historique, dossiers techniques, disponibilité des pièces de rechange, etc.) et ce par l'intermédiaire d'un ou plusieurs postes de travail fixes ou mobiles ;
- Enfin, la masse d'informations est brute : il est nécessaire d'en dégager quelques indicateurs permettant une **prise de décision** sous forme de compromis entre les pôles humain, économique et technique.

Notre travail est axé sur la résolution des conflits entre la production et la maintenance. La mise en place d'une G.M.A.O ne pourrait, en aucun cas, répondre à nos attentes. Malheureusement, quel que soit le planning proposé et le mode de déclenchement choisi, le planning de production n'est pas pris en compte vu qu'il n'y a aucune concertation entre les gestionnaires respectifs des deux services de production et de maintenance.

### **3.5. Les nouvelles approches de maintenance**

Les nouvelles approches de maintenance ont vu le jour pour aider la personne intervenante à réaliser le processus auquel elle est affectée. Ces méthodes, comme la télémaintenance, la maintenance productive totale ou la maintenance centrée sur la fiabilité sont présentées dans ce qui suit.

#### **3.5.1. La télémaintenance**

La télémaintenance est une forme évoluée de maintenance [KOL93, LAU96]. Elle est basée sur le principe suivant : les capteurs, mesurant des grandeurs intimement liées à l'état de la machine, sont reliés à une centrale de surveillance qui enregistre toutes les alarmes et les mesures [MON96, LAV96, LYO92]. Des tableaux synoptiques visualisent la localisation de

l'information. Cette technique permet d'une part, le suivi et l'enregistrement des données sur chaque machine pour des fins de comparaison et d'autre part, la détection d'aléas de fonctionnement. L'agent de surveillance qui constate une évolution d'une dégradation ou l'apparition d'un défaut, a la responsabilité de mettre hors service, de consigner la partie lésée de l'installation et d'alerter les agents d'intervention [MAL93]. Cette technique voit son application dans les chaînes de production automatisées ou auto-programmables. Avec l'évolution fulgurante de la technologie lors de la dernière décennie, la télémaintenance prend une place de plus en plus grande dans les entreprises manufacturières. Cette technologie permet de faire le contrôle et le suivi de l'évolution de l'état des machines de production à l'interne ou à l'externe.

### **3.5.2. La maintenance productive totale**

Nakajima [NAK87, NAK89] définit la Maintenance Productive Totale (en anglais *Total Productive Maintenance : TPM*) comme une approche où tous les employés participent à la maintenance préventive par des activités d'équipe. C'est la définition qui est adoptée d'emblée dans la littérature sur la TPM Il ajoute que le terme "Total" de TPM a trois significations : le rendement global des installations, un système global de réalisation et une participation de tout le personnel. La TPM vise à modifier la manière de penser des employés vis-à-vis de la maintenance et à améliorer leur niveau de connaissance.

Elle est définie en cinq points clés :

1. le fonctionnement optimal des installations;
2. un système exhaustif de maintenance préventive, incluant la maintenance autonome et la détection des micro-dégradations par un programme de propreté;
3. une approche multidisciplinaire (conception + production + maintenance);
4. l'implication de tous les employés et à tous les niveaux;
5. la réalisation des activités de maintenance préventive par petits groupes autonomes.

L'implantation du concept de la TPM doit s'effectuer progressivement, tel qu'exposé par Nakajima [NAK87]. Il propose une période de deux à trois ans aux membres de l'usine (incluant travailleurs et administrateurs) pour adopter cette philosophie. Il conseille d'essayer ce programme dans le cadre d'un projet-pilote avant de généraliser l'expérience. L'aspect principal à considérer, lors de l'implantation de la TPM, est le facteur humain. Peu de problèmes sont à prévoir du côté technique. Il est donc crucial de bien planifier et gérer le

facteur humain pour garantir la réussite d'un tel changement. Les pré-requis à la TPM semblent être plus du côté culture d'entreprise et du potentiel d'apprentissage des employés, que du côté technique de la maintenance. Les méthodes d'implantation proposées sont semblables mais non génériques.

### **3.5.3. La maintenance basée sur la fiabilité**

La Maintenance Basée sur la Fiabilité (MBF) (en anglais *Reliability Centred Maintenance : RCM*) a vu le jour dans l'industrie aéronautique au cours des années 60 [WHE96]. À la fin des années 50, le coût des activités de maintenance dans cette industrie était devenu exorbitant et a justifié une recherche spéciale sur l'efficacité de ces activités. En conséquence, un groupe de travail des forces armées américaines a pris en charge l'étude de nouvelles alternatives de l'entretien préventif. Ce groupe de travail a développé une série de directives pour les compagnies aériennes. Le premier ensemble a été émis en 1967 et a servi de base, entre autres, au programme d'entretien pour le Boeing 747. D'autres directives ont donné naissance à la maintenance basée sur la fiabilité (RCM), définie par Moubray [MOU97], comme un processus qui détermine les besoins en maintenance du composant dans son contexte opérationnel. Par la suite, la RCM2 a été présentée par Moubray [MOU97]. Elle ne présente pas une variation significative des principes initiaux de la RCM à un niveau théorique bien qu'il ait découvert quelques petites lacunes dans la logique (notamment le manque des principes initiaux de la RCM dans la considération explicite de l'impact potentiel d'une panne sur l'environnement, et la logique concernant le traitement pertinent des pannes cachées). La contribution de Moubray se situe dans les points suivants :

1. les questions de RCM ont été raffinées pour améliorer la clarté et la convivialité. Ceci permet d'appliquer plus facilement les principes de la RCM;
  2. le processus intègre une approche de gestion du changement pour le développement et la mise en place de nouvelles stratégies d'entretien du matériel;
  3. enfin il est bien davantage qu'un ensemble de principes d'ingénierie; il est conçu pour renforcer et mettre en valeur les qualifications des hommes de maintenance et des opérateurs.
- Le processus contient une partie formation et une partie mise en place.

## 4. Conclusion

La maintenance industrielle joue actuellement un rôle déterminant dans la conduite de la production. En effet, la recherche de l'accroissement des performances des ateliers de production, de plus en plus variés et complexes, conduit à transférer sur la fonction maintenance la responsabilité de garantir la disponibilité de ces systèmes. L'objectif de la maintenance devient alors l'identification réactive de l'élément défaillant (maintenance corrective) mais aussi la prévision des pannes afin de réduire la durée moyenne d'indisponibilité du système (maintenance préventive).

Dans ce chapitre, nous avons présenté la problématique générale de notre travail, qui concerne les systèmes de production manufacturiers et leur maintenance. Elle constitue une phase de réflexion et d'étude bibliographique. Dans la première partie, nous avons exposé le contexte général des systèmes de production et de la gestion de production, en insistant particulièrement sur les aspects typologie et évaluation des systèmes de production. La maintenance étant une fonction complexe, nous avons présenté dans la seconde partie les définitions des différentes formes de maintenance, l'architecture de cette fonction et sa position par rapport aux autres fonctions de l'entreprise. L'objectif premier de chapitre est la mise en place des deux concepts de base de cette thèse : la production et la maintenance.

Les études actuelles tendent à développer ces deux aspects de manière séparée, du fait de leurs objectifs antagonistes. En effet, une productivité accrue implique un système de production qui tourne à plein régime. Par contre, assurer la sûreté de fonctionnement du même système impliquerait des arrêts systématiques pour maintenir les machines, et ainsi pallier à d'éventuels arrêts. Il serait intéressant de faire converger les objectifs de productivité et de sûreté de fonctionnement des gestionnaires. Ce sont ces aspects que nous présenterons dans le chapitre suivant. De manière plus explicite, nous présenterons la problématique de l'ordonnancement liée aux activités de maintenance et de production de manière séparée, puis nous justifions la nécessité de mettre en place des relations de coopération, à ce niveau, par une résolution conjointe du problème de l'ordonnancement des activités de maintenance et de production, pour pallier aux conflits production/maintenance.





---

# CHAPITRE II

## *Ordonnancement des Activités de Production et de Maintenance*

---



---

## **CHAPITRE II.**

# **Ordonnancement des Activités de Production et de Maintenance**

---

**Résumé :** *Le caractère multi-objectif de l'ordonnancement est reconnu depuis longtemps, mais il est rarement admis que d'autres fonctions que la production devraient y participer. En fait, il est en pratique difficile de tenir compte des objectifs des autres fonctions dans le cadre des méthodes d'ordonnancement les plus classiques.*

*Les activités de production et de maintenance apparaissent à priori comme totalement antagoniste et par la même incompatibles puisqu'elles agissent sur les mêmes ressources. Ces deux activités ne peuvent donc pas être accomplies au même moment sur un même système, ce qui laisse augurer bien des conflits dans l'utilisation du système par l'un ou l'autre des services liés à la maintenance ou à la production. En effet, si la production a pour vocation d'utiliser un système dans le but de remplir un objectif de production, traduit en termes de temps et de nombre de tâches à accomplir. La maintenance quant à elle est une activité nécessitant le plus souvent l'arrêt du système pour atteindre son objectif de maintien de ce dernier en état de fonctionnement normal. Dans ce chapitre, nous présentons quelques notions de base concernant les problèmes d'ordonnancement dans les ateliers de production. La seconde partie est consacrée à l'ordonnancement de la maintenance, sa spécificité par rapport à celui de la production, ainsi que les objectifs devant être satisfaits. Nous présenterons par la suite le problème de l'ordonnancement conjoint production/maintenance. Nous y définissons ce qu'est un ordonnancement conjoint de la maintenance et de la production, ses stratégies ainsi que les enjeux d'un tel ordonnancement.*

*Enfin, nous définirons le contexte de notre étude, en termes d'orientations futures à donner à cette thèse, et ceci après un état de l'art où nous recensons les travaux qui se rapprochent le plus de notre travail, l'objectif étant de montrer l'intérêt de développer des ordonnancements conjoints production/maintenance.*

---

## 1. Introduction

L'ordonnancement est un champ d'investigation qui a connu un essor important ces quarante dernières années [BLA96, BRU98, LOP01], tant par les nombreux problèmes identifiés que par l'utilisation et le développement de nombreuses techniques de résolution.

Face à cette volonté de planifier la production, apparaît depuis peu une volonté d'améliorer l'ordonnancement de la maintenance. Cette activité, autrefois un peu délaissée au profit de la production, s'annonce aujourd'hui comme une voie d'amélioration de la productivité et de la rentabilité du système de production.

Dans un contexte de juste à temps, les arrêts accidentels du système de production sont considérés comme une source de perturbation et de perte de productivité. La disponibilité des équipements, au moment voulu, est une condition nécessaire au bon déroulement de la production et au respect des délais de livraison. La maintenance préventive est un moyen efficace pour maintenir un niveau de disponibilité acceptable. Cependant, le déploiement des activités de maintenance nécessite l'arrêt total des équipements de production, et par voie de fait la perturbation de l'ordonnancement en cours. Pour minimiser l'impact de ces arrêts sur la production, les activités de maintenance doivent être intégrées à la gestion de production. Ainsi, on est amené à considérer, simultanément, les contraintes des activités de maintenance et les exigences de gestion de production.

Malheureusement, que ce soit pour le plan de production ou le plan de maintenance, ni la maintenance ni la production n'est prise en compte, vu qu'il n'y a aucune concertation entre les gestionnaires respectifs des deux services.

Les travaux de cette thèse ont pour but de démontrer la nécessité de développer des algorithmes d'ordonnancement conjoint production/maintenance pour atteindre l'objectif d'optimisation de la sûreté de fonctionnement du système de production et ceci grâce à une bonne maintenance et par voie de fait celui de la maximisation de la rentabilité.

De même que le chapitre précédent, ce chapitre se présentera en deux parties : nous présenterons, dans la première, les problèmes d'ordonnancement de production, puis nous introduirons l'ordonnancement des activités de maintenance.

Nous commencerons par présenter de manière succincte les problèmes d'ordonnancement de la production auxquels nous nous intéressons dans cette thèse. L'objectif étant de présenter les notions fondamentales concernant les problèmes d'ordonnancement. Pour cela, il convient de définir ce qu'est un problème d'ordonnancement, d'exposer une typologie qui sera utilisée

dans la suite du document et de présenter quelques méthodes classiques de résolution de ces types de problème.

La seconde partie est consacrée à l'ordonnancement des activités de maintenance, sa spécificité par rapport à celui de la production, largement étudié dans la littérature, ainsi que les objectifs devant être satisfaits. Nous présenterons par la suite le problème de l'ordonnancement conjoint production/maintenance. L'origine du conflit étant les ressources de production nécessaires aux deux fonctions pour s'exécuter. Nous y définissons ce qu'est un ordonnancement conjoint de la maintenance et de la production, ses stratégies ainsi que les enjeux d'un tel ordonnancement.

Enfin, nous définirons les problématiques étudiées dans cette thèse, en terme de typologie d'atelier, de maintenance et de stratégie. L'objectif de cette démarche est de montrer l'intérêt de développer des ordonnancements conjoints production/maintenance.

## 2. Introduction aux problèmes d'ordonnancement de production

Bien que les problèmes d'ordonnancement de tâche soient inhérents à toute organisation, leur résolution est restée empirique jusqu'à 1958 où les méthodes PERT et la méthode des potentiels ont résolu les problèmes à contraintes de succession et de localisation temporelle [BAV02].

*« Le problème d'ordonnancement consiste à organiser dans le temps la réalisation d'un ensemble de tâches, compte tenu de contraintes temporelles (délais, contraintes d'enchaînement, etc.) et de contraintes portant sur l'utilisation et la disponibilité des ressources requises par les tâches »* [LOP01].

Dans les problèmes d'ordonnancement d'ateliers deux notions fondamentales sont utilisées: les **tâches** et les **machines** ou **ressources**. Le but est de *« programmer l'exécution des tâches en leur allouant les ressources requises et en fixant leur date de début »* [GHO93].

Un ordonnancement décrit l'exécution des tâches et l'allocation des ressources au cours du temps, et vise à satisfaire un ou plusieurs objectifs. Plus précisément, on parle de problème d'ordonnancement lorsqu'on doit déterminer les dates de début et les dates de fin des tâches, alors qu'on réserve le terme de problème de séquençement au cas où l'on chercherait seulement à fixer un ordre relatif entre les tâches qui peuvent être en conflit pour l'utilisation des ressources. Un ordonnancement induit nécessairement un ensemble unique de relations de séquençement. En revanche, la solution purement séquentielle d'un problème

d'ordonnancement recouvre une famille d'ordonnancements (éventuellement une infinité si le domaine de variation des dates est non borné ou réel).

D'autres définitions de l'ordonnancement existent dans la littérature, telles les définitions de Baker en 1974 [BAK74], de Carlier et Chrétienne en 1988 [CAR88], de Blazewicz & al en 1996 [BLA96] et de Brucker en 1998 [BRU98].

## 2.1. Modélisation des problèmes d'ordonnancement de production

La modélisation d'un problème d'ordonnancement passe par la spécification des tâches et leurs caractéristiques, des ressources et des critères à optimiser. Toutes ces données peuvent être représentées par la notation à trois champs  $\alpha | \beta | \gamma$  introduite par Graham & al en 1979 [GRA79], et reprise par Blazewicz & al en 1996 [BLA96]. Quelques extensions ont été introduites, dont celles données par Vignier & al en 1999 [VIG99] pour les problèmes de *flowshop* hybride et par T'Kindt en 1999 [TKI99] pour les problèmes d'ordonnancement multicritères.

Le champ  $\alpha$  décrit l'environnement machine, c'est-à-dire, la structure de l'atelier. Le champ  $\beta$  représente les contraintes du problème liées aux tâches. Le champ  $\gamma$  décrit le ou les objectifs sous forme de critères à optimiser.

Nous allons dans ce qui suit définir les quatre notions fondamentales qui interviennent dans un problème d'ordonnancement : les tâches (Travaux ou *Jobs* en anglais), les ressources, les objectifs et les contraintes. Une tâche se définit comme un ensemble d'opérations qui doivent être exécutées. Une ressource est un moyen matériel ou humain à disposition pour la réalisation d'un travail. Les contraintes représentent les limites imposées par l'environnement, tandis que l'objectif est le critère d'optimisation.

### 2.1.1. Les tâches

Une tâche est le processus le plus élémentaire. Elle est constituée d'un ensemble d'actions à accomplir, dans des conditions fixées, pour obtenir un résultat attendu et identifié, en termes de performances, de coûts et de délais (norme AFNOR NF X50-310).

Dans certains problèmes, les tâches peuvent être exécutées par morceaux. L'entrelacement des différents morceaux permettant de laisser, le moins possible, les ressources inactives. Dans d'autres cas, au contraire, on ne peut pas interrompre une tâche une fois commencée. On parle alors respectivement de problèmes **préemptifs** et **non préemptifs**.

Une tâche  $t_{ij}$  est localisée dans le temps par:

- $p_{ij}$ : La durée opératoire de la tâche  $i$  sur la machine  $j$ ;
- $r_{ij}$ : La date de début au plus tôt ou date de disponibilité de la tâche  $i$  sur la machine  $j$  ;
- $d_{ij}$ : La date de fin au plus tard ou «*deadline*» de la tâche  $i$  sur la machine  $j$ ;
- $t_{ij}$ : La date de début d'exécution de la tâche  $i$  sur la machine  $j$  ;
- $c_{ij}$ : La date de fin d'exécution de la tâche  $i$  sur la machine  $j$  ;
- $T_{ij}$ : Le retard vrai de la tâche  $i$  sur la machine  $j$  /  $T_{ij} = \max(0, c_{ij} - d_{ij})$ ;
- $U_{ij}$ : L'indice de retard Tel que  $U_{ij} = 1$  si la tâche est en retard, 0 sinon;
- $g_{ij}(u)$ : Le coût attaché à la fin d'exécution de la tâche  $i$  sur la machine  $j$ , à la date  $u$ .

La figure 2.1 illustre les caractéristiques temporelles présentées ci-dessus :

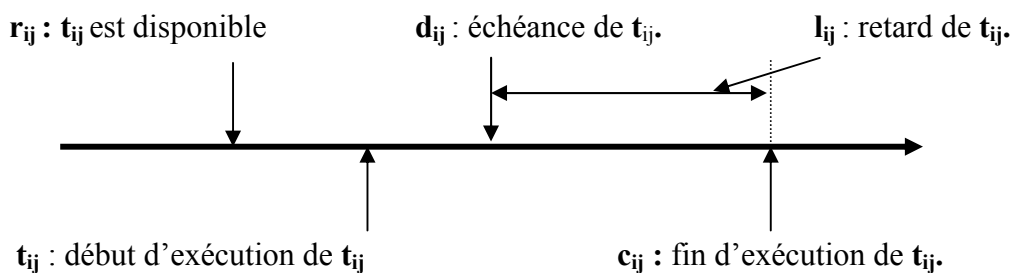


Figure 2.1: Caractéristique d'une tâche.

### 2.1.2. Les Ressources

Une ressource  $k$  est un moyen technique ou humain requis pour la réalisation d'une tâche. Elle est disponible en quantité limitée, et sa capacité est supposée constante.

Plusieurs types de ressources sont à distinguer. Une ressource est **renouvelable** si après avoir été utilisée par une ou plusieurs tâches, elle est à nouveau disponible en même quantité (les hommes, les machines, l'espace, l'équipement en général, etc.). La quantité de ressources utilisable à chaque instant est limitée. Dans le cas contraire, elle est **consommable** (matières premières, budget, etc.). La consommation globale (ou cumul) au cours du temps est limitée.

Par ailleurs, on distingue les ressources **disjonctives** (ou non partageables) qui ne sont allouées qu'à une tâche à la fois (machine-outil, robot manipulateur) et les ressources **cumulatives** (ou partageables) qui peuvent être utilisées par plusieurs tâches simultanément (équipe d'ouvriers, poste de travail).

### 2.1.3. Les critères d'optimisation

Lors de la résolution d'un problème d'ordonnancement, on peut distinguer entre deux types de stratégies, visant respectivement à l'optimalité des solutions par rapport à un ou plusieurs critères, ou à leur admissibilité vis-à-vis des contraintes. L'approche par optimisation suppose que les solutions candidates à un problème puissent être ordonnées de manière rationnelle selon un ou plusieurs critères d'évaluation numérique permettant d'apprécier la qualité des solutions. On cherchera à minimiser ou maximiser de tels critères. Les critères les plus utilisés sont donnés en le tableau 2.1.

Critères	Maximum	Somme	Somme pondérée
Fin de traitement	$C_{max} = \max_{i=1,n}(C_i)$	$\bar{C} = \frac{1}{n} \sum_{i=1}^n C_i$	$\bar{C}_w = \sum_{i=1}^n w_i C_i$
Décalage	$L_{max} = \max_{i=1,n} L_i$	$\bar{L} = \frac{1}{n} \sum_{i=1}^n L_i$	$\bar{L}_w = \sum_{i=1}^n w_i L_i$
Retard	$T_{max} = \max_{i=1,n} T_i$	$\bar{T} = \frac{1}{n} \sum_{i=1}^n T_i$	$\bar{T}_w = \sum_{i=1}^n w_i T_i$
Avance	$E_{max} = \max_{i=1,n} E_i$	$\bar{E} = \frac{1}{n} \sum_{i=1}^n E_i$	$\bar{E}_w = \sum_{i=1}^n w_i E_i$
Durée de flot	$F_{max} = \max_{i=1,n} F_i$	$\bar{F} = \frac{1}{n} \sum_{i=1}^n F_i$	$\bar{F}_w = \sum_{i=1}^n w_i F_i$
Nombre de retards		$\bar{U} = \frac{1}{n} \sum_{i=1}^n U_i$	$\bar{U}_w = \sum_{i=1}^n w_i U_i$

Tableau 2.1: Critères d'optimisation

Une caractéristique intéressante pour un critère est sa régularité. Un critère est dit *régulier* si pour deux ordonnancements différents  $O_1$  et  $O_2$  : si le fait que chaque tâche de  $O_1$  se termine au plus tard ou en même temps que la tâche correspondante dans  $O_2$ , entraîne que la valeur du critère d'optimisation pour  $O_1$ , est inférieur ou égale à la valeur du critère pour  $O_2$ .

Nous terminerons ce paragraphe par la définition de trois classes d'ordonnancement, qui sont les ordonnancements **actifs**, **semi-actifs** et **sans délai** [BAK74]:

- un ordonnancement est dit **semi-actif** (ou calé à gauche) si aucune tâche ne peut être exécutée plus tôt sans changer l'ordre d'exécution sur les ressources ou violer de contrainte ;
- un ordonnancement est dit **actif** si aucune tâche ne peut être exécutée plus tôt sans violer de contrainte ou retarder une autre tâche ;
- un ordonnancement est dit **sans délai** si à tout instant  $t$ , les ressources ne sont pas présentes en quantité suffisante pour pouvoir commencer une tâche disponible et exécutée plus tard dans l'ordonnancement.



Il a été démontré que pour tout critère régulier, les ensembles des ordonnancements semi-actifs et actifs sont **dominants**, c'est à dire qu'ils contiennent au moins une solution optimale [CHU95].

Dans la figure 2.2, on considère le problème d'ordonnement à une machine de deux tâches  $T_1$  et  $T_2$  avec  $p_1=2$ ,  $p_2=1$ ,  $r_1=1$  et  $r_2=0$ . L'ordonnement représenté dans la figure 2.2.a est semi-actif mais n'est ni actif ni sans délai alors que l'ordonnement représenté dans la figure 2.2.b est à la fois semi-actif, actif et sans délai.

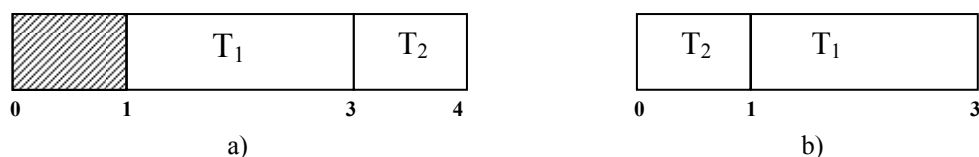


Figure 2.2: Exemples d'ordonnements semi-actifs et sans délai

Dans la figure 2.3, on considère le problème d'ordonnement à machines parallèles ( $m=2$ ) de trois tâches  $T_1$ ,  $T_2$  et  $T_3$  avec  $p_1=2$ ,  $p_2=2$ ,  $p_3=4$ ,  $r_1=1$  et  $r_2=r_3=0$ . L'ordonnement représenté dans la figure 2.3 est à la fois semi-actif et actif mais n'est pas sans délai.

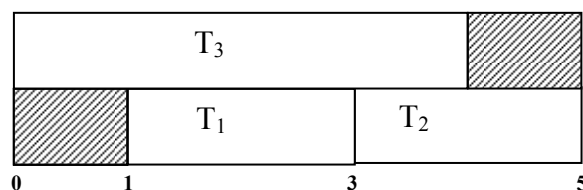


Figure 2.3: Exemple d'ordonnement actif qui n'est pas sans délai

#### 2.1.4. Les contraintes

Une **contrainte** exprime des restrictions sur les valeurs que peuvent prendre conjointement les variables représentant les relations reliant les tâches et les ressources. On distingue les contraintes temporelles et les contraintes de ressources.

Les contraintes temporelles intègrent:

- les contraintes de temps alloué, issues généralement d'impératifs de gestion et relatives aux dates limites des tâches (délais de livraison, disponibilité des approvisionnements) ou à la durée totale d'un projet ;
- les contraintes d'antériorité et plus généralement les contraintes de cohérence technologique, qui décrivent le positionnement relatif de certaines tâches par rapport à d'autres ;
- les contraintes de calendrier liées au respect d'horaires de travail, etc.

Les contraintes de ressources traduisent le fait que les ressources sont disponibles en quantité limitée. On distingue deux types de contraintes de ressources, liées à la nature disjonctive ou cumulative des ressources. Une ressource disjonctive ne peut être utilisée que par une tâche à la fois. Par contre dans une ressource cumulative les ensembles de tâches non réalisables simultanément sont de cardinalité quelconque.

Les problèmes d'ordonnancement sont très différents d'un atelier à l'autre, et il n'existe pas de méthode universelle permettant de résoudre efficacement tous les cas. Dans ce qui suit, nous décrivons les problèmes classiques d'atelier.

## 2.2. Les environnements machines

La classification des environnements machines dépend non seulement du type de machines rencontrées mais aussi des contraintes de type gamme<sup>2</sup> des travaux, qui dépend de la nature de l'atelier. Un atelier se définit par le nombre de machines qu'il contient et par son type. Dans cette section, nous nous limitons à la présentation des environnements machines les plus courants. Néanmoins nous présenterons plus en détails les problèmes d'atelier, vu que nous nous y intéressons dans cette thèse. Les différents types possibles sont les suivants [ESP98]:

### *- Les problèmes à une machine*

Dans ce type de problème, une seule ressource disjonctive (qui ne peut exécuter qu'une opération à un instant donné) est disponible pour exécuter les jobs (ou tâches puisque chaque job est composé d'une seule opération). Ce type de problème consiste à déterminer sur la machine, une séquence optimale de  $n$  tâches disponibles à l'instant 0 [BAK74].

### *- Les problèmes à machines parallèles*

Ces problèmes consistent à ordonnancer un ensemble de tâches sur des machines disjonctives en parallèle, une tâche nécessitant une seule machine pour son exécution. Un double problème doit être résolu : affecter les tâches aux machines et séquencer les tâches affectées à une même machine. Cette classe de problèmes peut être encore divisée en trois sous classes.

(i) *Machines identiques*. Il s'agit de problème à machines parallèles où toutes les opérations demandent le même temps sur toutes les machines.

(ii) *Machines uniformes*. Dans ce cas, les machines parallèles ont des vitesses différentes, mais ces vitesses sont constantes (elles ne dépendent pas de l'opération à réaliser). Si la

---

<sup>2</sup> La gamme de fabrication donne l'enchaînement logique des opérations

durée de référence d'une opération est  $P_i$  alors le temps demandé sur la machine  $j$  est  $P_i/C_j$  où  $C_j$  est une constante associée à la  $J^{\text{ième}}$  machine.

(iii) *Machines indépendante*. Ici la vitesse de chaque machine dépend de l'opération que l'on veut réaliser.

#### - Les problèmes d'atelier multi-machines.

Dans les problèmes d'ordonnement d'atelier, les ressources sont des machines ne pouvant réaliser qu'une tâche (ou opération) à la fois. D'autre part, chaque travail concerne une entité physique indivisible, appelée produit, ou lot lorsque plusieurs produits identiques sont regroupés. Une entité ne pouvant se trouver simultanément en deux lieux distincts, un même travail ne peut être exécuté qu'à raison d'une opération à la fois, sur une seule des machines.

Nous considéreront les problèmes d'ateliers multi-machines où les travaux comportent plusieurs opérations, selon que l'ordre des opérations composant un même travail est fixé et commun à tous les travaux (atelier à cheminement unique ou *flowshop*), qu'il soit fixé mais propre à chaque travail (atelier à cheminements multiples ou *jobshop*), ou enfin qu'il soit indéterminé (atelier à cheminements libres ou *openshop*).

(i) *Atelier à flot continu (flowshop)*. Dans ce type d'atelier, on dispose de  $n$  pièces qui doivent s'exécuter suivant le même ordre sur les  $m$  machines qui composent l'atelier. Ces jobs ont donc tous le même ordre de passage sur les machines. Si tous les temps d'exécution sont non nuls, l'atelier est de type *flowshop pur* (Fig. 2.4).

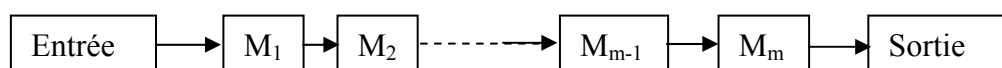


Figure 2.4 : Atelier de type *flowshop* pur.

(ii) *Atelier à tâches (jobshop)*. Dans un atelier *jobshop*, aucune séquence d'opérations n'est fixe, et une pièce peut circuler plusieurs fois sur le même processeur pour des tâches différentes. Les machines ne sont pas ordonnées et un nombre important de routages est possible. Le routage d'un job peut démarrer et se terminer sur n'importe quelle machine de l'atelier.

Le problème du *jobshop* se différencie du *flowshop* par un élément essentiel, le cheminement des pièces n'est pas unidirectionnel comme il l'est pour une chaîne de montage, par exemple. Cette différence se traduit également par un supplément de données

nécessaires. En effet, en plus des temps d'exécution sur les diverses machines, il faut connaître l'ordre dans lequel ces opérations doivent être effectuées.

(iii) *Atelier d'assemblage (assembly workshop)*. Une machine  $M_k$  peut effectuer un assemblage de deux pièces différentes, l'une provenant d'une machine  $M_i$  et l'autre d'une machine  $M_j$ . D'une manière générale, la symétrie entre les produits assemblés doit être respectée. De la même façon, un atelier de désassemblage est également introduit.

## 2.3. Les méthodes de résolution

Les méthodes de résolution des problèmes d'ordonnancement puisent dans toutes les techniques de l'optimisation combinatoire. Traditionnellement, le domaine de l'optimisation combinatoire a longtemps privilégié les méthodes exactes au détriment des méthodes approchées. Le recours à des modèles exacts permet de bénéficier de résultats théoriques forts accompagnant les notions de convergence et d'optimalité globale. De ce fait, plusieurs problèmes classiques (de taille raisonnable) peuvent être résolus de manière exacte, en un laps de temps raisonnable. Les méthodes exactes consistent généralement à énumérer l'ensemble des solutions de l'espace de recherche. Pour améliorer l'énumération des solutions, elles disposent de techniques pour détecter le plus tôt possible les échecs (calcul de bornes) et d'heuristiques spécifiques pour orienter les différents choix. Cependant le temps de calcul de ces méthodes est exponentiel ce qui explique qu'elles ne sont utilisables que sur des problèmes de petite taille.

Depuis la fin des années 80, les méthodes approchées, et plus particulièrement les métaheuristiques, suscitent un intérêt croissant pour leur capacité à fournir des solutions d'excellente qualité pour une consommation en temps de calcul réduite. La perte du caractère optimal se voit compensée par la diminution des temps de calcul et donc par un accroissement de la capacité de réaction. Les métaheuristiques bénéficient également d'autres avantages significatifs tels que leur rapidité d'adaptation à des modifications structurelles du problème.

Nous présenterons donc, cette section en deux parties où nous introduirons en premier lieu quelques méthodes exactes. Puis nous présenterons les métaheuristiques par approche à savoir : l'approche constructive, itérative ou par voisinage et enfin évolutive.

### 2.3.1. Les méthodes exactes

Ces méthodes sont généralement utilisées pour résoudre des problèmes de petite taille où le nombre de combinaisons est suffisamment faible pour pouvoir explorer l'espace des

solutions en un temps raisonnable. Ces méthodes assurent toujours l'optimalité de la solution trouvée. Elles garantissent donc la complétude de la résolution mais à coût élevé. Leur inconvénient majeur est le temps exponentiel de résolution qui croît en fonction de la taille du problème.

L'espace de recherche est représenté sous forme d'un arbre où chaque nœud correspond à l'affectation d'une valeur à une variable. Un chemin menant de la racine de l'arbre à une feuille représente une solution. En général et pour éviter justement d'énumérer toutes les solutions de l'espace de recherche et générer ainsi l'arbre complet, des mécanismes sont introduits (calcul de bornes) pour élaguer les chemins qui n'aboutissent pas.

Comme méthodes exactes appliquées au problème de l'ordonnancement de production, nous citerons la programmation mathématique, linéaire et dynamique ainsi que la recherche arborescente ou Procédure par Séparation et Évaluation (PSE) [CHA96, ESQ99].

Nous détaillons dans ce qui suit la Procédure par Séparation et Évaluation.

Le principe d'une PSE est de décomposer le problème en sous problèmes. Cela revient à construire un arbre où les sommets représentent les problèmes et les arcs sortants d'un sommet représentent sa décomposition en sous problèmes. L'objectif est d'arriver à un problème qui se résolve facilement en générant le moins de sommets possible. Elle repose sur quatre points :

- la séparation fait une partition (décomposition du problème en sous problèmes) de telle sorte que toute solution réalisable se trouve au moins dans une partie ;
- l'évaluation associe une borne de la valeur de la fonction objectif (critère d'optimisation) pour les solutions d'un sous problème ;
- le sondage permet de déterminer si un sommet doit être séparé ou non, en particulier si son évaluation montre qu'il ne donnera pas une solution meilleure que celles déjà trouvées alors il n'est pas nécessaire de le séparer ;
- la sélection (ou stratégie d'exploration) permet de choisir dans quel ordre séparer les sommets, en se basant sur l'évaluation faite ou en suivant une règle donnée.

### **2.3.2. Les métaheuristiques**

Les métaheuristiques forment une famille d'[algorithmes](#) d'[optimisation](#) visant à résoudre des problèmes d'[optimisation difficile](#) issus de la [recherche opérationnelle](#) pour lesquels on ne

connaît pas de méthode classique plus efficace. Ces méthodes sont suffisamment générales pour être appliquées à plusieurs catégories de problèmes d'optimisation combinatoire.

Cette section se concentre sur la description des trois grandes classes de métaheuristiques : les méthodes **constructives**, celles dites de **recherche locale ou itérative** et celles considérées comme **évolutives** [COS95].

Toutes les métaheuristiques s'appuient sur un équilibre entre l'intensification de la recherche et la diversification de celle-ci. D'un côté, l'intensification permet de rechercher des solutions de plus grande qualité en s'appuyant sur les solutions déjà trouvées et de l'autre, la diversification met en place des stratégies qui permettent d'explorer un plus grand espace de solutions et d'échapper à des minima locaux. Ne pas préserver cet équilibre conduit à une convergence trop rapide vers des minima locaux (manque de diversification) ou à une exploration trop longue (manque d'intensification).

### 2.3.2.1 L'approche constructive

L'objectif de ces méthodes est de construire progressivement un ordonnancement, solution du problème, en prenant à chaque étape une décision pour le placement d'une nouvelle tâche. L'idée consiste à diminuer la taille du problème à chaque étape, ce qui revient à restreindre l'espace des solutions admissibles  $X$  à un sous-ensemble de solution  $X^k \in X$  toujours plus petit. Une méthode constructive trouve une solution optimale lorsque chacun des sous-ensembles considérés contient au moins une solution optimale  $s^* \in X$ . Malheureusement, rares sont les cas où une telle condition est remplie avec certitude. La majorité des méthodes constructives sont de type glouton. Le type de recherche qui est à la base d'une méthode constructive est représenté dans la figure 2.5.

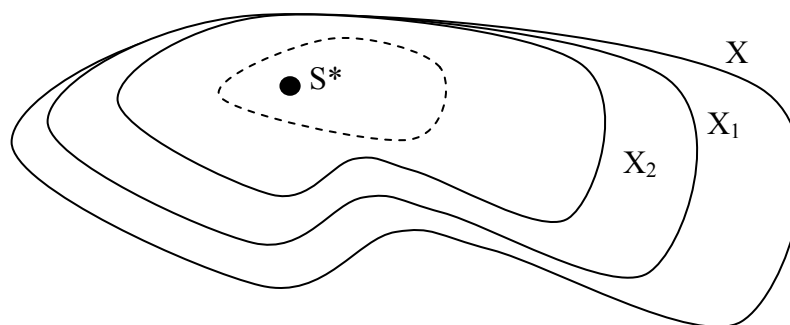


Figure 2.5: Exploration de  $X$  par une approche constructive

Les méthodes constructives se distinguent par leur rapidité et leur grande simplicité. En effet, on obtient très rapidement une solution admissible pour un problème donné sans avoir recours à des techniques hautement sophistiquées. L'inconvénient majeur de ces méthodes réside dans

la qualité des solutions obtenues, cela est dû au fait de vouloir opter à tout prix pour le meilleur choix possible à chaque étape.

Plusieurs heuristiques constructives sont dédiées au *flowshop* telles Palmer, GUPTA, CDS, RA et NEH. Ces heuristiques sont détaillées en annexe A.

### 2.3.2.2 L'approche itérative ou par voisinage

Les méthodes de recherche par voisinage sont des algorithmes itératifs qui explorent l'espace des solutions  $X$  en se déplaçant pas à pas d'une solution à une autre. Une méthode de ce type débute à partir d'une solution  $s_0 \in X$  choisie arbitrairement ou obtenue par le biais d'une méthode constructive. Le passage d'une solution admissible à une autre se fait sur la base d'un ensemble de modifications élémentaires qu'il s'agit de définir de cas en cas. Le voisinage  $N(s)$  d'une solution  $s \in X$  est défini comme l'ensemble des solutions admissibles atteignables depuis  $s$  en effectuant des modifications élémentaires. Les passages successifs d'une solution à une solution voisine définissent un chemin au travers de l'espace des solutions admissibles. Le processus d'exploration est interrompu lorsque un ou plusieurs critères d'arrêt sont satisfaits. Le fonctionnement d'une méthode de recherche basée sur le voisinage est illustré par la figure 2.6.

Les méthodes itératives ou de recherche locale diffèrent essentiellement par le système de voisinage utilisé et la stratégie de parcours du système de voisinage. Parmi les méthodes qui ont prouvé leur efficacité dans la résolution de problème d'optimisation combinatoire, nous pouvons citer les méthodes ascendante et descendante [PAP76, PAP82], le recuit simulé [KIR83] et la recherche tabou [GLO86] que nous détaillons dans ce qui suit.

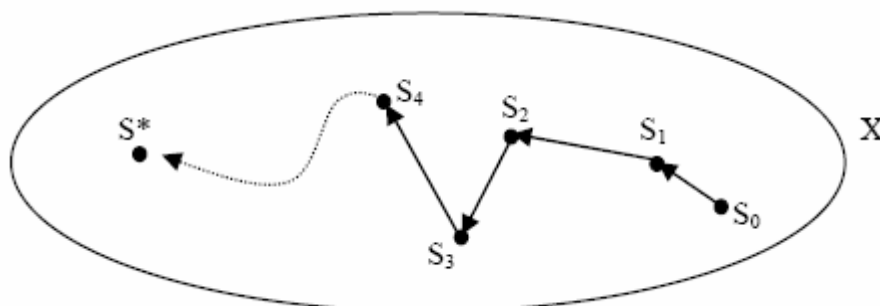


Figure 2.6 : Exploration de  $X$  par une approche séquentielle.

La recherche Tabou est une procédure itérative introduite par GLOVER en 1986 [GLO86, GLO89]. Les idées de bases de la recherche Tabou se retrouvent également dans le travail de

Hansen [HAN86]. Elle n'a aucune caractéristique stochastique et utilise la notion de mémoire pour éviter de tomber dans un optimum local.

L'idée consiste dans le déplacement d'une solution courante  $s$  vers une solution voisine  $s'$  de telle sorte que :  $f(s') = \min_{s'' \in N(s)} f(s'')$ ,  $f$  étant la fonction objectif qui permet d'évaluer les solutions. Lorsque l'on atteint un optimum local, la règle de déplacement donnée ci-dessus permet de choisir le moins mauvais des voisins et qui donne un accroissement aussi faible que possible de la fonction objectif.

L'inconvénient est que si un minimum local se trouve au fond d'une vallée profonde, il sera impossible de ressortir de celle-ci en une seule itération, et un déplacement de la solution  $s$  vers une solution  $s' \in N(s)$  avec  $f(s') > f(s)$  peut provoquer le déplacement inverse à l'itération suivante, et on risque ainsi de «cycler» autour de ce minimum local. C'est pour cette raison que la méthode Tabou s'appuie sur un deuxième principe qui consiste à garder en mémoire les dernières solutions visitées et à interdire le retour vers celles-ci pour un nombre fixé d'itérations. Le but de cette démarche est de donner assez de temps à l'algorithme pour lui permettre de sortir d'un minimum local.

Le processus itératif de la recherche Tabou est décrit en l'algorithme 2.1.

---

#### **Algorithme** Méthode générique de Tabou

---

**Procédure**  $\varphi$  : fonction de coût ;

**Variables locales** :  $S$  solution courante, liste tabou  $L$ , Meilleure solution  $M$ , itération courante  $K$ , nombre d'itération  $N$  ;

**Paramétrages** : Taille de la liste Tabou, Critère d'aspiration ; Solution initiale  $S_0$  ;

**Début**

Solution courante  $S \leftarrow S_0$  ;

Meilleure solution  $M \leftarrow S$  ;

$K \leftarrow 0$  ;

**Tant que**  $K < N$

**Faire**  $K \leftarrow K + 1$

Mise à jour de  $L$

Génération des candidats  $E$  par opération de voisinage

$C \leftarrow \text{meilleur}(E)$

**Si**  $(\varphi(S) < \varphi(M))$  ou  $C$  n'est pas tabou ou  $C$  vérifie l'aspiration

**Alors**  $S \leftarrow C$

**Sinon**  $E \leftarrow E \setminus C$

**Fin si**

**Fin Tant que**

**Fin.**

---

Algorithme 2.1: Méthode générique de la recherche Tabou [GLO89].

Plusieurs stratégies ont été proposées récemment afin d'améliorer l'efficacité de l'algorithme Tabou présenté jusqu'ici [GLO90]. L'intensification et la diversification sont deux d'entre



elles. L'intensification consiste à explorer en détail une région de l'espace de recherche  $X$  jugée prometteuse. Sa mise en œuvre réside le plus souvent en un élargissement temporaire du voisinage de la solution courante dans le but de visiter un ensemble de solutions partageant certaines propriétés. La diversification est une technique complémentaire à l'intensification. Son objectif est de diriger la procédure de recherche vers des régions inexplorées de l'espace  $X$ . la stratégie de diversification la plus simple consiste à redémarrer périodiquement le processus de recherche à partir d'une solution générée aléatoirement ou choisie judicieusement dans une région non encore non encore visitée de l'ensemble des solutions admissibles.

### 2.3.2.3 L'approche évolutive

Contrairement aux méthodes constructives et itératives qui font intervenir une solution unique (partielle ou non), les méthodes évolutives manipulent un groupe de solutions admissibles à chacune des étapes du processus de recherche. L'idée centrale consiste à utiliser régulièrement les propriétés collectives d'un ensemble de solutions distinguables, appelé population, dans le but de guider efficacement la recherche vers de bonnes solutions dans l'espace des solutions admissible  $X$ .

Après avoir généré une population initiale de solutions, aléatoirement ou par l'intermédiaire d'une méthode constructive, une méthode évolutive tente d'améliorer la qualité moyenne de la population courante en ayant recours à des principes d'évolution naturelle (Fig. 2.7).

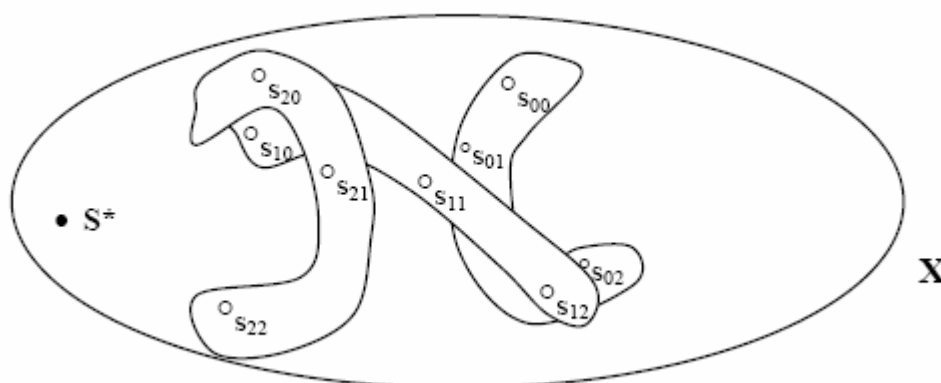


Figure 2.7 : Exploration de  $X$  par une approche évolutive

Les métaheuristiques évolutionnaires les plus récentes incluent les Algorithmes Génétiques [GOL89], la recherche dispersée (en anglais *Scatter search*) [GLO77] et les colonies de fourmis [DOR92]. Nous détaillons dans ce qui suit les Algorithmes Génétiques.

Les algorithmes génétiques (AGs) sont des méthodes évolutives, qui s'inspirent fortement des mécanismes biologiques liés aux principes de sélection et d'évolution naturelle.

Développés initialement par Holland & al en 1975 [HOL75] pour répondre à des besoins spécifiques en biologie, ces algorithmes ont été adaptés, par la suite, pour la recherche de solutions aux problèmes d'optimisation [GOL89]. Cette adaptation s'est opérée en développant une analogie entre un individu dans une population et une solution d'un problème dans un ensemble de solutions.

Les algorithmes génétiques utilisent une population d'individus (correspondants à des solutions) évoluant en même temps comme dans l'évolution naturelle en biologie. Pour chacun des individus, on mesure sa faculté d'adaptation au milieu extérieur (Fonction d'adaptation ou en anglais *fitness*). Les algorithmes génétiques s'appuient alors sur trois fonctionnalités :

- **la sélection** qui permet de favoriser les individus qui ont un meilleur *fitness* (le *fitness* sera la plus souvent la valeur de la fonction objectif de la solution associée à l'individu).
- **le croisement** qui combine deux solutions parents pour former un ou deux enfants (*offspring*) en essayant de conserver les bonnes caractéristiques des solutions parents.
- **la mutation** qui permet d'ajouter de la diversité à la population en mutant certaines caractéristiques (*gènes*) d'une solution.

La représentation des solutions (le codage) est un point critique de la réussite d'un algorithme génétique. Il faut bien sûr qu'il s'adapte le mieux possible au problème et à l'évaluation d'une solution.

Les algorithmes génétiques présentent l'avantage d'être modulaires. En effet, leur application ne nécessite pas une connaissance de la particularité du problème étudié. Néanmoins, leur efficacité dépend fortement du réglage des différents paramètres caractérisant ces algorithmes. De tels paramètres sont la taille de la population, le nombre maximal des générations, la probabilité de mutation  $p_m$ , et la probabilité de croisement  $p_c$ .

Le principe général de fonctionnement d'un algorithme génétique est donné par l'algorithme 2.2 :

---

#### Algorithme Méthode générique des AG

---

**Variables locales.**  $P_0$  : Population initiale de  $n$  individus ;

**Début**

---

---

$i \leftarrow 0$   
**Tant que** le critère d'arrêt n'est pas atteint  
**Faire**  $i \leftarrow i + 1$   
 Soit  $P_i$  une population vide,  
**Tant que**  $P_i$  ne contient pas  $n$  individus  
**Faire** Choisir aléatoirement  $I_1$  et  $I_2$  deux individus de  $P_{i-1}$ .  
 Appliquer l'opérateur de croisement sur  $I_1$  et  $I_2$  pour obtenir les enfants  $E_1$  et  $E_2$   
 Ajouter  $E_1$  et  $E_2$  à  $P_i$ .  
**Fin tant que**  
 Appliquer l'opérateur de mutation avec la probabilité  $p_m$  sur certains individus de  $P_i$   
 $P_i \leftarrow P_i \cup P_{i-1}$   
 Sélectionner  $p$  individus de  $P_i$  et supprimer les  $n$  autres.  
**Fin tant que**  
**Fin.**

---

#### Algorithme 2.2 : Principe de fonctionnement des AG.

L'inconvénient des AG est leur convergence prématurée vers un optimum. Après plusieurs générations, l'AG tend à converger rapidement et parfois même prématurément, vers la même solution. Ceci est dû au choix constant des meilleurs individus au niveau de la sélection. Ceci mène, au fil des itérations, vers une sur-dominance de certains individus et la perte trop rapide de solutions contenant des caractéristiques intéressantes. La conséquence en est la stagnation de l'AG dans un optimum local dû à la non diversité de la population de travail. Plusieurs techniques ont été proposées pour pallier à cet inconvénient et interviennent au niveau de la sélection tel que le *sharing*.

Le principe du *sharing* fut introduit par Holland [HOL75] dans le but d'avoir une meilleure exploration de l'espace de recherche et d'éviter la convergence prématurée des AG. Il est employé dans la sélection pour diversifier la population. Le *sharing* modifie la fonction d'adaptation pour réduire les effets de la sur-dominance de certains individus.

#### 2.3.2.4 L'approche hybride

La force d'un algorithme hybride réside dans la combinaison de deux principes de recherche fondamentalement différents, généralement itérative et évolutive pour mettre à profit les avantages des deux types de recherche. Le rôle de la méthode de recherche locale est d'explorer en profondeur une région donnée de l'ensemble des solutions, alors que la méthode de recherche évolutive introduit des règles de conduite générales qui guidera la recherche à travers l'espace des solutions  $X$ .

Les nombreuses adaptations qui ont été proposées dans la littérature comblent les déficiences principales des méthodes évolutives classiques dont les performances globales sont souvent

bien inférieures à celle d'une méthode séquentielle telle que la recherche Tabou ou le Recuit Simulé. Dans ce cadre plus spécifique, il est désormais établi qu'un AG simple n'est pas en mesure de fournir de bons résultats lorsque l'espace des solutions est très restreint [WID01a]. Les premiers algorithmes hybrides font référence aux travaux de Glover [GLO77], Grefenstette [GRE87] et Mühlenbein & al [MUH88]. Depuis de très nombreux croisements ont été proposés entre les algorithmes génétiques et d'autres métaheuristiques [LAS93] [DOR98, GHA00], en particulier avec les méthodes de recherche locale comme la méthode descendante, la méthode de recherche tabou ou le recuit simulé. Costa [COS95] a présenté dans sa thèse deux algorithmes génétiques hybrides, qui sont l'algorithme tabou évolutif et l'algorithme de descente évolutif.

### **2.3.2.5 Conclusion**

Les métaheuristiques sont généralement utilisées comme des méthodes génériques pouvant optimiser une large gamme de problèmes différents, sans nécessiter de changements profonds dans l'algorithme employé. En pratique, elles sont utilisées pour des problèmes ne pouvant être optimisés par des méthodes exactes. Le lecteur intéressé trouvera une comparaison des différentes méthodes, citées dans cette section, dans le récent état de l'art de Taillard et *al.* [TAI01].

Cependant une question reste inévitable : quelle est la meilleur métaheuristique ? Il n'est naturellement pas possible de répondre de manière directe et précise à cette question [WID01] : en effet, si certaines similitudes sont évidentes entre ces différentes approches (voisinages), elles diffèrent sur des points relativement délicats (liste des mouvements interdits de Tabou, opérateurs de croisement des algorithmes génétiques, ...). Des comparaisons s'avèrent difficiles et ce pour deux raisons : d'une part, il faut affiner les paramètres intervenant dans ces méthodes avec la même rigueur et, d'autre part, la qualité de la solution dépend du temps d'exécution.

Cependant, quelle que soit la métaheuristique utilisée, il faut constater qu'il est devenu possible de résoudre des problèmes d'optimisation combinatoire de taille intéressante. Les résultats déjà prometteurs des approches hybrides sont de bonne augure pour progresser de manière essentielle dans ce domaine.

## **3. Ordonnancement de la maintenance**

Le caractère multi-objectif de l'ordonnancement est reconnu depuis longtemps [BEN98], mais il est rarement, admis que d'autres fonctions, que la production, puissent y participer. Il

est en pratique difficile de tenir compte des objectifs des autres fonctions avec les méthodes d'ordonnement les plus classiques.

Cependant dans une entreprise industrielle, plusieurs services sont directement intéressés par l'ordonnement : la production, mais aussi les ventes, la maintenance, la gestion des approvisionnements ou encore la gestion des ressources humaines.

Dans ce qui suit, nous détaillons l'activité d'ordonnement de la maintenance, la spécificité de l'ordonnement de la maintenance par rapport à celui de la production, ainsi que certains des objectifs devant être satisfaits.

### **3.1. Modélisation de l'ordonnement de la maintenance**

L'ordonnement de la maintenance a pour rôle de planifier la réalisation des tâches de maintenance sur les machines et équipement de la production, c'est à dire de fixer leurs dates de début et leurs dates de fin et de réunir tous les moyens prévus pour leur réalisation (machines, personnel, outils, pièces de rechanges, etc.) [COU00]. Il intervient chronologiquement entre la fonction « Méthodes » chargée d'initier les tâches en affectant une durée à un travail (temps alloué) et la fonction « Réalisation » (ou Lancement) chargée de mettre en œuvre l'intervention au moment choisi par l'ordonnement. Les activités principales en ordonnancement de la maintenance sont la Programmation, le Lancement et l'Approvisionnement [MON96, GIA88].

La programmation est l'action d'intégrer une tâche en attente sur un planning en lui choisissant des dates de début et de fin. L'activité de lancement a pour mission de rassembler tous les moyens nécessaires, à la réalisation d'une tâche de maintenance, pour assurer leur disponibilité au moment choisi. L'ordonnement de la maintenance est également responsable de la disponibilité des « consommables » et des pièces de rechange, donc d'assurer les approvisionnements nécessaires.

La notion d'ordre de travail de maintenance peut être rapprochée de celle d'opération de production et la notion de ressource de maintenance peut être rapprochée de celle de ressource de production. Le problème d'ordonnement de la maintenance revient donc à un problème d'affectation de ressources de maintenance à la réalisation d'ordres de travail de maintenance sur des machines du système de production, les ressources de maintenances étant constituées des opérateurs de maintenance [PAZ94].

Parmi les méthodes utilisées pour l'ordonnement de la maintenance, nous citons la méthode PERT [HAR95] ou encore la méthode potentiel-tâche [BAV02]. Ces méthodes sont

généralement utilisées pour les gros travaux de maintenance et les révisions générales. Il s'agit de méthodes de gestion de projet qui conviennent bien à ce type de tâches non répétitives. L'ordonnancement des activités de maintenance est bien souvent réalisé de manière manuelle par les responsables de maintenance à l'aide de tableurs.

### **3.2. Les niveaux hiérarchiques**

L'ordonnancement de la maintenance, à l'instar de l'ordonnancement de la production, peut être décomposé de manière hiérarchique suivant trois niveaux : stratégique, tactique et opérationnel. Ces derniers correspondent respectivement aux horizons de prise de décision (long terme, moyen terme et court terme). A ces trois niveaux décisionnels sont associées trois activités de maintenance principales : établir la politique de maintenance, définir le plan de maintenance tactique et définir le plan de maintenance opérationnel [KAF01, ROB88].

#### ***- Niveau stratégique : établir la politique de maintenance***

Etablir la politique consiste à définir une structure, choisir une orientation et définir des objectifs. Les décisions prises ici sont limitées par des contraintes non négociables issues de la direction technique (budget, effectifs, espace, sécurité, etc.). La structure est établie en fonction des moyens mis à la disposition du service et des contraintes imposées, ses composantes étant le personnel, les ateliers, les outillages, etc. Le choix de l'orientation et la définition des objectifs doivent être le fruit d'un travail d'analyse réalisé en concertation avec la production, cette dernière ayant des exigences et des attentes, la maintenance ayant des capacités et des compétences [MON96].

#### ***- Niveau tactique : définir le plan de maintenance tactique***

La définition du plan de maintenance au niveau tactique est directement contrainte par les objectifs du niveau politique. Cette activité consiste à mettre en place un plan de maintenance et des méthodes de gestion de la structure physique de maintenance qui permettent d'atteindre ces objectifs. Les méthodes de gestion décrivent comment gérer chaque ressource de maintenance de la structure. Le plan de maintenance précise la forme d'action du service, c'est-à-dire les équipements à maintenir et le type de maintenance (préventive, curative), la proportion préventif/curatif visée, les formes de maintenance préventive (systématique, conditionnelle, intervalles et durées d'inspection prévues) et les formes de maintenance curative (dépannage, remise à neuf, amélioration).

Un échéancier de travaux préventifs spécifique à chaque équipement est ainsi obtenu avec les procédures d'action, celles-ci décrivent les opérations à réaliser, les ressources humaines et matérielles requises, les durées opératoires et les procédures de remise en service. Etablir un échéancier est une activité complexe : les paramètres pris en considération étant nombreux et souvent de nature stochastique, cela nécessite une analyse du contexte ainsi que l'évaluation des coûts générés. Les possibilités d'action étant très nombreuses, des choix bien souvent arbitraires doivent être faits, ce qui nécessite de consulter périodiquement les retours d'expérience. Ces consultations permettent soit de valider la stratégie adoptée, soit de modifier différents paramètres pour la corriger. L'échéancier ainsi obtenu devra être intégré à un planning de travaux correctifs au niveau décisionnel opérationnel.

***- Niveau opérationnel : définir le plan de maintenance opérationnel***

A ce niveau, le plan de maintenance tactique est mis en oeuvre. Ainsi, sont coordonnées en temps réel. Les actions en réponse aux événements aléatoires correspondant aux défaillances des équipements et les actions issues de la programmation de travaux préventifs consignés dans le calendrier établi au niveau hiérarchique supérieur.

Nous situons nos travaux à ce niveau opérationnel. Le plan de maintenance, établi au niveau tactique, décrit sur un planning, pour chaque équipement, les travaux préventifs spécifiques à effectuer. Au niveau opérationnel, l'ordonnancement de la maintenance correspond à l'évaluation du temps d'intervention et à la préparation de l'ordre de travail. Cette préparation consiste à définir l'ensemble des tâches élémentaires : les approvisionnements (matières, pièces, matériels, etc.), les travaux préparatoires, les travaux de réparation et la remise en service. Les connaissances concernant les caractéristiques d'engagement des équipements en production (importance dans le flux, taux d'utilisation, périodes de disponibilité, etc.) et des ressources de maintenance existantes vont permettre d'affecter, précisément, les tâches aux ressources. Cet ordonnancement est de nature statique, car il ne tient pas compte des tâches de maintenance corrective dont le caractère purement aléatoire nécessite un temps de réaction et d'analyse de la demande pour l'adapter au contexte du travail. Néanmoins, l'ordonnancement doit permettre d'intégrer ce type d'intervention dans le planning, suivant l'urgence de l'intervention par rapport à celle des travaux déjà programmés.

### **3.3. Les objectifs de l'ordonnancement de la maintenance**

Dans un service de maintenance, l'activité d'ordonnancement est particulièrement difficile à mettre en oeuvre en raison des multiples contraintes à prendre en considération. En effet,

l'ordonnancement de la maintenance doit gérer des problèmes bien spécifiques et plus particulièrement celui de sa dépendance de la production (arrêts de fabrication, taux d'utilisation des équipements) ou encore la prise en compte de la sécurité, le suivi des travaux sous-traités, la gestion des approvisionnements des fournitures et des pièces de rechange, la gestion de la disponibilité des moyens de manutention spéciaux, les différents niveaux d'urgence des travaux de maintenance corrective.

La spécificité de l'ordonnancement de la maintenance peut se voir à travers les points suivants [PAZ94] :

- La variété et l'incertitude des travaux de maintenance : **la durée d'une intervention** ne peut pas être connue tant que l'on a pas examiné l'équipement (diagnostic pour les interventions de maintenance corrective). L'évaluation théorique du temps d'intervention peut différer de la durée réelle de cette dernière. En effet, lors du déroulement de l'intervention, des imprévus (telle une défaillance identifiée durant l'intervention et qu'il faut réparer) peuvent allonger, de manière significative, la durée initiale de cette dernière ;
- **La date de réalisation** d'un ordre de travail de maintenance préventive peut être différée mais également être avancée par rapport à la date optimale ou à l'intervalle de tolérance de ce dernier [GIT94]. Si à l'arrivée d'une intervention de maintenance corrective, toutes les ressources de maintenance sont engagées pour l'intervention de maintenance préventive, un calcul de priorité intervient pour décider de la place d'insertion de l'ordre de travail dans le planning de maintenance. Ce calcul tient compte de la marge permise par l'intervalle de tolérance, des coûts de maintenance générés, des risques encourus sur le plan de la sécurité, de la possibilité de regroupement des interventions par familles ou encore de la volonté de perturber le moins possible le plan de production.

Les objectifs de l'ordonnancement de la maintenance peuvent être décomposés de manière hiérarchique en distinguant les objectifs internes à la maintenance des objectifs externes (Fig. 2.8 [COU00]). Les objectifs internes concernent le fonctionnement propre du service de maintenance et les objectifs externes sont imposés par d'autres services (principalement la production). Ils peuvent se retrouver en conflit de par leur nature. Leur satisfaction impose généralement aux ressources de maintenance d'intervenir sur les machines à des dates bien précises qui ne correspondent pas nécessairement à une optimisation de leur utilisation.



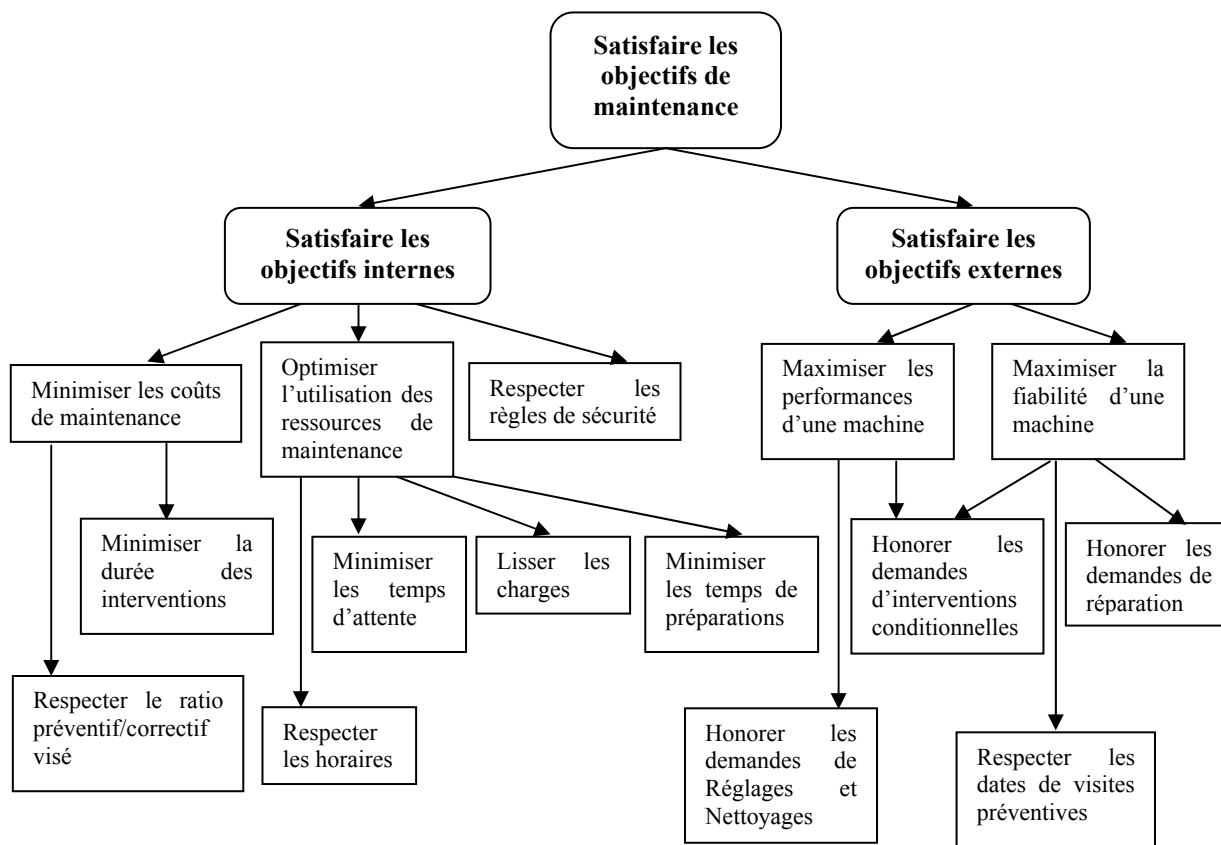


Figure 2.8: Structure hiérarchique des objectifs de l'ordonnancement de la maintenance.

***Les objectifs internes sont relatifs :***

- aux coûts de maintenance (respect du ratio préventif/correctif déterminé lors de la définition de la stratégie de maintenance ou encore la minimisation de la durée des interventions),
- à l'optimisation de l'utilisation des ressources de maintenance (tels que le respect des horaires des équipes, la minimisation des temps d'attente, le lissage des charges ou le regroupement des tâches de préparation afin d'en optimiser la durée),
- aux règles de sécurité (protection des intervenants pour les travaux dangereux).

***Quant aux objectifs externes, ils sont relatifs :***

- aux performances des machines (honorer les demandes de réglages et de nettoyage et respecter les demandes d'interventions conditionnelles),
- à la fiabilité d'une machine (honorer les demandes de réparation et d'interventions conditionnelles, respecter les dates d'interventions préventives).

#### **4. Ordonnancement de la production en présence de maintenance**

L'interaction entre la maintenance et la production et plus particulièrement à leur ordonnancement conjoint est, relativement, peu étudié et assez récent dans la littérature. Ces deux activités apparaissent a priori comme totalement antagoniste et par là même

incompatibles. En effet, si la production a pour vocation d'utiliser un système dans le but de remplir un objectif de production, traduit en termes de délais et de nombre de tâches à accomplir, la maintenance, quant à elle, est une activité nécessitant le plus souvent l'arrêt du système pour atteindre son objectif de maintien en état de fonctionnement normal.

Cependant le rapprochement entre ces deux fonctions semble, de plus en plus, naturel du fait que de plus en plus de petites tâches d'entretien sont intégrées dans les temps de production. L'objectif étant de planifier l'exécution des autres tâches de maintenance, en altérant le moins possible le plan de production, tout en respectant au mieux la périodicité de maintenance des équipements.

#### **4.1. La situation conflictuelle**

Dans la littérature actuelle portant sur l'ordonnancement de la production, les ressources (machines) sont toujours considérées comme disponibles à tout moment ou éventuellement durant certaines fenêtres de temps. Dès lors la planification de la maintenance n'est jamais prioritaire sur la production, pour effectuer des interventions préventives. Lorsque le responsable de la maintenance demande à immobiliser une ressource pour effectuer une opération de maintenance préventive, cela est généralement ressenti par les responsables de la production comme une perturbation supplémentaire et forcément malvenue. Cet arrêt de la machine a généralement été planifiée séparément des tâches de production et ne s'insère que très rarement dans un créneau horaire où elle ne nuirait pas à la productivité.

Cependant, les interventions de maintenance préventive ont un rôle prépondérant, puisqu'elles permettent au système de production de fonctionner de façon nominale. En effet, le coût engendré par une panne (impliquant une maintenance corrective, un arrêt non programmé de la production, des retards conséquents de livraison, etc.) est largement supérieur à celui d'un arrêt prévu de la production.

Les fonctions production et maintenance agissent sur les mêmes ressources à savoir les machines, cependant l'ordonnancement de leurs activités respectives est indépendant, et ne tient pas compte de cette contrainte de taille. La production a pour vocation d'utiliser le système dans le but de remplir des objectifs traduits en termes de temps et de nombre de tâches à accomplir. La maintenance quant à elle est une activité nécessitant le plus souvent l'arrêt du système pour atteindre son objectif de maintien du système en état de fonctionnement normal.

Généralement, il existe d'une part un ordonnancement de la production, destiné à satisfaire les contraintes de coût, qualité et délais. Et d'autre part, un plan de maintenance, visant à ordonnancer et affecter les ressources aux tâches de maintenance. Ces deux éléments ayant été établis séparément, leur intégration dans le fonctionnement de l'atelier pose un problème qui est souvent résolu par négociation entre les responsables respectifs des deux services et de manière séquentielle. Les besoins de la production sont donc quelque peu antagonistes. Pour satisfaire dans les délais les clients, il doit utiliser de façon optimale l'ensemble des installations, mais il doit également prévoir les interventions de maintenance dans le planning de production. Les solutions proposées ont pour but de planifier la maintenance préventive en fonction de la production et vice versa.

Quatre facteurs de complexité du problème sont à considérer :

- la typologie des systèmes de production : atelier à flux de production unique, parallèle ou hybride ;
- les tâches de maintenance : une seule tâche de maintenance périodique par machine, des tâches de maintenance pouvant être préemptées par la production, ou un ensemble de tâches de maintenance préventive par machine ;
- l'horizon de planification : à long terme (cas s'apparentant à des problèmes de gestion de projet), à moyen terme (sur un horizon de 6 mois à 2 ans) ou à court terme (horizon de 1 jour à 1 mois) ;
- les compétences des ressources humaines : un seul opérateur de maintenance qui intervient, plusieurs opérateurs de maintenance ou des équipes constituées en fonction des compétences individuelles et des besoins du système de production.

Les figures 2.9, 2.10 et 2.11 montrent bien que l'ordonnancement des activités de maintenance, à intervalles donnés, peut gêner l'ordonnancement des opérations de production, mais elle est nécessaire pour garantir la disponibilité de l'outil de production.

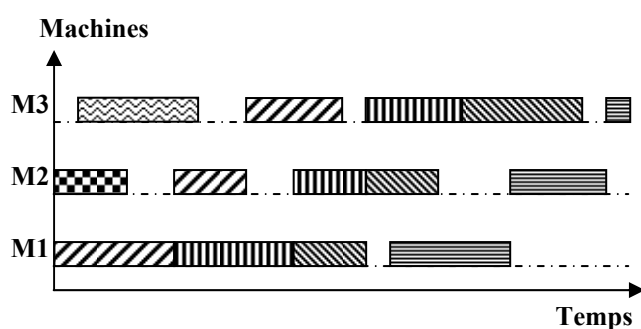


Figure 2.9: Ordonnement de la production.

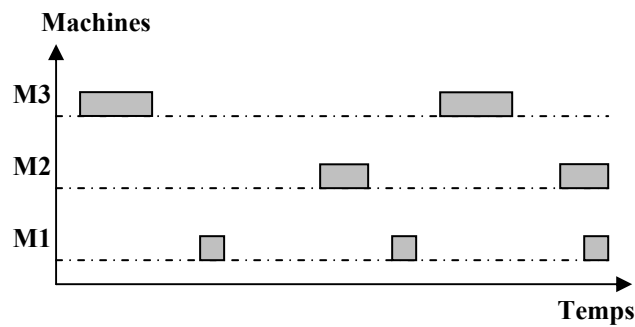


Figure 2.10: Planification de la maintenance.

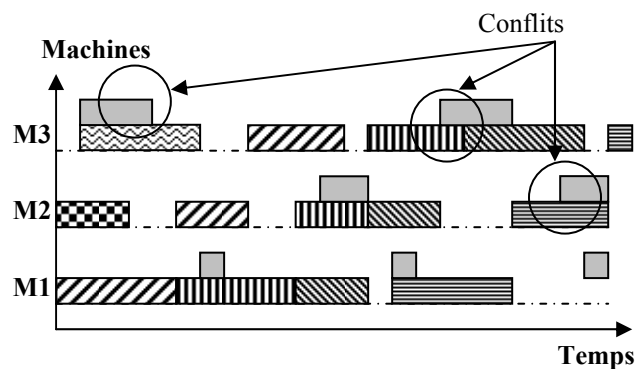


Figure 2.11: Conflits entre Maintenance et Production dans un ordonnancement séparé.

## 4.2. Les stratégies d'ordonnement conjoint

On recense dans la littérature plusieurs stratégies d'ordonnement conjoint qui vont être décrites ci-dessous et qui vise à résoudre les conflits structurels, qui existent entre ces deux activités, le plus efficacement possible. Trois stratégies d'ordonnement ont été recensées, l'ordonnement séparé, le séquentiel et l'intégré [LEE00].

### 4.2.1. L'ordonnement séparé

Actuellement, la maintenance et la production sont le plus souvent traitées de manière indépendante au sein de l'entreprise [BEM02]. Les ordonnancements correspondants à ces deux activités sont donc réalisés de manière séparée et interfèrent bien souvent l'un avec l'autre entraînant des retards dans la production ou dans la maintenance. Cette méthode implique la mise en place d'une communication accrue entre les services de maintenance et de production pour limiter les conflits dans l'immobilisation des ressources aussi bien humaines que matérielles.

#### 4.2.2. L'ordonnancement séquentiel

Cette politique consiste à planifier l'une des deux activités, maintenance ou production, et à utiliser cet ordonnancement comme une contrainte supplémentaire d'indisponibilité des ressources dans la résolution du problème d'ordonnancement de l'ensemble des deux types de tâches (Fig. 2.12). De manière générale, la maintenance est planifiée en premier, par la suite l'ordonnancement de la production est réalisé en prenant les opérations de maintenance comme des contraintes fortes d'indisponibilité des ressources [AGG02].

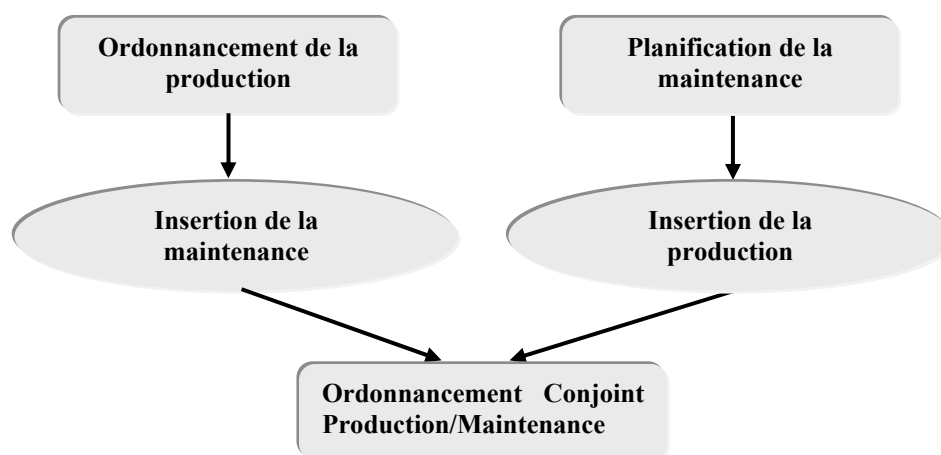


Figure 2.12 : Ordonnancement Séquentiel

#### 4.2.3. L'ordonnancement intégré

Cette politique consiste à créer un ordonnancement conjoint et simultané des tâches de maintenance et de production [BRA96] (Fig. 2.13). Une telle politique de planification limite les risques d'interférence entre la production et la maintenance et permet ainsi d'optimiser la qualité des ordonnancements. Cependant, cette politique n'est actuellement qu'au stade de recherche et de test, vu la différence de caractérisation des tâches de maintenance et de production. Néanmoins, elle offre un bon espoir de voir un jour disparaître les conflits

d'utilisation des ressources, mais elle implique la fusion des deux services production et maintenance, ou au moins la création d'un lien très fort entre les deux. De plus, un nombre non négligeable de problèmes est posé par la différence de caractérisation entre les tâches de maintenance et de production. Les premières possèdent bien souvent une date de début au plus tard en fonction des risques admissibles et ne possèdent pas de durée opératoire connue avec précision mais juste une durée moyenne (MTTR). Les secondes, quant à elles, sont caractérisées par une date de début au plus tôt, une date échue et une durée opératoire déterminée. Voilà autant d'obstacles à une planification commune aisée de ces deux types d'activités.

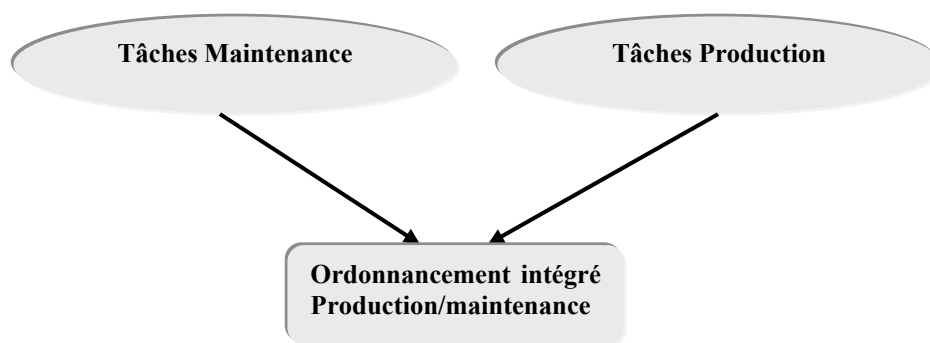


Figure 2.13: Ordonnancement Intégré

### 4.3. Enjeux d'un ordonnancement conjoint Production/Maintenance

Un des enjeux majeurs de l'ordonnancement conjoint de la production et la maintenance est la réduction des coûts de maintenance et de production. Brandolese et al [BRA96] ont réalisé une étude sur un système à machines parallèles dans l'ordonnancement desquelles ils intègrent des interventions de maintenance préventive. Ces dernières sont planifiées d'après la durée écoulée entre deux opérations successives de maintenance. Les coûts de maintenance préventive et ceux induits par les pannes sont à priori connus, les seconds étant bien entendu toujours supposés supérieurs aux premiers. Dans le cas contraire, on ne verrait pas l'intérêt d'introduire une politique de maintenance préventive.

Un autre enjeu de la planification conjointe de la maintenance et de la production consiste à obtenir un ordonnancement robuste, qui est susceptible de rester réalisable malgré l'apparition d'imprévus de production. Pour trouver une solution de ce type, on accepte généralement une certaine dégradation du critère d'optimisation afin de conserver un certain degré de liberté qui permet éventuellement des modifications mineures ou l'insertion d'opérations de maintenance, lorsque l'ordonnancement des opérations de production est

réalisé, afin d'avoir un ordonnancement robuste. C'est un ordonnancement que l'on peut situer entre un ordonnancement statique, complètement figé, et un ordonnancement dynamique, souvent remis en cause en partie ou totalement.

En ce qui nous concerne, nous pensons que la robustesse d'un ordonnancement conjoint production/maintenance peut être mesurée par sa réactivité à des événements aléatoires tels que des pannes ou encore par la constatation d'une dégradation d'un équipement induisant des interventions de maintenance corrective ou conditionnelle. Si les indices de fiabilité de la planification de chacune des tâches restent dans des proportions acceptables, on dira alors que l'ordonnancement est robuste.

La robustesse d'un ordonnancement fera l'objet du chapitre IV, nous y présenterons plus en détail une revue de la littérature dans le domaine.

#### **4.4. Discussions et conclusion**

La nécessité de faire partager les ressources entre les fonctions production et maintenance constitue une source de conflits et traduit l'existence d'un lien fort entre ces fonctions. En effet, trop souvent, les séquences d'opérations de maintenance et de production sont perçues comme antagonistes. Les périodes d'immobilisation des équipements nécessaires aux interventions des agents de maintenance, sont considérées comme perturbant réduisant les périodes d'utilisation des ressources et non comme un facteur favorisant leur bon déroulement. Ce type de conflit entraîne naturellement des querelles qui nuisent à la productivité globale de l'entreprise. La planification intégrée apporte une solution à ce conflit et entraînera sans aucun doute un changement de mentalité en déracinant l'idée préconçue : *«Moi je fabrique, toi tu ré pares»* et en mobilisant l'ensemble des compétences disponibles.

Nous sommes donc face à un problème multicritère, d'une part ordonnancer la production sous les contraintes de respect des délais, coût et qualité des produits et d'autre part planifier la maintenance sous les contraintes de sûreté de fonctionnement des équipements qui assurent la pérennité de l'outil de production. Ainsi la planification des activités de maintenance sur les machines, à intervalles donnés, peut gêner l'ordonnancement des opérations de production, mais elle est nécessaire pour garantir la disponibilité de l'outil de production. Des études récentes sur l'efficacité de la gestion en maintenance [LEE00] ont montré qu'un tiers des coûts de maintenance provient d'opérations inutiles ou mal effectuées. Cette inefficacité a pour raison principale, l'absence d'informations réelles qui permettraient de développer un

modèle de maintenance préventive capable de réduire ou d'éliminer les interventions inutiles et d'éviter les risques de pannes les plus graves des machines.

Notre travail s'inscrit dans ce souci d'optimiser l'ordonnancement de la production et de la maintenance en anticipant tout conflit pouvant se présenter entre ces deux services et ceci peut être réalisé en créant une certaine coopération entre les deux services production et maintenance.

L'ordonnancement conjoint de la production et de la maintenance n'a attiré que récemment l'attention des chercheurs. Un état de l'art sur ces travaux sera présenté dans la section suivante. Nous présenterons par la suite les problématiques étudiées dans cette thèse.

## **5. Etat de l'art : Ordonnancement conjoint production/maintenance**

La majeure partie de la littérature dédiée aux problèmes d'ordonnancement se place dans le contexte où les ressources nécessaires à l'exécution des tâches sont disponibles en permanence. Cette hypothèse n'est pourtant pas fidèle à la réalité des entreprises. En effet, les différentes ressources qu'elles soient humaines ou matérielles peuvent, pour diverses raisons, être indisponibles. Les indisponibilités des ressources peuvent être dues à une opération de maintenance sur les machines de l'atelier ou à des emplois du temps du personnel. Nous supposons dans cette étude qu'une indisponibilité est due à une opération de maintenance préventive systématique. Ces travaux ne considèrent que les opérations de production, et optimisent un critère bien précis.

Quant à la littérature consacrée au problème de l'ordonnancement de la maintenance, elle est relativement peu fournie. Les travaux dans le domaine sont principalement orientés vers le choix de stratégies de maintenance et peu d'auteurs abordent les problèmes organisationnels et fonctionnels qui sont pourtant essentiels [NOY99, SWA99].

Le fait que la maintenance prenne de l'importance au sein de l'entreprise a suscité d'autres recherches qui tiennent compte des opérations de maintenance, au cours de la résolution du problème d'ordonnancement posé. Les travaux qui abordent les relations entre la production et la maintenance sont le plus souvent des études de cas très particuliers, les approches génériques faisant défaut dans ce domaine.

Nous allons présenter dans cette section les travaux que nous avons recensés dans le domaine de l'ordonnancement conjoint production/maintenance. Nous introduirons, tout d'abord une classification des travaux par rapport à deux critères que nous avons jugés importants : l'interaction entre la politique de maintenance et la gestion de production puis entre la



production et la maintenance. Une synthèse de l'ensemble des travaux nous permettra de dégager les points essentiels des orientations à donner à nos travaux.

Deux grandes classes de travaux sont à distinguer. La première classe de travaux étudie l'interaction entre les **politiques de maintenance** et la **gestion de production**, l'objectif principal de cette première classe est l'optimisation simultanée du flux de production et de la politique de maintenance. La deuxième classe de travaux étudie l'interaction entre la **maintenance** et la **production** et plus particulièrement à leur **ordonnement conjoint**. C'est vers cet axe que nous avons choisi de diriger nos travaux. Ces deux classes se distinguent par le niveau hiérarchique d'intervention de chacune. La première classe des travaux se situe au niveau tactique et la seconde classe au niveau opérationnel.

## **5.1. Interaction entre politique de maintenance et gestion de production**

L'approche développée, dans cette classe, permet de prendre en compte les indicateurs de performance du système. Ces indicateurs sont exploités pour développer la politique de maintenance. Ils concernent, essentiellement, l'état du système de production tels que le niveau des stocks, l'âge des équipements et la demande prévisionnelle et la qualité des produits tels que le taux de rebus et le taux de reprise. La définition de la politique de maintenance doit prendre en compte le plan de production ainsi que la disponibilité des ressources nécessaires à la réalisation des actions de maintenance.

Tout d'abord, il existe les travaux de Buzacott & al [BUZ90], Dallery & al [DAL92] et Xie [XIE93] sur l'évaluation des performances des systèmes de production soumis à des pannes. L'impact des stratégies de maintenance sur la production au juste à temps est étudié dans les travaux de Abdounour & al [ABD95]. Les systèmes étudiés sont des lignes de fabrication, les systèmes d'assemblage et les systèmes gérés par Kanban. D'autres travaux (Boukas & al [BOU95], Gharbi & al [GHA00]), portant sur le contrôle de la production des systèmes soumis à la maintenance préventive, développent des approches analytiques et par simulation. Il existe, aussi, quelques travaux, sur l'optimisation des stratégies de maintenance et de gestion de stocks, appliqués à un système composé d'une seule machine et réalisant un seul produit. En particulier, Cheung & al [CHE97] considèrent l'optimisation simultanée du stock de sécurité et de la politique de maintenance de type âge. Il n'est pas souvent facile de simplifier et de résoudre analytiquement les modèles proposés. Ouali & al [OUA01] ont développé une politique intégrée de maintenance et de commande pour l'optimisation du

stock de produits finis d'une ligne de production. La complexité d'une telle approche réside dans la difficulté d'obtenir une forme analytique simple permettant d'optimiser la fonction de disponibilité ou la fonction économique.

## 5.2. Interaction entre la production et la maintenance

Cet aspect du problème est, relativement, peu étudié et assez récent dans la littérature. Nous le retrouvons décliné selon deux approches : globale (plusieurs machines) où les auteurs déterminent un ordonnancement global et restreint (une machine) où les auteurs étudient le problème uniquement dans le cas d'une machine et déterminent à ce moment là un ordonnancement partiel.

Dans la première approche Rishel & al [RIS96] proposent d'utiliser le système MRP (Material Requirements Planning) existant dans l'entreprise pour planifier simultanément la production et la maintenance. Le MRP définit des dates de lancement, de manière séparée, pour les tâches de production et les opérations de maintenance sans se soucier des conflits qui risquent de se poser au niveau de l'atelier (machines) et fournit un PDP (Plan Directeur de Production). Les auteurs ne réalisent pas exactement une optimisation, mais comparent la planification fournie par le système MRP en réponse à six jeux de données différents. Cela leur permet de choisir une politique optimale de maintenance.

Brandolese & al [BRA96] déterminent un ordonnancement conjoint de production et de maintenance fixant la date d'exécution de chaque travail et sur quelle machine il doit s'effectuer. Il inclut les interventions de maintenance, en les plaçant le plus près possible des périodes optimales fixées. Le but est d'optimiser la fonction objectif globale, en respectant les contraintes temporelles et en minimisant la durée totale de réalisation (Durées opératoires + Temps total de préparation (*set up time*) + Interventions de maintenance). Les auteurs proposent un modèle de description des différents coûts qui prend en compte les coûts de production, de préparation et de maintenance. Ils déterminent notamment la période optimale de maintenance ( $T^*$ ) qui minimise le coût de maintenance. Toute intervention faite avant ou après  $T^*$  accroît ce coût. Si  $T$  (période effective de maintenance) est grand, les pannes seront fréquentes ce qui engendre des coûts supplémentaires de maintenance corrective, et si  $T$  est petit, les interventions de maintenance préventive seront trop fréquentes par rapport aux besoins réels et induiront un coût trop élevé ;

BenDaya & al [BEN98] ont développé un modèle qui intègre dans le EPQ (Economic Production Quantity) la qualité et la politique de maintenance. Le système est soumis à des

inspections périodiques. Trois politiques différentes sont utilisées et les opérations de maintenance préventive sont coordonnées avec les inspections de contrôle de qualité de telle sorte qu'elles soient exécutées en parallèle. L'intervalle désignant la période de contrôle de la qualité est déterminé par une loi de Weibull. Le modèle détermine pour chaque politique EPQ le coût global associé (coût de set up, coût des stocks, coût du contrôle de qualité et coût de maintenance préventive), les dates d'interventions pour le contrôle de qualité et de maintenance préventive et le niveau de maintenance. Ces auteurs déterminent une politique d'intervention de maintenance préventive coordonnée avec le contrôle de qualité, mais pas un ordonnancement au sens propre.

Weinstein & al [WEI99] déterminent une politique de maintenance qui minimise les défaillances des équipements et maximise leur disponibilité. Ils proposent un modèle pour mettre en œuvre une politique de maintenance dans le but de minimiser son coût global (constitué des différents coûts associés aux activités de production, aux activités de maintenance, aux défaillances et au coût d'interruption du plan de production). Deux types de maintenance préventive sont considérés : la maintenance préventive systématique périodique (périodes d'interventions identiques) et séquentielle (périodes d'interventions inversement proportionnelles à la durée de vie de l'équipement maintenu). Le modèle d'évaluation de la politique de maintenance fait intervenir deux niveaux de décision, le premier (APP : *Aggregate Production Planning*) donne une planification grossière pour trouver une politique d'organisation et le second (MSP : *Master Scheduling Planning*) donne un ordonnancement de la production.

Par contre Anily & al [ANI99] ne s'intéressent qu'aux opérations de maintenance (pas de travaux à ordonnancer) dans un atelier à trois machines. Le but est de trouver une politique d'intervention optimale spécifiant à quelle période intervenir et sur quelle machine, tout en minimisant le coût moyen d'utilisation. La politique consiste à trouver un cycle optimal. Les auteurs ont démontré qu'une solution cyclique est optimale si elle respecte certaines propriétés.

Dans la seconde approche Qi & al [QI99], Graves & al [GRA99], Lee & al [LEE00] et Bennour & al [BEN01] proposent de construire un ordonnancement conjoint production/maintenance sur une machine.

Qi & al constituent des lots de travaux situés entre deux interventions successives de maintenance préventive pour placer l'opération de maintenance par rapport au dernier travail de chacun d'entre eux. Les travaux de Graves & al sont semblables à ceux de Qi & al. De la même façon, ils déterminent les lots de production à réaliser entre deux tâches de maintenance

successives. Cela revient à déterminer un ordonnancement de production tout en y intégrant au mieux les activités de maintenance. Les auteurs ont utilisé la programmation dynamique avec tri des travaux selon une règle heuristique (WSPT, EDD).

Lee & al déterminent un ensemble d'ordonnancement conjoint production/maintenance mais partiels pour chaque machine, sachant que chaque machine doit être maintenue une fois sur l'horizon considéré. Chacun de ces ordonnancements satisfait certains critères d'optimalité. L'ensemble ainsi construit réunit des ordonnancements admissibles parmi lesquels ils choisissent une solution.

Bennour & al proposent un modèle intégré représentant les opérations de production et les tâches de maintenance dans un même graphe afin de les ordonnancer de manière globale. Cette modélisation par les graphes potentiels-tâches leurs permet d'utiliser, lors de la résolution, les méthodes classiques d'ordonnancement. Leur travail s'est restreint à l'étape de modélisation.

### **5.3. Synthèse**

Pour l'ensemble de ces travaux, il n'existe pas de problématique type, ni de cadre de caractérisation systématique. La plupart des travaux concernant les relations Production/Maintenance utilisent des approches probabilistes dans le but de déterminer le meilleur moment pour planifier une opération de maintenance. L'enjeu majeur de cette planification commune est la réduction des coûts en fonction d'un compromis entre le coût de maintenance et le risque de perte de disponibilité des machines.

Les chercheurs ont diverses manières d'appréhender et de caractériser le problème de l'ordonnancement de la maintenance et de la production. Néanmoins la plupart des auteurs ont opté pour une approche intégrée et ont résolu un problème statique. Autrement dit, ces auteurs considèrent les opérations de maintenance et de production simultanément. Les caractéristiques (date de disponibilité, durée opératoire...) de l'ensemble des tâches à exécuter sont connues à priori.

Dans le cas de la maintenance préventive, l'objectif est de réaliser l'ordonnancement des tâches de maintenance. Pour ce faire, différentes politiques de planification peuvent être mises en place. Certains auteurs proposent de réaliser les tâches de maintenance au cours d'arrêts des machines programmés pour d'autres activités (inspections de contrôle de qualité par exemple) ([CHA97, BEN98, ANI99, SAF99, TSA01, GOP01]).

D'autres auteurs se positionnent au niveau de la planification et déterminent un planning des opérations de maintenance et de production, sans se préoccuper des conflits qui risquent d'apparaître. Les derniers, enfin, traitent des problèmes d'ordonnancement au sens propre relatif à une machine, à des machines parallèles et au *flowshop*. Ils construisent un ordonnancement respectant toutes les contraintes et optimisant un critère donné. Par contre, ils ne s'intéressent pas forcément au même niveau de décision ni aux décideurs et encore moins à l'interactivité qui existe entre eux. Dans le tableau 2.2, nous présentons un état de l'art non exhaustif sur ces travaux.

		Problème d'ordonnancement				
		Problème à une machine	Problème à machines parallèles	Flowshop à deux machines	Flowshop à plusieurs machines	Autres
Maintenance	Préventive	[QI99] [GRA99]	[BRA96] [CHA97] [LEE00]	[ESP01]	[SAN97] [AGG01]	[ANI99] [BEN98] [KOB97] [WEI99]
	Mixte			[CHU97] [CHU99]		[RIS96]

Tableau 2.2 : Classification des articles en fonction du type de maintenance et du problème d'ordonnancement.

Les problèmes à une machine ont un caractère fondamental. Ils peuvent former la base de problèmes plus complexes. La majorité des travaux étudiés dans la littérature ne s'intéressent qu'à des critères de production même pour les cas où les périodes d'indisponibilité sont dues à des interventions de maintenance préventive. D'autres auteurs se sont intéressés à l'étude des problèmes à machines parallèles. Ces machines peuvent être identiques ou pas. La majorité des problèmes traitant des machines parallèles sont démontrés *NP-difficile*. Ce qui a incité la majorité des chercheurs à proposer des algorithmes exacts pour les problèmes relaxés et des heuristiques pour résoudre des problèmes généralisés. Dans le cas du problème d'ordonnancement dans un atelier *flowshop* avec prise en compte des contraintes d'indisponibilité, la majorité des travaux se sont restreints au cas de deux machines étant donné la difficulté du problème général ( $m > 2$ ). Dans ce cas d'étude, seul l'aspect production est pris en compte.

En résumé, nous dirons que le contenu des articles est très varié. En effet les auteurs abordent des problèmes souvent différents en les caractérisant à leur manière. Ceci ne permet pas de faire une synthèse globale sur les travaux effectués dans le domaine «ordonnancement

production/maintenance». Nous avons donc opté une classification sur la base d'un certain nombre de critères qui nous semble représentatif du domaine étudié à savoir :

- le type de maintenance (préventive, corrective ou mixte),
- la stratégie d'ordonnancement conjoint (séparée, séquentielle ou intégrée),
- la solution obtenue (statique, dynamique ou robuste).
- le type d'atelier (*Flowshop*, *Jobshop*, *Openshop*, etc.)

D'une manière générale l'essentiel des travaux recensés traite le problème d'ordonnancement conjoint production/maintenance par l'approche intégrée pour obtenir des solutions statiques. Chung-Yee [CHU97, CHU99] a traité le problème de l'ordonnancement séquentiel production/maintenance dans le cas des ateliers de type *flowshop* à deux machines avec des contraintes de disponibilité dues à des pannes ou à l'exécution des tâches de maintenance préventive. La maintenance représente, dans ce cas, une contrainte forte. Le problème est traité dans un environnement déterministe et la période d'indisponibilité est connue à l'avance.

Espinouse & al [ESP01] ont traité le problème du *no-wait flowshop* à deux machines avec des contraintes de disponibilité. Ils démontrent que le problème est *NP-difficile*. Ils proposent une heuristique basée sur l'algorithme de Gilmore & Gomory dans un environnement déterministe où les périodes d'indisponibilité, pour la maintenance préventive, sont connues à l'avance.

Aggoune [AGG01] propose une méthode générale pour résoudre le problème de l'ordonnancement d'un *flowshop* à plusieurs machines avec des périodes d'indisponibilité sur chaque machine. L'objectif de l'ordonnancement est de déterminer la séquence qui satisfait les contraintes sur chaque machine et minimise le  $C_{max}$ . Il présente deux modèles pour prendre en compte les activités de maintenance. Pour le premier, les périodes d'indisponibilité sont connues, alors que pour le second, elles sont déterminées durant la construction de l'ordonnancement.

Enfin Sanmarti & al [SAN97] utilisent des modèles probabilistes qui permettent de planifier des interventions de maintenance préventive sur des lignes de production continues et semi continues de type *flowshop*. Ils considèrent que le modèle de chaque période de l'horizon de simulation permet de choisir entre produire et risquer d'avoir une défaillance ou maintenir et augmenter le coût de production.

Le tableau 2.3 présente une classification de ces travaux seulement en fonction de la stratégie de résolution (séquentielle, intégrée ou séparée). Nous n'avons pas jugé nécessaire de présenter la nature des solutions (statique, dynamique ou robuste) vu que seuls Sanmarti & al

[SAN97] traitent l'aspect robustesse de la solution. La majorité des travaux fournissent des solutions statiques.

Stratégie de résolution		
Séparée	Séquentielle	Intégrée
[ANI99]	[AGG01]	[BRA96]
[CHA97]	[BEN01]	[GRA99]
[GOP01]	[CHU97]	[LEE00]
[KOB97]	[CHU99]	[Q199]
[TSA01]	[ESP01]	[RIS96]
		[WEI99]

Tableau 2.3 : Classification des travaux en fonction de la stratégie de résolution

Certains articles n'ont pas pu être classés selon les critères retenus soit parce qu'ils traitent de la maintenance uniquement, soit parce qu'ils présentent des cas industriels :

Talal & al [TAL95] tentent de résoudre le problème de l'ordonnancement production/maintenance, dans une raffinerie, par une approche intégrée basée sur un modèle mathématique de programmation en nombre entier. Il s'agit en fait de déterminer le planning de maintenance des unités de cette raffinerie, sur la base d'un programme opérationnel de maintenance planifié sur un horizon H, sous les contraintes de maximisation du taux de productivité des unités et la satisfaction des contraintes de maintenance.

Pérès & al [PER97] proposent un logiciel de simulation pour l'évaluation des stratégies de maintenance. Le principe consiste à définir des produits virtuels devant passer sur les ressources du système à maintenir.

Ahmad & al [AHM98] ainsi que Paz & al [PAZ94] proposent un état de l'art sur les méthodes de résolution des problèmes d'ordonnancement de la maintenance. Les premiers présentent plusieurs approches telles que la programmation mathématique, les méthodes heuristiques, et plus récemment les systèmes experts, l'approche floue, les algorithmes génétiques et l'analyse simulée. Les seconds recensent l'ensemble des travaux qui traitent de l'ordonnancement de la maintenance pour ressortir avec une problématique du domaine avec les points traités et ceux qui restent à résoudre.

Deniaud [DEN99] a étudié un cas industriel dans le lequel une approche par simulation est employée afin de déterminer un planning de production et de maintenance sur des lignes de fabrication de turbines à vapeur, avec pour objectif de minimiser les effets de la maintenance préventive sur la production.

Quant à Gopalagrishnan [GOP01], il présente une méthode de recherche TABOU pour la résolution du problème de l'ordonnancement de la maintenance uniquement.

En conclusion de cette étude bibliographique, on remarque qu'il y a trois grands axes de recherche :

- Dans le premier, les auteurs s'occupent des problèmes d'ordonnancement au sens propre relatifs à une machine ([QI99, GRA99]), à des machines parallèles ([BRA96, LEE00]), et au *flowshop* ([SAN97]). Ce groupe construit un ordonnancement tout en en donnant les dates de début de chaque opération en respectant toutes les contraintes et en optimisant un critère donné ;
- Dans le deuxième, ils se positionnent à un niveau de décision plus élevé que l'ordonnancement appelé « niveau de planification » ([RIS96, WEI99]), qui revient à trouver un planning des opérations de maintenance et de production, donc à trouver une planification grossière sans se préoccuper des conflits qui risquent de se poser au niveau d'une machine donnée ;
- Dans le dernier, les auteurs étudient le problème pour déterminer une politique de maintenance optimale ([BEN98, ANI99]). Ce groupe détermine des politiques d'intervention pour une maintenance efficace, l'exemple de Bendaya & al [BEN98] est représentatif car il coordonne les opérations de maintenance avec les inspections de contrôle qualité. Cependant la plupart des auteurs ont opté pour une stratégie intégrée et ont résolu un problème d'ordonnancement statique.

## 6. Problématiques étudiées

L'étude bibliographique que nous avons présentée dans la section précédente, nous a permis de définir les orientations à donner à notre travail, par rapport au manque de références dans certains axes de recherche abordés. En effet, la majorité des travaux étudiés dans la littérature ne s'intéressent qu'à des critères de production même pour les cas où les périodes d'indisponibilité sont dues à des interventions de maintenance préventive.

Nos orientations concernent le type d'atelier, le type de maintenance ainsi que les variables de décision liées à cette dernière. Et enfin la stratégie de résolution adoptée. Dans ce qui suit, nous allons détailler chacun de ces aspects. Nous ferons ensuite, par rapport à cela, une synthèse où nous exposerons les orientations des chapitres suivants.



### **- *Flowshop de permutation***

Le *flowshop* est un modèle très présent dans la littérature depuis 1954, année où S.M.Johnson [JOH54] a proposé son algorithme de résolution pour le cas de deux machines. Ce modèle se définit par  $n$  tâches à ordonnancer sur  $m$  machines (voir Annexe A). Chaque tâche se décompose en  $m$  opérations qui doivent être exécutées sur l'ensemble des machines. Chaque opération doit attendre avant de commencer, que celle qui la précède soit achevée. De plus, l'ordre de passage des opérations sur l'ensemble des machines est le même pour toutes les tâches. En outre, pour le cas particulier du *flowshop* de permutation toutes les tâches sont disponibles à l'instant 0. Ce modèle connaît de nombreuses applications aussi bien dans l'industrie qu'en informatique, cependant, dans la littérature, très peu de travaux leurs sont consacrés d'où notre intérêt pour ce modèle vu les nombreuses applications industrielles de la maintenance.

### **- *Maintenance préventive systématique***

Le choix de la maintenance préventive systématique est une conséquence qui découle de l'ordonnancement vu que cette dernière est prédéfinie et facile à mettre en œuvre. Elle est la plus indiquée dans notre cas du fait de son caractère planifiable qui se traduit par des interventions planifiées ou des inspections périodiques, qui visent à agir pour prévenir la panne. Contrairement à la maintenance corrective qui agit pour réparer et donc après la survenue de la panne et donc de manière aléatoire.

L'aspect planifiable de la maintenance préventive systématique fait de ce type de maintenance la plus appropriée pour l'ordonnancement de la maintenance. Dans ce cas, la recherche d'une solution d'ordonnancement de la production va être corrélée à la recherche de la solution d'ordonnancement de la maintenance. Chaque mécanisme de recherche va être contraint par l'autre, justifiant ainsi le besoin d'une coopération étroite entre les deux fonctions.

Quand au choix des périodes de maintenance préventive, la littérature propose deux types : des périodes fixes prédéterminées, ou des périodes aléatoires générées lors de l'ordonnancement. Pour notre cas d'étude, nous avons travaillé avec la maintenance préventive déterministe dont les périodes sont données d'avance. Néanmoins, nous rajoutons des intervalles de tolérance autour de chaque période de maintenance, durant lesquels le coût de maintenance est faible. Ces intervalles donneront plus de flexibilité à la planification de la maintenance en cas de besoin.

### **- *Ordonnancement conjoint Production/Maintenance***

L'aspect économique nous a semblé le plus intéressant dans la mesure où c'est l'indicateur le plus important pour les entreprises. C'est pourquoi dans cette étude nous considérons un critère d'optimisation tenant en compte les deux aspects production et maintenance. Le rôle premier de la maintenance est d'augmenter la disponibilité des machines afin de réduire les coûts directs et indirects. Les coûts directs de cette maintenance sont ceux relatifs aux diverses pièces de rechange, main d'oeuvre, etc. Par contre, les coûts indirects sont essentiellement dus au manque à gagner engendré par un arrêt de production. On comprend alors que l'enjeu d'une bonne politique **conjointe** de production et de maintenance est très important pour les entreprises soucieuses d'avoir une meilleure maîtrise des coûts de production et de maintenance. Cette politique doit être le fruit d'une concertation conjointe entre les deux services respectives chargés de ces fonctions.

Il est très difficile dans le contexte actuel des entreprises industrielles, sauf refonte complète de leur organisation qui verrait les services production et maintenance fusionner en une seule entité, d'opter, de suite, pour une stratégie d'ordonnancement intégré. Cependant une stratégie séquentielle nous semble le meilleur choix à prendre actuellement. Néanmoins, comme la production reste la fonction prédominante dans les systèmes manufacturiers, c'est elle qui représentera une contrainte pour la planification de la maintenance. Ceci du fait qu'il existe des priorités entre les tâches de maintenance liées aux machines sur lesquelles elles doivent s'exécuter. Cela se fera en profitant des périodes d'inactivité des machines pour planifier le maximum de tâches de maintenance et minimiser ainsi les arrêts de production pour maintenance.

Nous nous intéressons au niveau planification et nous essayons de déterminer des ordonnancements conjoints production/maintenance, en faisant une gestion des conflits qui risquent de se poser lors de l'intégration de la maintenance dans l'ordonnancement de la production.

Nous allons considérer les deux options à savoir : la stratégie séquentielle supportée par deux services distincts et la stratégie intégrée supportée par un même et unique service Production/Maintenance. Nous n'avons pas la prétention de résoudre ainsi tous les conflits mais seulement de les ramener à un seuil acceptable qui perturberait le moins possible aussi bien la production que la maintenance.

La figure 2.14 présente la situation des problématiques étudiées dans cette thèse par rapport aux trois axes de recherche abordés dans l'étude bibliographique et relatifs à la typologie de la maintenance, la stratégie de résolution adoptée et enfin à la nature de la solution générée.

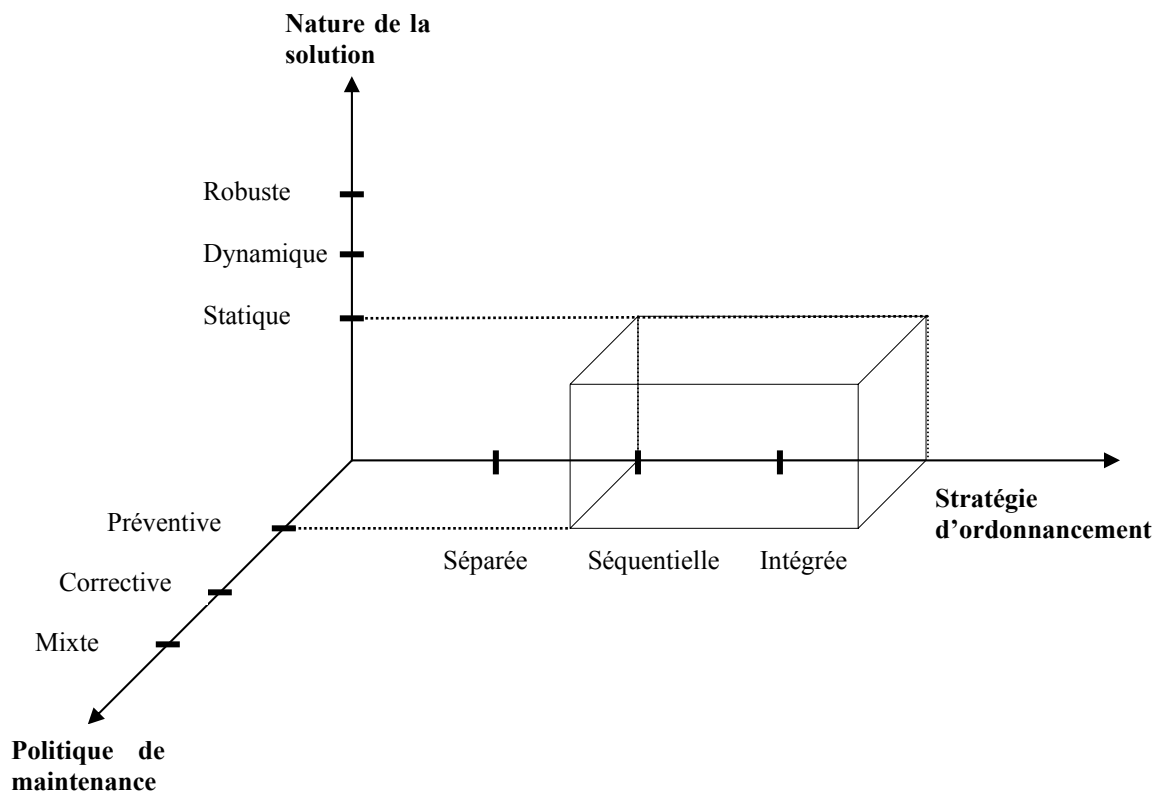


Figure 2.14: Problématiques étudiées

Ces trois problématiques vont être au cœur de notre travail. Elles vont servir de point de départ pour le chapitre suivant, dans lequel nous allons adapter des méthodes d'ordonnancement tel que la recherche Tabou et les algorithmes génétiques sur les problèmes d'ordonnancement conjoint production/maintenance. Pour notre étude nous avons opté pour l'ordonnancement de la production dans un premier temps puis l'insertion des tâches de maintenance dans un deuxième temps. Pour le problème du *flowshop*, on commencera par résoudre le problème dans le cas d'une machine puis celui de plusieurs machines. Les solutions générées sont de nature statique.

Les hypothèses qui sont faites pour la suite de notre travail, sont les suivantes :

- on est dans le cas discret,
  - c'est un *flowshop* de permutation où toutes les machines sont susceptibles d'être maintenues,
- on s'est intéressé à la maintenance préventive systématique,
- nous allons résoudre le problème par les stratégies séquentielle et intégrée,
  - on a un problème d'ordonnancement conjoint de type statique, donc on dispose initialement de toutes les données pour la résolution du problème.

De manière synthétique, nos travaux sont axés sur la recherche de planning de maintenance moyen terme. Il s'agit, comme le montre la figure 2.15, à partir d'un plan de production et d'un plan de maintenance préventive de fournir, de manière séquentielle ou intégrée, un planning des opérations de maintenance (pour une ou plusieurs équipes de maintenance) respectant à la fois les contraintes de production et les impératifs de remise en état des machines.

Les données du plan de production sont fournies par la gestion de production. Elles se présentent sous la forme d'un ensemble de lots de pièces à fabriquer. Chaque lot entre dans l'atelier à une date prévue et devra être livré à une date prévue. La durée opératoire nécessaire à la réalisation des lots dépendra de la taille du lot et de la nature des produits. Comme nous avons opté pour un atelier de type *flowshop* cela implique que les lots traversent l'atelier en passant successivement sur toutes les machines. Le plan de maintenance est caractérisé par un ensemble de tâches de maintenance. Chaque tâche de maintenance est liée à une machine et nécessite la présence d'une équipe de maintenance. Comme nous avons opté pour une maintenance préventive systématique périodique cela implique que les tâches de maintenance sont périodiques et leur périodicité est connue.

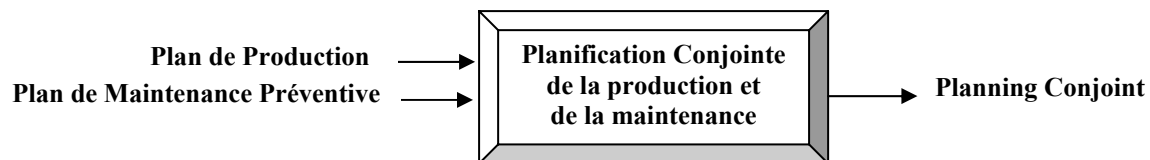


Figure 2.15: Ordonnancement conjoint Production/Maintenance

## 7. Conclusion

Nous avons présenté dans la première partie de ce chapitre l'ordonnancement des activités de production et de maintenance puis la spécificité de l'ordonnancement de la maintenance par rapport à l'ordonnancement de production ainsi que son importance au sein de l'entreprise. Dans la seconde partie, nous avons présenté un état de l'art sur les différents travaux réalisés dans ce domaine. La majorité des travaux qui ont traité le problème de l'ordonnancement avec prise en compte des indisponibilités des machines ne considèrent pas que ces indisponibilités soient dues à des interventions de maintenance. Peu de travaux se sont intéressés à la maintenance.

De cette étude bibliographique nous avons dégagé les orientations à donner à notre travail. Ces choix sont guidés par le manque de références dans certains axes abordés dans cette recherche bibliographique ainsi que de travaux de synthèse dans le domaine. Ces orientations sont traduites en terme de type d'atelier, de politique de maintenance ainsi que de stratégie de résolution à adopter. Le problème consiste alors à trouver un ordonnancement conjoint des tâches de maintenance et de production, qui respecte au mieux les contraintes du problème, et cela par l'adaptation des heuristiques d'ordonnancement de la production pour ce type de problème.

Nous présenterons donc dans le chapitre suivant l'adaptation des différentes heuristiques choisies ainsi que les résultats obtenus. Le but de ce travail est de montrer l'intérêt de telles approches pour la minimisation des conflits sur les ressources entre la production et la maintenance qui permettrait de faire des gains sensibles. Cette démarche peut être validée, dans une deuxième étape, par des outils de mesure de la robustesse de ces heuristiques que nous détaillerons dans le chapitre 4.



---

# CHAPITRE III

## *Ordonnancement Conjoint Production/Maintenance : Cas du Flowshop*

---





---

## ***CHAPITRE III.***

### ***Ordonnancement Conjoint***

### ***Production/Maintenance : Cas du flowshop***

---

***Résumé :*** *Le problème de l'ordonnancement de la production est un problème NP-complet [GAR 79, CAR88] et donc celui de l'ordonnancement conjoint de la production et la maintenance l'est aussi du fait que nous ajoutons une contrainte supplémentaire au problème qui est l'insertion des tâches de maintenance.*

*Nous présenterons dans ce chapitre les deux stratégies d'ordonnancement conjoint production/maintenance : la stratégie séquentielle, qui consiste à ordonner les tâches de production et à prendre, dans un deuxième temps, cet ordonnancement comme une contrainte forte pour l'insertion des tâches de maintenance, et la stratégie intégrée, qui consiste à créer un ordonnancement simultanée des tâches de production et de maintenance, ainsi que des hybridations pour chacune des deux stratégies. Nous déclinons ces stratégies selon les trois approches de résolution suivantes : constructive (RA, Gupta, CDS, Palmer et NEH), itérative (Tabou) et évolutive (AG). L'objectif de cette démarche est de démontrer que l'ordonnancement conjoint production /maintenance apporte un gain certain en terme de fiabilité des équipements, en contre partie d'une dégradation raisonnable de la performance.*

---

## 1. Introduction

La plus part des problèmes d'ordonnancement de production sont NP-complet [GAR79, CAR88]. L'ordonnancement conjoint de la production et la maintenance est à notre sens un problème complexe du fait de l'ordonnancement de deux activités différentes : la production et la maintenance préventive systématique avec des spécificités telle la périodicité des tâches de maintenance.

Les heuristiques et métaheuristiques de résolution des problèmes d'optimisation font l'objet de plusieurs recherches. Néanmoins le domaine de l'ordonnancement conjoint production/maintenance reste le parent pauvre de ces recherches. L'ensemble des travaux, que nous avons recensés dans le chapitre précédent, aborde le problème de manière très restreinte et se limitent, la plupart du temps, à une seule machine et une seule stratégie d'ordonnancement conjoint. Cette approche ne permet pas de faire un quelconque comparatif. L'approche que nous préconisons pour ce troisième chapitre consiste à conjuguer la classification des méthodes de résolution des problèmes d'ordonnancement et les stratégies de résolution de l'ordonnancement conjoint.

Nous présenterons donc, dans ce chapitre les deux stratégies d'ordonnancement conjoint production/maintenance : la stratégie séquentielle, qui consiste à ordonnancer les tâches de production et à prendre, dans un deuxième temps, cet ordonnancement comme une contrainte forte pour l'insertion des tâches de maintenance, et la stratégie intégrée, qui consiste à créer un ordonnancement simultanée des tâches de production et de maintenance. Nous déclinerons ces deux stratégies selon les trois approches de résolution suivantes (Fig. 3.1); où nous présenterons l'adaptation des heuristiques de chaque approche à notre problème pour chacune des deux stratégies d'ordonnancement conjoint :

- ➔ une approche constructive avec des heuristiques dédiées au *flowshop* telles Palmer, RA, GUPTA, CDS, NEH;
- ➔ une approche itérative avec la méthode de recherche Tabou ;
- ➔ une approche évolutive avec les algorithmes génétiques ;
- ➔ et enfin, nous proposerons quelques hybridations.

L'objectif de cette démarche est de démontrer que l'ordonnancement conjoint production /maintenance apporte un gain certain en terme de fiabilité des équipement en contre partie d'une dégradation raisonnable de la performance. La nécessité d'un compromis Performance/Sûreté de fonctionnement sera discutée dans le chapitre suivant.

En effet, nous pensons que la perte de performance due à l'insertion des tâches la maintenance dans l'ordonnancement de production est acceptable compte tenu du gain en sûreté de fonctionnement qu'apporte une maintenance préventive pour le système de production.

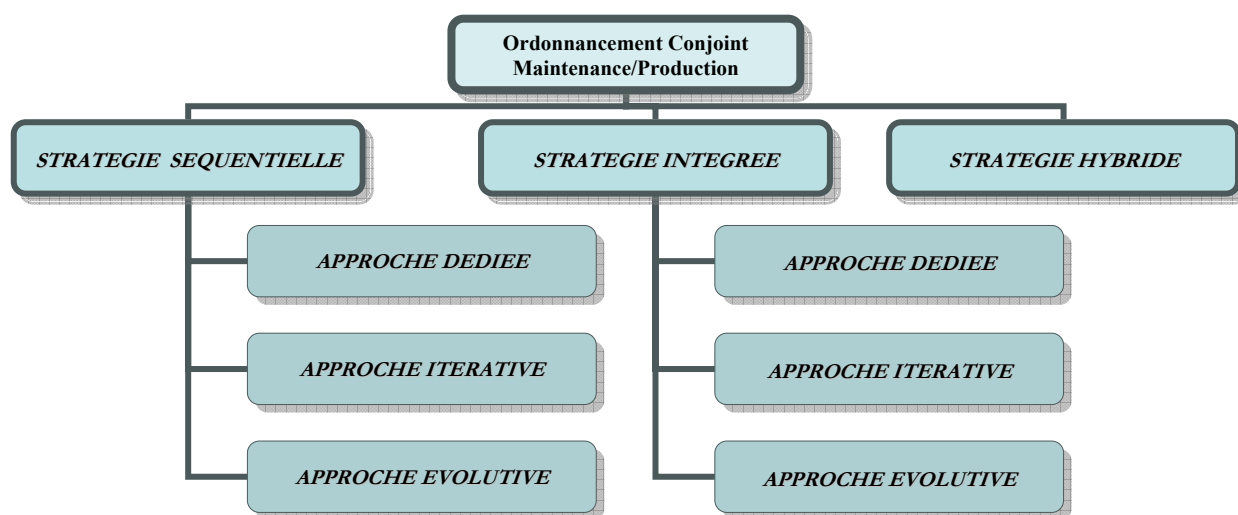


Figure 3.1 : Stratégies et approches de résolution.

Nous allons présenter dans ce chapitre l'adaptation des différentes heuristiques citées ci-dessus dans le cas de l'ordonnancement conjoint production/maintenance par les stratégies séquentielle et intégrée et comparer par la suite les différents résultats obtenus.

Ce chapitre est organisé comme suit : la première section est consacrée aux hypothèses, concernant la production et la maintenance, que nous avons formulées, dans le cadre de cette étude. Nous présentons dans les trois sections suivantes l'adaptation des heuristiques d'ordonnancement de la production en y intégrant les activités de maintenance préventive systématique, déclinée selon les deux stratégies d'ordonnancement conjoint production/maintenance : séquentielle et intégrée. A la fin de chaque section, une partie tests regroupe les différents résultats relatifs à l'approche étudiée. Nous présentons dans la dernière section une étude comparative des différentes approches.

## 2. Contexte de l'étude

Les données de notre problème sont les données relatives au *flowshop* de permutation, puis celles relatives à la maintenance préventive systématique, et enfin la fonction objectif à optimiser.

### 2.1. Les données de la production

Nous nous intéressons aux ateliers de type *flowshop* notés  $F_n / C_{\max}$  [BLA96]. Ce type d'atelier se rencontre très fréquemment en pratique. Nous nous restreindrons au cas du *flowshop* de permutation.

Soit  $m$  le nombre de machines dans la chaîne de production et  $n$  le nombre de lots à produire. Un lot est défini comme étant une séquence de  $m$  opérations élémentaires de production qui suivent l'ordre des machines. On notera par l'indice  $j$  une machine quelconque ( $1 \leq j \leq m$ ) et par l'indice  $k$  un lot quelconque ( $1 \leq k \leq n$ ). On notera également pour chaque lot  $k$ :

- $r_k$  : la date d'arrivée du lot  $k$  dans l'atelier.
- $d_k$  : la date de livraison au plus tard du lot  $k$ .
- $t_{kj}$  : la date de début d'exécution du lot  $k$  sur la machine  $j$ .
- $c_{kj}$  : la date de fin d'exécution du lot  $k$  sur la machine  $j$ .

On appellera  $C_{\max k} = \max(c_{kj})$ , la date de fin de la dernière opération du lot  $k$ .

- $p_{kj}$  : la durée opératoire du lot  $k$  sur la machine  $j$ .

### 2.2. Les données de la maintenance

Dans l'atelier *flowshop* nous considérons que chaque machine doit subir une ou plusieurs maintenances préventives systématiques déterministes dont les périodes sont données d'avance. Ces tâches de maintenance sont des interventions périodiques prévues toutes les  $T_j^*$  périodes ( $T_j^*$  dénote la périodicité optimale de la tâche de maintenance sur la machine  $j$ ).

Chaque tâche de maintenance préventive est caractérisée par une gamme de maintenance préétablie par le service maintenance ou par le constructeur de l'équipement considéré. Elles consistent en une suite d'opérations élémentaires dont la durée  $p_j$  est évaluée avec plus ou moins de certitude. Cette durée opératoire peut varier si le service de maintenance décide de ne pas exécuter toute la gamme de maintenance, à chaque occurrence de la tâche.

En pratique un certain écart (positif ou négatif) est toléré par rapport à la période idéale de maintenance  $T^*$ . Cette tolérance se matérialise par la mise en place d'un intervalle dit de tolérance autour de chaque période de maintenance, durant lesquels le coût de la maintenance, si elle est avancée ou retardée, est faible. Ces intervalles donneront plus de flexibilité à la planification de la maintenance en cas de besoin. La périodicité  $T_j^*$  de ces tâches est donc autorisée à varier dans un intervalle de tolérance noté  $[T_{\min_j}, T_{\max_j}]$ . Cet intervalle représente un compromis entre le coût de maintenance et le risque de perte de disponibilité de la machine. Si la période est inférieure à  $T_{\min_j}$  (Fig. 3.2-Zone 1), les interventions seront trop fréquentes, par rapport aux besoins réels de la machine en maintenance, et induisent ainsi un coût de maintenance trop élevé. Dans le cas contraire, c'est-à-dire une tâche de maintenance qui serait programmée après  $T_{\max_j}$  (Fig. 3.2-Zone 3), des pannes risquent d'apparaître et par conséquent les interventions de maintenance corrective aussi, ce qui est pénalisant en terme de disponibilité de la machine. Cet état de fait induit un accroissement du coût de la maintenance et une perte de productivité. Par contre, une intervention planifiée dans l'intervalle  $[T_{\min_j}, T_{\max_j}]$  va induire un coût de maintenance relativement constant, sachant que l'optimum est atteint à  $T_j^*$ .

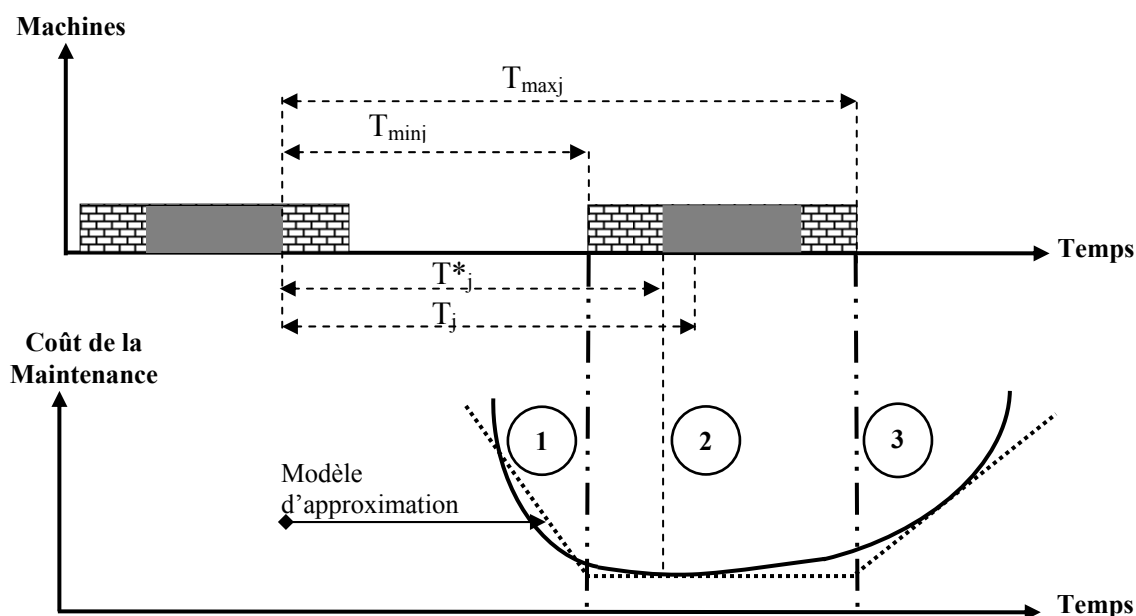


Figure 3.2: Intervalle de tolérance d'une tâche de maintenance.

On notera :

- $M_j$ : la tâche de maintenance  $i$  associée à la machine  $j$ . Une machine peut faire l'objet de plusieurs interventions différentes.  $M_j$  représente l'une de ces interventions qui sera répétée périodiquement.
- $T_j^*$ : la périodicité de la tâche  $M_j$ .
- $Tmin_j$ : la durée minimale séparant deux tâches de maintenances  $M_{jk}$  successives sur la machine  $j$ ;
- $Tmax_j$ : la durée maximale séparant deux tâches de maintenances  $M_{jk}$  successives sur la machine  $j$ ;
- $p'_j$ : la durée opératoire de la tâche  $M_j$ . Elle est supposée connue et constante.

Les données qui suivent concernent la  $k^{\text{ème}}$  occurrence de la tâche de maintenance  $M_j$ :

- $t'_{jk}$ : la date début d'exécution de la  $k^{\text{ème}}$  occurrence de la tâche  $M_j$ .
- $E'_{jk}$ : l'avance de la  $k^{\text{ème}}$  occurrence de  $M_{jk}$ .  $E'_{jk} = \max(0, (t'_{j-1k} + p'_{jk} + Tmin_{jk}) - t'_{jk})$
- $L'_{jk}$ : le retard de la  $k^{\text{ème}}$  occurrence de  $M_{jk}$ .  $L'_{jk} = \max(0, t'_{jk} - (t'_{j-1k} + p'_{jk} - Tmax_{jk}))$

L'intervalle de tolérance de la  $k^{\text{ième}}$  occurrence de la tâche de maintenance  $M_j$  sur la machine  $j$   $[I_{minjk}, I_{maxjk}]$  est déterminé de la manière suivante :

- $I_{minjk} = t'_{jk-1} + p'_{jk} + Tmin_j$ ;
- $I_{maxjk} = t'_{jk-1} + p'_{jk} - Tmax_j$ .

### 2.3. La fonction objectif

Le but est de proposer une méthode qui fournit un planning commun pour les tâches de production et de maintenance. L'objectif de l'optimisation consiste en un compromis entre la fonction objectif que l'on souhaite atteindre pour la production et la maintenance.

Les contraintes imposées par les clients à leurs fournisseurs s'expriment souvent en terme de délai, ce qui nous fait naturellement nous tourner vers la minimisation du temps total de fabrication (en anglais *makespan* ou  $C_{max}$ ). On notera  $f_1$  cette fonction objectif qui peut s'exprimer par :

$$f_1 = C_{max} = \text{Max}(c_{ij}) \quad (1)$$

Où  $c_{ij}$  est la date de fin d'exécution de la tâche  $i$  sur la machine  $j$

Du point de vue du fournisseur, le respect des contraintes impose un bon fonctionnement de son système de production. Ceci passe par le respect des périodes de maintenance prévues. On notera  $f_2$  cette fonction objectif qui peut s'exprimer par:

$$f_2 = \sum_{j=1}^m \sum_{k=1}^{kj} E'_{jk} + L'_{jk} \quad (2)$$

Avec

$E'_{jk}$  : l'avance de la  $k^{\text{ème}}$  occurrence de  $M_{jk}$ ;

$L'_{jk}$  : le retard de la  $k^{\text{ème}}$  occurrence de  $M_{jk}$ ;

$kj$  le nombre d'occurrences effectives de la tâche de maintenance  $M_j$ .

Pour optimiser les deux critères nous prenons en compte la fonction d'évaluation suivante :

$$f = \alpha f_1 + \beta f_2 \quad (3)$$

La linéarisation de la fonction objectif va permettre, dans une première étape, d'aborder le problème de manière simplifiée.  $\alpha$  et  $\beta$  sont des paramètres qui vont mesurer les contributions respectives de la production et de la maintenance dans la fonction objectif globale. Le but d'une telle démarche n'est pas de faire de l'optimisation multicritère mais uniquement de mesurer l'impact de la production ou de la maintenance sur la fonction objectif globale. Pour cela les paramètres  $\alpha$  et  $\beta$  peuvent dépendre du nombre de tâches, de la durée ou encore de la taille du problème traité. Ils sont indépendants et non complémentaires. Ils sont fixés par l'utilisateur suivant le degré d'importance qu'il donne à la production ou à la maintenance.

En référence aux approches de résolution en ordonnancement de production, trois approches de résolution sont présentées : une approche constructive mais dédiée au *flowshop* de permutation avec les heuristiques RA, Gupta, Palmer, CDS et NEH, une approche itérative avec la recherche Tabou et enfin une approche évolutive avec les AG. Nous déclinons chaque approche selon les deux stratégies d'ordonnancement conjoint : séquentielle et intégrée. Nous proposons par la suite une approche hybride.

### 3. La stratégie séquentielle

La résolution par la stratégie séquentielle se fait en deux étapes : l'étape « Ordonnancement de la production », pour laquelle nous n'apportons aucune adaptation puisque c'est une application directe des heuristiques d'ordonnancement choisies telles Palmer [PAL65], CDS [CAM70], Gupta [GUP71], RA [DAN77], NEH [NAW83], la recherche Tabou [GLO89] et

les AG [GOL89]. Puis l'étape « Insertion de la maintenance », où notre contribution est à situer à travers les heuristiques que nous proposons pour l'insertion de la maintenance. Du fait que nous traitons un atelier de type *flowshop* de permutation, nous proposerons d'abord des heuristiques pour le cas d'une seule machine telle l'heuristique de recherche en profondeur, puis pour le cas de plusieurs machines telles les heuristiques naïve, ascendante et descendante, en prenant l'ordonnancement de la production comme une contrainte forte du problème (Fig. 3.3). L'objectif étant d'optimiser la fonction objectif  $f$  qui prend en considération les critères de maintenance et de production en même temps (§ 3.2.3).

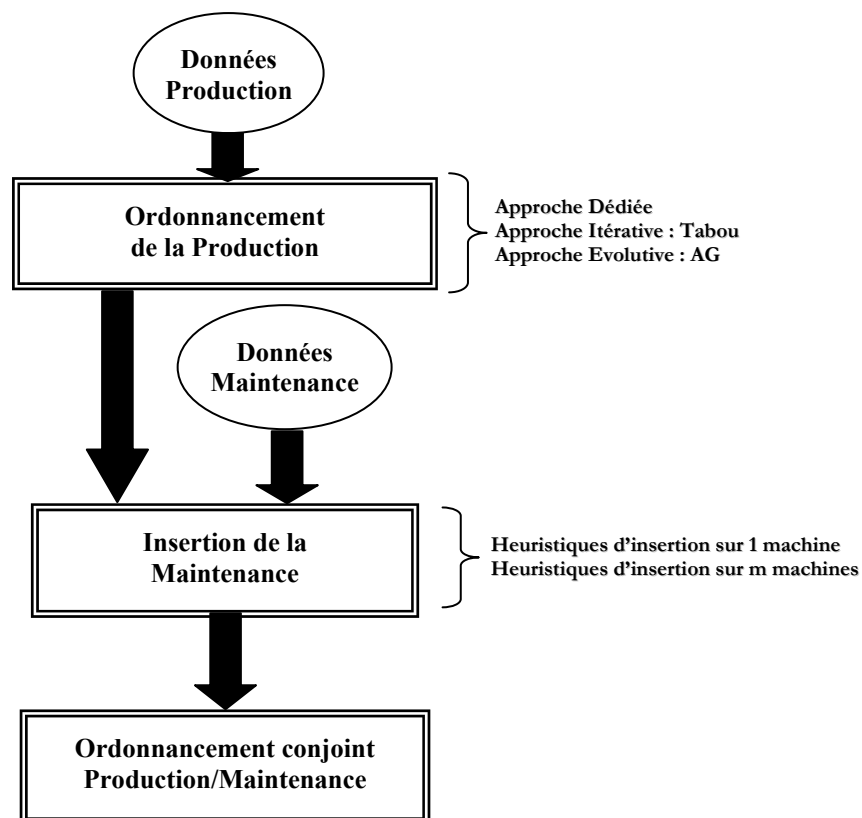


Figure 3.3: Ordonnancement conjoint Production/Maintenance par la stratégie séquentielle

### 3.1. Ordonnancement de la production

Nous allons décliner cette première étape, qui va consister en l'ordonnancement de la production, selon les trois approches de résolution. Nous présenterons uniquement le paramétrage que nous avons adopté pour les heuristiques choisies ci-dessus puisque aucune adaptation n'est nécessaire pour cette étape. Nous verrons dans la deuxième étape les différentes heuristiques utilisées lors de l'insertion de la maintenance dans la séquence de production obtenue à la fin de la première étape.



### 3.1.1. Représentation d'un ordonnancement

Comme notre atelier est de type *flowshop* de permutation, Le codage choisi pour la représentation d'un ordonnancement est le codage de permutation, où chaque solution est un vecteur (séquence) de tâches de taille  $n$  ( $n$  nombre de tâches).

Ce type de codage est parfaitement adapté à notre problème puisque une solution représente la séquence de passage de toutes les tâches de production, sur la totalité des machines.

On notera par  $S [i]=j$  la tâche numéro  $j$  qui sera exécutée au  $i^{\text{ème}}$  rang dans la séquence  $S$ .

Exemple

Pour 8 tâches de production, une solution peut être la séquence suivante :

Séquence S							
3	1	5	4	6	0	2	7

$S [2] = 1$  ;  $S [6] = 0$  ; etc.

### 3.1.2. Approche dédiée

De nombreuses contributions ont été consacrées à la conception d'algorithmes d'optimisation et d'heuristiques pour le problème du *flowshop* [FRE92]; la grande majorité d'entre elles utilisent des transformations mathématiques pour se ramener au problème de Johnson à deux machines [JOH54]. Nous présenterons dans cette partie les heuristiques de Palmer, CDS, de Gupta, RA et NEH. Elles sont présentées en détail en annexe A.

#### 3.1.2.1 Heuristique de Palmer

En 1965, Palmer [PAL65] a proposé un classement d'indice décroissant pour ordonner les tâches sur les machines en fonction des temps d'exécution. L'idée est de donner une priorité aux tâches ayant des temps d'exécution croissants dans leurs gammes opératoires.

#### 3.1.2.2 Heuristique CDS

En 1970, Campbell, Dudek et Smith [CAM70] ont proposé une méthode généralisant l'algorithme de Johnson. Son principe est de générer à partir du problème à  $m$  machines,  $m-1$  problèmes fictifs à deux machines équivalentes.

#### 3.1.2.3 Heuristique de Gupta

En 1971, Gupta [GUP71] a présenté une autre heuristique très similaire à celle de Palmer. Il s'est aperçu que la règle de Johnson donne une séquence optimale pour le problème à 3 machines en classant les tâches par ordre décroissant d'indices qu'il a définis.

#### **3.1.2.4 Heuristique RA**

En 1977, Dannebring [DAN77] a essayé de combiner les avantages de l'heuristique de Palmer et celle de Campbell et al [CAM70], en proposant une variante de l'heuristique CDS. Cette méthode est appelée procédure à accès rapide (Rapide Access procédure : RA). Il a proposé deux méthodes de post optimisation : la procédure à accès rapide avec recherche dans un voisinage proche (Rapid Access with Close order Search (RACS)), et la procédure à accès rapide avec recherche étendue (Rapid Access with Extensive Search (RAES)). Il a défini un voisin comme étant une nouvelle séquence obtenue par simple permutation de deux tâches successives.

#### **3.1.2.5 Heuristique NEH**

En 1983, Nawaz & al [NAW83] ont proposé un algorithme basé sur l'hypothèse qu'un lot ayant un temps total d'exécution élevé est prioritaire (le lot est positionné en priorité dans un ordonnancement partiel) par rapport à un lot dont le temps total d'exécution est plus faible dans le cas de la minimisation du  $C_{\max}$ .

### **3.1.3. Approche itérative : La recherche Tabou**

La recherche Tabou est une méthode itérative générale d'optimisation combinatoire introduite par Glover [GLO86] dans un cadre particulier et développée plus tard dans un contexte plus générale [GLO89]. La recherche Tabou se montre très performante sur un nombre considérable de problèmes d'optimisation combinatoire, en particulier les problèmes d'ordonnancement [LOP01].

Dans ce qui suit, nous présentons le paramétrage que nous avons utilisé pour notre problème. Nous présenterons, particulièrement, l'algorithme de génération d'une solution initiale au problème, puis la méthode de génération d'un voisinage pour une solution quelconque, la liste Tabou et enfin le critère d'arrêt retenu.

#### **3.1.3.1 Génération de la solution initiale**

Pour générer une solution initiale aléatoire de taille  $n$ , on l'initialise d'abord avec la séquence canonique  $\{0, 1, 2, \dots, n\}$ . Ensuite, on effectue des mutations sur cette solution. Une mutation consiste à générer aléatoirement deux positions puis à les permuter (Algorithme 3.1).

---

**Algorithme** Génération Aléatoire d'une Solution

---

**Variante locale**  $S$  : un vecteur de taille  $n$  ;  $nbrem$  : nombre de mutations,  $nbrem \in [1..n/2]$

**Début**

**Pour**  $i=0$  à  $n-1$

**Faire**  $S[i] \leftarrow i$  ;

**Fin Pour**

Générer Aléatoirement un nombre de mutation ( $nbrem$ ) ;

**Pour**  $k=0$  à  $nbrem$

**Faire** Générer Aléatoirement deux positions  $P_1$  et  $P_2$  /  $P_1, P_2 \in [0..n-1]$  et  $P_1 \neq P_2$  ;

Permuter  $S[P_1]$  et  $S[P_2]$

**Fin Pour**

**Fin.**

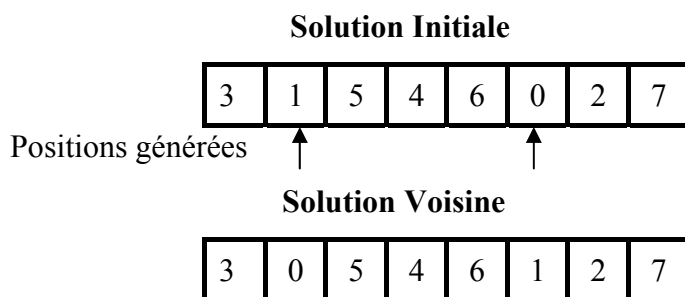
---

Algorithme 3.1: Génération aléatoire d'une solution.

### 3.1.3.2 Génération du voisinage : Voisinage par permutation

Dans ce type de voisinage, on génère deux positions aléatoirement dans la séquence de production et on inverse l'ordre des tâches de ces deux positions.

Exemple :



### 3.1.3.3 Liste Tabou

La liste Tabou représente un élément fondamental de la recherche Tabou. Elle permet de garder les solutions rencontrées au cours de la recherche. Cette liste permet d'éviter les blocages dans les minima locaux, en interdisant de passer, à nouveau, sur des configurations de l'espace de recherche, précédemment visitées, mais uniquement pour un laps de temps (mémoire à court terme). La taille de la liste est maintenue tout au long de l'évolution de la méthode, c'est-à-dire si sa taille est atteinte alors qu'une bonne solution se présente, on

élimine la solution la plus ancienne de la liste, pour l'y insérer. La taille de la liste dépend des performances de la recherche Tabou. Si elle est trop petite, elle engendre beaucoup de cycles. De même, si elle est trop grande, elle engendre un phénomène de dispersion de la recherche. Pour construire la liste Tabou deux techniques sont utilisées : soit on sauvegarde uniquement les mouvements qu'on s'interdit de reproduire pendant un certain nombre d'itérations, soit on sauvegarde la solution qu'on s'interdit de visiter pendant un certain nombre d'itérations. Nous avons opté pour la seconde technique.

### 3.1.3.4 Critère d'arrêt

La recherche s'arrête lorsqu'un certain nombre d'itérations (qui est une donnée du problème et connu à l'avance) ou bien lorsqu'on n'observe aucune amélioration pendant un certain nombre d'itérations (stagnation).

Ces étapes sont décrites dans l'algorithme 3.2 :

---

#### **Algorithme** Méthode de recherche Tabou pour l'ordonnancement de production

---

**Variable locale** S : Solution courante, S\* : Meilleure solution, S', S'' : Solutions intermédiaires ; f : Fonction objectif; NbreIter : Nombre d'itérations ;

**Début**

Générer aléatoirement une solution initiale S ;

Insérer S dans la liste Tabou ;

S\* ← S;

NbreIter ← 1;

**Tant que** le critère d'arrêt n'est pas atteint

**Faire** Générer un voisinage N(S) pour S;

S' ← meilleure solution de N(S);

**Si** f(S') < f(S)

**Alors** S ← S';

S\* ← S;

Insérer S' dans la liste Tabou ;

**Sinon** Tirer aléatoirement S'' de la liste Tabou ;

S ← S'';

Incrémenter le paramètre de stagnation ;

**Fin Si**

NbreIter ← NbreIter +1 ;

**Fin Tant que**

**Fin.**

---

Algorithme 3.2: Méthode de recherche Tabou

### 3.1.4. Approche évolutive : un algorithme génétique

Les problèmes de *flowshop* sont pour la plupart des problèmes difficiles à résoudre et les méthodes exactes ne parviennent pas à résoudre de grandes instances [ESP98]. Nous proposons de nous tourner vers les Algorithmes Génétiques (AG) qui ont déjà fait leurs preuves pour la résolution du problème de l'ordonnancement de production. Le problème qui nous intéresse est l'optimisation conjointe de la durée totale de production et du retard de la maintenance.

Nous ne rappellerons pas ici le mode de fonctionnement des AG cependant le lecteur pourra se référer aux travaux de Goldberg [GOL89]. Nous présentons uniquement le paramétrage que nous avons choisi pour le problème de l'ordonnancement conjoint production/maintenance par la stratégie séquentielle. Le codage choisi pour la représentation d'un individu est le codage de permutation (§3.3.1.1)

#### 3.1.4.1 Génération de la population initiale

La population initiale est formée d'un ensemble d'individus qui représentent des séquences de production, elle est générée comme suit (Algorithme 3.3):

---

##### Algorithme Génération Aléatoire

---

**Début**

**Pour**  $i=1$  jusqu'à la taille de la population demandée

**Faire** Générer un individu valide aléatoirement  
Appliquer des mutations sur cet individu

**Fin Pour**

**Fin.**

---

Algorithme 3.3: Génération aléatoire de la population initiale

#### 3.1.4.2 Fonction d'adaptation

Si l'on a une fonction à maximiser, la force d'un individu (valeur d'adaptation) peut être mesurée par une valeur qui varie dans le même sens que la fonction objectif du problème considéré, et dans le sens inverse s'il s'agit d'une minimisation [LOP01]. Comme nous traitons un problème de minimisation (minimiser le Makespan), nous avons choisi la fonction suivante :  $1 / (1 + \text{fonction objectif})$ . Le dénominateur est égale à  $(1 + \text{fonction objectif})$  pour éviter une division par zéro, dans le cas où la fonction objectif est nulle.

Notre choix de cette fonction et non d'une fonction linéaire (Max + 1 – fonction objectif) est motivée par le soucis de privilégier les meilleurs individus, lors de la phase de sélection, par rapport aux plus mauvais individus.

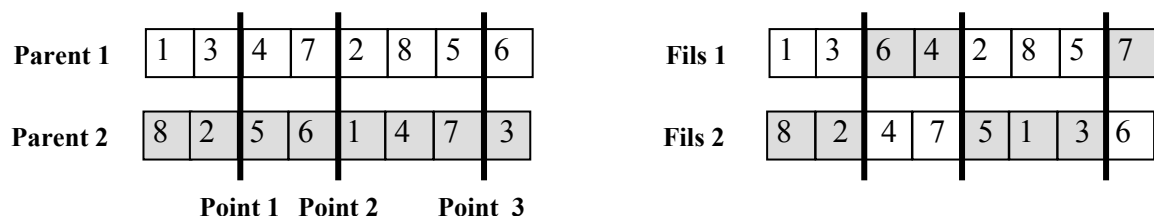
### 3.1.4.3 Sélection d'un individu

La sélection permet d'identifier les meilleurs individus et de les dupliquer proportionnellement à leurs valeurs d'adaptation, et de supprimer les plus mauvais. On trouve dans la littérature plusieurs opérateurs de sélection [GOL89], nous avons utilisé la roue de loterie qui fournit le rang de l'individu à sélectionner dans la population ainsi que la sélection par rang et par tournoi [DAV91].

### 3.1.4.4 Croisement

Il s'agit de l'opérateur principal d'un AG. A partir de 2 individus l'objectif est de les combiner pour en fabriquer un (ou deux) autre(s) intégrant les caractéristiques des individus de départ. Le type de croisement choisi est un croisement à k points [POR96]. Le principe est illustré sur l'exemple suivant et donné par l'algorithme 3.4.

Exemple :




---

#### Algorithme Croisement à k points

---

##### Début

Générer k points de croisement aléatoire (ceci fournit k+1 segments dans chaque parent) ;  
 Recopier les parties impaires (resp. paires) du premier parent dans le premier fils (resp. deuxième fils);

Compléter les segments manquants par les tâches manquantes dans leur ordre d'apparition dans le second parent.

##### Fin.

---

Algorithme 3.4 : Croisement à k points

### 3.1.4.5 Mutation

L'opérateur de mutation consiste à permuter deux positions choisies aléatoirement dans l'individu. On génère pour chaque individu un nombre aléatoirement  $r$  dans l'intervalle  $[0,1]$  s'il vérifie  $r < t_m$  alors l'individu sera muté (avec  $t_m$  taux de mutation). Ce taux contrôle le nombre de nouveaux individus introduits dans la population).

### 3.1.4.6 Remplacement

Le remplacement se fait, soit, entre la population sélectionnée et la population croisée, soit entre la population sélectionnée et la population mutée ou sans remplacement. Nous avons implémenté les stratégies suivantes:

- ***N\_meilleur*** : on choisit les  $N$  meilleurs individus entre la population courante et la population intermédiaire pour former la nouvelle population.
- ***K\_mauvais*** : on remplace les  $K$  mauvais individus de la population courante par les  $K$  meilleurs de la population intermédiaire.
- ***Sans remplacement*** : la population courante est complètement remplacée par les nouveaux individus.
- ***Aléatoire*** : on choisit aléatoirement  $N$  individus entre la population courante et la population intermédiaire pour former la nouvelle population.

### 3.1.4.7 Evaluation

Au niveau de la phase de remplacement, des individus de la population courante vont être remplacés par des individus de la population intermédiaire. Pour cela, on aura besoin des valeurs d'adaptation de ces derniers. La fonction d'adaptation retenue est  $1/(1+\text{fonction objectif})$  pour privilégier les meilleurs individus, lors de la phase de sélection, par rapport aux plus mauvais individus.

Il serait intéressant d'effectuer une évaluation uniquement sur les nouveaux individus qui seront choisis pour le remplacement (selon la stratégie de remplacement). Une amélioration est effectuée à ce niveau, elle consiste à changer le moment de l'évaluation des individus selon la stratégie de remplacement utilisée, comme il est indiqué ci dessous:

- ***Remplacement entre la population sélectionnée et la population croisée*** : effectuer l'évaluation sur les individus de la population croisée, et sur les individus de la population mutée non encore évalués.

- **Remplacement entre la population sélectionnée et la population mutée** : l'évaluation se fait uniquement sur les individus de la population mutée.
- **Sans remplacement** : dans cette stratégie, tous les individus de la population mutée seront injectés dans la nouvelle population. Donc l'évaluation va se faire pour ces nouveaux individus avant la phase de sélection pour la génération de la prochaine population.

#### 3.1.4.8 Critère d'arrêt

Le critère d'arrêt peut être soit le nombre de générations qui est définie par l'utilisateur, soit la stagnation de la meilleure solution après un certain nombre de générations.

Nous avons présenté dans la section précédente le paramétrage que nous avons utilisé pour les AG dans la première phase de la stratégie séquentielle. Nous avons utilisé certains paramètres de la littérature, et nous avons proposé certains autres avec de multiples possibilités au choix.

Dans ce qui suit, nous allons présenter la deuxième phase de cette stratégie, qui consiste en les différentes heuristiques pour l'insertion des tâches de maintenance tout d'abord sur une machine, puis sur plusieurs machines.

### 3.2. Insertion des tâches de maintenance

Comme nous traitons un problème de type *flowshop* de permutation, l'insertion des tâches de maintenance se fera d'abord sur une machine, ensuite sur la totalité des machines. Pour le cas à une machine nous présenterons deux heuristiques : l'heuristique de base proposée par Kaabi & al [KAA02] et l'heuristique de recherche en profondeur que nous avons proposée. Pour le cas à plusieurs machines nous présenterons les heuristiques que nous avons proposées telles l'heuristique naïve et les heuristiques ascendante et descendante.

#### 3.2.1. Résolution du problème à une machine

Plusieurs heuristiques ont été proposées pour l'insertion des tâches de maintenance lors de la résolution du problème d'ordonnancement conjoint de la production et de la maintenance par la stratégie séquentielle pour le cas à une seule machine. Nous présentons dans ce qui suit les heuristiques de base et de recherche en profondeur. L'heuristique de base consiste à insérer une tâche de maintenance strictement à  $T^*$ , à  $T_{\min}$  ou encore à  $T_{\max}$ . Par contre l'heuristique de



recherche en profondeur va rechercher le meilleur emplacement, dans l'intervalle de tolérance, pour y insérer la tâche de maintenance.

Pour rappel l'intervalle de tolérance  $[T_{\min}, T_{\max}]$  d'une tâche de maintenance représente l'écart (positif ou négatif) permis pour l'exécution de cette tâche de maintenance sachant que la date idéale d'exécution est atteinte à  $T^*$  avec  $T^* \in [T_{\min}, T_{\max}]$  (section 3.2.2).

### 3.2.1.1 Heuristique de base

L'heuristique de base (HB) consiste à faire ce que le service de maintenance fait manuellement. Il s'agit de programmer systématiquement les tâches de maintenance à leur période optimale  $T^*$  (période idéale pour l'insertion de la tâche de maintenance),  $T_{\min}$  ou  $T_{\max}$  (respectivement période minimale et maximale pour l'insertion de la tâche de maintenance). Si une tâche de production est programmée à la date prévue d'exécution de la tâche de maintenance, celle-ci sera reportée et ne commencera qu'à la fin de la tâche de production. On notera  $HB^*$  (respectivement  $HB_{\min}$  et  $HB_{\max}$ ) l'heuristique d'insertion des tâches de maintenance à la période  $T^*$  (respectivement à la période  $T_{\min}$  et  $T_{\max}$ ).

Comme les tâches de maintenance sont des tâches répétitives et séquentielles alors le calcul de la date d'insertion de la première tâche de maintenance se fera à partir de la date 0 (Fig. 3.4-1). Si à cette date  $t$  ( $t = 0 + \text{Période}$ ) l'insertion de la tâche de maintenance est possible (aucune tâche de production n'est programmée à cette date) alors la tâche sera insérée et une mise à jour des données de production est effectuée (Technique du décalage à droite), sinon (une tâche de production est programmée à cette date) elle sera retardée jusqu'à la fin d'exécution de la tâche de production (Fig. 3.4-2). Le calcul de la date d'insertion de la tâche de maintenance qui suit se fera à la date  $t'$  ( $t' = t + \text{durée opératoire de la tâche} + \text{Période}$ ). De même que pour la première, si à cette date l'insertion de la tâche de maintenance est possible alors la tâche sera insérée et une mise à jour des données de production est effectuée sinon elle sera retardée jusqu'à la fin d'exécution de la tâche de production (Fig. 3.4-3). Le processus sera répété jusqu'à ce qu'il n'y ait plus de tâche de maintenance à insérer.

Une amélioration de cette heuristique consiste à examiner chaque tâche de maintenance séparément. S'il y a conflit avec une tâche de production, on teste les emplacements avant et après la tâche de production, mais toujours en restant contraint par son intervalle de tolérance. Pour chacun de ces « deux » emplacements, on évalue l'ordonnement obtenu et on retient le meilleur. Le même processus est répété pour toutes les autres tâches de maintenance.

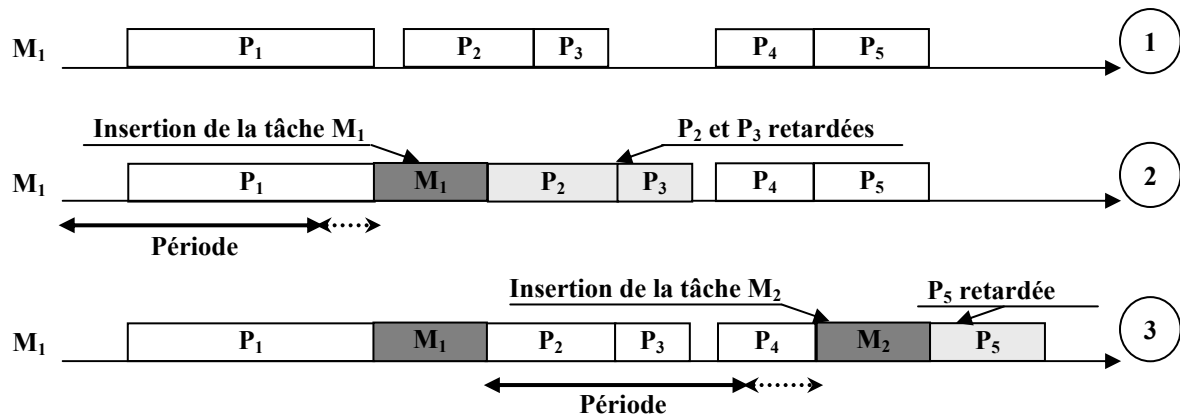


Figure 3.4 : Insertion de la maintenance sur une machine par l'heuristique HB\*

### 3.2.1.2 Heuristique de recherche en profondeur

L'heuristique de Recherche en Profondeur (HRP) tient compte de l'intervalle de tolérance (l'intervalle sur lequel une tâche de maintenance peut s'exécuter). Une tâche de maintenance peut être insérée sur tous les emplacements possibles de son intervalle de tolérance :

- à la fin de chaque tâche qui se trouve à l'intérieur de l'intervalle.
- au début de l'intervalle s'il n'existe aucune tâche qui s'exécute à cet instant.

L'insertion d'une tâche de maintenance peut se faire selon les deux cas suivants :

- L'avance et le retard des tâches de maintenance ne sont pas tolérés. Les tâches de maintenance sont insérées strictement dans leur intervalle de tolérance. C'est-à-dire toujours après  $T_{\min}$  et avant  $T_{\max}$ . Dans ce cas de figure le retard des tâches de maintenance est nul et donc  $f$  est égale à  $f_1$  (Fig. 3.5).
- L'avance et le retard des tâches de maintenance sont tolérés. L'insertion d'une tâche de maintenance peut se faire avant  $T_{\min}$  et après  $T_{\max}$  (Fig. 3.6).

Pour chaque emplacement possible d'une tâche de maintenance dans son intervalle de tolérance, on évalue l'ordonnancement obtenu par une fonction nommée  $g$ . Cette fonction donne une évaluation partielle de l'ordonnancement après insertion de cette tâche. Le choix du meilleur emplacement est celui qui optimise la fonction partielle  $g$ .

Dans le cas de la figure 3.5, trois emplacements sont possibles pour la première tâche de maintenance  $M_1$ , dans son intervalle de tolérance  $[T_{\min}, T_{\max}]$ . Soient  $g_1, g_2$  et  $g_3$  les trois fonctions partielles associées aux trois emplacements tel que  $g_3 < g_2 < g_1$ . Le meilleur

emplacement est donc le troisième ( $g_3$ ), en gardant l'emplacement de  $M_1$ . A partir de ce point, on cherche de la même façon le meilleur emplacement pour la tâche  $M_2$ . Le choix du meilleur emplacement est celui qui optimise la fonction  $f$ .

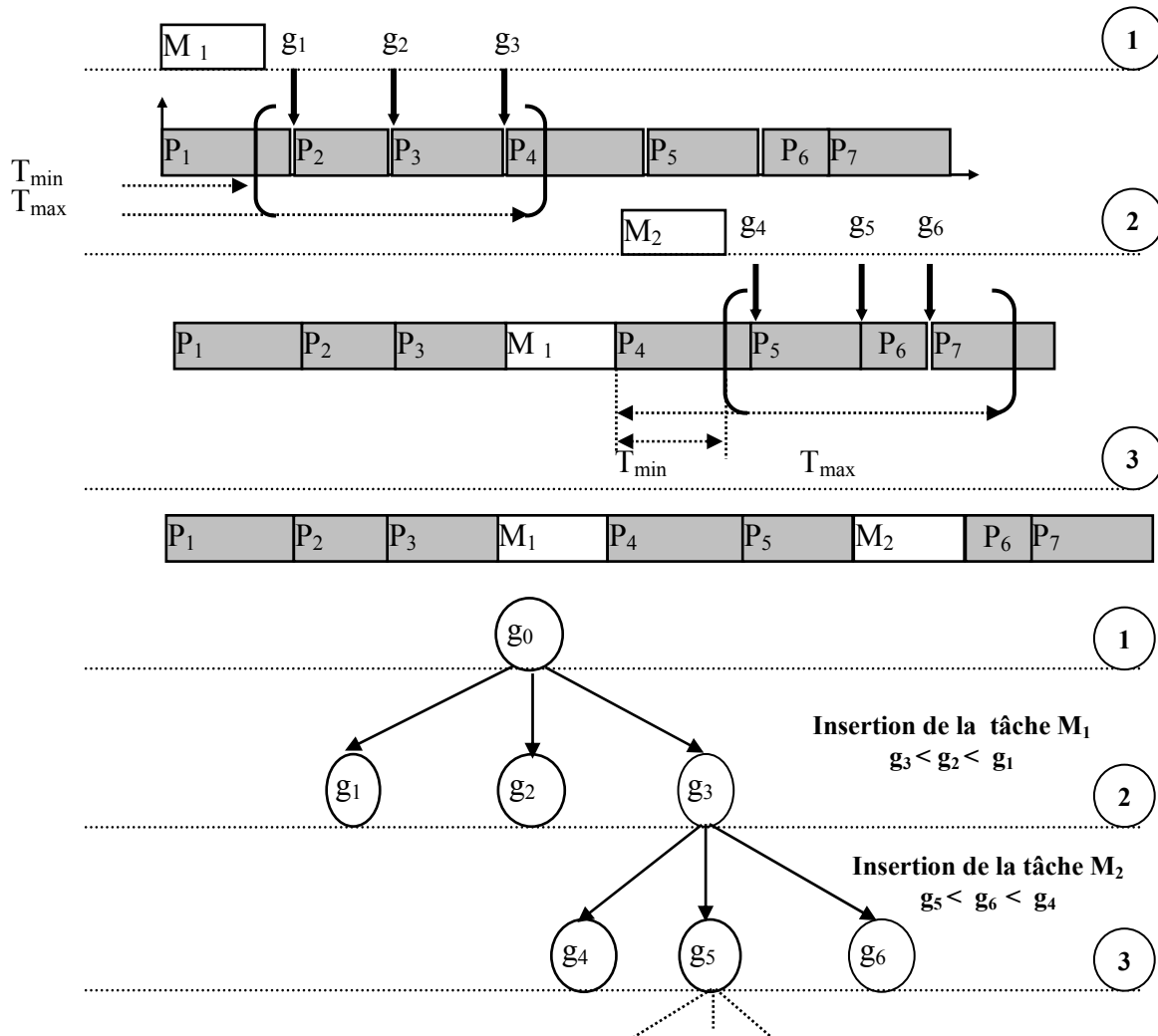


Figure 3.5 : Insertion de la maintenance par HRP : Avances/Retards non tolérés

Avec

$P_i$  : tâche de production.

$M_j$  :  $j^{\text{ième}}$  tâche de maintenance.

$g_i$  : fonction partielle  $i$  associée à l'emplacement possible  $i$  de la tâche de maintenance  $M_j$ .

Dans le cas où l'avance et le retard sont tolérés, trois cas peuvent se présenter :

**a) A l'intérieur de l'intervalle de tolérance**

Elle se fait de la même façon que celle du premier cas. La pénalité est nulle.

**b) Avant  $T_{\min}$  (tâche avancée)**

L'insertion de la  $i^{\text{ième}}$  tâche de maintenance sur une machine  $j$  ( $M_{ij}$ ) avant  $t'_{ij}$ , sa date de début au plus tôt, est possible si et seulement son avance ne dépasse pas une durée fixée qui est égale à un pourcentage de  $T_{\min j}$  :  $I_{\min ij} - t'_{ij} \leq k \times T_{\min j}$  ( $0 < k < 1$ ). Tel que :

- $t'_{ij}$  : l'instant d'insertion de la tâche  $M_{ij}$
- $I_{\min ij}$  : sa date de début au plus tôt.
- $T_{\min j}$  : la durée minimale séparant deux tâches de maintenance consécutives.

Une pénalité est imposée pour éviter d'avoir des tâches de maintenance trop rapprochées, c'est-à-dire que la durée minimale qui les sépare est inférieure à  $T_{\min j}$ . Cette pénalité est d'autant plus importante dans le cas où l'avance de ces tâches n'est pas aussi importante que la pénalité sur le retard des tâches de production. Cette pénalité d'avance est égale à  $\delta_j \times (I_{\min ij} - t'_{ij})$  ( $0 < \delta_j < 1$ ).

$\delta_j$  est un coefficient qui représente le taux de relaxation de la contrainte sur la durée minimale séparant deux tâches de maintenance sur la même machine. Il permet de tester un emplacement, au plus, avant  $T_{\min j}$ .

**c) Après  $T_{\max}$  (tâche retardée)**

Une tâche de maintenance  $M_{ij}$ , peut être retardée sur une machine  $j$ , si ce retard ne dépasse pas une certaine marge égal à  $(k \times T_{\max j})$ , c'est à dire  $t'_{ij} - I_{\max ij} \leq k \times T_{\max j}$  ( $0 < k < 1$ ). Tel que :

- $t'_{ij}$  : l'instant d'insertion de la tâche  $M_{ij}$ .
- $I_{\max ij}$  : sa date de fin au plus tard.
- $T_{\max j}$  : la durée maximale séparant deux tâches de maintenance consécutives.

Cette notion de marge garantit qu'aucune tâche de maintenance ne manquera, c'est-à-dire que dans chaque intervalle  $[T_{\min}, T_{\max}]$  il doit y avoir une tâche de maintenance. En effet, si la pénalité de retard d'une tâche de maintenance n'est pas aussi importante que la pénalité de retard d'une tâche de production, on risque facilement de sauter des tâches de maintenance. C'est-à-dire que sur un intervalle de temps  $T$ , aucune tâche de maintenance n'est programmée. De la même manière que précédemment, le retard d'une tâche de maintenance insérée après sa date de fin au plus tard sera pénalisée par  $\lambda_j \times (t'_{ij} - I_{\max ij})$ .

Le coefficient  $\lambda_j$  représente le taux de relaxation de la contrainte sur la durée maximale séparant deux tâches de maintenance, sur la même machine. Il permet de tester un emplacement, au plus, après  $T_{\max j}$ .

Dans le cas de la figure 3.6, en plus des emplacements possibles pour la première tâche de maintenance  $M_1$ , dans son intervalle de tolérance  $[T_{\min}, T_{\max}]$ , deux autres emplacements sont permis : avant  $P_2$  et après  $P_3$ . Soient  $g_1$ ,  $g_2$  et  $g_3$  les trois fonctions partielles associées aux trois emplacements tel que  $g_2 < g_1 < g_3$ . Le meilleur emplacement est donc le deuxième ( $g_2$ ). En fixant l'emplacement de  $M_1$  et à partir de ce point, on cherche de la même façon le meilleur emplacement pour la tâche  $M_2$ . Le choix du meilleur emplacement est celui qui optimise la fonction  $f$ .

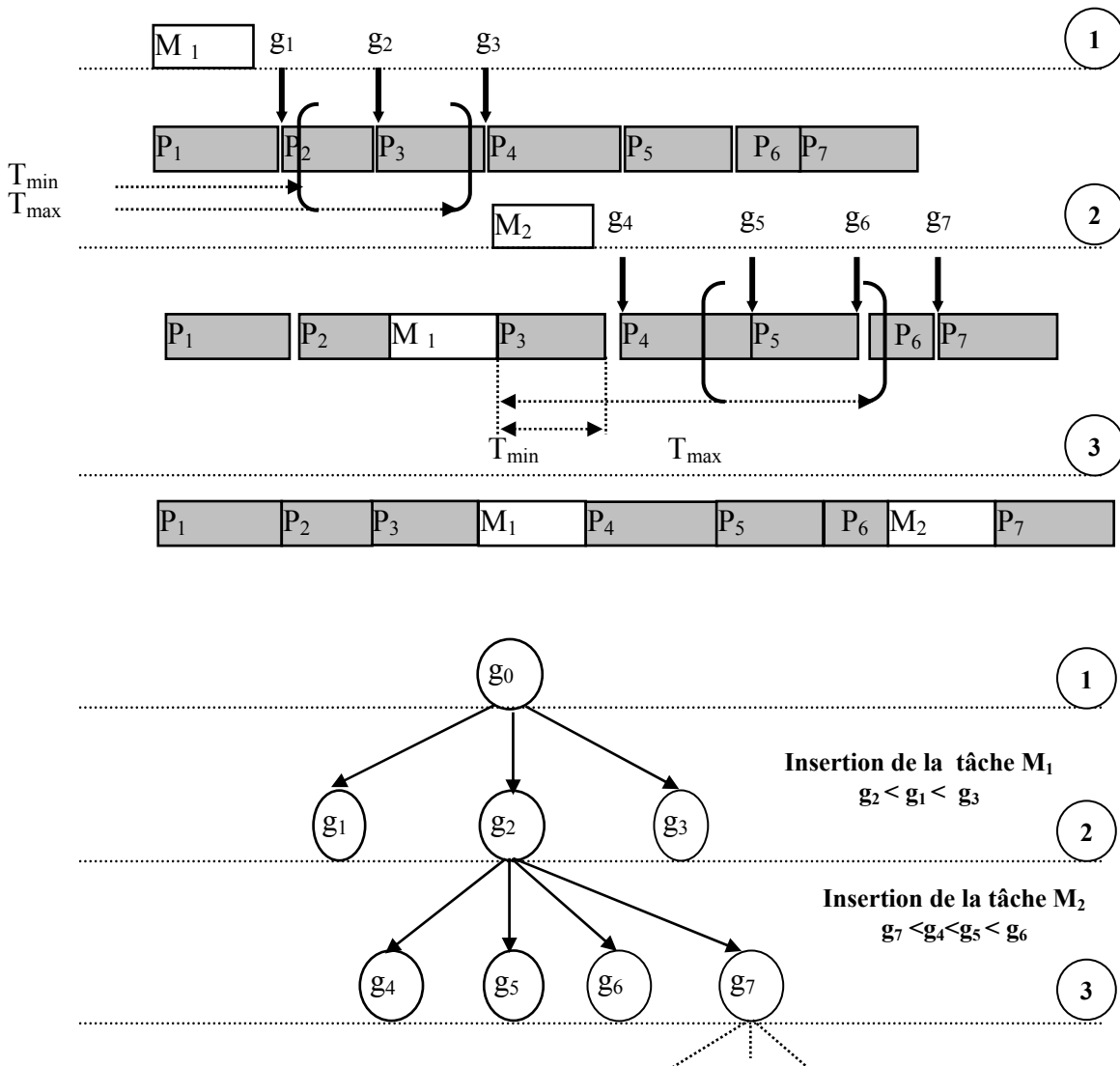


Figure 3.6: Insertion de la maintenance par HRP : Avances/Retards tolérés

Avec

$P_i$  : tâche de production.

$M_j$  :  $j^{\text{ième}}$  tâche de maintenance.

$g_i$  : fonction partielle  $i$  associée à emplacement possible  $i$  de la tâche de maintenance  $M_j$ .

### 3.2.2. Résolution du problème à plusieurs machines

L'insertion des tâches de maintenance sur la totalité des machines se fera selon trois heuristiques : Naïve, Ascendante et Descendante. Le principe général de ces heuristiques est dépendant de l'ordre d'insertion des tâches sur l'ensemble des machines. En effet, selon que l'insertion des tâches se fait de la première à la dernière machine ou inversement, certaines tâches risqueraient d'être décalées au-delà de leurs intervalles de tolérance. Ces trois heuristiques sont présentées ci-dessus.

#### 3.2.2.1 Heuristique naïve

Le principe de l'Heuristique Naïve (HN) est d'insérer toutes les tâches de maintenance sur toutes les machines en allant de la première jusqu'à la dernière tâche. L'insertion des tâches de maintenance sur une machine se fait selon l'heuristique de base, présentée dans la section 3.2.1.1. Cette heuristique permet d'insérer toutes les tâches de maintenances sur toutes les machines avec risque d'en retarder quelques unes. En effet, l'insertion de toutes les tâches de maintenance sur la première machine puis sur la suivante et le décalage des tâches de production qui en découle risquerait de retarder certaines tâches de maintenance. Pour remédier à ce problème, on parcourt l'ordonnancement pour replacer les tâches de maintenance dans leurs intervalles de tolérance par permutation avec les tâches de production. De même que pour l'heuristique de base (HB), on notera :  $HN^*$  (respectivement  $HB_{\min}$  et  $HB_{\max}$ ) l'heuristique d'insertion des tâches de maintenance, sur une machine, à la période  $T^*$  (respectivement à la période  $T_{\min}$  et  $T_{\max}$ ).

#### 3.2.2.2 Heuristique ascendante

Le principe de l'Heuristique Ascendante (HA) (Fig. 3.7-1) est d'insérer les tâches de maintenance de la première machine jusqu'à la dernière : on insère d'abord toutes les tâches de maintenance sur la première machine (Fig. 3.7-2), ensuite toutes les tâches de maintenance sur la deuxième (Fig. 3.7-3), jusqu'à la dernière machine.

L'insertion des tâches de maintenance, sur chaque machine, implique bien sûr la mise à jour (après chaque insertion si besoin) des données de production par application de la technique du décalage à droite.

L'insertion des tâches de maintenance sur une machine se fait selon l'heuristique HRP développée dans la section 3.2.1.2. En appliquant cette heuristique, chaque tâche de maintenance insérée ne sera plus décalée, puisqu'on commence de la première machine à la

dernière, et de la première tâche à la dernière tâche de maintenance, sur chaque machine. Par conséquent, toutes les tâches de maintenance seront insérées sans qu'il n'y ait de tâches manquantes. On dit qu'une tâche de maintenance est manquante sur une machine  $j$ , si la durée relative (on recommence à la date 0 après insertion de chaque tâche de maintenance) qui séparent deux tâches de maintenance successives est supérieure à  $T_{maxj} + p'_j$ .

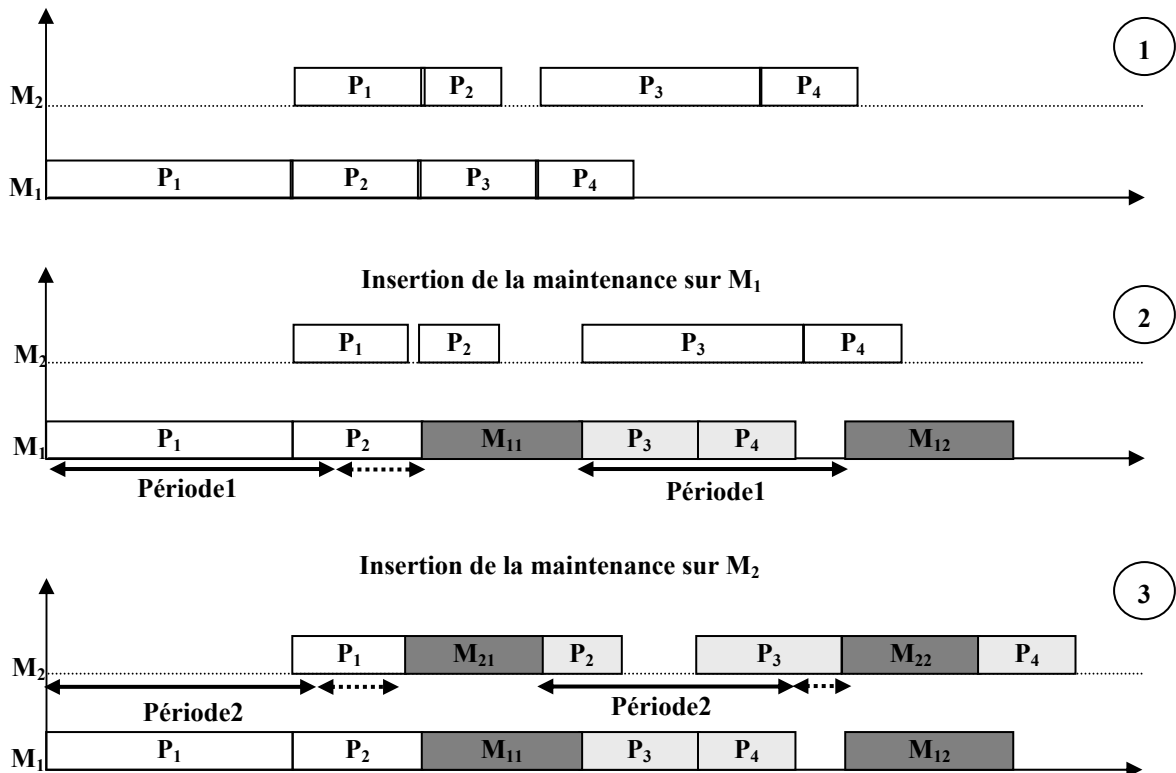


Figure 3.7 : Principe d'insertion de la maintenance par de l'heuristique HA

### 3.2.2.3 Heuristique Descendante

Le principe de l'Heuristique Descendante (HD) (Fig. 3.8-1) est d'insérer les tâches de maintenance de la dernière machine à la première : on insère les tâches de maintenance d'abord sur la dernière machine (Fig. 3.8-2), ensuite sur l'avant dernière jusqu'à la première machine. L'insertion des tâches de maintenance sur une machine se fait selon l'heuristique HRP développée dans la section 3.2.1.2.

L'insertion des tâches de maintenance, sur chaque machine, implique bien sûr la mise à jour (après chaque insertion si besoin) des données de production par application de la technique du décalage à droite. Cependant, contrairement à l'heuristique HA, une tâche de maintenance sur une machine  $M_j$  peut être décalée après l'insertion des autres tâches sur les machines

précédentes ( $M_i < M_j$ ), si cela optimise la fonction objectif. Par conséquent, il peut y avoir des tâches manquantes sur certaines machines (Fig. 3.8-3). Pour remédier à ce problème, après l'insertion de chaque tâche de maintenance, on parcourt toutes les tâches de maintenance qui sont sur les machines suivantes, et on vérifie s'il n'y a pas une tâche qui est trop éloignée de celle qui la précède (tâche manquante). Si ce cas se présente, on effectue des permutations (Fig. 3.8-4) avec les tâches de production qui la précèdent jusqu'à ce qu'elle soit suffisamment proche de la tâche de maintenance qui la précède (ceci consiste à la replacer dans son intervalle de tolérance). Sur une machine  $j$ , on dit qu'une tâche de maintenance est suffisamment proche de celle qui la précède si la durée relative (on recommence à la date 0 après insertion de chaque tâche de maintenance) qui les sépare est inférieur à  $T_{maxj} + p'_j$ .

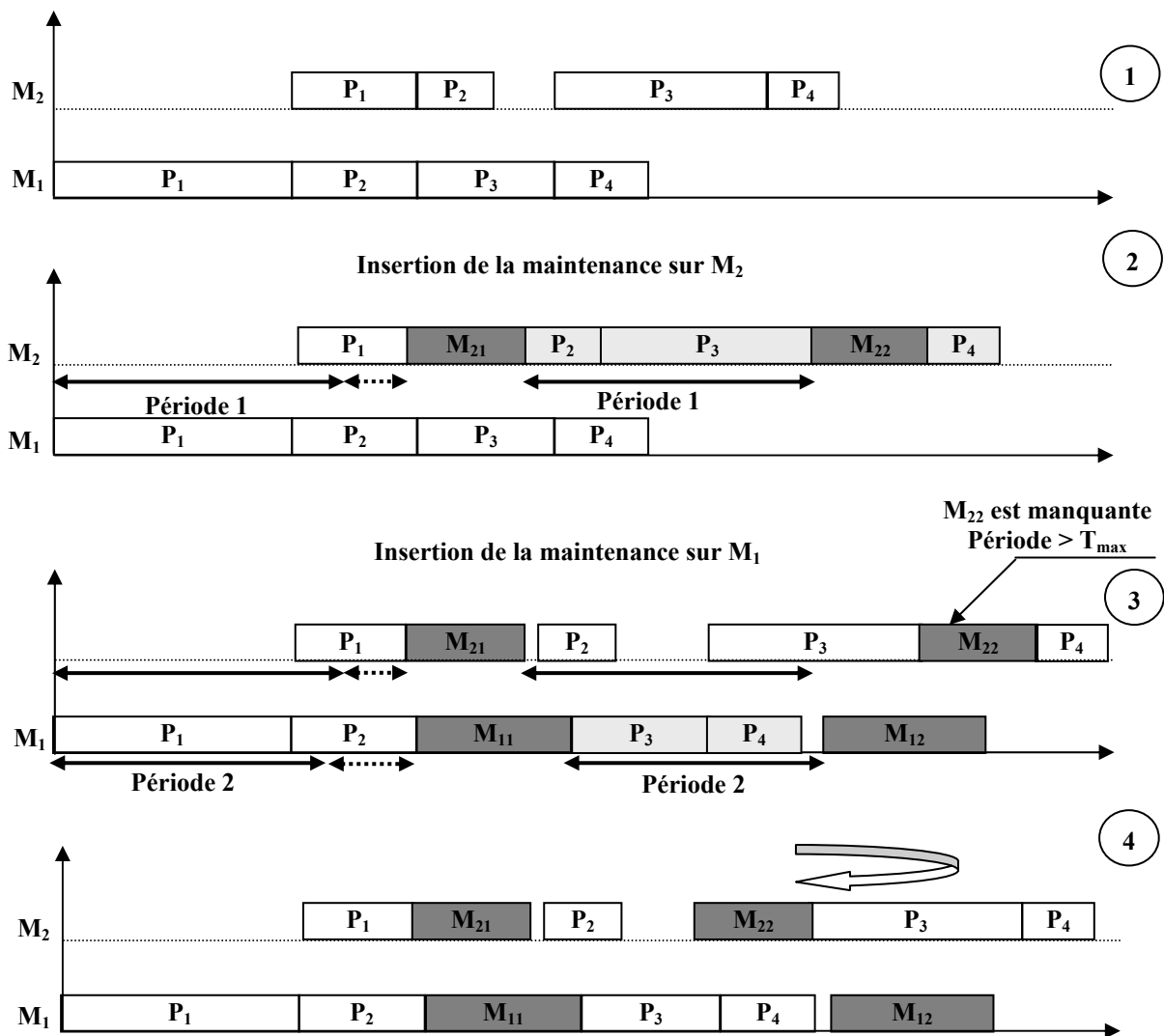


Figure 3.8 : Principe d'insertion de la maintenance par de l'heuristique HD



Dans ce qui suit, nous allons présenter les tests que nous avons effectués sur des benchmarks de Taillard [TAI93] ainsi que les résultats de ces tests.

### 3.3. Tests et Résultats

Nous allons présenter dans cette section les résultats obtenus par les différentes heuristiques de l'approche constructive (Palmer, CDS, Gupta, RA et NEH), itérative (recherche Tabou) et évolutive (AG) pour la résolution du problème de l'ordonnancement conjoint production/maintenance par la stratégie séquentielle.

Ces résultats sont obtenus après le déroulement des phases suivantes :

- Comparaison des résultats obtenus en exécutant les différentes heuristiques sur les différents problèmes.
- Comparaison des résultats après l'insertion des tâches de maintenance.

Les données de production sont des benchmarks de Taillard [TAI93] de différentes tailles. Les données de maintenance sont générées aléatoirement. Pour pallier le manque de benchmarks en maintenance préventive systématique, nous avons utilisé un générateur de tâches aléatoires. Les paramètres d'entrée du générateur sont : le nombre de machines, et les bornes inférieure et supérieure pour chaque paramètre d'une tâche de maintenance ( $T^*$ ,  $T_{\min}$  et  $T_{\max}$ ). Chacun de ces trois paramètres est lui-même borné par deux valeurs minimale et maximale, pour éviter d'avoir des valeurs identiques. Pour effectuer nos tests, nous avons généré une seule tâche de maintenance par machine pour chaque problème traité. De plus, on exécute la gamme complète à chaque occurrence de la tâche de maintenance, c'est-à-dire que la durée opératoire d'une tâche de maintenance est identique pour l'ensemble de ses occurrences.

La fonction objectif est la minimisation de la date de sortie de la dernière tâche (Makespan) pour la production, la minimisation de la somme des retards et avances des tâches de maintenance pour la maintenance (§ 3.3.2.3), et enfin la minimisation de la date de sortie de la dernière tâche après insertion de la maintenance pour les ordonnancements conjoint production/maintenance. Les contributions des fonctions objectif de la production et de la maintenance, dans la fonction objectif global  $f$ , sont égaux à 1 ( $\alpha = 1$  et  $\beta = 1$ ).

La recherche Tabou a été exécutée avec les paramètres suivants : nombre de générations : 50 ; taille du voisinage : 50 ; taille de la liste Tabou : 7 ; stagnation : 20. Pour chaque benchmark, le meilleur résultat est retenu.

Les résultats de l'algorithme génétique sont obtenus après plusieurs exécutions de la méthode. Le meilleur résultat est sauvegardé, ainsi que les paramètres associés. Les paramètres suivants sont les mêmes pour toutes les exécutions des algorithmes génétiques : taux de croisement : 0.7, taux de mutation : 0.01, stratégie de renouvellement : N\_meilleur, le remplacement se fait entre la population sélectionnée et la population croisée. Taille de la population : entre 50 et 100, nombre de simulation : 400. D'autre part les opérateurs suivants ont été utilisés : opérateur de croisement sur la production : k points pair et impair. Par ailleurs nous avons utilisé le *Sharing* avec les paramètres suivants :  $\alpha_{\text{Sharing}} = 0,99$  et  $\beta_{\text{Sharing}} = 4$ .

Ces données sont identiques pour l'ensemble des tests que nous avons effectués, et c'est pour cette raison que nous les présentons que dans cette section.

### 3.3.1. Ordonnancement de la production

Le graphe de la figure 3.9 présente non pas les résultats obtenus en appliquant les heuristiques de l'approche constructive (Palmer, CDS, Gupta, RA et NEH) sur les différents benchmarks, mais la déviation du critère d'optimisation de chaque heuristique par rapport à celui de l'heuristique NEH. L'idée de cette démarche est de représenter la déviation par rapport à l'optimum relatif obtenu par l'heuristique NEH. Sachant que Ponnambalam et al [PON01] concluaient dans leurs travaux que NEH est l'heuristique qui donne les meilleurs résultats par rapport aux heuristiques RA, CDS, Gupta et Palmer, pour l'ordonnancement de production dans un atelier de type *flowshop* de permutation.

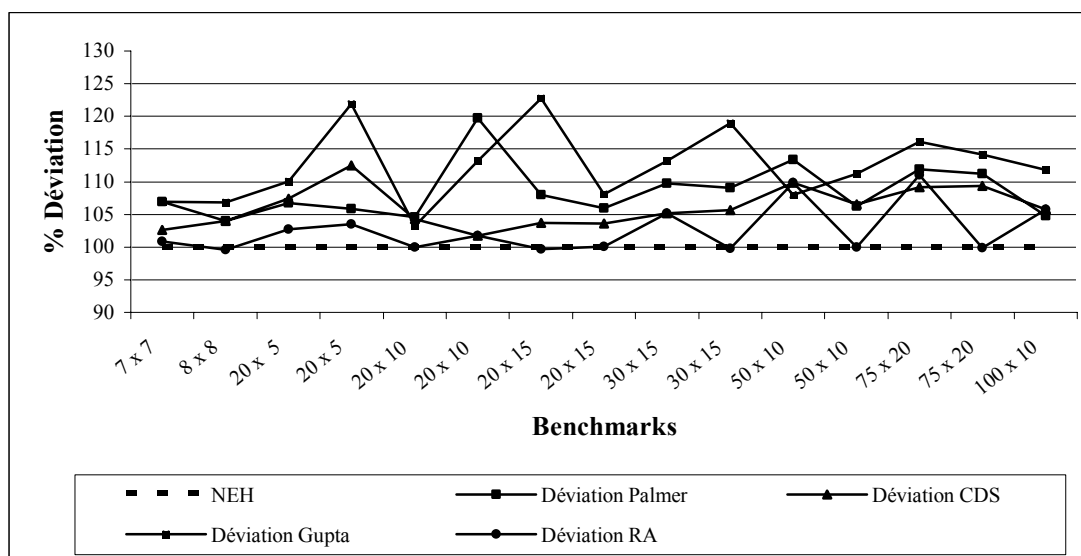


Figure 3.9 : Déviation des heuristiques constructives par rapport à NEH

On remarque que les déviations les plus importantes sont celles de l'heuristique de Gupta, une déviation de 20% dans le pire des cas. Sinon l'heuristique NEH reste celle qui donne les meilleurs résultats pour l'approche constructive.

### 3.3.2. Insertion de la maintenance

Les résultats après insertion des tâches de maintenance sur les séquences de production générées lors de la phase précédente sont représentés par le graphe de la figure 3.10. Nous représentons sur le graphe uniquement les meilleurs résultats après insertion des tâches de maintenance, par rapport à la valeur du  $C_{max}$  après insertion de la maintenance, par les différentes heuristiques d'insertion de la maintenance citées ci-après. C'est-à-dire que nous avons testé sur chaque séquence de production, toutes les heuristiques d'insertion des tâches de maintenance, et nous avons retenu le meilleur résultat pour la représentation graphique.

Les heuristiques considérées sont :

- les heuristiques  $HN^*$ ,  $HN_{min}$  ou  $HN_{max}$  ;
- l'heuristique HA avec et sans tolérance des avances/retards des tâches de maintenance
- l'heuristique HD avec et sans tolérance des avances/retards des tâches de maintenance

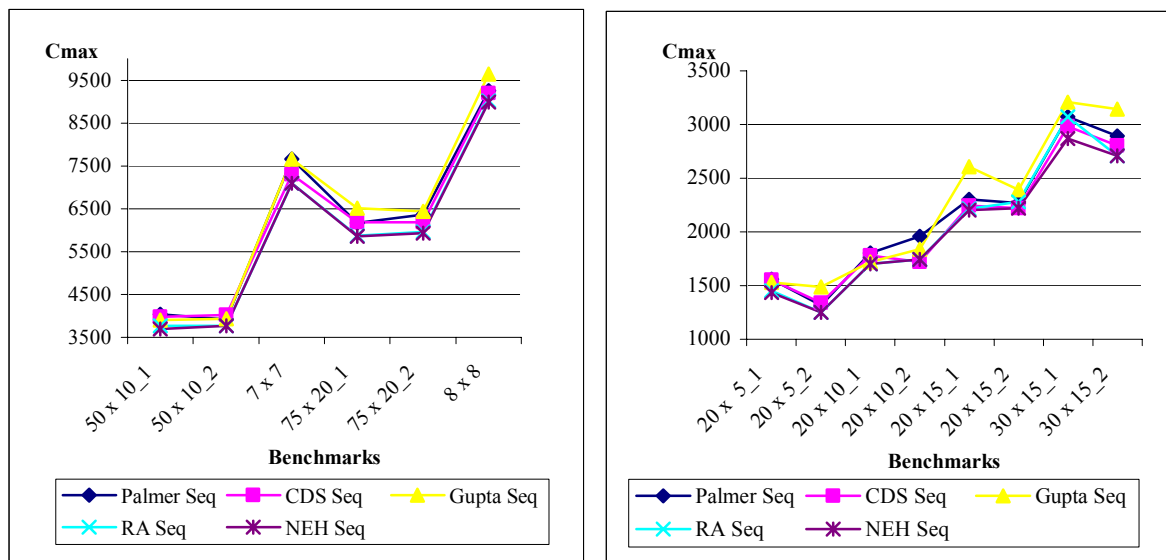


Figure 3.10: Ordonnancement conjoint P/M des heuristiques constructives

On remarque que dans plus de 50% des cas l'heuristique NEH optimise au mieux la fonction objectif (la date de sortie de la dernière tâche après insertion de la maintenance). NEH reste donc l'heuristique qui donne les meilleurs résultats même après insertion de la maintenance sur la séquence de production.

Néanmoins, on remarque que pour le Benchmark 20x10\_2, la date de sortie de la dernière tâche de production trouvée par l'heuristique NEH est inférieure à celle trouvée par CDS (Fig. 3.9), mais après insertion des tâches de maintenance, la date de sortie de la dernière tâche trouvée par l'heuristique CDS est inférieure à celle trouvée par NEH. Donc minimiser la date de sortie de la dernière tâche des séquences de production n'est pas forcément un gage d'optimalité après insertion de la maintenance (date sortie de la dernière tâche après insertion de la maintenance).

De même que pour l'ordonnancement de production, la figure 3.11 présente la déviation du critère d'optimisation de la deuxième phase de la stratégie séquentielle. Ceci du fait que même après insertion de la maintenance, l'heuristique NEH est celle qui optimise au mieux le  $C_{max}$ .

Il est important de noter qu'après l'insertion des tâches de maintenance le  $C_{max}$  est de moins bonne qualité. Ceci est dû à l'application de la règle de décalage à droite pour résoudre les chevauchements entre les tâches de production et de maintenance lors de l'insertion des tâches de maintenance. Et donc pour les figures 3.10 et 3.11, l'optimum relatif de NEH n'est pas le même.

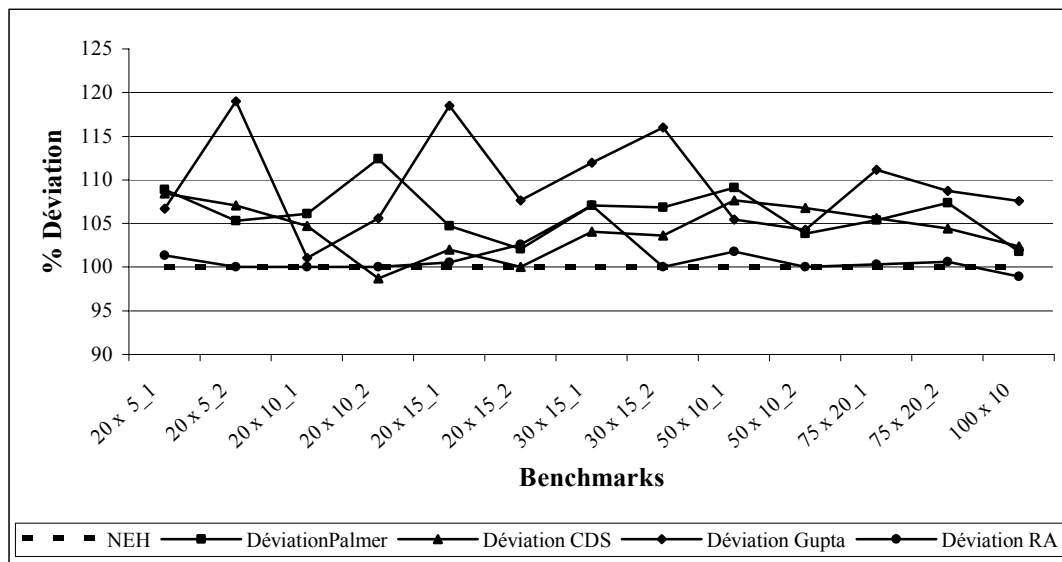


Figure 3.11 : Déviation par rapport à NEH après insertion de la maintenance

Le graphe de la figure 3.12 présente les résultats obtenus par les méthodes séquentielles : l'heuristique NEH (meilleure heuristique de l'approche constructive), la recherche Tabou et l'algorithme génétique.

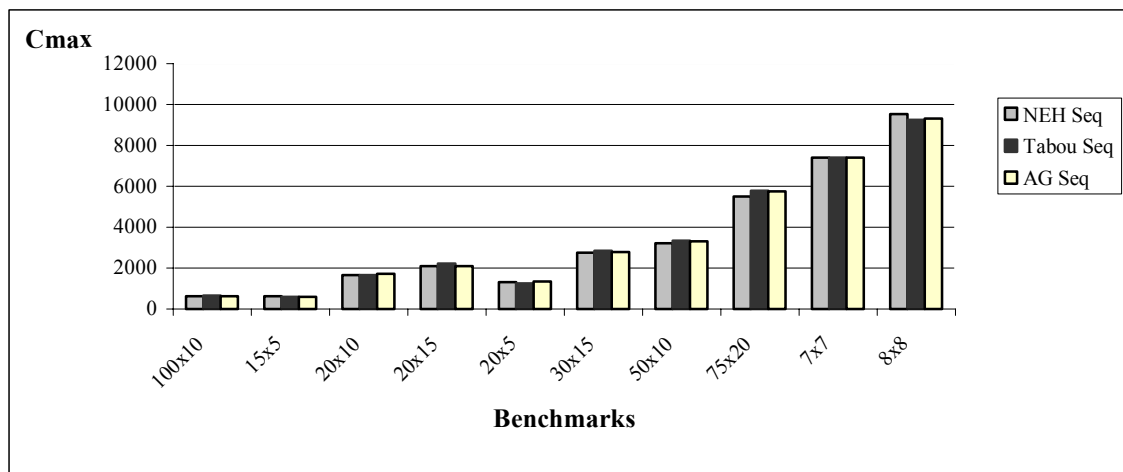


Figure 3.12: Ordonnancement conjoint P/M par la stratégie séquentielle

On remarque que 80% des meilleurs résultats sont obtenus par les algorithmes génétiques, 20% avec l'approche Tabou. D'autre part les meilleurs résultats de l'heuristique NEH sont réalisés pour les benchmarks de grande taille. Les algorithmes génétiques sont meilleurs que la recherche Tabou et l'heuristique NEH pour la stratégie séquentielle.

L'exécution des heuristiques HA et HD avec relaxation de la contrainte sur l'insertion (Tolérance des Avances/Retards des tâches de maintenance par rapport à l'intervalle de tolérance) fournit de meilleurs résultats que leurs exécutions sans relaxation de la contrainte. Cela est dû au fait, que si on tolère qu'une tâche de maintenance puisse être avancée (placée avant son  $T_{\min}$ ) ou retardée (placée après son  $T_{\max}$ ), on augmente systématiquement les emplacements possibles pour l'insertion, par rapport au cas où cette relaxation de contrainte n'est pas permise (heuristique sans tolérance des avances/retards).

### Remarque

Pour le benchmark TA20x10\_2, le  $C_{\max}$  de production trouvé par la méthode NEH (2295, séquence  $S_1$ ) est supérieur à celui trouvé par l'algorithme génétique (1300, séquence  $S_2$ ); mais après l'insertion des tâches de maintenance, le retard sur la séquence  $S_1$  (5525) est inférieur à celui de la séquence  $S_2$  (5563). Dans ce cas aussi optimiser la fonction objectif de production ne permet pas forcément de minimiser la fonction objectif globale.

Le tableau 3.1 présente les meilleurs résultats par stratégie d'insertion des tâches de maintenance sur les séquences de production.

Heuristiques d'insertion de la maintenance	Naïve			Descendante		Ascendante	
	T*	T <sub>min</sub>	T <sub>max</sub>	Avance/Retard Tolérés	Avance/Retard Non tolérés	Avance/Retard Tolérés	Avance/Retard Non tolérés
Pourcentage détaillé de meilleurs cas	0%	0%	4%	21%	12%	40%	23%
Total	4%			33%		63%	

Tableau 3.1 : Résultats par heuristique d'insertion de la maintenance

Les deux heuristiques Ascendante (HA) et Descendante (HD) qui tolère l'avance et le retard des tâches de maintenance fournissent de meilleurs résultats (respectivement 40% et 21%) par rapport à celles qui ne tolère pas l'avance et le retard des tâches de maintenance (respectivement 23% et 12%), du fait qu'une tâche de maintenance à plus d'emplacement possibles dans les première heuristiques que pour les secondes. Le plus mauvais résultat est réalisé par l'heuristique HN avec seulement 4%.

On constate que lorsque l'heuristique d'insertion est restrictive telle l'heuristique HN par exemple où on insère systématiquement à T\*, T<sub>min</sub> ou T<sub>max</sub>; l'incidence de cette politique d'insertion de la maintenance sur la qualité de l'ordonnancement de production est négative et de ce fait les résultats sont très mauvais. Par contre, lorsqu'on lève cette restriction en relaxant les contraintes, en tolérant l'avance et le retard des tâches de maintenance par exemple, l'incidence de cette politique sur l'ordonnancement de production est nettement meilleure et les résultats le sont aussi.

### 3.4. Synthèse

Dans cette section, nous avons présenté la résolution du problème de l'ordonnancement conjoint maintenance production par la stratégie séquentielle, dans un atelier de type *flowshop* de permutation. Cette dernière se présente en deux étapes : d'abord ordonnancer les tâches de production, puis insérer les tâches de maintenance, en considérant l'ordonnancement de production comme une contrainte supplémentaire du problème. La fonction objectif à optimiser, dans ce cas, est une fonction qui tient compte des contraintes de production et de maintenance.

Nous avons décliné l'étape ordonnancement de la production en trois approches. Une approche constructive où nous avons généré des ordonnancements de production par des heuristiques dédiées au *flowshop* telles RA, CDS, Gupta, Palmer et NEH. Une approche itérative, où nous avons présenté les paramètres de la recherche Tabou, que nous avons utilisé

pour générer un ordonnancement initial. Enfin une approche évolutive, où nous avons défini les opérateurs des AG, que nous avons utilisé, pour générer un ordonnancement initial.

La résolution du problème de l'ordonnancement de production, avec ces différentes heuristiques, est largement étudiée dans la littérature [PON01, LOP01, POR01]. Cependant, cette phase est indispensable pour la suite du problème, qui consiste à insérer les tâches de maintenance préventive systématique sur une séquence initiale de production.

Pour aborder cette seconde étape, de la résolution de notre problème, nous avons développé deux types d'heuristiques pour l'insertion des tâches de maintenance : des heuristiques pour l'insertion sur une machine, puis sur plusieurs machines. Nous avons utilisé deux heuristiques pour l'insertion sur une machine : l'heuristique HB développée par Kaabi [KAA04] et nous avons proposé l'heuristique de recherche en profondeur (HRP). L'heuristique HB consiste à insérer une tâche de maintenance strictement à  $T^*$ ,  $T_{\min}$  ou  $T_{\max}$ . Par contre l'heuristique HRP va rechercher le meilleur emplacement, dans l'intervalle de tolérance de chaque tâche de maintenance, avant de l'insérer. Dans les deux cas, la règle du décalage à gauche est appliquée pour résoudre les problèmes de chevauchement entre les tâches de production et de maintenance, s'ils se posent.

Par ailleurs nous avons proposé trois heuristiques pour l'insertion sur plusieurs machines : l'heuristique HN qui est une adaptation de l'heuristique HB pour le cas de plusieurs machines. L'heuristique HA qui consiste à insérer les tâches de maintenance de la première machine jusqu'à la dernière, et l'heuristique HD qui consiste à insérer les tâches de maintenance de la dernière machine à la première.

En terme de résultats obtenus, on peut dire que dans le cas de l'ordonnancement conjoint production/maintenance par la stratégie séquentielle, l'heuristique NEH reste l'heuristique qui donne les meilleurs résultats par rapport aux heuristiques dédiées au *flowshop* telles RA, CDS, Gupta et Palmer. Cependant les AG sont globalement meilleurs que l'heuristique NEH et la recherche Tabou.

De plus l'exécution des heuristiques HA et HD avec relaxation de la contrainte sur l'insertion, par la tolérance des avances/retards des tâches de maintenance par rapport à l'intervalle de tolérance, fournit les meilleurs résultats. En effet, force est de constater que l'incidence de l'insertion systématique (cas de l'heuristique HN avec ces trois variantes) sur la qualité de l'ordonnancement de production est négative. Par contre, la relaxation de cette contrainte, en autorisant la recherche du meilleur emplacement pour l'insertion de la maintenance, à la périphérie de l'intervalle de tolérance, induit une nette amélioration de la qualité de l'ordonnancement de production.

Par ailleurs, il est important de noter qu'optimiser la fonction objectif de production n'est pas forcément un gage d'optimalité pour la fonction objectif globale après insertion de la maintenance. L'optimisation en deux phases de la stratégie séquentielle : une première optimisation par rapport à la production pour obtenir une séquence de production, puis une seconde par rapport à la maintenance pour obtenir une séquence conjointe production/maintenance, détériore sensiblement la qualité de l'ordonnancement de production.

#### **4. La stratégie intégrée**

La résolution par la stratégie intégrée est basée sur le principe de la représentation conjointe de données de production et de maintenance avant de passer à la résolution proprement dite. Nous présenterons dans cette section l'adaptation des heuristiques choisies telles Palmer [PAL65], CDS [CAM70], Gupta [GUP71], RA [DAN77], NEH [NAW83], la recherche Tabou [GLO89] et AG [GOL89].

Notre contribution à cette partie de notre étude touche deux aspects : le premier concerne le principe même de la stratégie intégrée qui consiste en la prise en compte, lors de la résolution du problème, des deux types de tâches (production et maintenance), cependant la structure d'un ordonnancement adopté pour la stratégie séquentielle (section 3.3.1.1) s'avère inadéquate pour cette stratégie. Nous avons donc proposé une nouvelle structure à deux champs : une séquence de production qui représente l'ordre d'exécution des tâches de production et une matrice qui représente les emplacements d'insertion des tâches de maintenance dans la séquence de production. Cette structure va représenter un ordonnancement conjoint. L'autre aspect de notre contribution concerne les approches de résolution : la première contribution consiste en l'adaptation des heuristiques de l'approche constructive (Palmer, CDS, Gupta, RA et NEH) à notre problème ; pour cela nous avons proposé un ordre de précedence entre toutes les tâches, quelles soient de production ou de maintenance. La seconde consiste à proposer de nouveaux paramètres pour la recherche Tabou qui intègrent la maintenance. Enfin, la dernière contribution consiste à proposer de nouveaux opérateurs pour les AG qui intègrent aussi la maintenance.

Dans ce qui suit nous présentons la représentation d'un ordonnancement conjoint puis pour chaque approche de résolution les adaptations que nous proposons dans le cas de la stratégie intégrée.



## 4.1. Représentation d'un ordonnancement

Chaque individu est codé par une structure à deux champs : le premier champ est une séquence **S** qui représente l'ordre d'exécution des tâches de production. Le second est une matrice **M** qui représente les emplacements d'insertion des tâches de maintenance. L'élément **M[i,j]** de la matrice **M** représente l'emplacement d'insertion de la  $j^{\text{ième}}$  tâche de maintenance sur la  $i^{\text{ième}}$  machine dans la séquence **S**.

Exemple :

Séquence de production : <b>S</b>	1	9	3	8	5	6	7	4	2	0
Emplacements des tâches de maintenance : <b>M</b>	{	0	1	4	6					
1		2	5							
0		4	7	8						

$M[3,2] = 4$  signifie que la deuxième tâche de maintenance de la troisième machine s'insère à la position 4 (après la tâche de production 8) dans la séquence **S**. L'exécution des tâches sur les trois machines selon la codification de l'exemple précédant est la suivante :

$M_{ij}$  : la  $i^{\text{ième}}$  tâche de maintenance sur la machine  $j$ ,  $P_i$  : tâche de production  $i$ .

Machine 1 :  $M_{01}, P_1, M_{11}, P_9, P_3, P_8, M_{21}, P_5, P_6, M_{31}, P_7, P_4, P_2, P_0$

Machine 2 :  $P_1, M_{02}, P_9, M_{12}, P_3, M_{22}, P_8, P_5, P_6, P_7, P_4, P_2, P_0$

Machine 3 :  $M_{03}, P_1, P_9, P_3, P_8, M_{13}, P_5, P_6, P_7, M_{23}, P_4, M_{33}, P_2, P_0$

## 4.2. Approche dédiée

Le principe des heuristiques dédiées au *flowshop* de permutation, consiste à définir un ordre de précedence entre les tâches pour obtenir une séquence de production. Notre contribution dans le cas de la résolution du problème de l'ordonnancement conjoint par la stratégie intégrée, va consister en la définition d'un ordre de précedence entre toutes les tâches, qu'elles soient de production ou de maintenance. Ceci du fait que dans le cas de la stratégie intégrée, la résolution est simultanée et prend en compte, au départ, les deux types de tâches (production et maintenance), pour générer un ordonnancement conjoint.

L'approche que nous avons adoptée se base sur le principe que les tâches de maintenance préventive systématique sont périodiques et sont planifiées systématiquement toutes les  $T$  unités de temps ( $T$  variant dans l'intervalle  $[T_{\min}, T_{\max}]$ ). Partant de cet état, on prend en compte uniquement la première tâche de maintenance, sur chaque machine, lors de la génération de l'ordonnancement conjoint production/maintenance. Puis on insère les tâches de maintenance restantes selon l'heuristique HA définie dans la section 3.2.1.2.

Les sections suivantes présentent l'adaptation des différentes heuristiques de l'approche constructive à la résolution du problème de l'ordonnancement conjoint production/maintenance par la stratégie intégrée

#### 4.2.1. Heuristique de Palmer (resp. Gupta) intégrée

Comme l'heuristique de Gupta est une amélioration de l'heuristique de Palmer, nous présentons une seule adaptation pour les deux heuristiques. Cette adaptation consiste à déterminer une séquence de production  $S$  et pour chaque machine, l'emplacement de la première tâche de maintenance par l'heuristique de Palmer (resp. de Gupta). Sur cette séquence de base, l'insertion des tâches de maintenance restantes, sur chaque machine, s'effectuera selon le principe de l'heuristique HRP (section 3.2.1.2). Le principe de cette résolution est présenté dans l'algorithme 3.5.

---

#### Algorithme de Palmer (resp. Gupta) intégré

---

##### Début

Déterminer une séquence  $S$  de production ainsi que l'emplacement de la première tâche de maintenance par l'heuristique de Palmer (resp. de Gupta)

**Tant qu'**il reste des tâches de maintenance à insérer

**Faire** Insérer la tâche de maintenance avec l'heuristique HRP.

Insérer la tâche de production.

Passer à la prochaine tâche de production dans la séquence  $S$ .

**Fin Tant que**

**Fin.**

---

Algorithme 3.5: Heuristique de Pamler(resp. Gupta) intégrée

#### 4.2.2. Heuristique RA intégrée

Partant d'une solution initiale fournie par l'heuristique de Palmer ou de Gupta intégrée, la résolution de l'approche intégrée dans ce cas, consiste à permuter si possible deux tâches successives dans la séquence de la solution initiale. A chaque étape, on va choisir le meilleur

voisin pour le comparer avec la solution actuelle. La démarche de cette résolution est illustrée par l'algorithme 3.6:

---

**Algorithme de RA intégré**

---

**Variables locales**  $n$  : nombre de tâches de production ;  $m$  : nombre de machines ;  
**Début** Trouver une solution initiale sous optimale par l'une des heuristiques précédentes  
Mémoriser cette séquence dans le tableau MST ;  
 $i \leftarrow 1$  ;  
Trouver la meilleure solution parmi tous les voisins  
**Tant que**  $i < n+m-1$  (2)  
**Faire** **Si** les tâches de position  $i$  et  $i+1$  sont des tâches de production  
**Alors** Permuter les tâches  $i$  et  $i + 1$  dans MST;  
Evaluer la fonction objectif ;  
**Sinon Si** la permutation ne rend pas la tâche de maintenance en retard  
**Alors** Permuter les tâches  $i$  et  $i + 1$  dans MST;  
Evaluer la fonction objectif ;  
**Sinon**  $i \leftarrow i + 1$   
**Fin Si**  
**Fin Si**  
**Fin Tant que**  
Trouver la meilleure solution parmi tous les voisins.  
**Si** La valeur de la fonction objectif est inférieure à celle de MST;  
**Alors** Aller a (2).  
**Sinon** Le tableau MST fournit la solution de l'heuristique RA intégrée.  
**Fin Si**  
**Fin.**

---

Algorithme 3.6: Heuristique RA intégrée

#### 4.2.3. Heuristique CDS intégrée

La résolution de problème d'ordonnancement conjoint de la production et de la maintenance dans le cas de l'heuristique CDS par la stratégie intégrée a été résolu de la manière suivante : on considère que les tâches de maintenance sont des tâches de production telle que la durée d'exécution d'une tâche de maintenance est égale à sa durée d'exécution sur la machine sur laquelle elle doit être exécutée et égale à zéro sur le reste des machines.

Après avoir appliqué l'heuristique CDS comme dans le cas séquentiel, on insère toutes les tâches de maintenance restantes sur la totalité des machines, par l'heuristique HRP.

Ce principe de cette résolution est présenté dans l'algorithme 3.7.

**Algorithme CDS intégré**

**Variabes locales**  $m$  : le nombre de machines ;  $P_{kj}$  : durée d'exécution du travail  $k$  sur la machine  $j$  ;  $P^j_{k1}$  : durée d'exécution du travail  $P_k$  sur la  $j^{\text{ième}}$  machine équivalente.

**Début** On ne considère que la première et la dernière machine (machine  $m$ ), soit pour un travail  $k$  :  $P^1_{k1} = p_{k1}$  ;  $P^1_{k2} = P_{km}$

Les deux machines équivalentes sont respectivement constituées des deux premières et des deux dernières machines, sans tenir compte des machines intermédiaires. De manière générale à l'étape  $j$ , on a les durées suivantes sur les deux machines équivalentes :

$$P^j_{k1} = \sum_{i=1}^j P_{ki} \quad P^j_{k2} = \sum_{i=m-j+1}^m P_{ki} \quad j = 1, \dots, m-1.$$

Pour chacun de ces problèmes fictifs, on insère toutes les tâches de maintenance sur toutes les machines, ensuite on détermine la séquence optimale par la règle de Johnson.

Conserver la meilleure solution parmi les  $m-1$ .

**Fin.**

## Algorithme 3.7: Heuristique CDS intégrée

**4.2.4. Heuristique NEH intégrée**

La deuxième étape de l'heuristique NEH est l'ordonnancement des tâches à exécuter dans une liste  $L$  (la définition d'un ordre de précédente). L'application de cette méthode pour la résolution du problème d'ordonnancement conjoint par la stratégie intégrée, nécessite la définition d'un ordre de précédente entre toutes les tâches à ordonner (tâches de production et de maintenance). Cet ordre est clairement défini entre les tâches de production, pour le cas d'une minimisation du Makespan (§ Annexe A).

Pour les tâches de maintenance, l'ordre est prédéfini sur chaque machine (maintenance préventive systématique), et il est indépendant entre les machines.

Reste le problème de la définition d'un ordre entre les tâches de production et de maintenance. Pour le résoudre, nous proposons le principe suivant : la recherche du meilleur emplacement pour une tâche de production à un instant  $t$  se fait à partir de sa date début au plus tôt, jusqu'à la dernière tâche insérée à cet instant  $t$ . Et la recherche du meilleur emplacement pour la  $k^{\text{ième}}$  tâche de maintenance sur la machine  $j$  se fait à l'instant  $t'$  ( $t' \geq t$ ).

Cette recherche se fait dans un intervalle  $[t_1, t_2]$  selon l'heuristique HRP avec :

$t'$  : date de sortie de la dernière tâche sur la machine  $j$ ,

$t''$  : date de fin au plus tard de la tâche de maintenance  $M_j$  sur la même machine  $j$ .

$$t_1 = I_{\min j} - K \times T_{\min j}.$$

$$t_2 = I_{\max j} + K \times T_{\min j}.$$

$K$  : la marge de tolérance ( $0 \leq K \leq 1$ ).

$I_{\min j}$  : La date de début au plus tôt de la tâche  $M_j$

$I_{\max j}$  : La date de fin au plus tard de la tâche  $M_j$

L'insertion d'une tâche de production ou de maintenance provoque le décalage des tâches qui la succèdent. Par conséquent, des tâches de maintenance peuvent s'éloigner de leurs intervalles de tolérances, par rapport à celles qui la précède. Ceci peut provoquer le phénomène des tâches manquantes<sup>3</sup> sur certaines machines (Fig. 3.13). Pour remédier à ce problème, nous avons utilisé le principe suivant : à chaque insertion d'une tâche (de production ou de maintenance) on vérifie toutes les tâches de maintenance qui la succède, si une tâche est très éloignée par rapport à la tâche de maintenance qui la précède (tâche manquante), on effectue des permutations avec les tâches de production qui la précèdent jusqu'à ce que elle soit assez proche de la tâche de maintenance qui la précède (pratiquement on replace la tâche de maintenance dans son intervalle de tolérance).

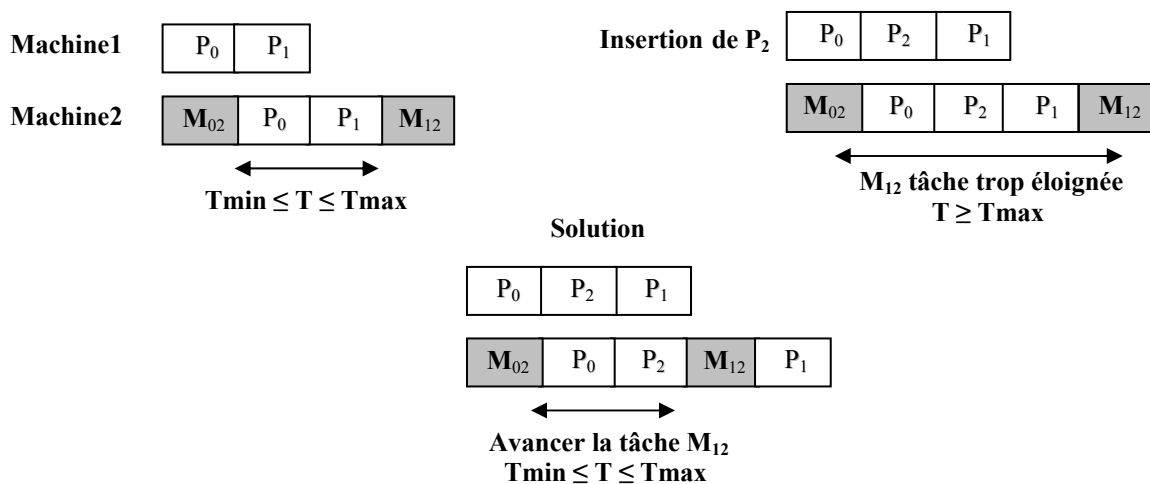


Figure 3.13: Insertion d'une tâche de maintenance

### 4.3. L'approche itérative : La recherche Tabou intégrée

<sup>3</sup> On dit qu'une tâche de maintenance est manquante sur une machine  $j$ , si la durée relative (on recommence à la date 0 après insertion de chaque tâche de maintenance) qui sépare deux tâches de maintenance successives est supérieure à  $T_{\max j} + p_j$  (section 3.2.2).

Dans cette section, nous présentons les nouveaux paramètres de la recherche Tabou que nous proposons dans le cas de l'ordonnancement conjoint production/maintenance par la stratégie intégrée. Ce paramétrage présente la particularité de travailler, avec non pas une séquence de production mais, avec une séquence conjointe production/maintenance.

Dans ce qui suit, nous présentons le principe de ce nous paramétrage.

#### 4.3.1. Génération de la solution initiale

Deux approches sont possibles soit la solution initiale est fournie par l'utilisateur, soit elle est générée aléatoirement.

La recherche Tabou peut tenter d'améliorer des solutions générées par d'autres méthodes. Cela est dû au fait qu'elle offre la possibilité de commencer par une solution fournie en entrée, comme elle peut également commencer par une solution partielle (séquence de production générée par une des méthodes de résolution du problème d'ordonnancement de production). Dans ce cas, une solution initiale complète est obtenue après insertion des tâches de maintenance sur cette séquence, selon l'une des heuristiques présentées dans la section 3.3.2. Enfin, la méthode peut débiter son processus de résolution par une solution initiale générée de manière aléatoire. Nous avons proposé pour ce cas un algorithme pour la génération aléatoire d'une solution conjointe production/maintenance (Algorithme 3.8).

---

#### Algorithme Génération aléatoire de la solution initiale

---

##### **Début**

Générer une séquence de production aléatoirement (Algorithme 3.1).

Insérer sur cette séquence les tâches de maintenance selon l'une des heuristiques d'insertion des tâches de maintenance développées dans la section 3.3.2.

##### **Fin.**

---

Algorithme 3.8: Génération aléatoire d'une solution conjointe production/maintenance

#### 4.3.2. Génération du voisinage

Le déplacement d'une solution à une autre dans le voisinage peut se faire en effectuant des décalages dans la séquence de production, des décalages dans la séquence de maintenance, ou sur les deux en même temps. Nous avons définis deux principes de décalages permettant de générer des solutions voisines par rapport à une solution. Le premier concerne les tâches de production et le second les tâches de maintenance. Nous présentons dans la section suivante le principe de chacun.

#### 4.3.2.1 Génération du voisinage par décalage des tâches de maintenance

Une tâche de maintenance peut avoir plusieurs emplacements possibles dans son intervalle de tolérance. Il est donc intéressant de définir le voisinage d'une solution comme étant l'ensemble des emplacements possibles pour l'insertion d'une ou plusieurs tâches de maintenance, sur une ou plusieurs machines. Le déplacement d'une solution à une autre dans le voisinage se fait alors selon l'algorithme 3.9 :

---

**Algorithme** Génération du voisinage par décalage des tâches de maintenance

---

**Variabes Locales**  $t_m$  : Taux de mutation sur les machines ( $0 \leq t_m \leq 1$ ) ; Sens = 0 ou 1

**Début**

**Pour** Chaque Machine  $M_j$

**Faire** Générer un nombre aléatoire  $r$

**Si**  $r < t_m$  (la machine  $M_j$  est sélectionnée)

**Alors** Sélectionner une tâche de maintenance aléatoirement de la machine  $M_j$ .

Générer un nombre aléatoire Sens

**Si** Sens = 0

**Alors** Décaler vers la droite la  $r^{\text{ième}}$  tâche de maintenance de  $M_j$ .

**Sinon** Décaler vers la gauche la  $r^{\text{ième}}$  tâche de maintenance de  $M_j$ .

**Fin Si**

**Fin Si**

**Fin Tant que**

**Fin.**

---

Algorithme 3.9: Génération du voisinage par décalage des tâches de maintenance

Après un certain nombre d'itérations ( $nbiter$ ), il serait intéressant de guider la recherche vers d'autres espaces, et cela en changeant l'ordre d'exécution des tâches de production. Le principe de cette amélioration est présenté dans l'algorithme 3.10.

---

**Algorithme** Exploration du voisinage

---

**Variable Locale**  $nbiter$  : nombre d'itération ;  $S$  : solution choisie du voisinage

**Début**

**Si** l'itération courante est multiple de  $nbiter$

**Alors** Choisir deux tâches de production aléatoirement  $P_{r1}$  et  $P_{r2}$  dans  $S$ .

Permuter l'emplacement de ces deux tâches.

Garder les mêmes emplacements pour les tâches de maintenance.

**Fin Si**

**Fin.**

---

Algorithme 3.10: Exploration du voisinage

#### **4.3.2.2 Génération du voisinage par décalage des tâches de production**

Le but de cet opération est de créer de nouveaux individus en changeant l'ordre d'exécution des groupes de tâches de production, mais en gardant le bon emplacement des tâches de maintenance à l'intérieur de chaque groupe.

Le choix d'une solution du voisinage se fait selon l'une des stratégies suivante : Best move, First Improve ou Aléatoirement.

#### **4.3.3. Fonction d'aspiration**

Si aucune solution du voisinage n'est sélectionnée (toutes les solutions sont taboues), alors on sélectionne aléatoirement une solution de la liste Tabou et on continue le processus de recherche avec cette solution.

#### **4.3.4. Critère d'arrêt**

La méthode tente d'améliorer la solution initiale au bout d'un nombre d'itérations maximum (nbiter). On arrête le processus de recherche si la solution ne s'améliore pas pendant un certain nombre d'itération (Nbstag), ou bien, si on atteint le nombre maximal d'itération (nbiter).

### **4.4. Approche évolutive : un algorithme génétique intégré**

Nous présentons dans cette section les nouveaux opérateurs des AG que nous proposons dans le cas de l'ordonnancement conjoint production/maintenance par la stratégie intégrée. Ces opérateurs présentent la particularité de travailler, avec non pas une séquence de production mais, avec une séquence conjointe production/maintenance.

Les opérateurs de croisement et de mutation sur la production ont été utilisés, en combinaison avec les opérateurs de croisement et de mutation sur la maintenance, pour définir de nouveaux opérateurs de croisement et de mutation conjoint production/maintenance.

Dans ce qui suit, nous allons présenter le principe des opérateurs de croisement et de mutation sur la maintenance que nous avons développés, puis celui des opérateurs conjoints issus de la combinaison des principes de chacun des opérateurs de croisement et de mutation.



#### 4.4.1. Croisement sur la maintenance

Pour créer un fils valide à partir de deux parents, ce premier doit hériter des informations sur la production, des informations sur la maintenance, ou les deux en même temps, d'où la nécessité de définir des opérateurs de croisement sur la production et sur la maintenance. Dans ce qui suit, nous présentons uniquement les opérateurs de croisement sur la maintenance que nous avons définis. Les opérateurs de croisement sur la production sont les opérateurs définis pour les AG de production classique présentés dans la section 3.1.4.

Nous nous sommes inspirés de l'opérateur de croisement à  $k$  points (Algorithme 3.5) pour définir deux nouveaux opérateurs de croisement sur la maintenance : un premier opérateur de croisement horizontal à  $k$  points pairs et un deuxième opérateur horizontal à  $k$  points impairs. La dénomination d'horizontal vient du fait que nous travaillons sur la matrice  $M$  de la structure d'un individu, comme définit dans la section 4.1, où chaque ligne représente l'emplacement des tâches de maintenance sur chaque machine.

L'opérateur de croisement horizontal à  $k$  points pairs consiste à générer  $k$  points aléatoirement puis à effectuer un croisement à  $k$  points pairs classique, comme définis par l'algorithme 3.5, mais uniquement sur les tâches de maintenance. Le principe de cet opérateur est présenté dans l'algorithme 3.11.

---

#### **Algorithme** Croisement horizontal à $k$ points pairs

---

**Début** Générer  $k$  points aléatoires ( $k \in [0..m]$ )

Les emplacements des tâches de maintenance sur les machines qui se trouvent dans des parties paires du premier parent sont copiés vers le fils1, et les emplacements des tâches de maintenances qui se trouvent dans les parties restantes (impaires) du fils1 sont copiés à partir des parties impaires du deuxième parent.

**Fin.**

---

Algorithme 3.11: Croisement horizontal à  $k$  points pairs

Le principe du croisement horizontal à  $K$  points impairs est le même que le croisement horizontal à  $K$  points pairs, sauf que dans ce cas les parties impaires sont copiées à partir du premier parent et les parties paires à partir du deuxième. Le choix de l'opérateur de croisement est aléatoire.

Le principe du croisement sur la maintenance horizontal à  $k$  points pairs est illustré par l'exemple suivant à un point de croisement ( $r = 1$ ) suivant :

	<b>Père 1</b>		<b>Fils 1</b>
r →	<b>Machine1</b> 1 2 3 4 <b>Machine2</b> 0 1 5 <b>Machine3</b> 0 2 5 7		<b>Machine 1</b> 1 2 3 4 <b>Machine 2</b> 0 1 5 <b>Machine 3</b> 2 3 4
	<b>Père 2</b>		<b>Fils 2</b>
r →	<b>Machine 1</b> 0 3 6 <b>Machine 2</b> 1 4 5 6 <b>Machine 3</b> 2 3 4		<b>Machine 1</b> 0 3 6 <b>Machine 2</b> 1 4 5 6 <b>Machine 3</b> 0 2 5 7

Nous avons définis d'autres opérateurs de croisement en combinant les principes des opérateurs de croisements sur la production et ceux sur la maintenance. Cette combinaison nous a permis de définir les opérateurs de croisements suivants:

**Opérateur 1:** On prend l'emplacement des tâches de maintenance du premier parent, et l'ordre d'exécution des tâches de production du deuxième parent.

**Opérateur 2:** On prend l'emplacement des tâches de maintenance du premier parent, l'ordre d'exécution des tâches de production et un croisement pair entre les séquences d'exécution des tâches de production des deux parents.

**Opérateur 3:** On prend l'emplacement des tâches de maintenance du premier parent, l'ordre d'exécution des tâches de production et un croisement impair entre les séquences de production des deux parents.

**Opérateur 4:** L'ordre d'exécution des tâches de production est celui qui apparaît dans le premier parent, et l'emplacement des tâches de maintenance est un croisement horizontal à K points pair entre les emplacements des tâches de maintenance des deux parents.

**Opérateur 5:** L'ordre d'exécution des tâches de production est celui qui apparaît dans le premier parent, et l'emplacement des tâches de maintenance est un croisement horizontal à K points impair entre les deux parents.

**Opérateur 6:** L'ordre d'exécution des tâches de production est un croisement pair entre les deux parents, et l'emplacement des tâches de maintenance est un croisement horizontal à K points pair entre les deux parents.

**Opérateur 7:** L'ordre d'exécution des tâches de production est un croisement pair entre les deux parents, et l'emplacement des tâches de maintenance est un croisement horizontal à K points impair entre les deux parents.

**Opérateur 8:** L'ordre d'exécution des tâches de production est un croisement impair entre les deux parents, et l'emplacement des tâches de maintenance est un croisement horizontal à K points pair entre les deux parents.

**Opérateur 9:** L'ordre d'exécution des tâches de production est croisement impair entre les deux parents, et l'emplacement des tâches de maintenance est un croisement horizontal à K points impair entre les deux parents.

#### 4.4.2. Mutation sur la maintenance

Comme nous manipulons une séquence conjointe de tâches de production et de maintenance, la mutation peut se faire aussi bien sur la production que sur la maintenance.

Le choix des individus qui vont subir une mutation se fait selon le même principe que l'algorithme génétique de production défini dans la section 3.1.4.

Dans ce qui suit, nous présentons uniquement les opérateurs de mutation sur la maintenance que nous avons définis. Les opérateurs de mutation sur la production sont les opérateurs définis pour les AG de production classique comme présentés dans la section 3.1.4.

Nous avons proposé deux principes de mutation sur la maintenance : la mutation aléatoire et la mutation verticale. La mutation aléatoire consiste à décaler aléatoirement vers la gauche, ou vers la droite, une ou plusieurs tâches de maintenance. Le principe de cette mutation est défini dans l'algorithme 3.12.

---

#### **Algorithme** Mutation aléatoire sur la maintenance

---

**Variables Locales**  $t_r$  : Taux de mutation sur les machines ( $0 \leq r \leq 1$ ) ; Sens = 0 ou 1

**Début**

**Pour** Chaque Machine

**Faire** Générer un nombre aléatoire  $r \in [0..1]$ .

**Si** ( $r \leq t_r$ )

**Alors** Choisir aléatoirement une tâche de maintenance sur cette machine.

Générer un nombre aléatoire **Sens** (0 ou 1).

**Si** Sens=0

**Alors** Décaler la tâche de maintenance vers la gauche.

**Sinon** Décaler la tâche de maintenance vers la droite.

**Fin Si**

**Fin Si**

**Fin Pour**

**Fin.**

---

Algorithme 3.12: Mutation aléatoire sur la maintenance

La mutation verticale consiste à permuter les emplacements des tâches de production qui se trouvent dans deux parties différentes, générées aléatoirement, en conservant les emplacements des tâches de maintenance, qui se trouvent à l'intérieur de chaque partie par rapport aux tâches de production qui se trouvent dans cette même partie. Son efficacité augmente lorsque les tâches de maintenance sont bien placées par rapport aux tâches de production. La dénomination verticale vient du fait que nous travaillons sur la séquence de production  $S$  de la structure d'un individu, comme défini dans la section 4.1.

Nous avons définis deux opérateurs de mutation verticale : la mutation verticale à un point et la mutation verticale à deux points.

Le principe de la mutation verticale à un point consiste à générer aléatoirement une position  $p$  dans la séquence de production  $S$  de la structure d'un individu. Puis permuter les emplacements des deux groupes de tâches de production, qui se trouvent respectivement entre la position 0 et  $p$  (premier groupe) et entre  $p+1$  et  $n$  (deuxième groupe). Ce principe est présenté dans l'algorithme 3.13.

---

#### Algorithme Mutation verticale à un point

---

**Variables Locales**  $n$  : nombre de machines ;  $a$  : nombre aléatoire ( $0 \leq p \leq n$ ) ;

**Début** Générer un nombre aléatoire  $p$

Permuter les emplacements des deux groupes de tâches de production qui se trouvent respectivement entre la position 0 et  $p$  (groupe 1) et entre  $p+1$  et  $n$  (groupe 2).

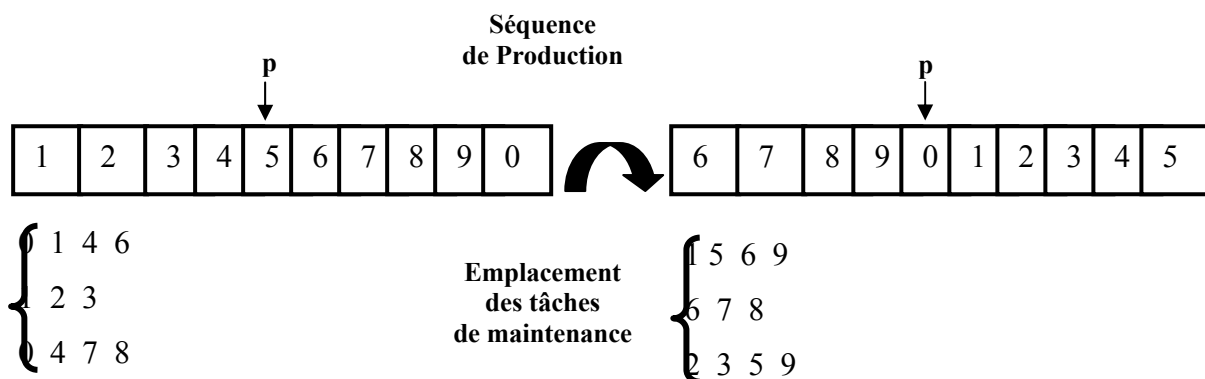
Permuter les emplacements des tâches de maintenance de manière que les emplacements des tâches qui se trouvent dans chaque groupe reste les mêmes par rapport aux tâches de production qui se trouvent dans le même groupe.

**Fin.**

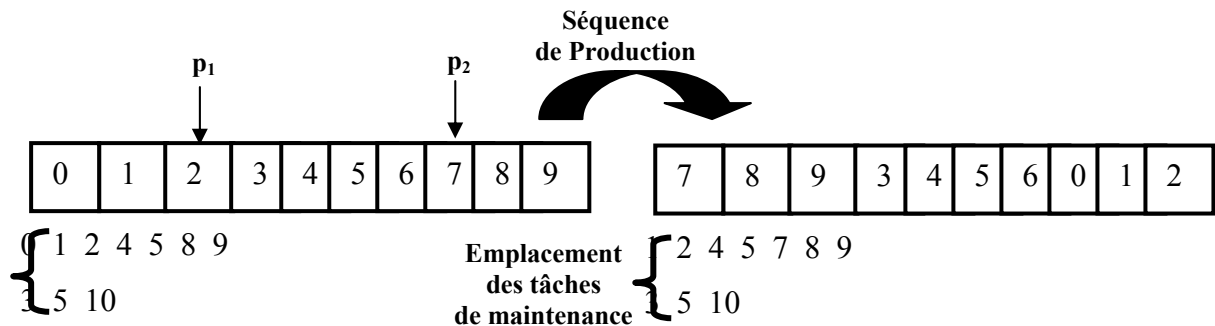
---

Algorithme 3.13: Mutation verticale à un point

L'exemple suivant illustre le principe de cet opérateur pour 3 machines et 10 tâches avec  $p = 4$



Le principe de la mutation verticale à deux points est le même que celui de la mutation verticale à un point. L'exemple suivant illustre le principe de cet opérateur avec 10 tâches et 2 machines pour :  $p_1=2$  et  $p_2=7$



Dans ce qui suit, nous allons présenter les tests que nous avons effectués sur des benchmarks de Taillard [TAI93] ainsi que les résultats de ces tests.

#### 4.5. Tests et résultats

Nous présentons les résultats de la stratégie intégrée en deux volets : le premier concerne la déviation des performances des heuristiques de l'approche constructive par rapport à l'optimum relatif de l'heuristique NEH, et dans le second la comparaison entre les approches constructive, itérative et évolutive.

La figure 3.14 présente la déviation des heuristiques de l'approche constructive par rapport à l'optimum relatif fournit par l'heuristique NEH.

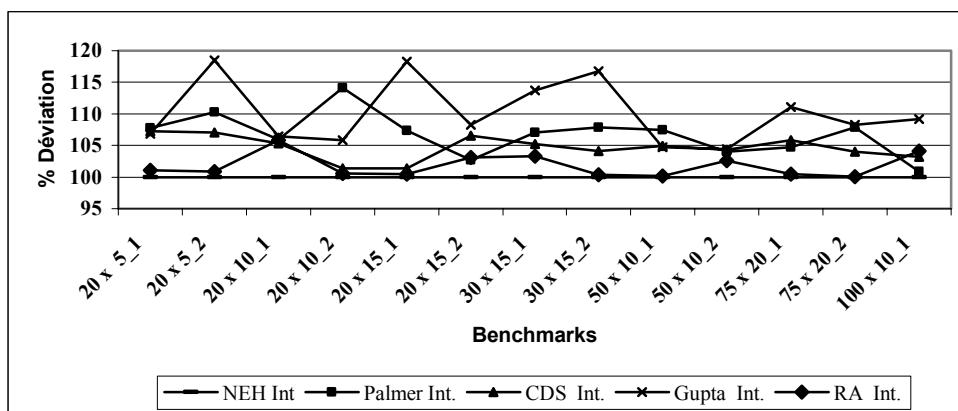


Figure 3.14 : Déviation par rapport à l'heuristique NEH

Pour la stratégie intégrée, le constat reste le même, plus de 80% des meilleurs résultats sont donnés par l'heuristique NEH, qui reste donc l'heuristique qui donne les meilleures performances même pour l'ordonnancement conjoint Production/Maintenance par la stratégie intégrée. La déviation varie dans l'intervalle [0, 20%] et de même que pour la stratégie séquentielle, l'heuristique de Gupta est celle qui dévie le plus par rapport à l'optimum relatif fourni par l'heuristique NEH. L'heuristique RA est celle qui dévie le moins.

Avant de présenter la partie de l'étude relative à la comparaison entre les approches dédiée, itérative et évolutive, nous avons effectué deux études préalables à cela. Ces études portent sur l'efficacité des nouvelles stratégies et des nouveaux opérateurs, que nous avons définis pour aborder le problème par la stratégie intégrée. Nous avons effectué des tests sur les mêmes benchmarks définis pour l'ensemble de l'étude [TAI93] pour la recherche Tabou et les AG afin de déterminer les meilleurs paramètres à adopter.

Le tableau 3.2 présente les meilleurs résultats par stratégies pour la recherche Tabou.

Paramètres de la méthode Tabou	Génération de la solution initiale		Génération du voisinage par décalage	
	Aléatoire	Fournie	sur la Production	sur la Maintenance
Pourcentage des meilleurs cas	21%	79%	43%	57%

Tableau 3.2 : Résultats par paramètre de la recherche Tabou intégrée

La solution initiale fournie a donné les meilleurs résultats du fait que cette solution est sous optimale vu qu'elle est générée par une heuristique constructive sur laquelle on insère la maintenance. Pour ce qui est du voisinage, la stratégie de décalage des tâches de maintenance a fournie les meilleurs résultats vu qu'une tâche de maintenance possède la latitude d'être exécutée sur l'ensemble des emplacements de son intervalle de tolérance.

Pour les AG, nous avons effectué 100 tests avec les mêmes paramètres en faisant appel à tous les opérateurs de croisement et de mutation que nous avons proposé dans le cas de la stratégie intégrée. Le tableau 3.3 présente les meilleurs résultats par opérateurs.

Opérateurs des AG	Croisement sur la maintenance		Mutation sur la maintenance		
	Horizontal à k points pairs	Horizontal à k points impairs	Aléatoire	Verticale à 1 point	Verticale à 2 points
Pourcentage des meilleurs cas	51%	49%	11%	57%	32%

Tableau 3.3: Résultats par opérateurs des AG intégrés

Pour les opérateurs de croisement sur la maintenance, nous ne pouvons affirmer que l'un est meilleur que l'autre. Le croisement horizontal à  $k$  points pair a fourni 51% des meilleurs résultats par rapport à l'opérateur à  $k$  points impairs. Nous pensons que les deux opérateurs sont techniquement aussi bon l'un que l'autre. Par contre, en ce qui concerne les opérateurs de mutation, l'opérateur de mutation verticale à 1 point a fourni, sans conteste, les meilleurs résultats. Le principe de permuter deux groupes dans la séquence  $S$ , et d'effectuer la même opération sur la matrice associée  $M$ , semble être efficace.

Le graphe de la figure 3.15 présente l'heuristique NEH (meilleure heuristique de l'approche constructive), la recherche Tabou et les AG. Nous avons exécuté la recherche Tabou avec les stratégies les mieux adaptées à notre problème, déduites du tableau 3.2. Et de même pour les AG, que nous avons exécutés avec les opérateurs, les mieux adaptés à notre problème, déduits du tableau 3.3.

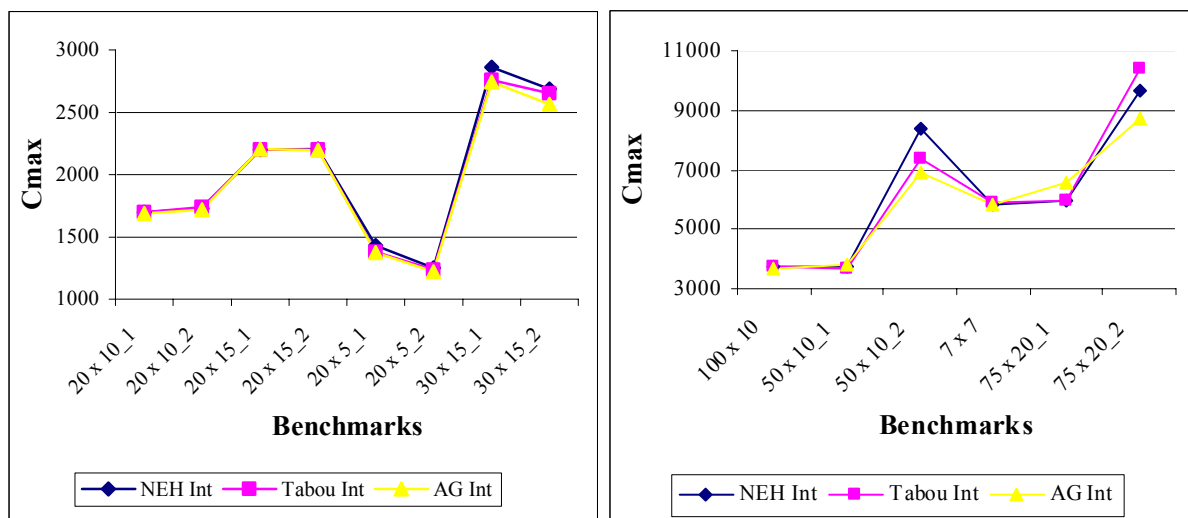


Figure 3.15 : Ordonnement conjoint P/M par la stratégie intégrée

On constate là encore que les AG fournissent globalement les meilleurs résultats, même dans le cas de l'ordonnement conjoint Production/Maintenance, puisque plus de 60% des meilleurs résultats sont obtenus par les AG intégrés et 20% par la méthode Tabou intégrée.

De plus la plus part des résultats sont obtenus en prenant l'heuristique HA comme méthode d'insertion des tâches de maintenance pour la génération de la solution initiale.

## 4.6. Synthèse

Dans cette section, nous avons présenté la résolution du problème de l'ordonnancement conjoint maintenance production par la stratégie intégrée, dans un atelier de type *flowshop* de permutation. A la différence de la stratégie séquentielle, elle résout le problème, de manière simultanée, en une seule phase, en considérons les tâches de production et de maintenance en même temps.

Notre contribution à ce niveau touche deux aspects : le premier concerne le principe même de la stratégie intégrée qui consiste en la prise en compte, lors de la résolution du problème, des deux types de tâches (production et maintenance), cependant la structure d'un ordonnancement adopté pour la stratégie séquentielle (section 3.3.1.1) s'avère inadéquate pour cette stratégie. Nous avons donc proposé une nouvelle structure à deux champs : une séquence de production qui représente l'ordre d'exécution des tâches de production à laquelle on associe une matrice qui représente les emplacements d'insertion des tâches de maintenance dans la séquence de production. Le deuxième aspect concerne les approches de résolution : une première contribution pour les heuristiques dédiées au *flowshop* de permutation, qui consiste à définir un ordre de précédence entre toutes les tâches, qu'elles soient de production ou de maintenance. L'approche que nous avons adoptée se base sur la principe que les tâches de maintenance préventive systématique sont périodiques et sont planifiées systématiquement toutes les  $T$  unités de temps ( $T$  variant dans l'intervalle  $[T_{\min}, T_{\max}]$ ). Partant de cet état, on prend en compte uniquement la première tâche de maintenance, sur chaque machine, lors de la génération de l'ordonnancement conjoint production/maintenance. Puis on insère les tâches restantes selon l'heuristique HA (§ 3.2.1.2). La seconde concerne la recherche Tabou et les AG, à travers la définition d'un paramétrage adéquat pour la stratégie intégrée. Nous avons proposé, dans le cas de la recherche Tabou, de nouvelles stratégies pour la génération d'une solution initiale conjointe, ainsi que pour l'exploration du voisinage. Pour les AG, comme nous manipulons une séquence conjointe de tâches de production et de maintenance, le croisement et la mutation peuvent se faire aussi bien sur la production que sur la maintenance. Nous avons, donc, proposé de nouveaux opérateurs de croisement et de mutation sur la maintenance. Les opérateurs de croisement et de mutation sur la production restent les opérateurs classiques de la méthode.

En terme de résultats, on peut dire que dans le cas de l'ordonnancement conjoint par la stratégie intégrée, les AG ont donné les meilleurs résultats par rapport à la recherche Tabou et l'heuristique NEH.



L'étude effectuée sur les opérateurs des AG a permis de dégager les meilleurs opérateurs parmi ceux que nous avons proposés dans le cas de la stratégie intégrée. Nous avons donc exécuté les AG intégrés en effectuant un croisement horizontal à  $k$  points pairs et une mutation verticale à un point. Pour ce qui est de la recherche Tabou nous avons exécuté la méthode en générant la solution initiale par la stratégie aléatoire, et le voisinage par décalage des tâches de production.

## 5. La stratégie hybride

La force d'un algorithme hybride réside dans la combinaison de deux principes de recherche fondamentalement différents, pour mettre à profit les avantages des deux types de recherche. En complément de cette étude, nous proposons deux méthodes hybrides : un algorithme génétique hybride et la recherche Tabou hybride.

### 5.1. Un algorithme génétique hybride

Cet algorithme est une extension de l'algorithme génétique développé pour la résolution du problème de l'ordonnement de la production. La nouvelle génération est formée après l'insertion des tâches de maintenance. L'idée de cette hybridation consiste à manipuler des séquences de production, et à chaque itération de l'algorithme génétique, insérer sur les séquences de production, les tâches de maintenance. Les opérateurs de reproduction (croisement et mutation) s'appliquent uniquement sur les séquences de production.

La génération de la population initiale, la sélection, le croisement, la mutation et le remplacement sont définis de la même façon que pour l'algorithme génétique développé pour l'ordonnement de la production (§ 3.3.1.4). Seules les phases évaluation et renouvellement se feront après insertion des tâches de maintenance (Fig. 3.16). Le principe de cette démarche est présenté dans l'algorithme 3.14.

---

#### **Algorithme AG hybride**

---

##### **Début**

Génération de la population initiale

Insertion des tâches de maintenance et Evaluation des individus conjoints

Sélection, Croisement, Mutation

Insertion des tâches de maintenance et Evaluation des individus conjoints

Renouvellement et retour à la phase sélection.

##### **Fin.**

---

Algorithme 3.14 : AG hybride

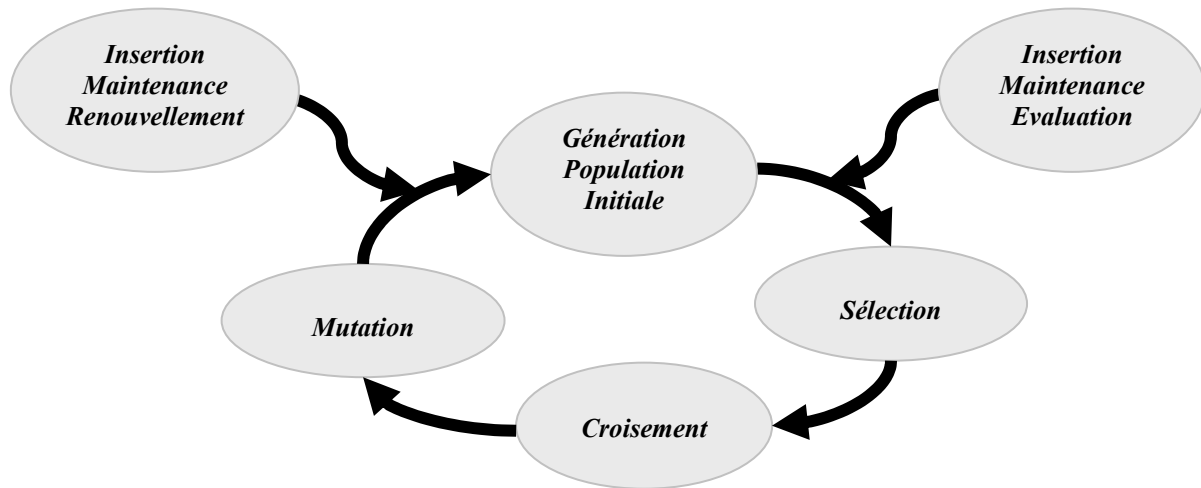


Figure 3.16: Principe de fonctionnement de l'AG hybride

Pour chaque individu de la population sélectionnée, on insère toutes les tâches de maintenance selon deux stratégies : Déterministe ou Aléatoire.

Dans ce qui suit nous allons présenter ces deux stratégies.

#### **a) Stratégie déterministe**

L'heuristique d'insertion des tâches de maintenance sur les séquences de production, est la même pour toutes les itérations de l'algorithme. Elle peut être effectuée par : l'heuristique HN (selon  $HN^*$ ,  $HN_{\max}$  ou  $HN_{\min}$ ), l'heuristique HA, ou l'heuristique HD. Pour les heuristiques HA et HD, on peut tolérer l'avance et le retard des tâches de maintenance ou bien ne pas le tolérer. Ces heuristiques sont détaillées dans la section 3.2. de ce chapitre.

#### **b) Stratégie aléatoire**

Cette stratégie permet de guider la recherche vers d'autres régions de l'espace de recherche, et cela en changeant la méthode d'insertion des tâches de maintenance sur les séquences de production. Contrairement à la stratégie déterministe, l'insertion des tâches de maintenance n'est pas la même pour toutes les itérations, elle est déterminée de façon aléatoire de la manière suivante (Algorithme 3.15):

---

**Algorithme** Stratégie aléatoire d'insertion de la maintenance
 

---

**Variables locales**  $r$  : nombre aléatoire ;  $r \in \{0, 1, 2\}$ ;

**Début**
**Pour** Chaque évaluation d'un individu

**Faire** Générer un nombre aléatoire  $r$ 

     **Si**  $r = 0$  **Alors** Utiliser l'heuristique HN

     **Si**  $r = 1$  **Alors** Utiliser l'heuristique HA

     **Si**  $r = 2$  **Alors** Utiliser l'heuristique HD.

**Fin Pour**
**Fin.**


---

Algorithme 3.15: Stratégie aléatoire d'insertion de la maintenance

## 5.2. La recherche Tabou hybride

Cette méthode consiste à hybrider la méthode Tabou avec l'algorithme génétique et l'heuristique NEH.

Les heuristiques HN, HA et HD placent les tâches de maintenances sur la séquence de production sans tenir compte des prochaines tâches à insérer. Dans ce cas, le choix d'un meilleur emplacement pour une tâche, se fait avant l'insertion de la totalité des tâches. Ce choix peut ne plus être le meilleur après l'insertion des tâches restantes. L'hybridation de la recherche Tabou avec l'algorithme génétique et l'heuristique NEH, a pour but d'améliorer la solution obtenue par ces dernières méthodes, en repositionnant quelques tâches de maintenance afin de trouver leurs meilleurs emplacements (Fig. 3.17).

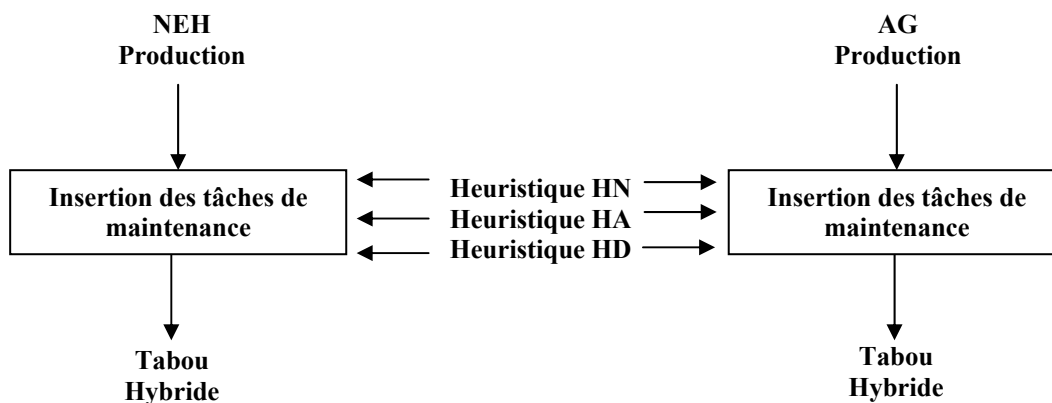


Figure 3.17 : Hybridation de Tabou

### 5.3. Tests et résultats

Le graphe de la figure 3.18 présente les résultats des tests effectués pour la stratégie hybride sur les algorithmes génétiques et la recherche Tabou. On constate que la recherche Tabou est meilleure que les AG, pour les types d'hybridation présentés.

Généralement les méthodes hybrides donnent les meilleurs résultats, le principe de combiner deux types de recherche différents est à l'origine de ces bons résultats. La technique de l'hybridation présentée dans la section 3.3 et qui consiste à fournir à une méthode quelconque une solution de départ de bonne qualité, pour pouvoir commencer la recherche a donné de bons résultats dans le cas de la recherche Tabou.

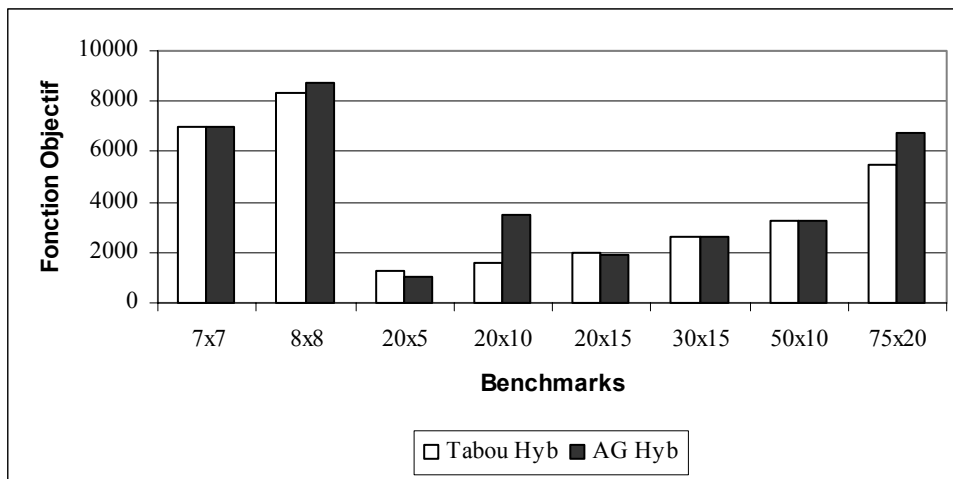


Figure 3.18: Ordonnancement conjoint P/M par la stratégie hybride

La plupart des meilleurs résultats sont trouvés par l'algorithme génétique hybride. Le problème de l'algorithme génétique hybride est le temps de calcul qui est considérable par rapport aux autres méthodes, et la difficulté de trouver, pour chaque type de problème, les bons paramètres qui permettent de trouver une bonne solution.

## 6. Etude comparative

Nous avons présenté après chaque stratégie d'ordonnancement conjoint production/maintenance (séquentielle et intégrée) des tests sur des benchmarks de Taillard [TAI93] et les résultats associés ainsi que la déviation des heuristiques de chaque approche par rapport à l'optimum réalisé par la meilleure heuristique. L'étude que nous présentons dans cette section compare les approches (constructive, itérative et évolutive). Ainsi les meilleures

heuristiques, par approche, sont comparées pour déterminer la meilleure heuristique pour la résolution du problème de l'ordonnancement conjoint production/maintenance.

Il serait intéressant de voir à ce niveau la perte de performance subie par la séquence de production après insertion des tâches de maintenance pour chacune des stratégies séquentielle et intégrée, dans le cas de NEH (meilleure heuristique de l'approche constructive), la recherche Tabou et les AG vu qu'ils optimisent au mieux la séquence de production. Nous présentons dans les figures 3.19, 3.20 et 3.21 cette perte de performance.

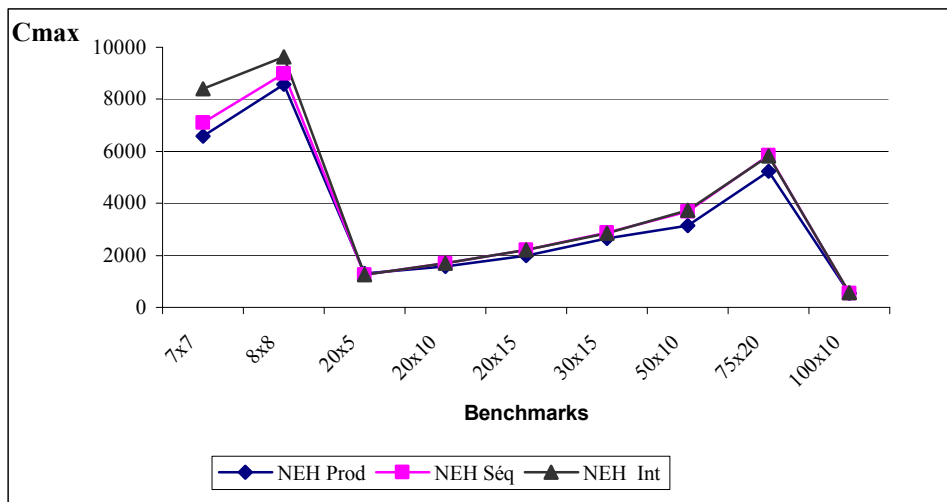


Figure 3.19 : Perte de performance de l'heuristique NEH.

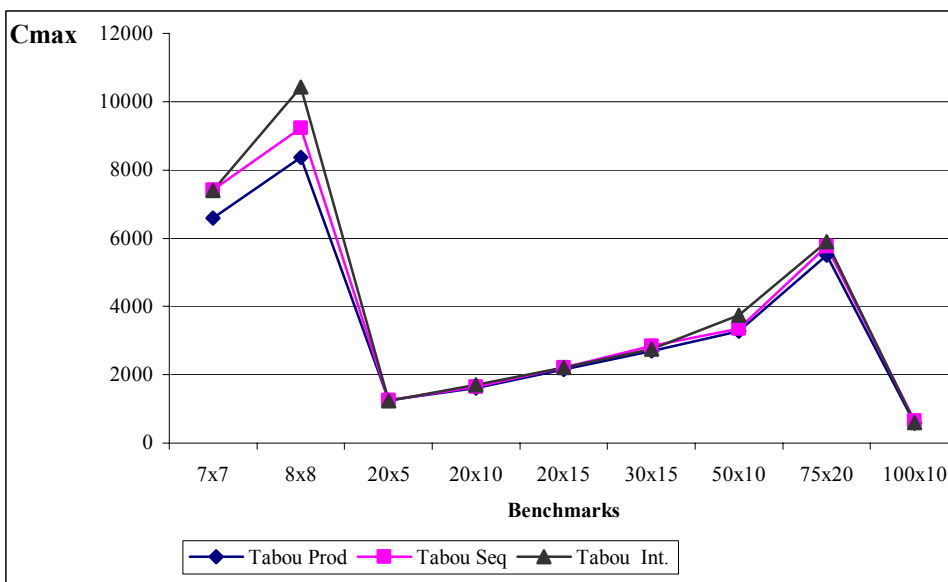


Figure 3.20: Perte de performance de la recherche Tabou

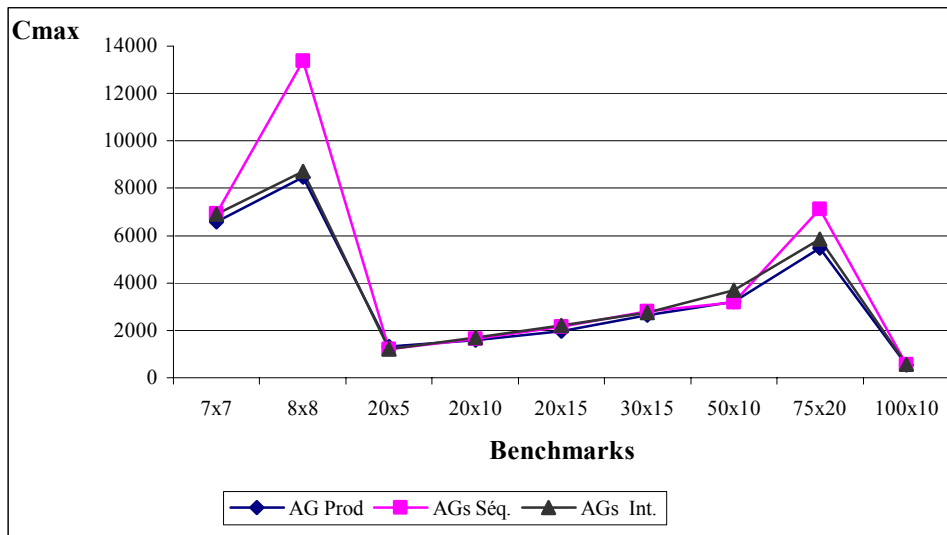


Figure 3.21: Perte de performance de l'algorithme génétique

Pour la stratégie séquentielle, on remarque que le décalage entre les séquences de production avant et après insertion des tâches de maintenance reste acceptable, tant que les benchmarks sont de petite et moyenne taille (moins de 10 machines). Au-delà, le décalage est très important, ceci s'explique par le fait que plus il y a de machines, et plus l'insertion des tâches de maintenance provoque des décalages dans la séquence de production et donc la perte de performance est très importante.

Pour la stratégie intégrée, le constat reste le même, la perte de performance reste acceptable pour les benchmarks de petite et moyenne taille. Pour les benchmarks de grande taille l'ordonnancement devient caduc. L'insertion des tâches de maintenance provoque des décalages très importants.

Le tableau 3.4 présente un récapitulatif des meilleures performances pour la première étape de la stratégie séquentielle : l'étape de l'ordonnancement de production pour les heuristiques NEH, la recherche Tabou et AG. Partant de ce constat nous identifions les meilleurs écarts par rapport à l'ordonnancement de production, pour les stratégies séquentielle (Etape insertion de la maintenance) et intégrée.

On constate que la stratégie intégrée est meilleure que la stratégie séquentielle. Ce constat est dû au fait que la stratégie séquentielle se déroule en deux étapes, une première étape pour obtenir un ordonnancement de production, qui constitue une contrainte forte, ensuite l'insertion de la maintenance. Par contre, pour la stratégie intégrée, le processus de recherche d'un ordonnancement conjoint tient compte dès le départ des contraintes de production et de maintenance.

Benchmarks	Production		Stratégie Séquentielle		Stratégie Intégrée	
	Production	Algorithme	Écart (%)	Algorithme	Écart (%)	Algorithme
7x7	6590	AG	3,35%	AG	3,06%	AG
8x8	8366	Tabou	6,22%	NEH	3,54%	AG
20x5	1249	Tabou	-0,47%	AG	-0,27%	AG
20x10	1603	AG	0,37%	AG	0,83%	AG
20x15	1972	AG	19,4%	AG	22,8%	AG
30x15	2643	AG	15,3%	AG	10,2%	AG
50x10	3153	NEH	0,36%	AG	5,45%	AG
75x20	5227	Tabou	5,41%	NEH	5,99%	Tabou
100x10	530	AG	0,22%	NEH	0,25%	NEH

Tableau 3.4: Tableau récapitulatif des meilleurs écarts par rapport à la production

L'algorithme génétique séquentiel est globalement meilleur par rapport à l'heuristique NEH séquentielle et la recherche Tabou séquentielle. La recherche Tabou a fourni les plus mauvais résultats. Cela est dû au fait qu'elle commence par une solution aléatoire qui converge vers un optimum local. Cette méthode devient intéressante en l'exécutant en hybridation.

L'algorithme génétique intégré a fourni sept meilleurs résultats sur les neuf problèmes traités. L'heuristique NEH a fourni un meilleur résultat et la méthode Tabou a fourni meilleur un résultat également.

Les meilleurs résultats pour chaque approche (constructive, itérative et évolutive) ainsi que pour chacune des deux stratégies (séquentielle et intégrée) sont comparés dans le schéma de la figure 3.22.

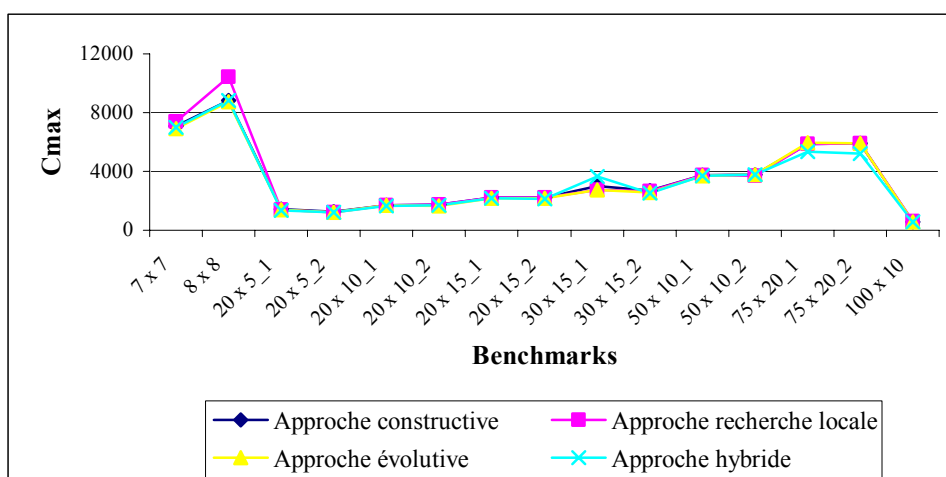


Figure 3.22: Comparaison entre toutes les approches

On remarque d'après le graphe que neuf résultats sont obtenus par l'approche hybride, quatre résultats par l'approche évolutive, un résultat par l'approche recherche locale et un résultat par l'approche constructive.

On constate que même pour l'ordonnancement conjoint production/maintenance, l'approche hybride est celle qui optimise au mieux le problème traité. Le principe de combinaison de deux types de recherche différents donnent les meilleurs résultats pour le problème de l'ordonnancement conjoint production/maintenance.

## 7. Conclusion

Dans ce chapitre, nous avons étudié le problème de l'ordonnancement conjoint de la production et de la maintenance préventive systématique, dans un *flowshop* de permutation. Le but est d'optimiser une fonction objectif qui prend en compte les critères d'optimisation de la production et ceux de la maintenance. Pour le résoudre, nous nous sommes intéressés aux heuristiques dédiées à ce type d'atelier (RA, CDS, GUPTA, Palmer et NEH), les algorithmes génétiques et la recherche Tabou qui ont prouvé leurs aptitudes de résolution en ordonnancement de production. De plus, nous avons abordé ce problème selon les deux stratégies d'ordonnancement conjoint les plus intéressantes : la stratégie intégrée et la stratégie séquentielle. Cette démarche est illustrée par la figure 3.23.

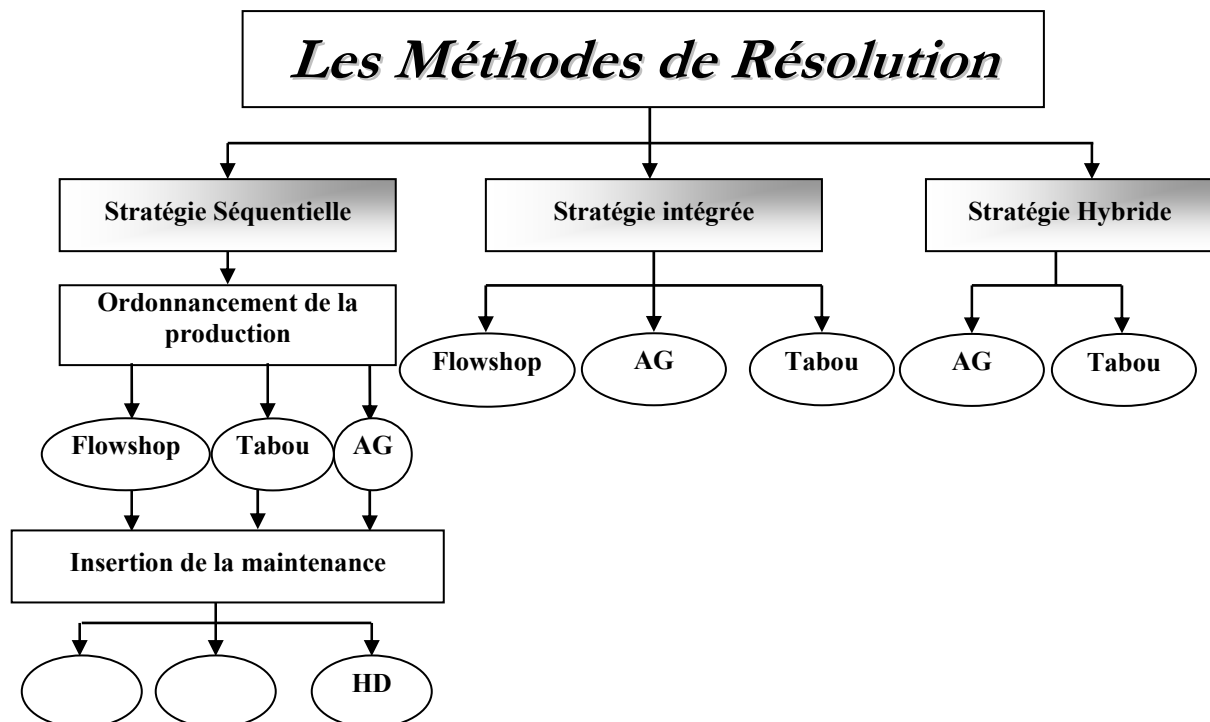


Figure 3.23 : Démarche de résolution



Une étude comparative entre toutes les méthodes implémentées a montré que la stratégie intégrée semble meilleure que la stratégie séquentielle du fait de la prise en compte simultanée des tâches de production et de maintenance. C'est ce qui explique entre autre qu'un bon ordonnancement de production, ne le reste pas systématiquement après insertion de la maintenance, dans le cas de la résolution par la stratégie séquentielle. L'optimisation en deux phases de la stratégie séquentielle : une première optimisation par rapport à la production pour obtenir une séquence de production, puis une seconde par rapport à la maintenance pour obtenir une séquence conjointe production/maintenance, détériore sensiblement la qualité de l'ordonnancement de production. Dans le cas de la stratégie intégrée, l'optimisation simultanée de la production et de la maintenance donne de meilleurs résultats.

Notre apport concerne deux aspects. D'une part le développement d'heuristiques pour l'insertion de la maintenance, dans le cas de la stratégie séquentielle. Nous avons proposé l'heuristique HRP pour l'insertion dans le cas à une machine et les heuristiques HN, HA et HD pour le cas à plusieurs machines. Force est de constater que l'incidence de l'insertion systématique (cas de l'heuristique HN avec ses trois variantes) sur la qualité de l'ordonnancement de production est négative. Par contre, la relaxation de cette contrainte, en autorisant la recherche du meilleur emplacement pour l'insertion de la maintenance, à la périphérie de l'intervalle de tolérance (cas des heuristiques HA et HD), induit une nette amélioration de la qualité de l'ordonnancement de production.

D'autre part, pour la résolution par la stratégie intégrée nous avons proposé d'une part une nouvelle structure à deux champs pour la représentation d'un ordonnancement conjoint : une séquence de production qui représente l'ordre d'exécution des tâches de production à laquelle on associe une matrice qui représente les emplacements d'insertion des tâches de maintenance dans la séquence de production. D'autre part, pour chaque approche de résolution, nous avons proposé une adaptation adéquate. Ainsi nous avons adapté les heuristiques de l'approche constructive en proposant un ordre de précedence entre toutes les tâches de production ou de maintenance, de nouveaux paramètres qui intègrent l'aspect maintenance dans le cas de la recherche Tabou, et enfin de nouveaux opérateurs génétiques dans le cas des AG. Dans le but d'améliorer les solutions trouvées par les différentes méthodes développées, une hybridation est proposée entre ces dernières.

Dans la majorité des cas, toutes stratégies confondues, les AG ont donné les meilleurs résultats, viennent ensuite la recherche Tabou et les heuristiques dédiées. Ceci s'explique par le principe de recherche de chaque méthode en lui-même. En effet le principe de recherche à

la base de l'approche évolutive est plus performant que celui de l'approche itérative et constructive.

L'objectif du chapitre suivant sera l'étude de la robustesse de ces ordonnancements conjoints production/maintenance. Notre souci majeur reste l'étude de l'impact de l'insertion de la maintenance dans les ordonnancements de production. Cet impact a été étudié sous l'angle du rapport gain en sûreté de fonctionnement, performance. Dans le chapitre suivant, nous allons essayer de l'étudier sous l'angle de la contribution à la robustesse des ordonnancements conjoints par rapport aux ordonnancements de production dans un milieu incertain.

---

# CHAPITRE IV

## *Robustesse des Ordonnancements Conjointes Production/Maintenance*

---



---

## ***CHAPITRE IV.***

### ***Robustesse des ordonnancements conjoints Production/Maintenance***

---

---

***Résumé :*** *L'objectif de ce chapitre est d'évaluer la contribution de la maintenance à la robustesse des ordonnancements conjoints Production/Maintenance (§ Chap. 3) pour démontrer que la perte de performance due à l'insertion de la maintenance dans l'ordonnancement de production est compensée par accroissement de la robustesse. Nous étudierons deux autres points : le premier relatif aux critères assurant un certain degré de robustesse, s'ils sont optimisés. Le second est relatif aux méthodes d'ordonnancement robuste, qui génèrent des solutions robustes, en milieu incertain.*

*Nous y présentons un état de l'art succinct sur la robustesse des ordonnancements ainsi que les définitions retenues, dans la littérature, de ce qu'est la robustesse. Nous donnons une classification des perturbations puis un tour d'horizon des approches d'ordonnancement en présence de perturbations. Nous les classons en trois catégories : approches réactives, approches proactives ou robustes et approches mixtes.*

*Nous présentons par la suite la problématique de ce chapitre relatif à la robustesse et la maintenance. Nous y définissons les perturbations, ainsi que les niveaux et classes associés, pris en compte lors des tests, puis les critères d'optimisations retenues. La dernière partie est consacrée aux tests. Enfin nous concluons par les résultats de cette dernière étape.*

---

## 1. Introduction

Les heuristiques étudiées, dans le chapitre précédent, se préoccupent seulement de trouver une solution optimisée dans un état bien déterminé où les données du problème sont connues dès le départ. Mais on s'est rapidement rendu compte que ces mêmes heuristiques d'ordonnement ne donnent le résultat escompté que dans un cas déterministe. Et contrairement à l'hypothèse, souvent faite dans les études traitant de problèmes d'ordonnement, l'environnement réel d'application de l'ordonnement est très rarement, sinon jamais aussi déterministe que le sont les hypothèses du modèles de résolution, et de ce fait, les données ne sont connues que partiellement ou totalement inconnues.

La meilleure façon d'éviter ces incertitudes est de les intégrer à l'ordonnement lui même, et de les incorporer dès la résolution du problème. Les premières études dans le domaine ont portées sur la *sensibilité* des ordonnements [GRA69] pour étudier l'influence d'une perturbation de données sur la solution. Après cela, d'autres approches ont vu le jour et qui consistent à réagir pendant la recherche de l'ordonnement pour corriger les erreurs éventuelles et adapter la solution aux nouvelles données si le besoin en est ressenti. La tendance actuelle s'oriente vers l'introduction de *la flexibilité* dans le processus de construction des ordonnements. La réponse minimale consiste à proposer un algorithme déterministe et à mesurer la sensibilité des ordonnements construits face aux aléas : c'est l'étude de la *robustesse* [JEN01]. Si les performances s'avèrent insuffisantes ou les aléas trop perturbateurs, il faut se donner davantage de liberté : *stabiliser* l'ordonnement proposé.

L'ensemble des travaux portant sur l'étude de la robustesse des ordonnements se basent sur deux facteurs importants : les perturbations prises en compte et la mesure de robustesse utilisée. En général, les perturbations les plus étudiées sont les pannes machines et l'incertitude des durées opératoires. Quant aux mesures de robustesse, les mêmes auteurs suivent en général l'une des deux classes d'expériences suivantes : la première utilise une seule mesure de robustesse avec plusieurs méthodes de résolution (les approches de réordonnement sont les plus utilisées), alors que la seconde utilise une ou peu de méthodes de résolution mais portent sur plusieurs mesures de robustesse. Le point commun aux deux classes précédentes est que la mesure de robustesse est connue à l'avance. De cette littérature le terme maintenance est inexistant. Ces travaux s'intéressent essentiellement aux ordonnements de production.

L'approche que nous proposons pour ce chapitre est une approche qui tient compte de la présence de perturbations lors du calcul d'un ordonnancement conjoint production/maintenance. Notre objectif est, d'une part, évaluer la contribution de la maintenance à la robustesse des ordonnancements conjoint Production/Maintenance. Et d'autre part d'essayer d'identifier, parmi les méthodes d'ordonnements conjoints développées dans le chapitre précédent, celles susceptibles de générer des ordonnancements robustes. Enfin d'identifier parmi les critères d'optimisation classiques de potentiels critères de robustesse, qui permettraient, s'ils sont optimisés lors de la génération de l'ordonnement, d'assurer une certaine robustesse de ce dernier face à des incertitudes lors de son exécution.

Ce chapitre est organisé en trois grandes parties. Dans la première partie, nous présentons un état de l'art succinct sur la robustesse en ordonnancement de production. Nous y présentons des définitions retenues, dans la littérature, de ce qu'est la robustesse, les méthodes d'ordonnements en présence de perturbations ainsi qu'une classification des perturbations. Nous présenterons par la suite la problématique de ce chapitre relatif à la robustesse et la maintenance. Nous y définissons le modèle que nous proposons pour cette étude, les perturbations prises en compte. Puis les critères d'optimisations retenues pour les tests que nous avons effectué. Enfin nous concluons le chapitre par les résultats retenus de cette dernière.

## **2. Sensibilité, Robustesse et Flexibilité**

Le terme robustesse paraît être bien établi dans la littérature, tandis que la flexibilité n'a été introduite que récemment par [BRA00, JEN00]. Malgré l'utilisation fréquente de la robustesse par les auteurs, sa signification reste néanmoins variable d'un auteur à un autre.

La recherche de robustesse est une préoccupation de plus en plus présente en Recherche Opérationnelle et Aide à la Décision (RO-AD) où elle donne lieu à des démarches et concepts de plus en plus riches et diversifiés. Même si ce qualificatif a généralement le sens de « *qui résiste à l'a peu près* » [ROY04], il recouvre pourtant des significations différentes et parfois obscures. Les « à-peu-près » peuvent se rapporter à la façon de modéliser, au caractère restrictif de certaines hypothèses, au mode d'attribution de valeurs à des données et/ou des paramètres, ...

Le recours à cette notion pour éclairer des décisions soulève de multiples questions. Tout d'abord qu'est ce qui doit être robuste ? Corrélativement, vis-à-vis de quoi, vis-à-vis de quel à peu près, cette robustesse doit-elle être appréciée ? Enfin, pourquoi cette préoccupation de robustesse a-t-elle de l'importance ?

En RO-AD, la notion de robustesse est souvent apparentée (parfois même assimilée) à celle de flexibilité, sensibilité et même, dans certains cas, équité. Dans ce qui suit nous rapportons certaines définitions d'auteurs des termes de sensibilité, robustesse et flexibilité, dans le contexte de l'ordonnement de production.

### **- Sensibilité**

C'est l'étude de la dégradation de la solution par rapport à la solution optimale quand les données sont sujettes à des perturbations. Graham a effectué la première analyse sur la sensibilité en 1969 [GRA69]. L'analyse de sensibilité tente de répondre aux questions « Que se passe-t-il si ... ? ». Elle s'intéresse aux aléas plus qu'à l'incertitude en tant que telle : les données sont fixées mais peuvent être perturbées, et il existe une solution initiale, un ordonnancement de référence, le plus souvent optimal. L'analyse de sensibilité cherche à mesurer la dégradation de performance si un aléa identifié se produit ; la phase d'exécution n'est pas son objet, il s'agit de mesurer la robustesse d'un algorithme statique. Hall & al [HAL04] présentent une classification et de nombreux résultats pour l'analyse de sensibilité en ordonnancement. Il est possible bien sûr d'étendre l'analyse à une incertitude véritable sur les données prises simultanément [PEN01], tant que l'étude est restreinte à la phase statique et à certaines catégories d'aléas (il paraît impossible par exemple de prendre en compte les pannes).

### **- Robustesse**

La robustesse généralise la sensibilité. Il s'agit de modéliser les incertitudes et d'en tenir compte lors de la prise de décision. En effet les incertitudes ne sont pas de simples déviations temporelles des données, mais font partie du futur. La meilleure manière de manipuler les incertitudes et par conséquent de prendre des décisions dans l'incertain est de les modéliser et les considérer à part entière dans les modèles de résolution. L'objectif alors est de trouver des solutions dont les performances sont peu sensibles aux perturbations sur les données qui peuvent survenir. Plusieurs définitions de la robustesse sont données, selon le contexte et le problème.



Dans la littérature, les auteurs ont des définitions différentes de la robustesse. Ainsi dans le domaine des statistiques, Huber [HUB81] caractérise la robustesse comme étant une insensibilité à toutes déviations par rapport aux hypothèses. D'un point de vue de la conception de produit, la notion de robustesse fait référence à l'insensibilité relative des performances fonctionnelles d'un produit, lorsque ses conditions de fabrication et d'utilisation dévient de leurs spécifications [KAC85]. En recherche opérationnelle et plus particulièrement dans le contexte de l'aide à la décision (RO-AD), Roy [ROY97] propose le concept de conclusion robuste qui vise à élaborer des éléments de réponse à un problème auquel est confronté un décideur, éléments de réponses qui sont obtenus en prenant en compte les incertitudes éventuelles sur les paramètres du problème. Dans la théorie de la décision, l'action la plus robuste est celle qui ne conduit pas à une perte importante quelque soit l'événement qui va se produire [POM01].

Le qualificatif « robuste » s'applique à des objets aussi variés que données, méthode ou alors solution :

- La robustesse des données peut être vue comme leur degré de fiabilité et d'adéquation avec les données réelles.
- Une méthode est dite robuste si elle génère un ordonnancement de hautes performances en présence d'incertitudes (temps de traitement, la disponibilité des machines...etc.).
- Par contre une solution est dite robuste si elle est à la fois optimisée en performance et en robustesse. Pour cela, on doit boucler phases de conception et d'évaluation des performances et de la robustesse. On est alors amené à effectuer un compromis entre robustesse et performance [CHE02]. La robustesse peut être définie, à ce niveau, comme la capacité à préserver certaines propriétés d'intérêt face à des perturbations.

En ordonnancement de production, une solution est dite robuste si elle est toujours admissible ou bien elle est toujours à au plus  $\epsilon\%$  de l'optimum [BIL05].

En ordonnancement de production le terme robuste est très utilisé surtout dans un environnement dynamique, dans lequel il est souvent nécessaire de modifier ou réordonnancer le plan existant à cause des différents aléas. Le Pape [PAP95] définit la robustesse d'un ordonnancement comme étant "*sa capacité à satisfaire des exigences de performance d'une façon prévisible*". Leon & al. [LEO94] définissent un ordonnancement robuste comme étant "*un ordonnancement insensible aux perturbations qui surgissent dans l'atelier pour une règle de pilotage donnée*". Jensen [JEN00] définit la robustesse comme étant "*une qualité de l'ordonnancement supposé rester acceptable si quelques changements imprévus se*

*produisent*". Enfin Shafaei & al [SHA99a] l'ont défini comme "*un moyen d'évaluer la performance actuelle des méthodes d'ordonnement dans un environnement stochastique*". Par contre pour Davenport & Beck [DAV00], il n'existe pas de définition d'ordonnement robuste mais plutôt une compréhension générale informelle de ce qu'est la robustesse (i.e., la capacité d'un ordonnancement prédictif d'être performant malgré des événements inattendus). Il est difficile de donner une définition type de la robustesse en ordonnancement de production, sachant que toutes les définitions citées précédemment sont différentes mais complémentaires. Nous proposons une définition consensuelle proposée par Billaut & al [BIL05] « *un ordonnancement est robuste si sa performance est peu sensible à l'incertitude des données et aux aléas* ». Il faut comprendre performance dans le sens très large de qualité de la solution pour le responsable, ce qui englobe naturellement la valeur de cette solution relativement à un critère donné, mais également la structure même de la solution proposée. La *robustesse* d'un ordonnancement caractérise donc cette performance.

### **- Flexibilité**

*"La flexibilité d'une solution d'un problème d'ordonnement peut être définie comme l'existence de modifications possibles sur la solution entraînant éventuellement, une perte de performance acceptable"* Groupe Flexibilité du Ghota, 2002 [GOT02].

Cette définition correspond à ce que Artigues et al [ART02] appellent « flexibilité statique ». On parle aussi de flexibilité dynamique qui correspond à la capacité de l'algorithme réactif à satisfaire les objectifs qui lui sont assignés, lorsque les contraintes fluctuent (suite aux aléas), en exploitant au mieux la flexibilité statique introduite proactivement.

On distingue plusieurs types (ou degrés) de flexibilité statique [GOT02] on parle de :

- Flexibilité sur *les temps* lorsque les dates de début et de fin peuvent varier ;
  - Flexibilité sur *les ordres* quand il existe plusieurs séquences admissibles permettant d'exécuter un ensemble de tâches sur une même ressource de différentes façons. Cette flexibilité entraîne forcément une flexibilité sur les temps ;
  - Flexibilité sur *les ressources* lorsqu'on peut changer l'affectation des ressources aux tâches ;
  - Flexibilité sur *les modes d'exécution* lorsqu'il est possible de changer le mode d'exécution d'une ou plusieurs tâches, par exemple, autoriser la préemption, la duplication, le changement de gamme, etc.

### **- Flexibilité & Robustesse**

Un préordonnancement qui a tendance à mieux s'exécuter que les ordonnancements ordinaires après l'occurrence de l'incertitude et l'introduction d'un simple décalage des tâches est dit *robuste*, pendant qu'un ordonnancement qui a tendance à mieux s'exécuter après l'occurrence de l'incertitude et l'utilisation des techniques de réordonnancement qui introduisent un changement dans l'ordre des tâches est dit *flexible*.

Il est difficile de joindre les termes Robustesse et Flexibilité car souvent un ordonnancement robuste peut être flexible à un certain degré. La robustesse veut dire que l'ordonnancement reste acceptable si des aléas perturbent l'exécution de l'ordonnancement. Tandis que la flexibilité signifie la possibilité d'introduire des changements dans l'ordonnancement.

Un exemple d'ordonnancement robuste est présenté dans la figure 4.1. Deux ordonnancements A et B sont soumis à une même panne sur la machine  $M_1$ . On remarque que le Makespan initial des deux ordonnancements est le même, mais après perturbation (panne) et l'application du Right Shifting (règle de décalage à droite), le nouveau Makespan obtenu plus petit que celui de B, ainsi on dira que A est plus robuste que B.

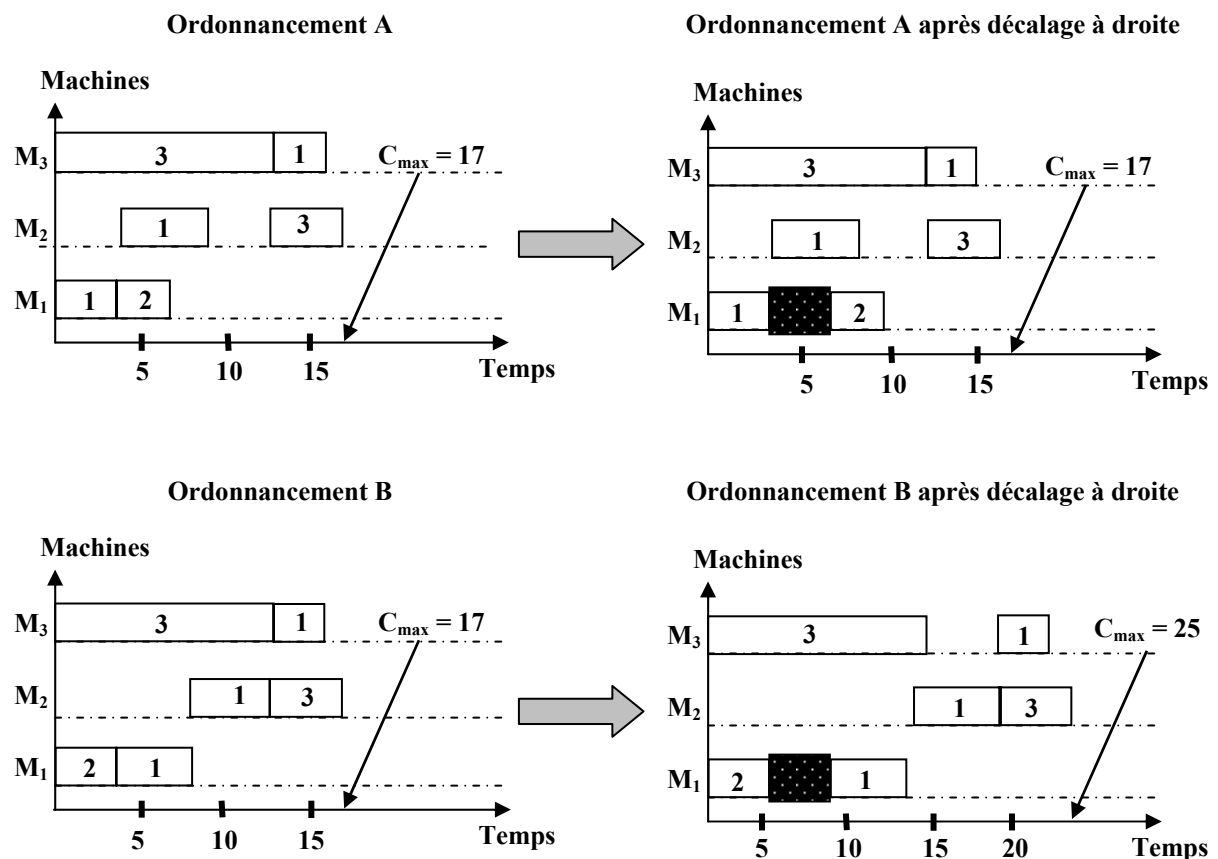


Figure 4.1 : Exemple d'un ordonnancement robuste [JEN01a]

Un exemple d'ordonnancement flexible est présenté dans la figure 4.2. De même, deux ordonnancements A et B de même Makespan, sont soumis à une même panne sur la machine

$M_3$ . L'unique différence entre eux est l'ordre des opérations 1 et 2 sur la machine  $M_3$ . Des méthodes de réordonnement (voisinage) ont été appliquées sur les ordonnancements A et B après apparition de la panne, et le Makespan obtenu par A est inférieur à celui de B. Ainsi A est dit plus flexible que B.

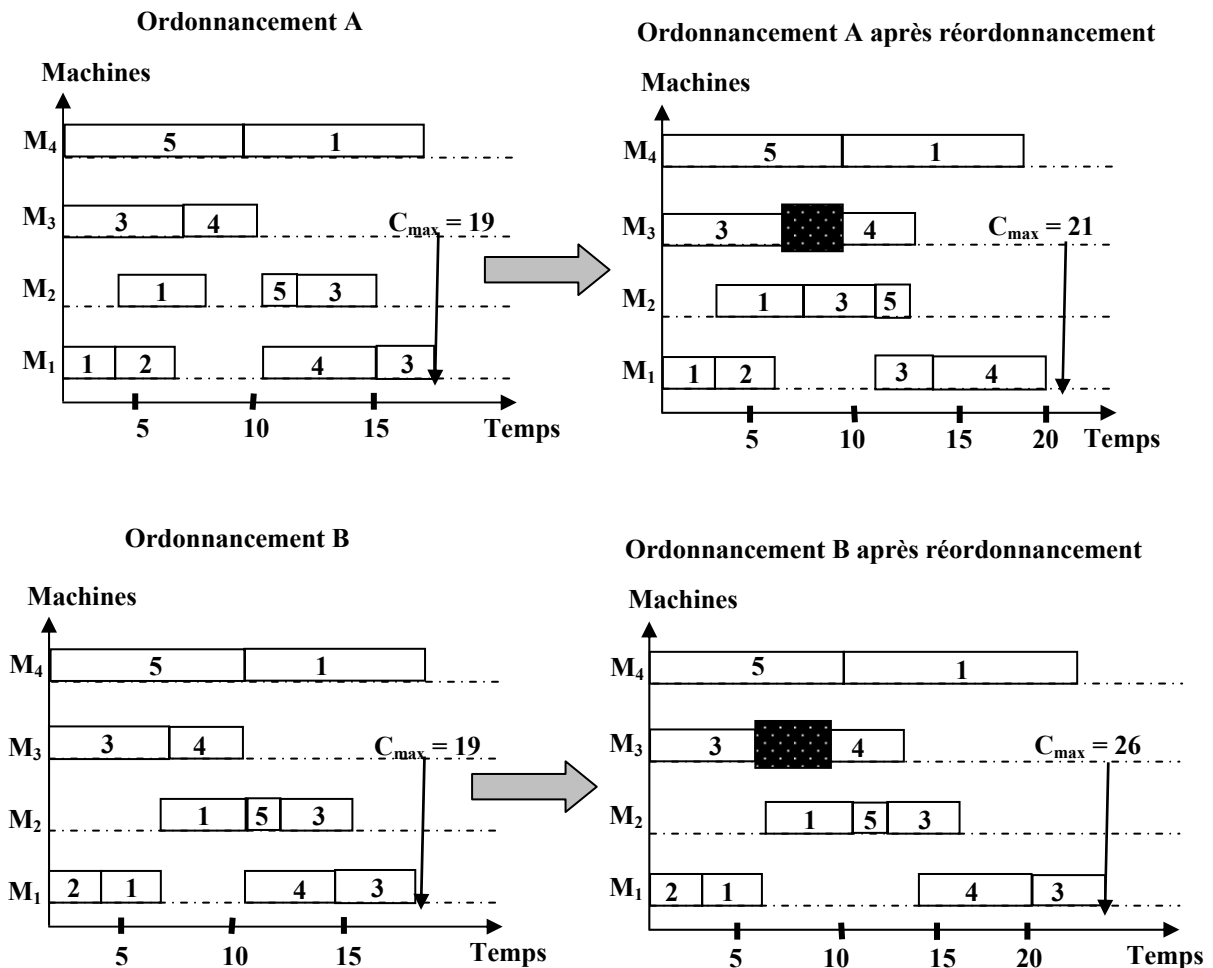


Figure 4.2 : Exemple d'ordonnement flexible [JEN01a]

Dans la plupart des applications de l'ordonnement (gestion d'atelier, parallélisme, gestion de projet), il existe toujours une incertitude plus ou moins grande sur les données associées au problème considéré et sur la stabilité du système étudié. Il peut s'agir de la durée des tâches, mais aussi de la disponibilité des ressources, pour ne prendre que ces deux exemples. L'incertitude n'est alors levée que lors de l'exécution effective des tâches. L'approche déterministe est inadaptée à des problèmes où l'ampleur de l'incertitude est trop importante. L'approche stochastique postule une connaissance en fait très fine de l'incertitude (modèle aléatoire), et se heurte à de grosses difficultés calculatoires.

Il existe une autre approche qui consiste à proposer des schémas globaux de résolution couplant une partie hors ligne (avant exécution) et une partie en ligne (durant l'exécution). Ces processus sont caractérisés par leur flexibilité, relativement au type d'aléas qu'ils peuvent prendre en compte lors de l'exécution, et leur robustesse.

### 3. Ordonnancement en présence de données incertaines

Plusieurs approches d'ordonnancement et de réordonnancement ont été proposées dans la littérature pour tenir compte de la présence de données incertaines et des perturbations qui peuvent survenir en temps réel dans l'atelier. On peut distinguer deux politiques d'ordonnancement en présence de perturbations, dont l'objectif commun est de générer un ordonnancement robuste, c'est à dire apte à absorber des perturbations par rapport au fonctionnement nominal. Cette robustesse peut être prise en compte dynamiquement par une régulation en ligne du nombre des tâches à ordonnancer, cette approche du problème est dite réactive, ou statiquement pour analyser une évolution fonctionnelle ou organique. Cette dernière est dite proactive.

Nous présentons dans cette section tout d'abord une classification des perturbations, puis une classification des approches d'ordonnancement en présence de perturbations citées ci-dessus.

#### 3.1. Classification des perturbations

Une perturbation est définie, d'une manière générale, comme tout événement opérationnel non prévu pouvant affecter les performances d'un ordonnancement généré, dans un environnement statique où toutes les données du problème sont connues à l'avance et aucun changement dans l'environnement n'est envisagé. La perturbation intervient lors de la mise en œuvre de ce dernier.

Selon Baillet [BAI94] les perturbations peuvent être *internes* ou *externes* au système de production:

- Les perturbations internes sont des événements opérationnels imprévus qui se produisent au sein du système de production, par exemple, un prolongement de la durée d'une ou plusieurs tâches, ou une panne machine, ou une absence inopinée d'un opérateur ou, une erreur d'information, etc.
- Les perturbations externes sont des événements affectant les entrées et les sorties du système de production. On distingue alors les perturbations dues aux fournisseurs, comme la variation des dates de disponibilité de la matière première, et des perturbations dues aux

clients qui passent et modifient fréquemment leurs commandes, au dernier moment, au détriment de celles planifiées depuis longtemps.

Selon Artigues et al [ART02] la classification des perturbations dépendant du niveau de connaissance de l'incertitude. Ils identifient trois types d'incertitudes dans les environnements manufacturiers :

- ➔ Des incertitudes complètement inconnues, ou bien aléas imprévisibles, qui sont des événements imprévus pour lesquels aucune information n'est disponible à l'avance, par exemple, une grève ou un accident dans le lieu de travail ou une absence inopinée d'un ouvrier, etc. Dans ce cas, il est impossible d'en tenir compte dans la phase de pré calcul d'un ordonnancement prédictif initial.
- ➔ Des suspicions du futur qui sont données grâce à l'intuition et l'expérience du décideur. Ce sont des incertitudes difficilement quantifiables et par conséquent il est difficile de les intégrer dans des algorithmes d'ordonnancement. Des systèmes d'aide à la décision interactifs sont plus appropriés pour ce type d'incertitudes.
- ➔ Des incertitudes partiellement connues, ou bien aléas prévisibles, dont quelques informations sont disponibles avant l'exécution. Par exemple, on connaît le type de perturbations qui peuvent arriver (pannes machines, arrivée des tâches en retard, etc.). Plus d'informations peuvent être déduites grâce à l'historique de l'atelier, comme par exemple, les lois de distribution de la fréquence d'occurrence et de la durée des pannes. Ces informations peuvent être incluses lors du calcul d'une solution initiale au problème considéré.

Les données associées à un problème d'ordonnancement sont les durées, les dates d'occurrence de certains événements, certaines caractéristiques structurelles, et les coûts. Aucune de ces données n'est exempte de facteurs d'incertitudes. Mehta et al [MEH99] ainsi que Cowling et al [COW02] ont considéré essentiellement deux classes d'incertitudes : Celles liés *aux ressources* et celles liés *aux travaux*. C'est cette classification que nous avons adoptée pour la suite de ce travail, et que nous définissons dans la section suivante.

### 3.1.1. Perturbation sur les ressources.

Les perturbations prises en compte dans cette catégorie sont :

- Les Pannes machines. Elles peuvent suivre des lois de probabilité ou être complètement aléatoires. On peut inclure aussi, dans cette catégorie, le facteur humain, l'opérateur travaillant sur la machine [ALC02, LAW97] (i.e. on peut considéré l'indisponibilité d'un ou plusieurs opérateurs comme une panne machine, car la machine commandée par cet opérateur est indisponible). La durée de la panne d'une machine est vue comme la durée de l'indisponibilité de l'opérateur. Les pannes machines peuvent être classées par niveaux, chaque niveau détermine une grandeur des pannes envisagées pour ce niveau (le nombre de pannes dans l'atelier ou le nombre de machines affectées), par exemple (le niveau 1 : entre 2 et 3 pannes) et (niveau 2 : plus de 3 pannes). Cette classification se fait pour mieux étudier le comportement de l'ordonnancement et évaluer la robustesse de ce dernier pour chaque niveau [SHA99a, JEN01, PIS96, BAL02] ;
- Le Changement de Ressources. Il concerne : la modification dans les matières premières pour la fabrication d'un produit ou le changement d'un ou plusieurs outils d'une machine et ceci dans le but d'accomplir une autre tâche par la même machine ne pouvant s'effectuer avec le même outil [MIG95, LAW97] ;
- Manque de ressources. Il concerne : soit le manque de matière première ou d'outils de fabrication. Ce type de perturbation est rencontré dans les ateliers ne disposant pas de stock, ou les ateliers n'ayant pas une livraison adéquate et régulière [LAW97].

### **3.1.2. Perturbation sur les travaux**

Parmi les perturbations les plus fréquentes et les plus étudiées, on peut trouver :

- Ajout d'un nouveau travail : dans les situations réelles, on peut inclure des commandes (travaux) urgentes et ceci pendant l'exécution de l'ordonnancement. [LAW97] ;
- Annulation d'un travail : cette situation est rencontrée quand un travail n'est plus nécessaire dans le déroulement de l'atelier [LAW97] ;
- Changement de la priorité dans les travaux : cette situation est rencontrée seulement si les travaux n'ont pas la même importance. Donc, un changement dans leur ordre de priorité conduit inévitablement à un changement dans les affectations des ressources et l'ordre de passage des travaux par les machines [MIG95] ;
- La charge d'un atelier : on désigne par cette notion le degré de liberté d'exécution de chaque tâche dans l'atelier. On dit qu'un atelier est chargé si la tâche n'a pas beaucoup de liberté d'exécution (i.e. les dates de début et de fin d'exécution coïncident ou il n'y a pas

un grand écart avec les dates de début d'exécution au plus tôt et de fin d'exécution au plus tard respectivement) ;

- Modification d'un travail : En général, cette modification concerne le changement dans les durées opératoires des différentes tâches d'un travail ou d'un ensemble de travaux (élargissement ou rétrécissement de la durée opératoire) [LAW97, SHA99a]. Ce changement est dû à plusieurs facteurs, on peut citer l'état de la machine, qui peut influencer sur tous les produits passant sur cette dernière, si la machine est âgée ou n'est pas dans un bon état les opérations effectuées sur cette dernière seront prolongées avec un certain degré. Dans ce cas, on dit que le changement est lié aux machines. Ce changement peut être lié au travail lui-même, on peut décider par exemple que les durées opératoires de certaines tâches seront rétrécies ou élargies, cela est dû à la non importance ou l'importance de la qualité des produits concerné par le rétrécissement/élargissement de leurs durées de fabrication.

L'ensemble des auteurs qui ont traitées les incertitudes dans les ordonnancement ont étudié principalement deux perturbations : perturbations sur les équipements et plus particulièrement **les pannes machine** ou encore les perturbations concernant les travaux et plus particulièrement le **changement dans les durées opératoires**. L'ensemble des travaux recensés, dans cette section, concernent ces deux types de perturbation du fait de la simplicité de mise en œuvre de l'un (perturbation sur les durées opératoires) ou encore parce que le deuxième (pannes machines) est régi par des lois de probabilités connus.

Un état de l'art sur les méthodes d'ordonnement de production en présence de perturbations est présenté dans la section suivante, les travaux que nous y recensons traitent des aspects relatifs à l'ordonnement de production mais en présence de perturbation.

### **3.2. Classification des méthodes d'ordonnement en présence de perturbations**

Plusieurs approches d'ordonnement et de réordonnement ont été proposées dans la littérature pour tenir compte de la présence de données incertaines et des perturbations qui peuvent survenir en temps réel dans l'atelier. Mehta et Uzsoy [MEH99] proposent une classification de ces approches en quatre catégories : les approches totalement réactives, les approches prédictives réactives, les approches robustes et les approches à base de



connaissances. Davenport et Beck proposent en 2000 [DAV00] une deuxième classification : approches réactives, approches proactives et approches mixtes.

Les approches réactives selon Davenport et Beck [DAV00] se basent sur des informations mises à jour en temps réel selon l'état du système et éventuellement sur un ordonnancement initial calculé à priori sans intégrer de connaissances sur ce qui va se passer dans l'atelier. Ces approches correspondent exactement aux approches totalement réactives et les approches prédictives réactives selon Mehta et Uzsoy [MEH99]. En outre la définition des approches proactives donnée par Davenport et Beck correspond exactement à celle donnée aux approches robustes par Mehta et Uzsoy. Cependant, la classification de Mehta et Uzsoy ne permet pas de classer les approches mixtes de Davenport et Beck.

Les approches à base de connaissances peuvent être considérées comme des approches réactives où l'algorithme réactif consiste à sélectionner, chaque fois qu'une décision doit être prise, une règle de réordonnement ou de séquençement parmi un ensemble d'alternatives selon l'état de l'atelier et les connaissances acquises dans des expériences antérieures.

Dans la suite du chapitre nous adopterons la classification de Davenport et Beck.

### 3.2.1. Les approches réactives

*« Les approches réactives décident quand et où les activités doivent être exécutées, et ce en se basant sur les informations mises à jour concernant l'état du système et éventuellement sur un ordonnancement prédictif initial »* [DAV00].

On distingue les approches totalement réactives ou réactive pure qui n'utilisent pas d'ordonnement prédictif initial et les approches prédictives réactives qui utilisent à la fois un algorithme prédictif qui calcule un ordonnancement initial hors ligne, sans tenir compte de la présence de perturbations, et un algorithme réactif qui tient compte du nouvel état du système pour recalculer une nouvelle solution chaque fois qu'un aléa survient.

Dans l'approche réactive pure, aucun ordonnancement initial n'est donné à l'atelier. Les décisions sont prises localement en utilisant dans la plupart des cas des règles de priorité. Dans les approches prédictives réactives, un ordonnancement prédictif est généré initialement sans tenir compte de futurs aléas. Enfin, quand très peu d'hypothèses sont faites sur les données, les décisions à prendre à chaque instant sont très difficiles à évaluer à priori, et c'est le domaine de l'ordonnement en ligne (en anglais *on-line scheduling*).

Nous présentons dans ce qui suit deux techniques d'ordonnement issues de l'approche réactives : la première basée sur le voisinage et la seconde sur le réordonnement.

### 3.2.1.1 Technique basée sur le voisinage

Cette technique est inspirée des travaux récents sur les fonctions d'optimisation continues [BRA00]. Dans les applications actuelles, on peut trouver des imprécisions dans le passage d'une solution idéale à son implémentation [JEN01]. Dans ce cas, une solution robuste est tirée à partir des pics larges dans le graphe de la fonction objectif, alors que les solutions non robustes sont sur les pics étroits (Fig. 4.3). Les solutions avoisinantes de la solution optimale doivent se trouver sur les pics larges du graphe. Ceci parce que sur un pic large la valeur de la fonction objectif est toujours meilleure que celle d'un pic étroit.

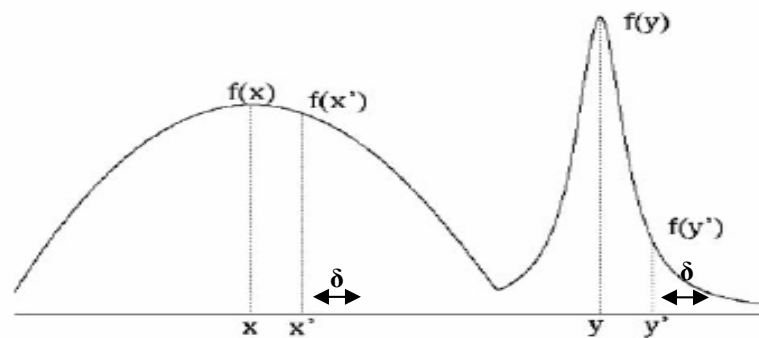


Figure 4.3 : Graphe de la fonction d'optimisation continue

On remarque qu'avec la même perturbation  $\delta$  sur les solutions  $x$  et  $y$ , notée  $(x+\delta, y+\delta)$ , la fonction objectif  $f(y)$  a diminuée plus que  $f(x)$ .

Dans le problème d'ordonnement, l'idée est que lorsqu'une panne survient, on puisse trouver un ordonnancement proche du l'ordonnement initial (minimum de permutations de tâches par rapport à l'ordonnement original), pouvant s'exécuter partiellement autour de la panne. Il existe plusieurs algorithmes pour la génération du voisinage. Parmi ces algorithmes, on peut trouver les algorithmes génétiques et leurs dérivées [JEN01a].

### 3.2.1.2 Technique basée sur le réordonnement

Dans cette approche, un ordonnancement prédictif est généré initialement sans tenir compte de la présence de futurs aléas. Il permet en particulier de prévoir l'arrivée dans l'atelier de composants secondaires de manière raisonnable peu avant la date où les tâches vont les utiliser. Cet ordonnancement est ensuite soumis à l'atelier pour exécution. En présence d'aléas, cet ordonnancement est soit remplacé par un autre ordonnancement recalculé en utilisant les

nouvelles données du problème, on parle alors de réordonnement, ou bien modifié et réparé pour tenir compte du nouvel état de l'atelier.

Elle doit assurer :

- La disponibilité d'ordonnements utilisables à l'instant de l'apparition d'aléas : Avant l'occurrence de perturbations, un ordonnancement était entrain de s'exécuter, mais après, un réordonnement doit être lancé. Une partie de l'ordonnement s'exécutant au moment de l'occurrence de la perturbation peut être utilisé comme espace de recherche initial dans le cas où le réordonnement est dépendant de l'ordonnement original ;
- Quand on réordonne, il est souhaitable que le nouvel ordonnancement soit le plus proche possible de l'ordonnement original ;
- Le processus de réordonnement ne doit pas être long.

Le réordonnement peut être exécuté de plusieurs manières [ALO02]:

1. *Le réordonnement continu* qui consiste à recalculer un nouvel ordonnancement chaque fois qu'une perturbation surgit rendant l'ordonnement initial non réalisable. Il permet bien évidemment, s'il est applicable, d'obtenir de très bonnes performances en fin d'exécution. Cependant, il nécessite beaucoup de ressources de calcul.

2. *Le réordonnement périodique* est appliqué à des intervalles réguliers dans le temps. Si une perturbation a lieu entre deux instants de réordonnement successifs, des règles simples de réparation de l'ordonnement courant sont utilisées. Ce type de réordonnement requiert moins de temps de calcul que le réordonnement continu et donne une idée plus claire de ce qui va être exécuté au moins entre deux instants successifs de réordonnement global. Cependant, il peut donner de moins bonnes performances surtout si une ou plusieurs perturbations importantes surgissent entre deux instants de réordonnement.

3. *Le réordonnement événementiel* combine les avantages des deux derniers types de réordonnement. Le réordonnement est aussi appliqué à des intervalles réguliers. Lors de l'occurrence d'une perturbation entre deux instants de réordonnement successifs, il faut décider si on utilise une règle simple de réparation ou si il est nécessaire de faire un réordonnement global. Cette décision est difficile à faire et nécessite une analyse fine de la situation et/ou l'expérience du décideur.

En outre plusieurs méthodes de réordonnement sont à distinguer [ALO02]:

1. *La méthode de décalage à droite* (en anglais *right-shift rescheduling*) consiste, pour le cas d'une panne, à retarder dans le temps les opérations affectées par la panne en maintenant la même séquence de départ au niveau de chaque machine. Une opération se trouve affectée par la panne si elle est exécutée sur la machine en panne ou à cause de contraintes de précédence. Cette méthode est très simple (même triviale) et permet de préserver la stabilité de l'atelier. Cependant, elle ne permet pas, dans la majorité des cas, d'obtenir de bonnes performances.

2. *Le réordonnement complet* consiste à considérer les opérations non encore exécutées et construire un nouvel ordonnancement en tenant compte de l'état de l'atelier (disponibilité des machines, nouvelles caractéristiques de tâches, etc.). Ce type de méthode favorise bien évidemment la performance de l'ordonnement réalisé mais peut contribuer facilement à l'instabilité de l'atelier.

3. *Le réordonnement multi-objectifs* consiste à considérer l'efficacité de l'ordonnement réalisé et la stabilité de l'atelier dans le processus de recalcul d'un nouvel ordonnancement suite à une perturbation.

4. *Le réordonnement par retour vers l'ordonnement initial* (en anglais *match up rescheduling*) consiste à recalculer une partie de l'ordonnement prédictif initial de façon à ce que le nouvel ordonnancement revienne sur l'ordonnement initial à un point donné dans le temps. Ceci nécessite la présence de temps morts dans l'ordonnement initial.

Dans la plupart des méthodes rencontrées dans la littérature, on a considéré que, indépendamment de la fréquence de réordonnement, le réordonnement multi-objectifs et le réordonnement par retour vers l'ordonnement initial sont les plus intéressants. En effet, dans ces méthodes, on essaie de modifier le moins possible l'ordonnement initial pour assurer ainsi une certaine prédictibilité du système de production tout en espérant préserver de bonnes performances.

### **3.2.2. Les approches proactives ou robustes**

Contrairement aux approches réactives, dans les approches proactives ou robustes, la connaissance de l'incertitude est utilisée par l'algorithme statique pour construire un seul ordonnancement (ordonnement de référence, en anglais *baseline schedule*) ou une famille d'ordonnements. On trouve aussi dans la littérature le terme d'approche prédictive, la différence étant que l'ordonnement construit par un algorithme statique dans ce cas, ne

tient pas compte des incertitudes. Nous ne considérons dans ce qui suit que les approches au moins proactives.

Le but de l'approche proactive est de prendre en considération les incertitudes lors de la mise en œuvre de l'ordonnancement prédictif original. Cette prise en compte des informations incertaines est utile pour rendre l'ordonnancement prédictif plus robuste. Selon cette approche un ordonnancement robuste est caractérisé par :

- La possibilité de rester valide sous une large variété de perturbations.
- Le fait de remettre en cause les hypothèses de départ ne remet pas en cause l'exécution de l'ordonnancement.

Dans ce qui suit, nous commencerons par définir le *slack time* et les trois méthodes qui découlent de cette technique. Cette technique prend en considération les perturbations (incertitudes) afin de produire des ordonnancements capables d'absorber un certain nombre d'évènements non programmés (non prévus), et ceci sans faire appel à des méthodes de réordonnancement.

### ***Principe du Slack time***

Le *Slack Time* d'une opération dans un ordonnancement, est un temps "tampon" (Temps supplémentaire ou *Buffer Time* en anglais), ajouté à l'opération dans lequel cette dernière peut être retardée sans dégrader les performances (par exemple le *Makespan*) de l'ordonnancement si l'ordre du traitement est gardé (sans réordonnancement) [JEN01].

L'idée principale derrière cette technique est d'offrir à chaque activité un temps supplémentaire (*Extra Time* en anglais) pour s'exécuter dans des cas incertains. Ce *slack* peut être absorbé sans réordonnancement [DAV01]. Il est défini par la formule suivante :

$$\text{Slack}_i = p_i + d_i - t_i$$

avec :

$t_i$  : la date début au plus tôt de l'activité  $i$ .

$d_i$  : la date fin au plus tard de l'activité  $i$ .

$p_i$  : la durée opératoire de l'activité  $i$ .

1. *La protection temporelle* [GAO95] étend la durée de chaque activité (opération), en se basant sur des statistiques sur les incertitudes de la ressource sur laquelle elle s'exécute. Les ressources qui ont une probabilité de défaillance non nulle, sont considérées comme des ressources *fragiles*. La durée d'exécution d'une opération nécessitant pour son exécution de

telles ressources est étendue pour lui offrir un temps supplémentaire, et dans lequel on peut prendre en charge la panne (la maintenance de la machine).

Le problème devient alors, un problème d'ordonnement avec des durées opératoires protégées. L'idée derrière l'extension des durées opératoires est que durant l'exécution de l'ordonnement, les slack<sup>4</sup> fournissent un temps qui peut être utilisé dans le cas d'une panne machine.

2. *Time Window Slack (TWS)*. La protection temporelle transforme le problème original en un nouveau problème qui peut être résolu par plusieurs techniques d'ordonnement. Cependant, l'inconvénient majeur de cette technique est que le slack ajouté à chaque activité ne permet pas de résoudre directement le problème. Cela peut mener à des situations où il est impossible de partager le slack entre plusieurs activités [DAV01]. Pour éviter une telle situation, *TWS* traite directement la disponibilité de l'activité et son slack pendant la résolution du problème, cela est plus intéressant que d'inclure le slack comme une partie de la durée d'exécution d'une activité. L'avantage de cette méthode est qu'il y a plus d'information concernant le slack pendant la résolution du problème.

3. *Focused Time Window Slack (FTWS)*. Ni la protection temporelle, ni *TWS*, ne prennent en considération l'emplacement de l'activité sur l'axe de l'ordonnement (en anglais Scheduling Horizon).

*FTWS* utilise les incertitudes statistiques pour concentrer le slack sur les périodes de l'horizon de l'ordonnement qui en ont besoin le plus en cas de panne. La distribution de probabilité  $P(N(\mu_{TBF}(R), \delta_{TBF}(R)) \leq t)$  donne la probabilité qu'une panne survient durant un intervalle  $[0, t]$ . Une approximation de la courbe de cette distribution est utilisée pour déterminer le volume du slack affecté à une activité. Le volume du slack augmente avec la probabilité d'avoir une panne (une activité avec forte probabilité d'être interrompue aura un grand slack). Alors, le slack d'une activité *A* est calculé comme fonction de probabilité qu'une panne survienne avant ou durant l'exécution de *A* et de la durée de la panne prévue.

### 3.2.3. Les approches proactives réactives

« Un système d'ordonnement capable de gérer les incertitudes utilise vraisemblablement non seulement une approche proactive mais aussi une approche réactive » [DAV00].

Il est naturel de coupler une approche proactive, lorsqu'elle propose une famille d'ordonnements, avec une phase dynamique élaborée : au fur et à mesure de la

connaissance des valeurs réelles des données et éventuellement suite à un aléa, un algorithme dynamique non trivial est utilisé pour choisir parmi les ordonnancements préalablement sélectionnés en statique, ceux ayant les meilleures performances. Cette approche, qui permet de réagir aux conditions réelles est appelée approche proactive/réactive.

En effet il est très difficile de tenir compte proactivement de toutes les perturbations qui surviennent réellement, sauf peut être si on sacrifie les performances de l'ordonnancement généré hors ligne en insérant par exemple un très grand nombre de temps morts. Ceci n'est évidemment pas désiré. Par conséquent, il faut prévoir un algorithme réactif, plus élaboré que ceux utilisés dans les approches proactives ou robuste, pour pouvoir réagir vis-à-vis des perturbations que l'ordonnancement initial ne peut pas absorber. Cependant, l'algorithme réactif est généralement contraint par un temps de réponse limité. C'est pourquoi, on ne doit pas s'attendre à une décision optimale de l'algorithme réactif et il est ainsi préférable de ne pas avoir à l'utiliser pour toutes les perturbations qui surviennent. En d'autres termes, il est intéressant d'avoir un algorithme proactif qui introduit de la *flexibilité* dans les solutions lors de la phase de calcul. Cette flexibilité introduite permet de minimiser le besoin de procédures complexes dans l'algorithme réactif en limitant par exemple l'espace de recherche tout en garantissant l'obtention de solutions performantes [ALO02].

Deux méthodes ont été développées dans ce contexte : la méthode ORABAID (ORdonnancement d'Atelier Basé sur une AIdé à la Décision) [DEM77, ART97] et la méthode PFSL (*Preprocess First Schedule Later*) [WU99].

### 3.3. Synthèse

L'ensemble des travaux portant sur l'étude de la robustesse des ordonnancements de production se basent sur deux facteurs importants : les perturbations prises en compte et la mesure de robustesse utilisée. Même si elles diffèrent au niveau de leurs environnements, les auteurs suivent, en général, l'une des deux classes d'expériences suivantes: la première utilise une seule mesure de robustesse avec plusieurs méthodes de résolution (les approches de réordonnancement sont les plus utilisées), alors que la deuxième classe utilise une ou peu de méthodes de résolution mais portent sur plusieurs mesures de robustesse. Le point commun aux deux classes précédentes est que la mesure de robustesse est connue à l'avance. L'objectif

---

<sup>4</sup> Temps supplémentaire ou *Buffer Time* en anglais

étant soit de trouver une (ou plusieurs) mesure qui assure un certain degré de robustesse ou alors de démontrer que la méthode de génération de l'ordonnement est robuste.

Dans ce qui suit, nous présentons quelques travaux intéressants se rapportant à la robustesse des ordonnancements.

Jensen [JEN01] considère un atelier de type Jobshop de taille soumis à des pannes machine. Il propose une mesure de robustesse basée sur le voisinage d'un ordonnancement. L'idée de base est de minimiser le coût d'un ensemble d'ordonnements situés autour de l'ordonnement considéré. Il montre que cette mesure permet d'améliorer la robustesse de tous les problèmes considérés (minimiser la durée totale, le plus grand retard, la somme des dates de fin et la somme des retards). Les pannes ont été simulées lors de l'exécution de l'ordonnement (La date de panne coïncide avec la date de début d'une opération de l'atelier et le temps d'arrêt des machines est constant). Il arrive à la conclusion que l'utilisation de la mesure de robustesse basée sur le voisinage est meilleure, en général pour toutes les mesures de performances, en terme de robustesse, spécialement par rapport à celle basé sur le retard. L'expérience a permis aussi, de prouver que la méthode du voisinage donne de bons résultats sur les problèmes avec peu de points critiques (parties de l'ordonnement qui ne peuvent être changées sans dégrader les performances de l'ordonnement).

Léon & al [LEO94] ont proposé pour le même problème une méthode qui optimise différentes mesures de robustesse basées sur les marges pour améliorer la performance moyenne de l'atelier.

Davenport & al [DAV01] considèrent un atelier de type Jobshop de taille 6x6 et optimisent la somme des retards en utilisant les méthodes dites robustes (section 3.2.2). L'atelier est soumis à des pannes machine. Neuf niveaux d'incertitudes ont été définis (par rapport au niveau d'incertitude et le nombre de machines susceptibles de tomber en panne). Ils optimisent deux mesures de robustesse : la moyenne des retards simulés et la moyenne des différences absolues entre les retards prédits et les retards simulés. Pour la première mesure de robustesse les résultats des simulations ont montré que les méthodes avec protection donnent de meilleurs résultats que la méthode sans protection. Pour la deuxième mesure de robustesse, les résultats varient selon les niveaux d'incertitudes.

Shafaei & al [SHA99] considèrent un atelier de type Jobshop soumis à trois types de perturbations : Perturbation sur les durées opératoires, les pannes machines et la charge de l'atelier. L'objectif est d'étudier, d'une part les performances des différentes règles de priorité, et d'autre part l'effet des intervalles de réordonnement, la charge et les goulots sur la robustesse d'un ordonnancement ;



Lawrence & al [LAW97] ont utilisés 53 ateliers génériques de type Jobshop soumis à des perturbations liées aux travaux. Ils ont défini 10 niveaux de perturbation, sans inclure celui du cas déterministe, puis ils les ont classés en 3 classes (Basse, Moyenne, Elevée). Ils comparent des méthodes de résolution optimales et des méthodes heuristiques (Shifting Bottleneck et les règles d'ordonnancement) pour déterminer quelle méthode utiliser et dans quelle situation afin d'avoir un ordonnancement efficace et robuste. En même temps, ils espèrent vérifier si les bonnes politiques d'ordonnancement déterministes le restent dans le cas où les durées opératoires sont incertaines.

### **3.4. Discussions et conclusion**

Dans cette section, nous avons présenté différentes approches pour résoudre le problème de l'ordonnancement en présence de perturbations. Ces approches ont été classées selon la terminologie proposée par Davenport et Beck en 2000 [DAV00] : approches réactives, approches proactives et approches mixte. L'efficacité de ces approches dépend du degré de connaissance de l'incertitude et du type de perturbations traitées.

Les approches réactives se basent sur des informations collectées en ligne sur l'état du système et les caractéristiques des tâches à exécuter et éventuellement sur un ordonnancement prédictif calculé hors ligne. Le calcul de cet ordonnancement prédictif ne tient pas compte de la présence de perturbations. Ces approches échouent lorsqu'on veut prédire les performances du système et/ou planifier d'autres activités internes ou externes à l'atelier. Néanmoins, lorsque l'incertitude est complètement inconnue (les caractéristiques des tâches ne sont pas maîtrisées et/ou l'atelier est fortement perturbé), il est clair que les approches totalement réactives sont plus appropriées surtout si elles se basent sur des règles de priorité avec garantie de performance.

Les approches proactives ou robustes sont envisagées lorsqu'on est en présence d'incertitude connue (informations disponibles sur le type des perturbations, les lois de distributions des caractéristiques de perturbations, etc.). Notons qu'une approche proactive ne permet pas toute seule de prendre en compte hors ligne toutes les perturbations qui surgissent réellement dans l'atelier. Il faut lui associer un algorithme réactif permettant de réagir vis à vis des aléas qui ne peuvent pas être absorbés par l'ordonnancement prédictif initial. Pour que cet algorithme n'ait pas à recalculer un nouvel ordonnancement chaque fois qu'une perturbation surgit, il faut introduire de la flexibilité dans la solution initiale donnée à l'atelier.

Par ailleurs l'état de l'art sur l'ordonnement de production en présence de perturbations a permis de dégager deux axes de recherche. Le premier vise à prouver la robustesse de certaines méthodes d'ordonnement en utilisant plusieurs mesures d'optimisation. Le deuxième, par contre se donne pour objectif de trouver des mesures de robustesse.

La démarche que nous présentons dans cette thèse est aussi une démarche proactive, qui se démarque par rapport aux travaux que nous avons recensés dans la section précédente par le fait que les ordonnements considérés sont des ordonnements conjoints production/maintenance, pour lesquels on propose d'étudier la contribution de la maintenance à leur robustesse, en présence de perturbations liées aux durées opératoires. Notre objectif n'est pas de proposer des algorithmes proactifs, mais plutôt de démontrer que la génération d'ordonnements conjoints production/maintenance est une démarche proactive.

De plus, les deux axes précédents sont étudier : d'une part essayer d'identifier, parmi les méthodes d'ordonnements conjoints production/maintenance présentées dans le chapitre précédent, des méthodes robustes. Et d'autre part d'identifier des mesures potentielles de robustesse.

## **4. Etude de la robustesse des ordonnements conjoints**

### **4.1. Introduction**

Nous avons présenté dans la section précédente une classification des méthodes d'ordonnements en présence de perturbations. De cette littérature le terme maintenance est inexistant. L'ensemble des travaux étudiés concerne l'ordonnement de production uniquement. La maintenance et la production sont les fonctions pivots de l'entreprise, malheureusement leurs ordonnements respectifs sont disjoints et ont des objectifs fréquemment antagonistes, même si les deux fonctions oeuvrent toutes les deux à atteindre l'objectif ultime de la productivité maximale. L'ordonnement de la maintenance provoque l'immobilisation des équipements alors que le second doit optimiser leurs exploitations.

L'idée de départ de cette thèse fût de démontrer que l'ordonnement conjoint de la production et de la maintenance permet, d'une part, le respect des contraintes de production et de maintenance, et d'autre part un gain en sûreté de fonctionnement. L'idée que nous allons développer dans ce chapitre, et qui découle naturellement de la précédente, est d'évaluer la contribution de la maintenance à la robustesse d'un ordonnement conjoint, en présence de perturbation.

Nous développerons pour cette étude trois aspects : le premier est relatif à l'étude de la contribution de la maintenance à la robustesse des ordonnancements conjoints, les deux autres sont relatifs à l'identification de méthodes d'ordonnancements conjoints robustes et de critères qui assurent un certain degré de robustesse à l'ordonnement conjoint.

Pour cela nous commencerons par une description globale du modèle que nous proposons pour l'étude de la robustesse des ordonnancements conjoints en présence de perturbations. Nous présenterons la méthodologie que nous proposons pour concevoir le modèle adopté. La définition du modèle et de son implémentation est un préalable à cette étude. La définition du modèle passe par la description du problème d'ordonnement considéré. Pour la phase implémentation du modèle, le générateur d'algorithmes génétiques conjoints production/maintenance, développé dans le chapitre précédent, produit des ordonnancements de production et des ordonnancements conjoints production/maintenance qui seront perturbés, puis la déviation des performances de l'ordonnement conjoint, en présence de perturbations, par rapport à l'ordonnement conjoint initial, est calculée pour élaborer une classification des ordonnancements.

## **4.2. Modèle pour l'étude de la robustesse des ordonnancements conjoints production/maintenance**

Dans cette section, nous présentons d'une façon globale le modèle que nous proposons pour l'étude de la contribution de la maintenance à la robustesse des ordonnancements conjoint production/maintenance en présence de perturbations ainsi que la méthodologie que nous avons adoptée pour la conception de ce modèle. Notre objectif est de donner une vue d'ensemble sur ce qui est présenté dans la suite du chapitre.

### **4.2.1. Contexte de l'étude**

Comme nous l'avons déjà noté, notre travail s'inscrit dans le cadre des approches proactives dont le principe a été présenté dans la section 3.2 de ce chapitre. Nous traitons d'une façon générale les problèmes de *flowshop* de permutation. Nous allons aborder cette section par les hypothèses que nous avons formulées pour cette partie de l'étude puis exposer l'approche adoptée.

Soient :

- O un ordonnancement quelconque ;

- $C^*(O)$  : la valeur optimale du critère d'optimisation  $C$  de l'ordonnement  $O$  ;
- $Ca(O)$  : la valeur acceptable du critère d'optimisation de l'ordonnement  $O$  (non optimale) ;
- $C(O)$  : la valeur mesurée du critère d'optimisation de l'ordonnement  $O$  (non optimale) ;
- $R(O)$  : la mesure de la robustesse de l'ordonnement  $O$ . La robustesse d'un ordonnement doit être synonyme d'une certaine optimalité de l'ordonnement. C'est à dire  $O$  est robuste  $\Leftrightarrow C(O) \in [C^*(O), Ca(O)]$
- $d(O)$  : la variation de la valeur du critère d'optimisation par rapport à l'optimal :

$$d(O) = C(O)/C^*(O) ; d(O) \in [0,1]$$

$$C(O) \in [C^*(O), Ca(O)]$$

Comme nous allons étudier la contribution de la robustesse aux ordonnements conjoints de manière statique, nous ferons une distinction entre la robustesse et la flexibilité d'un ordonnement. La figure 4.4 présente un schéma global de résolution couplant une partie hors ligne (avant exécution) et une partie en ligne (durant l'exécution). La partie hors ligne génère des ordonnements robustes  $R(O)$ , qui seront injectés dans la partie en ligne (ordonnement réactif). L'indicateur  $R(O)$  est statique et donne donc une mesure absolue. Toutefois on peut mesurer la robustesse de cet ordonnement en ligne, mais à ce moment la mesure sera relative vu qu'on ne prendra pas en compte toutes les tâches mais uniquement celles qui restent à exécuter. Une amélioration de la valeur de cet indicateur est possible par une relaxation des contraintes de performance de l'ordonnement initial jusqu'à un certain degré au-delà duquel l'ordonnement deviendrait caduc. L'idée étant de fournir, au départ, à la partie en ligne (réactif) un ordonnement qui possède une certaine capacité à absorber les aléas. Par la suite, si les aléas sont importants, une phase réactive avec un réordonnement est envisageable.

Il est important de souligner à ce stade de l'étude que nous nous situons dans la partie hors ligne du modèle global de résolution de la figure 4.4, couplant une partie hors ligne (avant exécution) et une partie en ligne (durant l'exécution). Notre modèle se restreint à la génération d'un ordonnement conjoint production/maintenance, hors ligne. Le modèle proposé va mesurer la contribution de la maintenance à la robustesse de ce dernier.

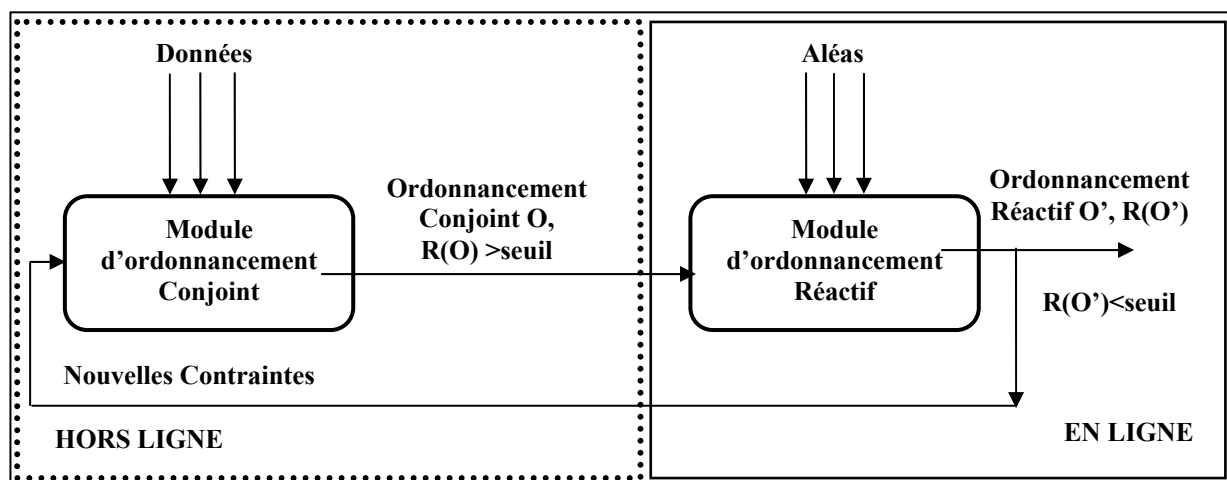


Figure 4.4: Situation du modèle par rapport au schéma global.

Notre modèle ne propose donc pas des algorithmes proactifs pour la génération d'ordonnements robustes mais se propose d'inscrire **la génération d'ordonnement conjoint production/maintenance comme une démarche proactive**. C'est-à-dire que nous allons démontrer que l'intégration de la maintenance lors de la génération des ordonnancements de production, est à mettre au profit de la robustesse de ces derniers.

Dans ce qui suit, nous allons présenter cette approche et comment nous comptons la mettre en œuvre.

#### 4.2.2. Présentation globale de l'approche

La complexité des mécanismes de dégradation des équipements a fait en sorte que la durée de vie de ces derniers a toujours été traitée comme une variable aléatoire. L'absence de données fiables et d'outils efficaces de traitement de ces données a réduit la fonction maintenance à des tâches de dépannage, et par le fait même, à une fonction dont les coûts ne cessent d'augmenter et dont la contribution à la performance de l'entreprise n'est pas évidente. Cet état de fait a incité plusieurs entreprises à adopter des approches plutôt réactives, n'étant pas en mesure de justifier économiquement les avantages que peut procurer la mise en place d'une maintenance préventive. Mais actuellement, les entreprises sont de plus en plus sensibilisées à l'importance des coûts induits par les défaillances accidentelles des systèmes de production. Alors que la maintenance, jusqu'à très récemment, était considérée comme un centre de coûts, il est de plus en plus évident qu'elle peut contribuer d'une manière significative à la performance globale de l'entreprise.

La mise en place d'une maintenance préventive est à priori « négative » pour la productivité du système du fait de l'immobilisation des équipements qu'elle entraîne. Si le rendement en terme de fiabilité des équipements est mesurable, le compromis entre la performance et la sûreté de fonctionnement est à atteindre. Pour répondre à cela, deux possibilités sont envisagées. Soit nous mesurons la dérive du système sans maintenance préventive, sachant que d'une part une action corrective ne ramène jamais le système à son état nominal ; et que d'autre part, beaucoup d'actions correctives nuisent grandement au rendement du système. Soit nous essayons d'évaluer la contribution de la maintenance, dans le rendement global de l'entreprise, par rapport à la plus-value qu'elle génère en terme de robustesse du système de production.

La première proposition a été largement étudiée dans la littérature. Certains auteurs ont étudié l'impact des politiques de maintenance sur le processus industriel, nous citerons en particulier les travaux de Wang et Pham [WAN96], Whisterkamp [WHI98], Sherwin [SHE00], et Vineyard & al [VIN00]. D'autres auteurs se sont intéressés à des cas pratiques, nous citerons en particulier les travaux de Sanmarti & al [SAN95], Ashayeri & al [ASH96], et Dumont et Morel [DUM01].

Nous avons donc opté pour la deuxième proposition, et l'objectif du modèle que nous proposons est d'évaluer la contribution de la maintenance à la robustesse des ordonnancements conjoints, où nous chercherons à vérifier si le respect des périodicités optimales des tâches de maintenance garantit un seuil minimal de robustesse.

Pour ce faire, nous nous sommes posé les trois questions suivantes :

### ***1. La Maintenance contribue-t-elle à la robustesse des ordonnancements conjoints ?***

La réponse à cette question permettra de déterminer si la maintenance contribue à la robustesse de l'ordonnement en l'améliorant ou en la dégradant. L'intégration des tâches de maintenance dans l'atelier dégrade certains critères d'ordonnement tel le *makespan* en retardant quelques tâches de production. Cela est valable seulement dans un environnement statique, car dans un environnement dynamique, où on peut envisager des perturbations, elle doit l'améliorer en optimisant la fiabilité des machines (une machine récemment entretenue aura une probabilité faible de dégradation de ses performances).

L'objectif est d'arriver à démontrer que la perte de performance des ordonnancements conjoints production/maintenance due à l'insertion de la maintenance est inférieure à la perte de performance des ordonnancements de production uniquement, ceci dans le cas de l'occurrence de perturbations.

## ***2. Existe-t-il des méthodes d'ordonnements conjoints robustes ?***

Cette question va permettre d'identifier une méthode d'ordonnement robuste. Les méthodes que nous allons étudier sont : les heuristiques dédiées, la recherche Tabou et les algorithmes génétiques (ces méthodes sont présentées dans le chapitre 3). Elles sont déclinées selon les deux stratégies d'ordonnement conjoint : séquentielle et intégrée. Ces méthodes sont dites robustes si elles génèrent un ordonnancement de hautes performances en présence d'incertitudes (temps de traitement, la disponibilité des machines...etc.).

## ***3. Existent-il des critères d'optimisation pouvant influencer sur la robustesse des ordonnancements?***

Cette question ouvre un horizon sur la possibilité de l'étude des critères d'optimisation ayant une influence sur la robustesse des ordonnancements. Cette question voit tout son intérêt dans le cas où ni la maintenance et ni les méthodes d'ordonnements n'influencent, ou influencent peu, la robustesse des ordonnancements, mais qu'il existe malgré cela des ordonnancements robustes. Alors, le but consiste à chercher les autres critères d'optimisation responsables de cette amélioration.

L'objectif est d'arriver à identifier parmi quatre critères d'optimisation classiques d'éventuelle(s) mesure(s) dont l'optimisation assure un certain degré de robustesse à l'ordonnement.

Les deux dernières questions abordent deux aspects très important liés à l'étude de la robustesse : la robustesse vu sous l'angle de méthodes robustes qui génèrent des solutions robustes et des critères robustes dont l'optimisation assurent un certain degrés de robustesse aux solutions générées.

Un autre aspect de cette étude concerne l'effet du choix de la stratégie de génération des ordonnancements conjoints sur leurs performances. Pour cette phase, nous utiliserons les stratégies de génération d'ordonnement déjà développées, dans le chapitre précédent, les stratégies séquentielle et intégrée. Pour rappel, la première stratégie consiste à générer au départ un échantillon de production en utilisant seulement les données de production. Ensuite, on insère les tâches de maintenance dans l'ordonnement de production déjà générée. Quant à la seconde, elle consiste à générer, cette fois-ci, en utilisant les tâches de production et les tâches de maintenance en même temps, un ordonnancement conjoint.

Pour répondre à ces trois questions, nous avons adoptée une méthodologie que nous présentons dans la section suivante.

### 4.2.3. Méthodologie adoptée pour la conception du modèle

Le processus global que nous proposons pour concevoir un modèle pour l'étude de la contribution de la maintenance à la robustesse des ordonnancements conjoints en présence de perturbations est en quatre phases (Fig. 4.5) : échantillonnage, perturbation, classification et validation.

- **Echantillonnage.** Nous avons généré trois échantillons d'ordonnements sous optimaux avec les algorithmes génétiques développés dans le chapitre précédent. Le premier échantillon est constitué d'ordonnements de production uniquement (AGP) issu de la première phase de la stratégie séquentielle (§ Chap. 3). Le second quant à lui est constitué d'ordonnements conjoints production/maintenance (AGS) issus de la deuxième phase de la stratégie séquentielle (après insertion de la maintenance). Le dernier enfin est constitué d'ordonnements conjoints production/maintenance mais issus de la stratégie intégrée (AGI). Le but recherché par cette distinction entre les méthodes de génération des ordonnancements conjoints est de voir si la méthode de génération a une influence quelconque sur la robustesse des ordonnancements.

Pour les aspects évaluation de la contribution de la maintenance à la robustesse des ordonnancements conjoints ainsi que de la recherche de critères de robustesse, les trois échantillons sont constitués d'ordonnements issus des algorithmes génétiques développés pour le chapitre 3. Par contre pour l'aspect recherche de méthode(s) robuste(s) les échantillons sont constitués des ordonnancements issus de chaque méthode étudiée dans le chapitre précédant (RA, Gupta, CDS, Palmer, NEH, Tabou et AG).

- **Perturbation.** Cette phase consiste à perturber les ordonnancements de chaque échantillon. Un ordonnancement peut subir des perturbations sur trois niveaux différents selon qu'il appartient à l'un des trois échantillons : à la génération de l'ordonnement de production (la première phase de la génération d'un ordonnancement conjoint par la stratégie séquentielle : échantillon AGP), après l'insertion des tâches de maintenance (la deuxième phase de la stratégie séquentielle : échantillon AGP) et enfin après la génération de l'ordonnement dans le cas de la stratégie intégrée (échantillon AGI).

Nous effectuons un processus de perturbation pour chaque ordonnancement original<sup>5</sup> issu de chacun des trois échantillons (AGP, AGS et AGI). Un processus de perturbation consiste à

---

<sup>5</sup> Un ordonnancement original est un ordonnancement non perturbé



simuler des perturbations sur les ordonnancements originaux de chaque échantillon (en moyenne 100 simulation sont effectuées) avec le même protocole de perturbation. La performance associée à chaque ordonnancement perturbée consiste en la moyenne algébrique de ces 100 simulations.

A la fin de chaque processus de perturbation, on calcule les écarts entre les mesures des ordonnancements originaux par rapports aux ordonnancements perturbés.

- **Classification.** Pour chaque échantillon, on réalisera un classement des ordonnancements par rapport aux mesures initiales. Cette phase va permettre d'identifier des mesures ou des méthodes que nous qualifierons de "robustes. Ainsi que de comparer les résultats des simulations des échantillons issues des stratégies séquentielle et intégrée.

- **Validation.** Cette phase a pour objectif de valider les conclusions de la phase précédente. Cette validation se fera comme suit : nous allons générer des échantillons d'ordonnancements optimisés par rapport aux mesures retenues comme robustes et issues de la phase précédente. Si après perturbations et classification des ordonnancements nous aboutissons aux mêmes résultats alors nous pourrons affirmer que ces mesures sont robustes. Bien sûr, si aucune mesure n'est retenue à la fin de la phase de classification, la phase de validation n'aura pas lieu d'être.

Ce processus est présenté dans la figure 4.5.

Dans ce qui suit, nous allons présenter l'approche que nous avons développée en suivant la méthodologie présentée dans cette section. Cette présentation est faite pour les trois questions posées dans la section 4.2, dont les spécificités sont détaillées dans la section suivante ainsi que les restrictions faites dans chaque cas. Nous développerons par la suite les notions fondamentales relatives à notre modèle. Nous définirons en particulier les mesures de robustesse retenues dans notre problème ainsi que les perturbations utilisées dans la phase perturbation du modèle (section 4.2.2). Nous définirons enfin, le concept de mention attribuée à un ordonnancement que nous avons utilisé dans la phase Classification (section 4.2.2).

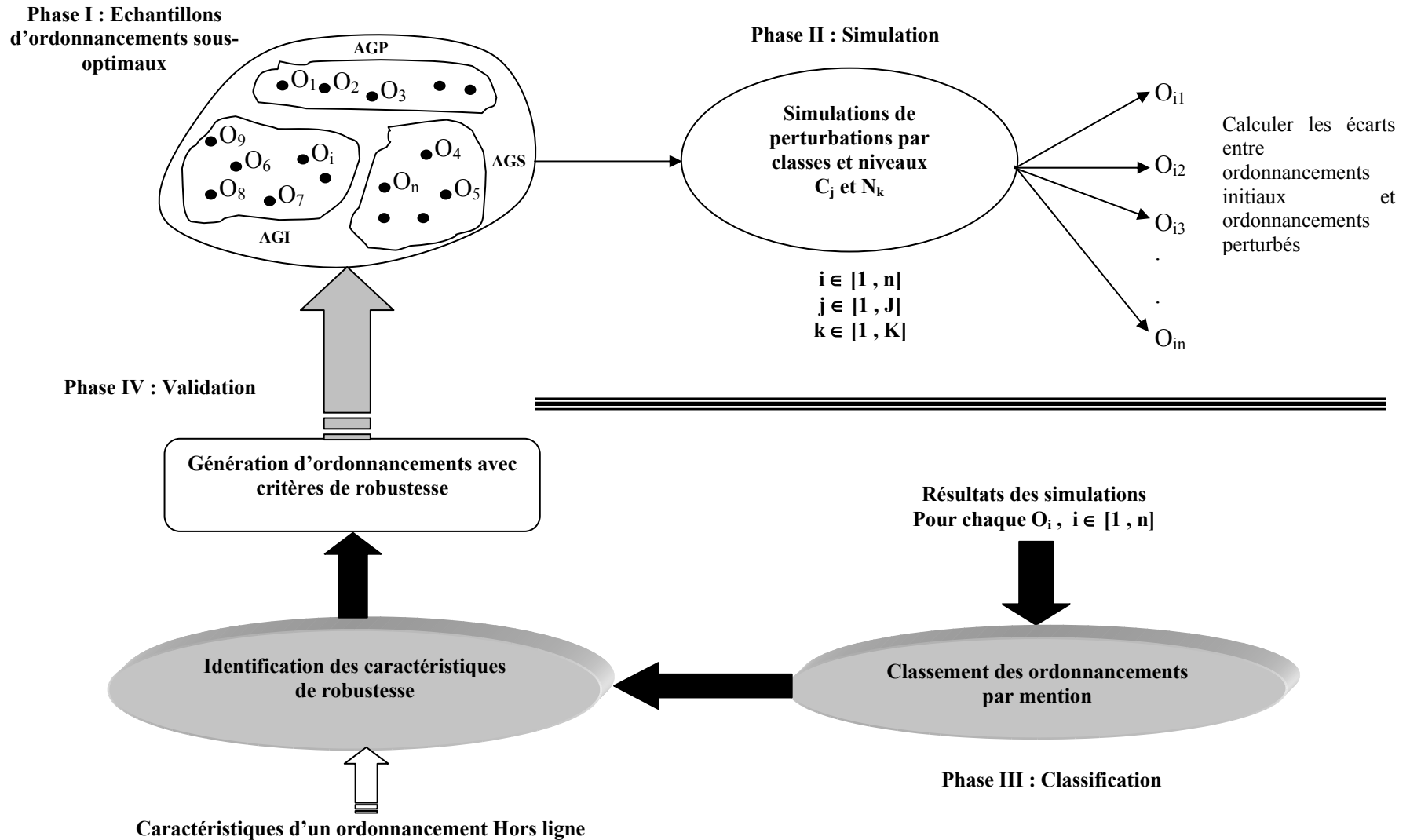


Figure 4.5: Démarche Classification – Validation

### 4.3. Définition du modèle

La mise en place de notre modèle nécessite la définition de certains concepts utilisés tels l'environnement du problème, les mesures de performances retenues et enfin le protocole de perturbation que nous avons adopté.

Dans ce qui suit, nous définissons ces concepts.

#### 4.3.1. Définition du problème et de son environnement

Nous considérons de manière générale tout problème d'ordonnancement classique dans lequel l'affectation des opérations aux machines est déjà faite. Et plus particulièrement le problème de *flowshop* de permutation. La description de ces problèmes a été donnée dans la section 2.3 du chapitre 2. Cette restriction n'est pas due au fait que le modèle ne serait pas valable pour des problèmes plus complexes, mais que le générateur d'ordonnements conjoints développé pour le chapitre 3 traite uniquement ce type de problèmes.

L'atelier est sujet à des perturbations sur les durées opératoires qui peuvent être modélisées par un rallongement sur cette durée aléatoire (augmentation des  $p_{is}$ ). Nous ne tenons pas compte des caractéristiques des perturbations (fréquence, durée, etc.).

#### 4.3.2. Définitions des mesures de performances retenues

Lors de la génération des échantillons d'ordonnements, on prendra en considération les mesures suivantes : la durée de l'ordonnement ( $C_{\max}$ ), Encours, *Idle Time*, la somme des retards de production ( $\sum T_i$ ), la somme des retards et avances des tâches de maintenance. Le choix des critères à optimiser pour la production n'est pas aléatoire dans le sens où il existe un lien entre le  $C_{\max}$  et les autres critères. Ils sont liés entre eux mais non dépendants l'un de l'autre. Si le premier varie, a priori, les autres varient dans le même sens. La fonction objectif est une fonction multicritère simplifiée.

Dans ce qui suit nous présentons en détail chacune de ces mesures.

##### ***Le Makespan ( $C_{\max}$ ):***

C'est un critère à minimiser, il représente la date de fin d'exécution de la dernière tâche de l'atelier. Il est donné par la formule suivante :  $C_{\max} = \text{Max}(c_{ij})$  où  $c_{ij}$  est la date de fin d'exécution de la tâche  $i$  sur la machine  $j$

**Encours :**

Minimiser la moyenne des temps de séjour des tâches de production, ou plus généralement

l'encours, dans un système revient à minimiser la valeur :  $Encours = \frac{1}{n} \sum_{i \leq n} (C_{im} - t_{i1})$

**Idle Time:**

Il s'agit des durées d'oisiveté des machines donc de l'atelier. Le temps à vide est la durée séparant deux tâches successives sur la même machine. La moyenne des temps à vide (en

anglais *idle time*) est défini par  $IdleTime = \frac{1}{m} \times \sum_{j=1}^m \sum_{i=1}^{k_j+n} (t_{i+1j} - C_{ij})$

Avec

$n$  : nombre de tâche de production.

$k_j$  : nombre de tâche de maintenance sur la machine  $j$ .

$m$  : nombre de machines dans l'atelier.

$c_{ij}$  : date de fin d'exécution de la tâche  $i$  sur la machine  $j$ .

$t_{i+1j}$  : date de fin d'exécution de la tâche  $i$  sur la machine  $j$ .

**La Somme des Retards :  $\sum T_i$** 

C'est un critère à minimiser. Il est donné par  $\sum T_i = \sum_{i=1}^n Max(o, C_{im} - d_i)$

Avec :

$C_{im}$  : date de fin d'exécution de la tâche de production  $i$  sur la dernière machine.

$n$  : le nombre de tâches de production.

**La fonction objectif  $f$** 

La fonction objectif  $f$  est présenté de manière détaillée dans la section 2.3 du chapitre 3. Pour notre modèle d'étude de la robustesse des ordonnancements conjoints, en présence de perturbations, elle combine un des critères d'optimisation de la production cités ci-dessus, noté  $f_1$  et une fonction  $f_2$  exprimant le respect des périodes de maintenance prévues.

$$f_2 = \sum_{j=1}^m \sum_{k=1}^{k_j} E'_{jk} + L'_{jk}$$

Avec :

$k_j$  : le nombre de tâches de maintenance effectif de chaque tâche de maintenance  $M_j$ .

$m$  : nombre de machines.

La fonction objectif est donnée par  $f = \alpha f_1 + \beta f_2$  où  $\alpha$  représente la contribution de la production et  $\beta$  celle de la maintenance. Nous avons utilisé  $\alpha=1$  et  $\beta=1$

Lors de l'étude des échantillons de production uniquement, la fonction objectif est obtenue en mettant le coefficient  $\beta$  à 0.

### 4.3.3. Définition du protocole de perturbation

Le type de perturbation pris en compte dans notre étude est la perturbation des durées opératoires des travaux selon des *niveaux* et des *classes*.

Un niveau détermine l'intensité d'élargissement de la durée opératoire d'une tâche de production. Une classe de perturbation représente le nombre de tâches de l'ordonnancement affectées par les perturbations.

Ce protocole fait référence aux travaux de Lawrence & al [LAW97]. Les résultats de leurs travaux sont exposés en détails en section 4.3.3. La différence dans notre modèle est que les classes et les niveaux ne sont pas statiques mais dynamiques. C'est-à-dire l'élargissement de la durée opératoire, ainsi que le nombre de tâches à perturber, peuvent être différents, pour un même niveau et une même classe de perturbation. Ils sont générés par une variable aléatoire uniforme bornée par le niveau et la classe de perturbation retenus.

#### 4.3.3.1 Les niveaux de perturbation

Nous avons défini, pour notre étude, 10 niveaux de perturbation selon des intervalles de longueurs égales à 10% de la durée opératoire originale de la tâche à perturber. Cette perturbation est obtenue en rallongeant la durée opératoire avec un pourcentage proportionnel à cette dernière. Chaque niveau comprend un intervalle de longueur 10, dans lequel les perturbations peuvent varier. Cette variation est obtenue par un coefficient noté  $CoefPert_{ij}$  pour une tâche (i, j) (Fig. 4.6). Ce coefficient est généré par une loi uniforme  $U(a,b)$  où a et b sont les bornes du niveau de perturbation.

Le tableau 4.2 définit les différents niveaux de perturbation pris en compte ainsi que les variables uniformes associées à chaque niveau.

Niveau	Intervalle	Intensité
1	[ 0%, 10%]	CoefPert <sub>1</sub> → U [0, 0.1]
2	] 10%, 20%]	CoefPert <sub>2</sub> → U [0.1, 0.2]
.	.	
.	.	
.	.	
9	] 80%, 90%]	CoefPert <sub>9</sub> → U [0.8, 0.9]
10	] 90%, 100%]	CoefPert <sub>10</sub> → U [0.9, 1]

Tableau 4.1: Les niveaux de perturbation

Les nouvelles durées opératoires des tâches de production seront obtenues de la manière suivante (Fig 4.6) :  $P'_i = P_i \times (1 + CoefPert_i)$

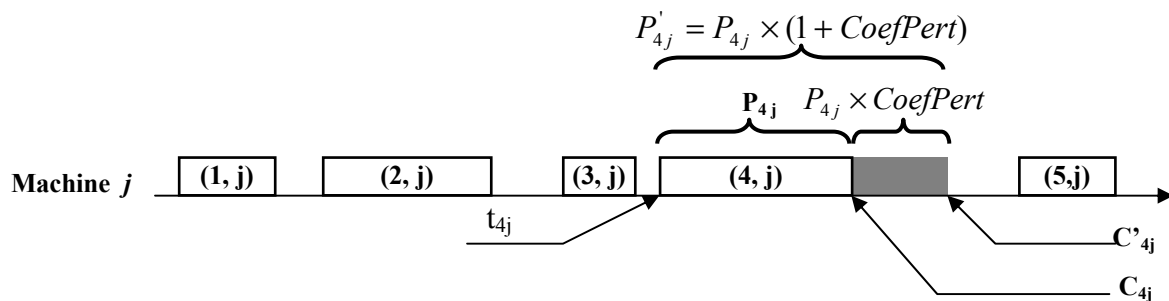


Figure 4.6 : Exemple de perturbation d'une tâche

Avec :

$P'_i$  : La nouvelle durée opératoire perturbée de la tâche  $i$ .

CoefPert <sub>$i$</sub>  : Le coefficient de perturbation de la tâche  $i$ .

#### 4.3.3.2 Les classes de perturbation

Trois classes de perturbation sont définies (Faible, Moyenne et Forte). Une classe représente le nombre de travaux qui vont être perturbés. Les tâches à perturber sont choisies aléatoirement dans un intervalle suivant une loi uniforme  $U [a, b]$ ,  $a$  et  $b$  étant les bornes de l'intervalle (Tab. 4.3).

Classe	Intervalle	Nombre / Classe
Faible	[0%, 40%]	$u \rightarrow U [0, 0.4] / n \times u$ Travaux perturbés
Moyenne	] 40%, 70%]	$u \rightarrow U ] 0.4, 0.7] / n \times u$ Travaux perturbés
Forte	] 70%, 100%]	$u \rightarrow U ] 0.7, 1] / n \times u$ Travaux perturbés

Tableau 4.2: Les classes de perturbation

Dans ce qui suit, nous allons définir pour chaque question posée dans la section 4.2.1, son environnement, ce qu'on appelle déviation ainsi que les différentes approches utilisées, pour chaque cas, pour mesurer la déviation de l'ordonnancement conjoint perturbé par rapport à l'ordonnancement prédictif initial. Ci-dessous, sont donnés les détails pour ces trois cas.

### 1. La Maintenance contribue-t-elle à la robustesse des ordonnancements conjoints?

L'objectif est d'arriver à démontrer que la perte de performance des ordonnancements conjoints production/maintenance du à l'insertion de la maintenance est inférieure à la perte de performance des ordonnancements de production uniquement, ceci dans le cas de l'occurrence de perturbations.

Pour aborder ce problème nous allons suivre la démarche suivante :

- Dans cette étude nous nous restreindrons à un seul critère d'optimisation  $C_{max}$ .
- Pour un problème donné<sup>6</sup> :

Solution théorique	$\left. \begin{array}{l} \text{AGP} \rightarrow C_{maxp} \\ \text{AGS} \rightarrow C_{maxs} \\ \text{AGI} \rightarrow C_{maxi} \end{array} \right\}$	$\left. \begin{array}{l} \Delta_{p-s}\% = \overline{C_{maxs}} - C_{maxp} / C_{maxp} \\ \Delta_{p-i}\% = \overline{C_{maxi}} - C_{maxp} / C_{maxp} \end{array} \right\}$
Solution Perturbée	$\left. \begin{array}{l} \text{AGP} \rightarrow \overline{C_{maxp}} \\ \text{AGS} \rightarrow \overline{C_{maxs}} \\ \text{AGI} \rightarrow \overline{C_{maxi}} \end{array} \right\}$	

Le calcul de la déviation entre la solution théorique et la solution perturbée dans le cas d'ordonnements de production et d'ordonnements conjoints, va permettre de mesurer l'impact de la maintenance sur les ordonnancements conjoints.

### 2. Existent-il des critères d'optimisation pouvant influencer sur la robustesse des ordonnancements?

L'objectif est d'arriver à identifier parmi quatre critères d'optimisation classiques ( $C_{max}$ , somme des retards, encours et iddle time) d'éventuelle(s) mesure(s) dont l'optimisation assure un certain degré de robustesse à l'ordonnement.

---

<sup>6</sup> P : Production, S : Séquentielle, I : Intégrée

Pour aborder ce problème nous allons suivre la démarche suivante :

Pour un problème donné<sup>7</sup> :

$$\begin{array}{l} \text{Solution théorique} \\ \left. \begin{array}{l} \text{AGP} \rightarrow (C_{\max}, \Sigma T_i, \text{Encours}, \text{Iddle Time})_P \\ \text{AGS} \rightarrow (C_{\max}, \Sigma T_i, \text{Encours}, \text{Iddle Time})_S \\ \text{AGI} \rightarrow (C_{\max}, \Sigma T_i, \text{Encours}, \text{Iddle Time})_I \end{array} \right\} \end{array}$$

$$\begin{array}{l} \text{Solution Perturbée} \\ \left. \begin{array}{l} \overline{\text{AGP}} \rightarrow (\overline{C_{\max}}, \overline{\Sigma T_i}, \overline{\text{Encours}}, \overline{\text{Iddle Time}})_P \\ \overline{\text{AGS}} \rightarrow (\overline{C_{\max}}, \overline{\Sigma T_i}, \overline{\text{Encours}}, \overline{\text{Iddle Time}})_S \\ \overline{\text{AGI}} \rightarrow (\overline{C_{\max}}, \overline{\Sigma T_i}, \overline{\text{Encours}}, \overline{\text{Iddle Time}})_I \end{array} \right\} \end{array}$$

$$\Delta_{p-s}\% = \overline{C_{is}} - C_{ip} / C_{ip}$$

$$\Delta_{p-i}\% = \overline{C_{ii}} - C_{ip} / C_{ip}$$

$$C_i \in \{C_{\max}, \Sigma T_i, \text{Encours}, \text{Iddle Time}\}$$

Le calcul de la déviation entre la solution théorique et la solution réelle (après perturbation), dans le cas d'ordonnements de production et d'ordonnements conjoints, pour chacun des quatre critères retenus, va permettre de mesurer la sensibilité de ce dernier par rapport aux perturbations, selon les étapes suivantes :

1. Simulation des perturbations avec un niveau et une classe ( $n_i, c_j$ ), et pour chaque critère  $C_i$
2. Calcul de la valeur des nouveaux critères après perturbations  $C_i$  et des déviations
3. Classification

### 3. Existe-t-il des méthodes d'ordonnements conjoints robustes ?

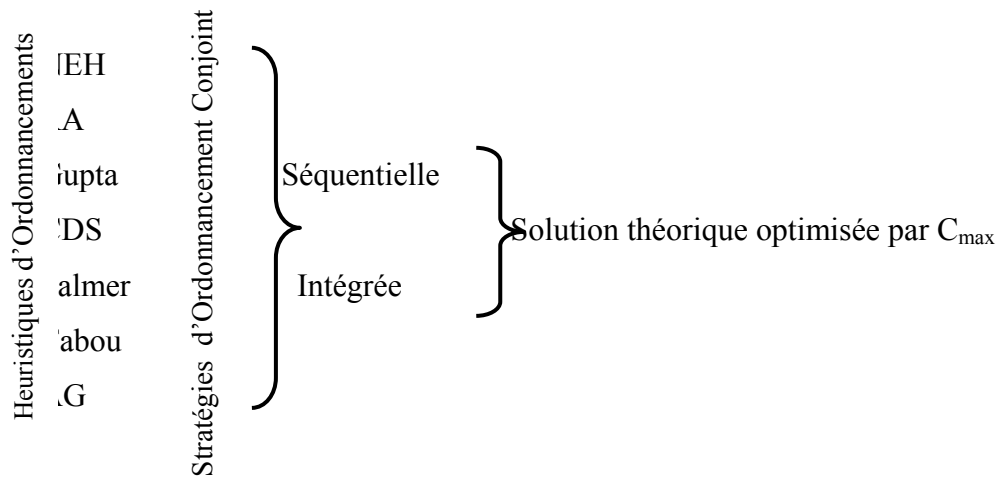
Pour cette partie aussi, nous nous restreindrons dans les simulations à un seul critère d'optimisation  $C_{\max}$ .

De plus si la réponse à la première question est positive, nous travaillerons uniquement avec les échantillons mixtes production/maintenance : AGS et AGI. Chaque échantillon ne va pas être généré par les AG mais par les méthodes d'ordonnements développées dans le chapitre précédent selon le schéma suivant :

---

<sup>7</sup> P : Production, S : Séquentielle, I : Intégrée.





Le processus de simulation est le suivant :

- Simulation des perturbations avec un niveau et une classe ( $n_i, c_j$ )
- Calcul du nouveau  $C_{\max}$  après perturbation :  $\overline{C_{\max}}$
- Classement des méthodes sur la base des pourcentages de cas acceptables

#### 4.3.4. Définition de la mention d'un ordonnancement

Pour étudier la contribution de la maintenance à la robustesse des ordonnancements, nous allons utiliser, dans nos tests, des hiérarchies et des statistiques. Les hiérarchies utilisent, comme critère de classification, des valeurs énumérables (connues à l'avance). Or la valeur du critère de classification que nous utilisons dans la construction des arborescences n'est pas connue à l'avance (voir le schéma de classification de la section 4.2.2). Pour pallier à cela, nous procéderons à la subdivision de la fonction objectif en sous intervalles. Dans le but de donner une homogénéité aux mentions obtenues, nous avons utilisé la différence entre la valeur de la performance de l'ordonnancement perturbé et celle de l'ordonnancement original, au lieu d'utiliser seulement celle de l'ordonnancement perturbé.

La construction des intervalles des mentions est guidée par le rapport coût/performance. Le coût désigne le coût de la génération d'un nouvel ordonnancement. Quant à la performance, elle représente l'augmentation permise dans la valeur de la fonction objectif, après les perturbations. Dans la majorité des expériences dans le domaine de la robustesse, la marge permise d'augmentation de la valeur de la fonction objectif, après perturbations est de 50%.

Donc, nous avons fixé comme marge d'augmentation maximale de la fonction objectif, la valeur de 50%. Au-delà de cette valeur, la génération d'un nouvel ordonnancement sera moins coûteuse que l'utilisation de cet ordonnancement. Les bornes des intervalles des mentions

représentent les marges d'augmentation de la valeur du critère de performance de l'ordonnement perturbé.

Les mentions ainsi que les intervalles correspondants, sont illustrés dans le tableau 4.4.

Mention	Intervalle
Très Bien (TB)	[0 , 5%]
Bien (B)	]5%, 15%]
Moyen (Moy)	]15%, 35%]
Mauvais (M)	]35%, 50%]
Très Mauvais (TM)	> 50%

Tableau 4.3 : Sous-Intervalles des mentions

#### 4.4. Implémentation du modèle

Dans les sections précédentes, nous avons donné quelques définitions nécessaires pour la conception du modèle pour l'étude de la contribution de la maintenance à la robustesse des ordonnements conjoints production/maintenance en présence de perturbations. Dans ce modèle, les perturbations affectent deux types d'ordonnements : des ordonnements de production et des ordonnements conjoints production/maintenance.

Dans ce qui suit nous allons présenter les deux types de perturbation à considérer lors de la phase perturbation, présentée dans la section 4.2.2, de notre modèle. La première concerne les ordonnements de production et la seconde les ordonnements conjoints production/maintenance. Pour chacun des trois aspects que nous voulons étudier et que nous avons présenté en section 4.2 de ce chapitre, nous avons adopté la démarche de perturbation suivante pour l'ordonnement de production puis pour l'ordonnement conjoint production/maintenance. Cette distinction entre les deux types d'ordonnements est faite pour étudier l'impact des perturbations sur des ordonnements de production, donc sans maintenance, par rapport aux ordonnements conjoints, donc avec maintenance.

Dans cette section, nous commencerons par définir la notion de chevauchement des tâches quelles soient de production ou de production et de maintenance. Puis présentons le principe de perturbation (élargissement des durées aléatoires) des deux types d'ordonnements étudiés : production et conjoint production/maintenance, ainsi que les techniques utilisées pour résoudre le problème de chevauchement des tâches dans les deux cas.

##### 4.4.1. Le chevauchement entre tâches

L'élargissement des durées opératoires des tâches de production, peut provoquer des chevauchements entre deux ou plusieurs tâches de production. Un chevauchement entre deux tâches est défini par "*l'entrelacement ou la superposition de deux tâches après décalage à droite de la première pour une raison quelconque*". Le rallongement des durées opératoires est une des raisons qui causent les chevauchements.

Deux types de chevauchement sont possibles : le **Chevauchement Horizontal** (entre deux tâches se succédant sur la même machine) et le **Chevauchement Vertical** (entre deux tâches d'un même job). Le chevauchement horizontal affecte l'exécution de deux tâches qui se succèdent sur la même machine, alors que le chevauchement vertical affecte l'exécution de la même tâche mais sur deux machines qui se succèdent.

Comme le montre la figure 4.7, la tâche (2, j+1) s'exécutant sur la machine  $M_{j+1}$  doit s'exécuter après la tâche (2, j). Or la perturbation de cette tâche a provoqué un chevauchement **vertical** avec la tâche (2, j+1). Par contre, la tâche (5, j) doit commencer après la tâche (4, j). Or la perturbation de cette tâche a provoqué un chevauchement **horizontal** ( $c'_{4j} > t_{5j}$ ).

Pour y remédier, La méthode de décalage à droite (en anglais right-shift rescheduling) est appliquée car elle résout le problème du chevauchement tout en préservant l'ordre des tâches. Et ceci pour maintenir le séquençement original si on veut faire une comparaison valide entre les performances de l'ordonnancement original et toutes ses dérivés obtenues après la perturbation de ce dernier.

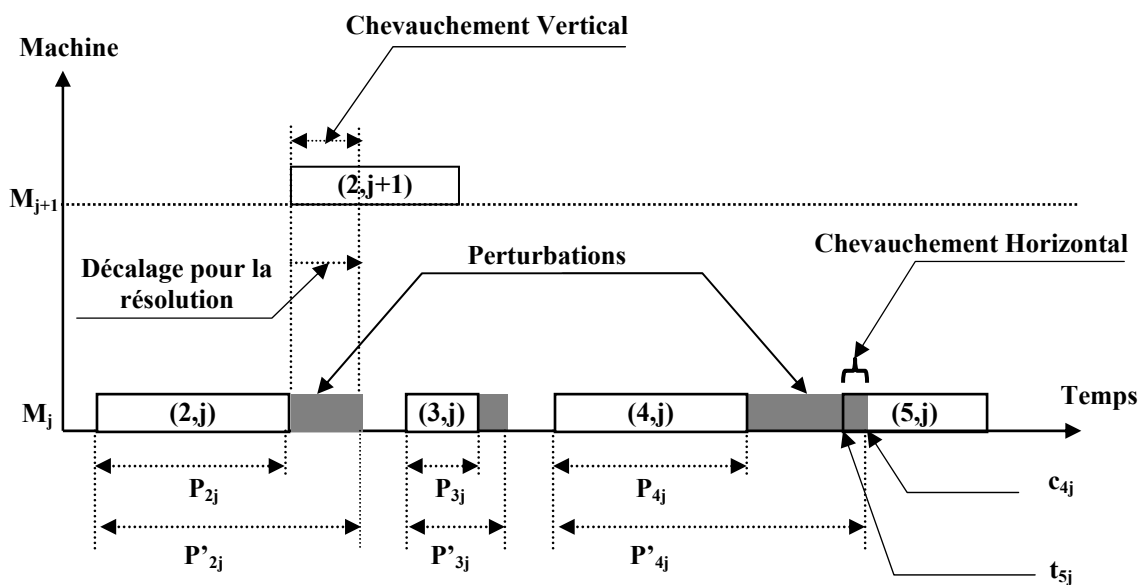


Figure 4.7: Chevauchement horizontal et vertical entre deux tâches

Nous allons présenter dans ce qui suit le principe de perturbation de chaque type d'ordonnement étudiés : ordonnancement de production et ordonnancement conjoint production/maintenance, ainsi que les techniques utilisées pour résoudre, dans chaque cas, les chevauchements induits par les perturbations.

#### 4.4.2. Procédure de perturbation d'un ordonnancement de production

La procédure de perturbation d'un ordonnancement de production est donnée par l'algorithme 4.1. Elle permet de générer un ordonnancement perturbé selon les niveaux et classes de perturbation. La variable NbOp détermine une classe de perturbation, elle représente le nombre de tâches à perturber, et Niv représente un niveau de perturbation variant de 0 à 10. Pour la mise à jour des nouvelles durées opératoires, on modifie les durées opératoires des tâches perturbées. Les chevauchements entre tâches provoqués par le rallongement des durées opératoires de ces tâches sera résolu par application de la règle de décalage à droite (en anglais *right-shift rescheduling*), comme nous l'avons présenté dans la section précédente. Cette procédure est appliquée pour chaque mesure d'optimisation retenue (section 4.3.2 de ce chapitre).

---

#### Algorithme Perturbation Ordonnement de Production

---

**Variables** NbOp : Classe de perturbation, Niv : Niveau de perturbation, m : nombre de machines, n : nombre de tâches

**Début**

**Pour** Niv=0 à 10

**Faire Pour** Chaque intervalle appartenant à l'ensemble des classes  $\{[0, 0.4], ] 0.4, 0.7], ] 0.7, 1]\}$ . Soient a, b leurs bornes.

**Faire**

Générer NbOp  $\sim U [a, b]$

**Pour** l=1 à NbOp\*n

**Faire** Générer i  $\sim U [0, n]$  et j  $\sim U [0, n]$

$$CoefPert = \frac{Niv}{10} \quad \text{et} \quad P'_{ij} = P_{ij} \times (1 + CoefPert)$$

Mise à jour de l'ordonnement avec prise en considération de la nouvelle durée opératoire et les chevauchements.

**Fin Pour**

Calculer le critère d'optimisation pour l'ordonnement (§ 4.3.2).

**Fin Pour**

**Fin Pour**

**Fin.**

---

Algorithme 4.1: Principe de perturbation de l'ordonnement de production

Dans la section suivante nous présenterons la procédure de perturbation d'un ordonnancement conjoint production/maintenance ainsi que le principe de résolution des chevauchements, dans les ordonnancements conjoint, entre les tâches de production et ceux de maintenance.

Comme la règle de décalage à droite n'est pas applicable, dans ce cas du fait de la spécificité des tâches de maintenance qui ne peuvent être décalées sans risque d'être déplacées en dehors de leur intervalle de tolérance, nous présenterons les règles que nous avons utilisées pour résoudre ce problème.

#### 4.4.3. Procédure de perturbation d'un ordonnancement conjoint

Après maintenance des machines, leur fiabilité va augmenter, induisant ainsi l'absorption de certains niveaux de perturbations. Nous avons simulé cette absorption par la diminution des perturbations. La figure 4.8 donne l'évolution de l'intensité de perturbation dans une durée de temps bornée par deux opérations de maintenance successives sur la même machine.

Nous avons modélisé l'effet de la maintenance sur la fiabilité d'une machine par deux intervalles: un Intervalle de Fiabilité (IF) et un Intervalle de Perturbation (IP) représentés par les deux parties discontinues sur le graphe.

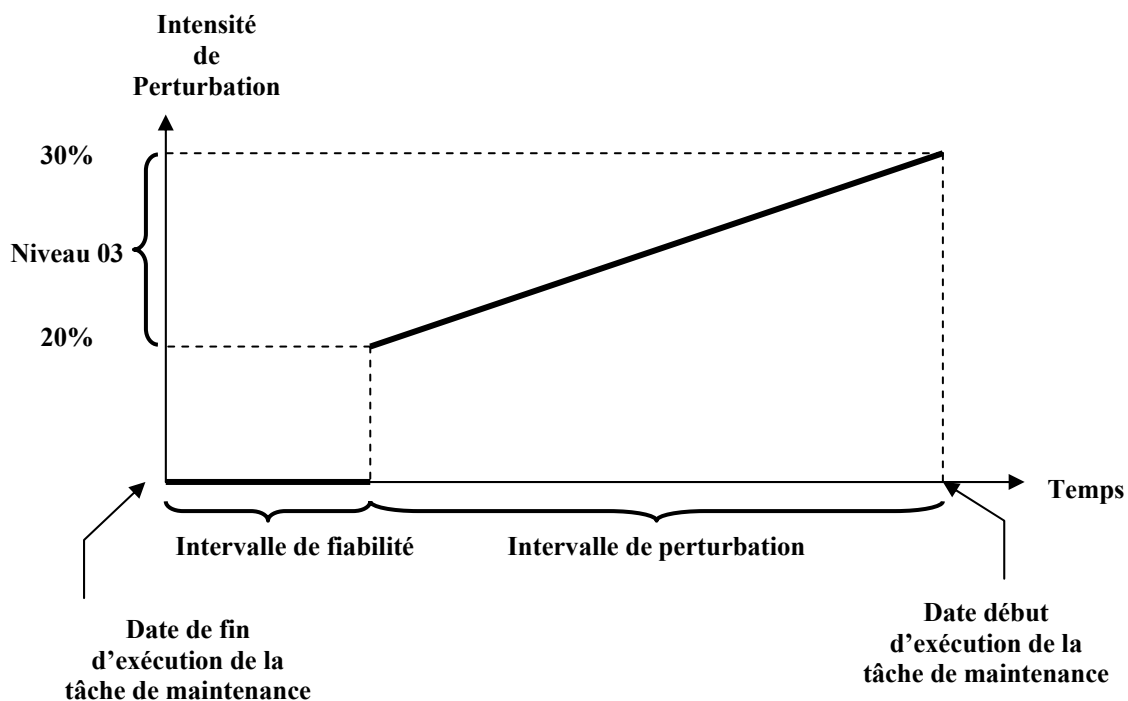


Figure 4.8 : Evolution de l'intensité des perturbations d'un niveau

Dans l'intervalle de fiabilité, on ne tolère aucune perturbation sur les durées opératoires des tâches de production, car la machine a été récemment maintenue, donc fiable. Cet intervalle est une donnée de maintenance, il dépend seulement de la machine concernée par la maintenance. Par contre, dans l'intervalle de perturbation, l'intensité de perturbation commence de la borne inférieure de l'intervalle du niveau de perturbation concerné et augmente jusqu'à ce qu'elle atteigne la borne supérieure de cet intervalle.

La figure 4.9 présente les intervalles de fiabilité et de perturbation sur l'ordonnement conjoint production/maintenance. Le graphe précédent sera constant et égal à la borne supérieure du niveau de perturbation dans le cas d'une machine non entretenue (échantillon de production).

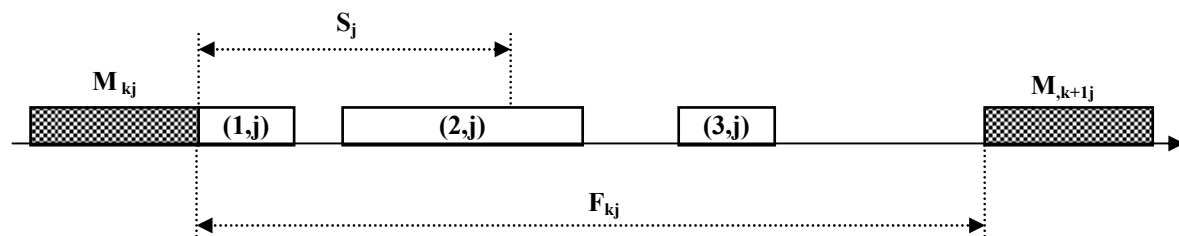


Figure 4.9 : Intervalle de fiabilité

Avec :

$M_{kj}$  :  $K^{\text{ième}}$  tâche de maintenance sur la machine  $j$ .

$F_{kj}$  : Durée séparant les tâches de maintenance  $M_{kj}$  et  $M_{k+1j}$ .

$S_j$  : Intervalle de Fiabilité de la machine  $j$ , représentant la période durant laquelle aucune perturbation n'a lieu sur cette machine après avoir effectué une tâche de maintenance  $M_{jk}$ .

Les situations qu'on peut rencontrer sont :

- (1,  $j$ ) ne subit aucune perturbation car elle se situe dans l'intervalle de Fiabilité.
- (2,  $j$ ) ne subit aucune perturbation car on prend comme hypothèse de ne pas perturber les tâches commençant avant la fin de  $S_j$ .
- (3,  $j$ ) peut subir des perturbations, l'intensité de la perturbation est fonction de son éloignement de la tâche de maintenance qui la précède.

#### 4.4.3.1 Algorithme de perturbation d'un ordonnancement conjoint

Afin d'introduire l'effet de la maintenance sur les perturbations des durées opératoires, nous avons été amené à modifier la procédure de perturbation déjà appliquée à l'échantillon de production et présentée dans la section précédente.

On prend un intervalle de fiabilité  $S_j$  après chaque tâche de maintenance. Cet intervalle est lié à la machine concernée par la maintenance, i.e., il est différent d'une machine à une autre, de manière formelle l'intervalle de fiabilité doit être inférieur à la période de maintenance (durée séparant deux tâches de maintenance successives sur la même machine dans l'ordonnancement), i.e.,  $S_j \leq F_{kj}$ .

Le coefficient de perturbation  $\text{CoefPert}$  (section 4.3.3) dépend :

- du niveau (l'intensité de la perturbation à prendre en considération) et de la classe de perturbation (la décision de perturber ou non l'opération) définis dans la section 4.3.3.
- de l'emplacement (la distance) de la tâche de production par rapport à la dernière tâche de maintenance effectuée sur la même machine. Par exemple, si on décide de perturber les deux tâches (4,j) et (5,j) de la figure 4.10, cette dernière subira une perturbation plus intense par rapport à la première.

Pour cette dernière raison nous avons décidé de subdiviser l'intervalle dans lequel une tâche peut subir une perturbation (intervalle de perturbation  $I_{jk}$ ) en dix sous-intervalles. Ces sous-intervalles permettent de localiser les tâches de productions, par rapport aux tâches de maintenance, et de concrétiser l'effet des tâches de maintenance sur les tâches de production i.e. la diminution de l'effet des perturbations prises en compte (Fig. 4.10).

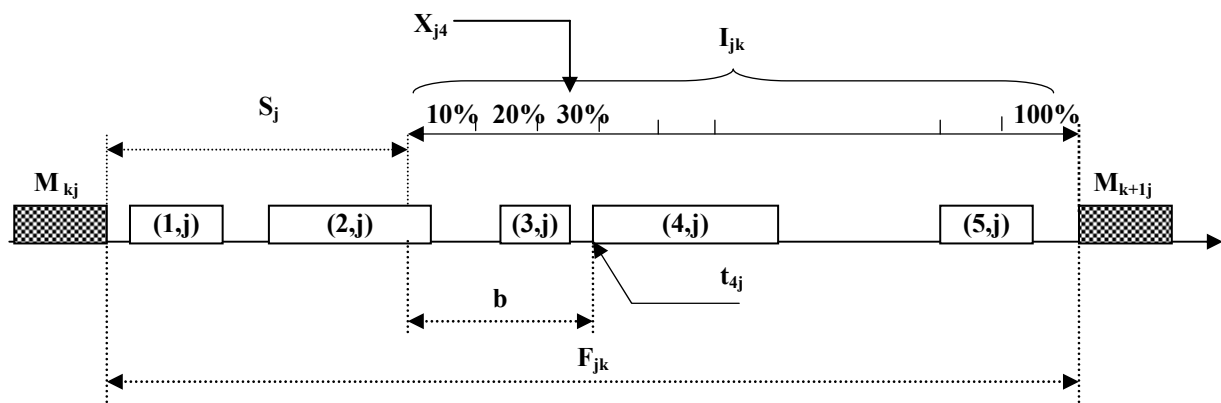


Figure 4.10 : Intervalle de Fiabilité et de Perturbation

Si on veut perturber la tâche (4, j), alors la nouvelle durée opératoire  $P'_{4j}$  sera obtenue de la manière suivante :  $P'_{4j} = P_{4j} \times (1 + CoefPert_{4j})$

**(i) Détermination du coefficient de perturbation**

Comme le coefficient de perturbation dépend d'une part du niveau et de la classe de perturbation et d'autre part de l'emplacement de la tâche à perturber (par rapport à la dernière tâche de maintenance effectuée sur la même machine), sa valeur est donnée par :

$$CoefPert_{ij} = X_{ij} \times Niv$$

Avec

$X_{ij}$  : le sous-intervalle dans lequel se trouve la tâche i (voir le schéma précédent).

Niv : La borne supérieure du niveau de perturbation pris en compte, exprimé en pourcentage.

**(ii) Valeur de  $X_{ij}$**

$X_{ij}$  permet de déterminer la durée de temps à ajouter par rapport au niveau de perturbation traité. Pour cela, on aura besoin de connaître la date de début d'exécution  $t_{ij}$  de la tâche de production (i,j) . Les étapes nécessaires pour trouver cette valeur sont :

Soit b (Fig. 4.10) la durée séparant la date d'exécution de la tâche (i, j) et la date de fin de l'intervalle de fiabilité  $S_j$  sur la même machine :  $b = t_{ij} - (C_{kj} + S_{kj})$ .

*Sous-intervalle de la tâche (i, j).*

On calcule la durée réservée pour les perturbations en utilisant la relation  $I_{kj} = F_{kj} - S_{kj}$  . En effectuant une règle de trois on pourra trouver le nombre de subdivisions x se trouvant dans b, afin de trouver la subdivision dans laquelle se trouve la tâche (i,j), de la manière suivante :

$$x = \left[ \frac{b \times 10}{I_{kj}} \text{ Subdivisions} \right] \text{ Partie entière.}$$

Enfin, on remplacera x dans la formule  $X_{ij} = \frac{x}{10}$ , on aura alors :

$$X_{ij} = \frac{\lfloor t_{ij} - (C_{kj} + S_{kj}) \rfloor \times 10}{F_{kj} - S_{kj}} = \frac{t_{ij} - (C_{kj} + S_{kj})}{F_{kj} - S_{kj}}$$

Donc la formule générale du coefficient de perturbation est donné par :

$$CoefPert_{ij} = \frac{t_{ij} - (C_{kj} + S_{kj})}{F_{kj} - S_{kj}} \times Niv\%$$



L'algorithme 4.2 permet de générer les 30 ordonnancements perturbés selon les niveaux et classes de perturbation puisqu'il y a 10 niveaux et 3 classes de perturbations possibles. La variable NbOp détermine la classe de perturbation. Pour la mise à jour des nouvelles durées opératoires, on modifie seulement les durées opératoires des tâches de production qui ont été sélectionnées pour être perturbées, ainsi que les tâches de production et de maintenance affectées par cette modification (possibilité d'utilisation des techniques de résolution de chevauchement). Cet algorithme est appliqué pour chaque mesure d'optimisation.

---

**Algorithme** Perturbation ordonnancement Conjoint Production/Maintenance

---

**Variabes** NbOp : Classe de perturbation, Niv : Niveau de perturbation, m : nombre de machines, n : nombre de tâches ; a et b les bornes.

**Début**

**Pour** Niv=0 à 10

**Faire Pour** Chaque intervalle appartenant à l'ensemble des classes

{[0, 0.4], [0.4, 0.7], [0.7, 1]}.

Générer NbOp ~ U [a, b]

**Pour** l=1 à NbOp\*N

**Faire Générer** i ~ U [0, n]

Générer j ~ U [0, m]

Chercher  $M_k$  i.

**Si**  $t_{ij} < C_{kj}^M + S_{kj}$

**Alors**  $P'_{ij} = P_{ij}$

**Sinon**  $CoefPert_{ij} = \frac{t_{ij}^P - (C_{kj}^M + S_{kj})}{F_{kj} - S_{kj}} \times Niv\%$

$P'_{ij} = P_{ij} \times (1 + CoefPert_{ij})$

**Fin Si**

Mise à jour de s Ordonnements.

**Fin Pour**

Calculer les nouvelles performances pour l'ordonnement.

**Fin Pour**

**Fin Pour**

**Fin.**

---

Algorithme 4.2: Principe de perturbation d'une ordonnancement conjoint

#### 4.4.3.2 Résolution des chevauchements entre tâches

La perturbation d'une seule tâche peut provoquer tout un changement dans l'ordonnement, car des chevauchements peuvent survenir lors de l'accroissement des durées opératoires, les deux types de chevauchement (vertical et horizontal) déjà traités peuvent être rencontrés. Le décalage à droite (en anglais *Right Shifting*) sera utilisé dans le cas du chevauchement

vertical. Mais pour le chevauchement horizontal, on peut rencontrer deux cas de chevauchements selon le type de la tâche mise en cause (tâche de maintenance ou tâche de production).

Comme le montre la figure 4.11, dans le cas où le chevauchement est entre deux tâches de production, un décalage suffit à rétablir la situation. La tâche (5, j) doit commencer après (4, j). Or la perturbation de la tâche (4, j) a provoqué un chevauchement ( $C'_{4j} > t_{4j}$ ). Dans ce cas, on doit décaler la tâche (5, j).

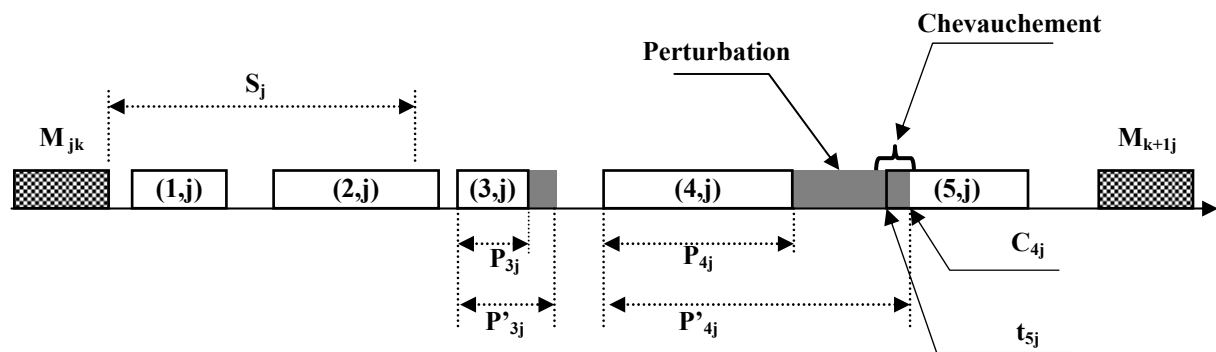


Figure 4.11 : Chevauchement horizontal entre deux tâches de production

Par contre dans le cas où le chevauchement est entre une tâche de production et une tâche de maintenance deux cas se présentent :

- Si le décalage de la tâche de maintenance concernée ne provoque pas le dépassement de son  $T_{max}$  (Fig. 4.12), alors un décalage à droite sera effectué.
- Si le décalage de cette tâche provoque le dépassement de son  $T_{max}$ , le décalage ne se fera pas. Une permutation de la tâche de maintenance et de la tâche de production qui a provoqué le chevauchement (Fig. 4.13) sera appliquée pour résoudre le problème de chevauchement. La tâche de production qui a subi une permutation ne subira pas de perturbation, car elle se situera juste après la tâche de maintenance, donc dans la période où aucune perturbation ne sera prise en compte. La date de début d'exécution de la tâche de maintenance sera choisie de telle sorte qu'elle ne survienne pas avant son  $T_{min}$  et sera dans le meilleur emplacement possible.

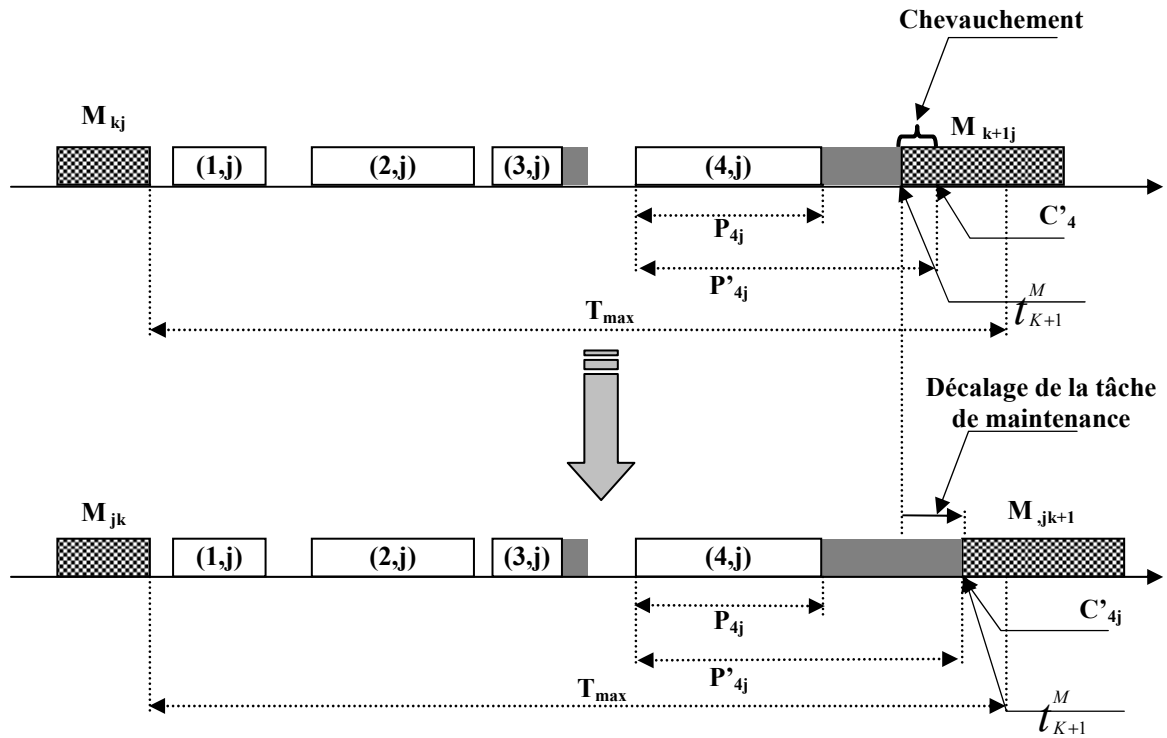


Figure 4.12 : Résolution du chevauchement P/M par décalage à droite (Right Shifting)

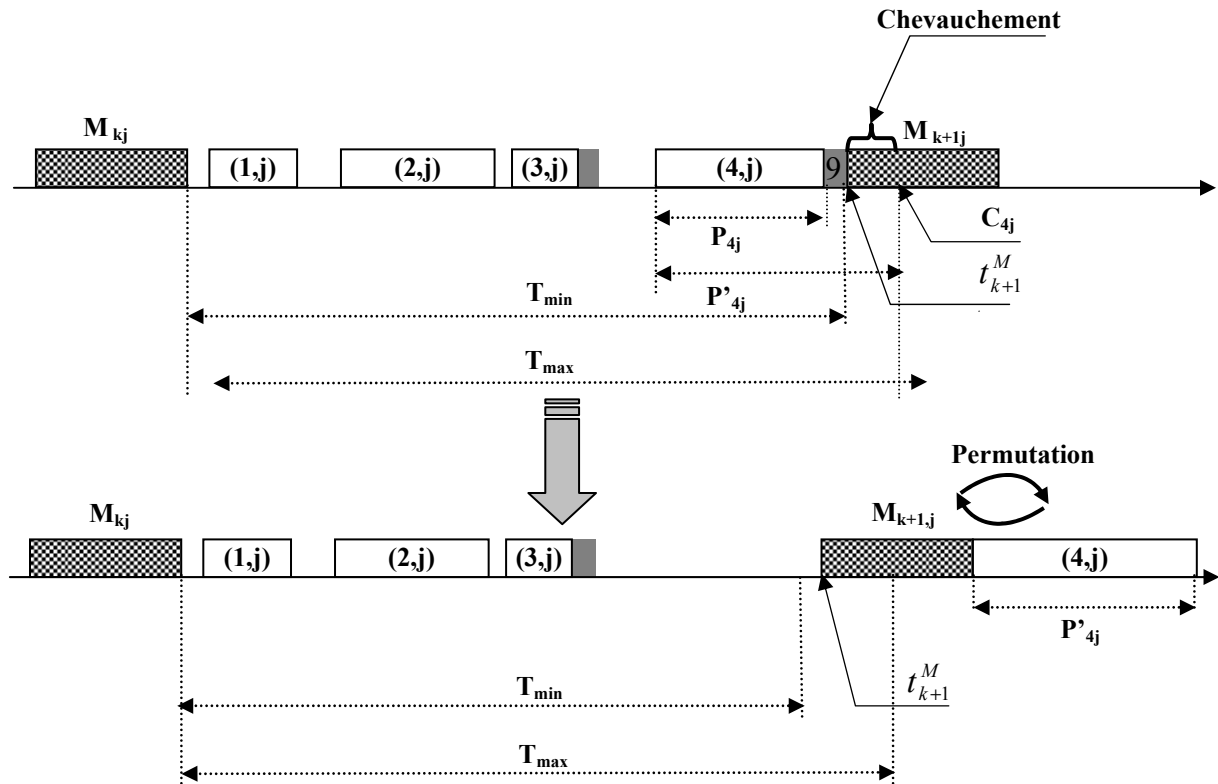


Figure 4.13: Résolution du chevauchement P/M par permutation

Dans le premier cas (résolution du chevauchement entre tâche de production et tâche de maintenance par décalage à droite, Fig. 4.12), on remarque que les critères d'optimisation somme des retards et  $C_{\max}$  se dégradent forcément à cause des décalages des tâches de maintenance et de production.

Dans le deuxième cas (résolution du chevauchement entre tâche de production et tâche de maintenance par permutation, Fig. 4.13), on remarque qu'il y a accroissement de la somme des retards, mais que le  $C_{\max}$  est amélioré, cela est dû à la non perturbation de la tâche de production permutée avec la tâche de maintenance.

Reste à comparer entre les taux de dégradation de ces critères pour l'échantillon de production et production/maintenance, afin de déterminer quel type d'échantillon est plus sensible à ces chevauchements.

#### **4.5. Mise en œuvre du modèle**

Dans les sections précédentes nous avons donné quelques définitions nécessaires pour la conception d'un modèle pour l'étude de la robustesse des ordonnancements conjoint production/maintenance, en présence de perturbations. Dans ce modèle, nous générons des perturbations sur les durées aléatoires sur deux types d'ordonnements : ordonnancement de production et ordonnancement conjoints. Ces ordonnancements sont des algorithmes génétiques, développés dans le chapitre 3, pour la production et adaptés dans le cas conjoint production/maintenance. L'algorithme génétique de production sert de base de comparaison pour l'ordonnement conjoint production/maintenance. Le but de cette démarche est l'étude de la contribution de la maintenance à la robustesse de ces derniers.

La mise en œuvre du modèle concerne les phases Perturbations, Classification et Validation. La phase Echantillonnage est représentée par le générateur d'ordonnements conjoints, développé dans le cadre des travaux développés dans le chapitre précédent. La phase de Classification va permettre de répondre aux trois questions à la base de notre étude : contribution de la maintenance à la robustesse des ordonnancements conjoints, identification d'éventuelles méthodes robustes et des critères susceptibles d'améliorer la robustesse des ordonnancements conjoints de production et de maintenance. Par contre, la phase Validation va effectuer le processus inverse pour confirmer si les ordonnancement générés en respectant ces critères sont bien robustes.

Avant de présenter les étapes de ce processus, nous définissons trois notions utilisées dans ces étapes. La première notion est **l'ordonnancement original** qui désigne un ordonnancement (individu) non encore perturbé issu de l'un des trois échantillons : AGP, AGS ou AGI. N'importe quel ordonnancement appartenant à l'un de ces trois échantillons est considéré comme ordonnancement original. Quant à la deuxième notion qui est **l'ordonnancement perturbé**, elle désigne un ordonnancement mais après sa perturbation par un niveau et une classe donnés. Pour chaque ordonnancement original, nous avons 30 ordonnancements perturbés correspondants. En effet, nous avons défini 10 niveaux et 3 classes de perturbations. La combinaison de ces deux critères donne 30 types de perturbations possibles.

La dernière notion est **une simulation** qui représente l'instanciation d'un ordonnancement perturbé. La différence entre deux simulations, effectuées sur le même ordonnancement, se situe au niveau du choix des tâches à perturber car si le protocole de perturbation reste le même, les tâches à perturber peuvent être différentes d'une simulation à une autre du fait que dans une classe la génération du nombre de tâche à perturber est dynamique.

Maintenant, nous décrivons les étapes suivies afin d'avoir la performance d'un ordonnancement perturbé. Nous commençons par perturber un ordonnancement original d'un échantillon donné selon les classes et niveaux. Ces perturbations engendreront 30 ordonnancements perturbés. Les simulations des ordonnancements de chaque échantillon sont réalisées 100 fois. Les résultats présentés donneront les moyennes des 100 simulations.

Afin d'étudier la répartition de la performance de l'ordonnancement perturbé, nous attribuons une **mention** à cet ordonnancement après perturbation selon le tableau 4.3. D'autre part, nous désignons par ordonnancement acceptable tout ordonnancement dont la déviation de sa performance, après perturbation, par rapport à celle de l'ordonnancement original est restée dans un intervalle acceptable. Sa mention est alors : Très Bien (TB), Bien (B) ou Moyenne (Moy). Il sera rejeté, si la mention qu'il a obtenue après perturbation est Mauvaise (M) ou Très Mauvaise (TM).

Le processus de classification que nous présentons ci-après est général et identique pour les trois méthodes de génération des ordonnancements : AGP, AGS et AGI. Donc, il suffit de le décrire pour une seule méthode de génération. Ce processus suit les trois étapes suivantes (Fig. 4.15) :

**1. Génération des échantillons d'ordonnancements originaux.** Pour chaque mesure (§4.3.2), nous générons un échantillon de 20 ordonnancements obtenu par l'une des deux stratégies de

génération (Séquentielle : première et deuxième étape ou Intégrée). A la fin de cette étape, nous aurons cinq sous échantillons pour chacun des trois échantillons originaux de Production ou Conjoint Production/Maintenance selon la stratégie et la mesure de performance.

**2. Perturbations des ordonnancements.** La deuxième étape consiste en la perturbation des échantillons originaux, obtenus à la fin de l'étape précédente. Le processus de perturbation de chaque échantillon génère ainsi un échantillon d'ordonnements perturbés selon les 10 niveaux et les 3 classes de perturbation. Cette étape génère pour chaque échantillon original, cinq sous échantillons perturbés (un sous échantillon par mesure).

**3. calcul des performances.** Lors de cette étape, on calcule les performances des divers ordonnancements perturbés, issus des 5 échantillons, selon un niveau et une classe de perturbation donnée.

A la fin de cette étape, nous aboutissons à la subdivision des divers ordonnancements des cinq échantillons en cinq groupes. Chaque groupe possède une mention (TB, B, Moy, M, TM définie dans la section 4.3.4). Ces mentions sont utilisées, comme critère de classification intrinsèque, dans l'étape de la hiérarchisation des cinq échantillons. Cette hiérarchisation est intrinsèque car nous n'effectuons la hiérarchisation que des individus obtenus avec la même méthode de génération.

Dans le but d'effectuer une comparaison entre les performances, des divers ordonnancements perturbés, obtenus avec les trois méthodes de génération, nous utilisons des statistiques sur les mentions de ces ordonnancements perturbés. Les statistiques nous permettent d'appréhender l'effet de la maintenance sur la robustesse des ordonnancements. Ce processus est présenté dans la figure 4.14.

Le processus de validation est appliqué uniquement dans le cas où il y a émergence d'un critère générique à partir de l'étape de classification. Il est identique au précédent dans la gestion des perturbations, sauf sur trois points. Le premier se trouve au niveau de la génération des échantillons originaux (AGP, AGS, AGI). Car nous prendrons en compte, cette fois-ci, lors de la génération d'un échantillon dans ce schéma, le respect de(s) critère(s) de classification. Ce(s) critère(s) de classification est (sont) obtenu(s) à partir du schéma de classification précédent. La deuxième différence réside au niveau de l'utilisation de comparaison entre les résultats obtenus dans l'étape de classification et les résultats obtenus de cette étape. Il existe deux situations de comparaison : la première comparaison concerne la hiérarchisation et la deuxième concerne les statistiques.

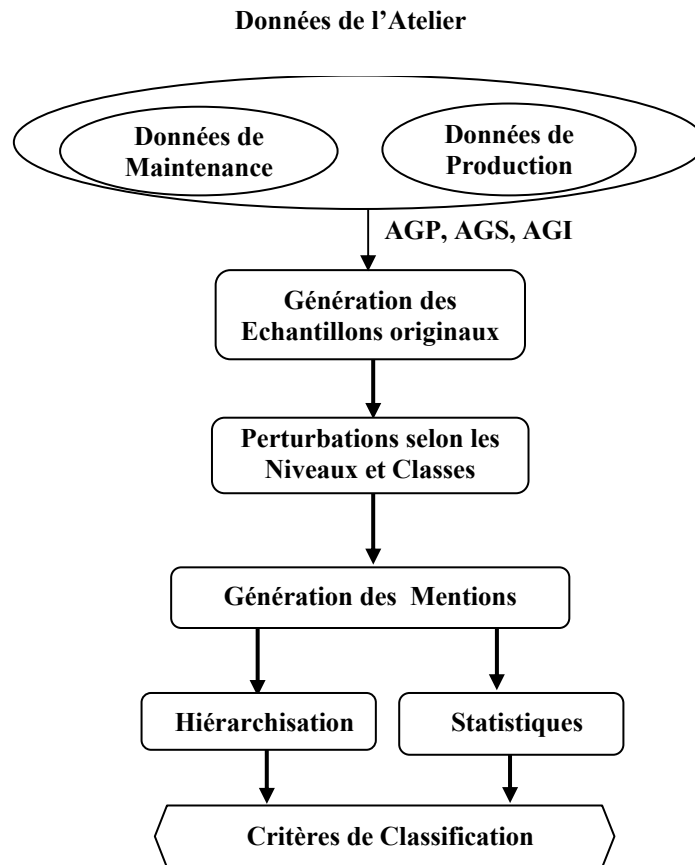


Figure 4.14: Schéma de Classification

## 5. Résultats et Synthèse

Les données de production sont des benchmarks de type Taillard [TAI93], Carlier [CAR78] et Reeves [REE95] de différentes tailles. Les données de maintenance sont générées aléatoirement. Pour pallier le manque de benchmarks en maintenance préventive, nous avons utilisé un générateur de tâches aléatoire. Il reçoit en paramètres : le nombre de machines, et les bornes inférieure et supérieure pour chaque paramètre d'une tâche de maintenance ( $T^*$ ,  $T_{\min}$  et  $T_{\max}$ ). Chacun de ces trois paramètres est lui-même borné par deux valeurs minimale et maximale, pour éviter d'avoir des valeurs identiques. Pour effectuer nos tests, nous avons généré une seule tâche de maintenance par machine pour chaque problème traité. De plus, on exécute la gamme complète à chaque occurrence de la tâche, c'est-à-dire que les durées opératoires des tâches de maintenance sont toutes identiques.

La fonction objectif est la minimisation de la date de sortie de la dernière tâche (Makespan) pour la production, et la minimisation de la somme des retards et avances des tâches de maintenance pour la maintenance (§ 3.3.2.3). Les paramètres  $\alpha$  et  $\beta$  sont égaux à 1. Pour

l'aspect recherche de mesure(s) assurant la robustesse, la fonction objectif de production sera selon les cas étudiés :  $C_{\max}$ ,  $\sum Ti$ , Encours, Idle Time.

Les paramètres de génération des populations d'ordonnements par les AG sont les mêmes pour toutes les instances de la génération et pour toutes les stratégies (Séquentielle et Intégrée). Les valeurs des différents paramètres concernant les échantillons sont : la taille de chaque échantillon est égale à 20, avec un nombre de génération égal à 40, en utilisant la sélection par roue de loterie. Nous avons opté pour le croisement *pair ordre* à deux points de croisements. Pour les taux de mutation et de croisement, nous avons choisi respectivement 0.01 et 0.7. Nous avons choisi de ne pas utiliser de stratégie de remplacement, et d'utiliser comme stratégie de renouvellement les N meilleurs. Quant à l'insertion de la maintenance, nous avons opté pour l'heuristique HN\* (3.3).

Pour les paramètres des perturbations, nous avons utilisé cent simulations par expérimentation. Pour les intervalles de fiabilité, nous avons utilisé deux types d'intervalles : un intervalle de fiabilité faible à 5% de  $T^*$  et un intervalle de fiabilité fort à 25% de  $T^*$ . Cette marge entre les deux intervalles va mettre en exergue l'impact de cet intervalle.

Pour pouvoir exploiter le volume très important de cette étude (20 ordonnancements par échantillon, 10 niveaux et 3 classes, 5 mesures de performance et de plus pour chaque instance de ce triplet nous effectuons 100 simulations), nous avons utilisé le classifieur See5.

See5 (<http://www.rulequest.com/>) peut traiter des problèmes de classification avec n'importe quel nombre d'attributs. See5 permet de prédire la classe d'appartenance d'un individu en se basant sur ces valeurs d'attributs. A partir des données d'entrée, See5 construit un classificateur qui s'occupera de la prédiction. Le classificateur construit peut être exprimé sous forme d'un arbre de décision ou d'un ensemble de règles d'apprentissage. See5 se base essentiellement sur deux fichiers de données : le premier contient les noms des attributs ainsi que leurs instances, le deuxième contient les valeurs des attributs pour chaque exemple d'apprentissage. L'utilisation de See5 est facilitée par une interface graphique permettant de lire les fichiers d'entrée et d'afficher les résultats de classification des individus.

Notre choix s'est porté sur le classifieur See5 pour son aptitude à prédire la classe d'appartenance d'un individu. Les attributs que nous avons retenus sont : les mesures, les niveaux de perturbation, les classes de perturbation.

Le classifieur a généré quatre arbres de décision par méthode de génération (une arborescence par mesure retenue). Une mesure (ou attribut) est susceptible d'améliorer la robustesse d'un ordonnancement si elle se trouve dans la racine de chacune des classifications



et que cette dernière possède une petite profondeur, i.e., un attribut est un critère de robustesse s'il agit indépendamment du niveau et de la classe de perturbation.

Nous désignons par ordonnancement acceptable tout ordonnancement dont la déviation de sa performance, après perturbation, reste inférieure à 35%. Sa mention est alors : Très bien, Bien ou Moyen. Il sera rejeté si sa mention après perturbation est Mauvais ou Très mauvais

Dans ce qui suit nous présenterons les résultats par rapport aux trois questions que nous nous sommes posé pour aborder cette étude.

### **5.1. La maintenance contribue-t-elle à la robustesse des ordonnancements conjoints Production/Maintenance?**

Le premier constat que nous pouvons faire, est l'apport de la maintenance à la robustesse des ordonnancements. Si nous prenant n'importe quel atelier (petit ou grand), nous remarquons une nette stabilité des performances des ordonnancements conjoints production/maintenance par rapport à ceux de production uniquement. En moyenne 75% des ordonnancements de production perturbés (niveaux et classes confondus) sont rejetés. De plus parmi les 25% des cas restants, les mentions varient, en général, entre Moy et B. Par contre, dans le cas des méthode de génération d'ordonnancement conjoints Production/Maintenance (AGS et AGI), le niveau de perturbation à partir duquel les performances deviennent très mauvaise est nettement supérieur à celui de production, donc il y a augmentation du pourcentage d'ordonnancements acceptables par rapport à celui de la production, environ 55% des cas pour l'AGS et 64% des cas pour l'AGI sont acceptables.

La figure 4.15 présente la distribution du pourcentage de cas acceptables pour les trois échantillons étudiés : AGP, AGS et AGI.

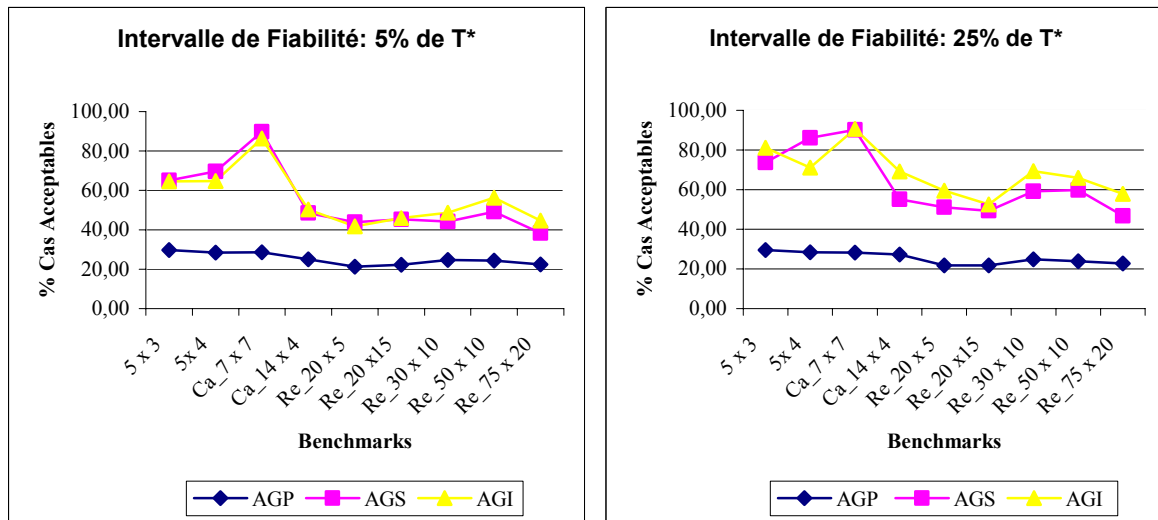


Figure 4.15: Distribution des cas acceptables par échantillons et intervalle de confiance

Les résultats pour le deuxième intervalle de tolérance (25% de  $T^*$ ) sont meilleurs que le premier (5% de  $T^*$ ), du fait de l'élargissement de la période de «grâce» après une maintenance et dans laquelle on n'admet pas de perturbation. D'autre part, pour l'intervalle de tolérance à 5% de  $T^*$ , les distributions des échantillons AGS et AGI sont globalement similaires. Pour le second intervalle, l'échantillon AGI donne les meilleurs résultats pour les petits benchmarks, et l'échantillon AGS pour les plus grands. Enfin, il est important de noter que les résultats pour les petits benchmarks (AGS et AGI) sont meilleurs que ceux des grands benchmarks. Ceci en raison de la forte densité de ces derniers en machines ou en tâches, et de ce fait des perturbations, même dans les niveaux et classes les plus faibles, détériorent, de manière sensible, les performances des ordonnancements originaux. La moyenne des cas acceptables pour les petits benchmarks est de 65%, contre 55% pour les plus grands.

D'autre part, nous avons constaté que la méthode de génération de l'ordonnement conjoint (AGS ou AGI) influait aussi sur la robustesse de ces derniers. En effet, pour les ordonnancements conjoints possédant des intervalles de fiabilité faibles (5% de  $T^*$ ), l'échantillon AGS est meilleur. Par contre, pour ceux avec un intervalle de fiabilité fort (25% de  $T^*$ ) l'échantillon AGI est le mieux adaptée.

En conclusion de cette première étape, nous dirons que, compte tenu des tests que nous avons effectués, les méthodes de génération d'ordonnements conjoints production/maintenance (AGS et AGI), génèrent plus de cas acceptables qu'avec la méthode de génération de production uniquement (AGP) et ceux-ci quelque soit la taille de l'atelier. L'insertion des tâches de maintenance dans les ordonnancements de production se fait certes au dépend de la

performance de ces derniers (en restant bien entendu dans des limites très acceptable) mais elle apporte un gain certain, en terme de robustesse. En effet l'apport de la maintenance se situe au niveau de l'intervalle de fiabilité qu'elle assure à la machine maintenue et qui lui permet d'absorber certaines perturbations. D'autre part, nous avons constaté que dans le cas d'utilisation d'intervalles de fiabilité fort, les ordonnancements générés par l'échantillon AGI sont plus robustes que ceux générés par l'échantillon AGS, et que c'est l'inverse dans le cas d'intervalles de fiabilité faible. Enfin, il est clair que la maintenance n'a aucun effet sur des perturbations liées au personnel, à la matière première ou encore aux commandes. La maintenance est efficace dans le cas de perturbations liées soit aux équipements soit aux travaux.

Au vu de cette conclusion nous nous restreindrons à l'étude des échantillons d'ordonnements conjoints production/maintenance, pour les deux points suivants.

#### *Remarque*

Les pics que l'on remarque dans la figure 4.15 s'expliquent par le fait qu'à partir des niveaux 3 et 4, les mentions TB et B cèdent la place à la mention Moy qui devient plus fréquente et par conséquent le nombre de cas acceptables augmente. D'un autre côté, pour obtenir le nombre de cas acceptables par niveau, nous faisons le cumul des cas pour les trois classes (Faible, Moyenne et forte).

## **5.2. Existe-t-il un critère dont l'optimisation rend l'ordonnement robuste ?**

Le point important relatif à ce deuxième aspect de notre étude est qu'il n'y a pas émergence d'un critère de robustesse parmi les mesures retenues pour cette étude. En effet la taille des benchmarks ainsi que les niveaux et classes de perturbation sont inhérent à la classification. Les ateliers de grandes tailles (75x20, 50x10, 30x10) engendrent une arborescence totalement dépendante du niveau et de la classe de perturbation. Le niveau se situe à la racine, suivi de la classe, dans quelques cas le critère d'optimisation apparaît dans l'arborescence à la troisième position. Quant aux ateliers de petites tailles (5x4, 5x3), nous avons obtenu quelques classifications dont la racine est le critère d'optimisation suivi, en général, du niveau et de la classe de perturbation. Les critères cités avec le plus grand nombre de mentions acceptables sont la moyenne des temps à vide (Idle Time) et l'encours, car dans la plupart des cas ils ont la mention TB et B, et ceci indépendamment du niveau et de la classe de perturbation.

Les grands ateliers sont sensibles aux perturbations du fait que plus le nombre de tâches est important et plus la marge de manœuvre, en présence de perturbations, est limitée. De plus, plus le niveau de perturbation est élevé et le nombre de tâches perturbées important, plus l'ordonnement devient rapidement caduc. Par contre si l'atelier est de petite taille (peu de machines et peu de tâches) l'encours et la moyenne des temps à vide sont importants parce qu'ils représentent cette marge de manœuvre qui est susceptible d'absorber les perturbations et d'assurer à l'ordonnement une certaine stabilité de sa performance, et donc de le rendre robuste.

Au vu de ces remarques nous représentons dans les figures 4.16 et 4.17 le pourcentage de cas acceptables pour chaque mesure retenue pour l'expérimentation, de plus deux intervalles de fiabilité ont été utilisés (5% de  $T^*$  et 25% de  $T^*$ ). Ce pourcentage représente la moyenne obtenue tout niveau et classe de perturbation confondus. De ce fait, les bons résultats ne reflètent que pas l'émergence d'un critère mais uniquement l'influence des bons résultats obtenus par les premiers niveaux de perturbation associés aux classes faible et moyenne, en fait les cas de figures les plus simples.

La figure suivante présente la distribution du pourcentage des cas acceptables par mesure et intervalle de confiance pour l'échantillon AGS et ceci tous niveaux et classes de perturbation confondus.

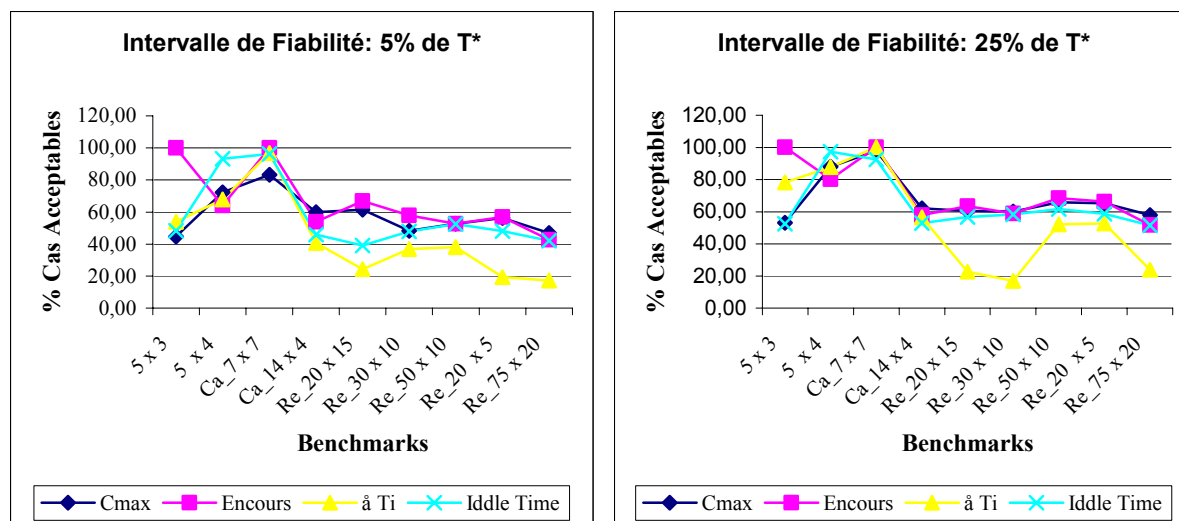


Figure 4.16: Distribution des cas acceptables par critère et IF<sup>8</sup> pour l'AGS. On remarque que l'encours et la moyenne des temps à vide (Idle Time) sont les critères qui obtiennent les meilleurs résultats. Le  $C_{max}$  et la  $\sum Ti$  sont les deux critères les plus

<sup>8</sup> IF : Intervalle de Fiabilité

défavorables à la robustesse. Du fait que les perturbations agissent sur le rallongement des durées opératoires et que pour résoudre les conflits qu'engendrent les perturbations nous utilisons la technique du décalage à droite (*right shifting*), qui consiste à opérer un décalage à droite des tâches jusqu'à résoudre le conflit généré par la perturbation, les critères dont la performance se détériore le plus rapidement par cette technique, certes simple à appliquer, sont le  $C_{max}$  et la  $\sum Ti$ .

D'autre part deux points sont à noter : le premier concerne l'impact de l'intervalle de fiabilité. On remarque à ce niveau une amélioration du pourcentage des cas acceptables sans pour autant modifier la distribution générale. Par contre, on constate une nette détérioration de la  $\sum Ti$  dans le cas d'un intervalle de fiabilité de 25% de  $T^*$ .

Le second point concerne la taille des benchmarks où la encore on constate que pour les benchmarks de petite taille, l'encours et la moyenne des temps à vide (Idle Time) sont les plus favorables à la robustesse. Pour les benchmarks de grande taille, la distribution des quatre critères est globalement semblable est défavorable à l'aspect robustesse.

La figure 4.17 présente la distribution du pourcentage des cas acceptables par mesure et intervalle de confiance pour l'échantillon AGI et ceci tous niveaux et classes de perturbation confondus.

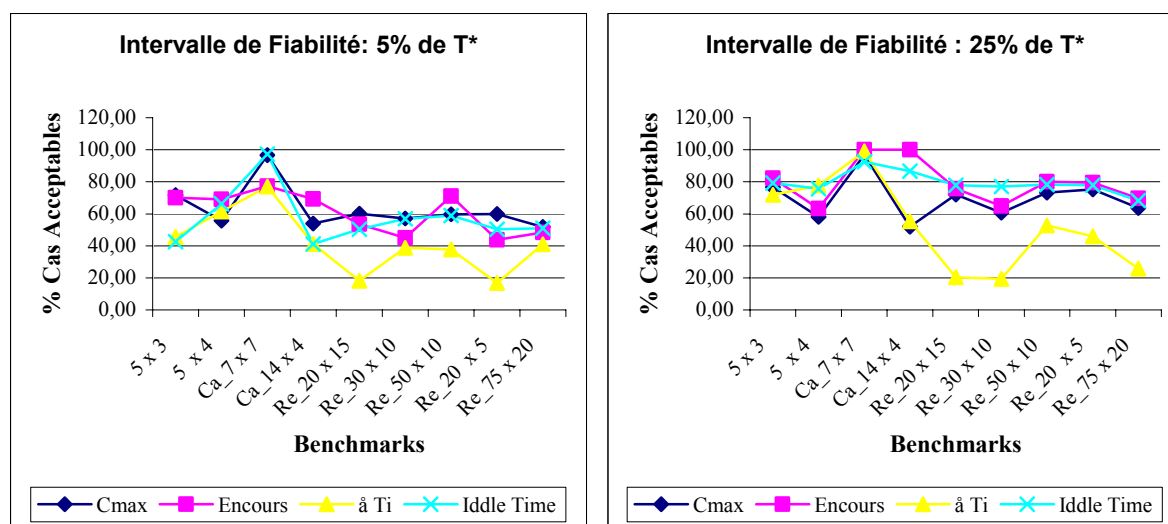


Figure 4.17: Distribution des cas acceptables par critère et IF pour l'AGI

Là encore le constat reste le même pour l'échantillon AGI. Même si l'encours et la moyenne des temps à vide (Idle Time) ont obtenu les meilleurs résultats, ils restent totalement dépendants des niveaux et classes de perturbations. D'autre part la taille des benchmarks joue un rôle important aussi. C'est ce qui explique des pourcentages proches de 100% pour les

petits benchmarks. Pour les premiers niveaux de perturbation et les classes faible ou moyenne, il y a émergence de critère de robustesse. Mais au-delà, cette affirmation devient caduque.

En effet il y a deux tendances dans la figure 4.17, une tendance pour les petits benchmarks dont les résultats sont meilleurs que pour les benchmarks de grande taille. D'autre part, pour la deuxième tendance, on constate une nette détérioration de certains critères ( $C_{\max}$  et  $\sum Ti$ ). Enfin, on remarque que dans l'intervalle de fiabilité à 25% de  $T^*$ , les résultats sont meilleurs sans pour autant avoir une influence quelconque sur la tendance générale des distributions.

En conclusion de cette deuxième partie de l'étude, nous ne pouvons confirmer l'émergence d'un critère, parmi ceux retenus pour l'étude, qui assure la robustesse des ordonnancements conjoint production/maintenance. Et de ce fait, il n'y aura pas de phase validation pour ce processus de classification.

Les critères restent totalement dépendants des niveaux et classes de perturbations d'une part et de la taille des benchmarks d'autre part. Nous ne pouvons que noter que l'encours et la moyenne des temps à vide ont obten les meilleures mentions pour les deux échantillons étudiés. Et que globalement les résultats de l'échantillon AGI sont meilleurs que l'échantillon AGS.

Pour ce qui est de l'intervalle de fiabilité, nous avons remarqué que pour l'intervalle à 25% de  $T^*$ , les résultats sont meilleurs que l'intervalle à 5% de  $T^*$  pour l'encours et la moyenne des temps à vide (Idle Time). Par contre il y a une nette détérioration du  $C_{\max}$  et  $\sum Ti$ .

### 5.3. Existe-t-il des méthodes d'ordonnements robustes ?

La dernière partie de cette étude relative aux méthodes robustes n'a pas été concluante. En effet, il n'y a pas eu d'émergence de méthodes robustes parmi les méthodes étudiées : AG, Tabou, NEH, CDS, RA, Gupta et Palmer.

Les arborescences obtenues après classification sont fonction du protocole de perturbation (niveaux et classes de perturbation) et de la taille des problèmes. Ainsi la robustesse est inhérente aux benchmarks de petite taille pour les cas les plus favorable du protocole (premiers niveaux de perturbation associés aux classes faible et moyenne).

La figure 4.18 présente la distribution des cas acceptables par méthode d'ordonnement et par intervalle de fiabilité pour l'échantillon AGS.

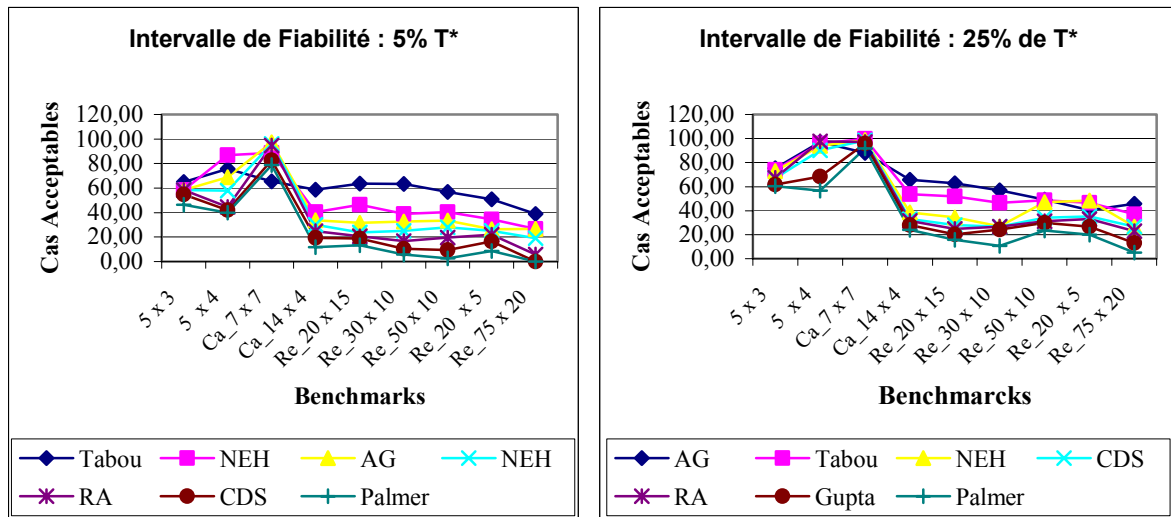


Figure 4.18: Distribution des cas acceptables par méthode et IF pour l'AGS

La tendance générale va vers une implication totale des classes et des niveaux de perturbation d'une part, et de la taille des problèmes d'autre part, dans la distribution générale des courbes. On remarque là aussi que globalement, pour les benchmarks de petite taille les résultats sont meilleurs que ceux des benchmarks de grande taille. Par contre au-delà du niveau 4, la tendance est à la détérioration exponentielle de tous les résultats, toutes méthodes confondues. D'autre part les méthodes des approches évolutive et itérative (AG et Tabou) ont donné de meilleurs résultats que ceux des méthodes de l'approche constructive (NEH, Palmer, CDS, RA, Gupta). Néanmoins l'allure générale de toutes les courbes est une exponentielle inversée. D'autre part, les résultats pour l'intervalle de fiabilité fort (25% de T\*) sont meilleurs que ceux de l'intervalle de fiabilité faible, ceci en raison de l'élargissement de l'intervalle de fiabilité (ou encore de non perturbation) et qui permet d'inhiber toutes perturbations si elles occurrent dans cet intervalle.

La figure 4.19 présente la distribution des cas acceptables par méthode d'ordonnancement et par intervalle de fiabilité pour l'échantillon AGI.

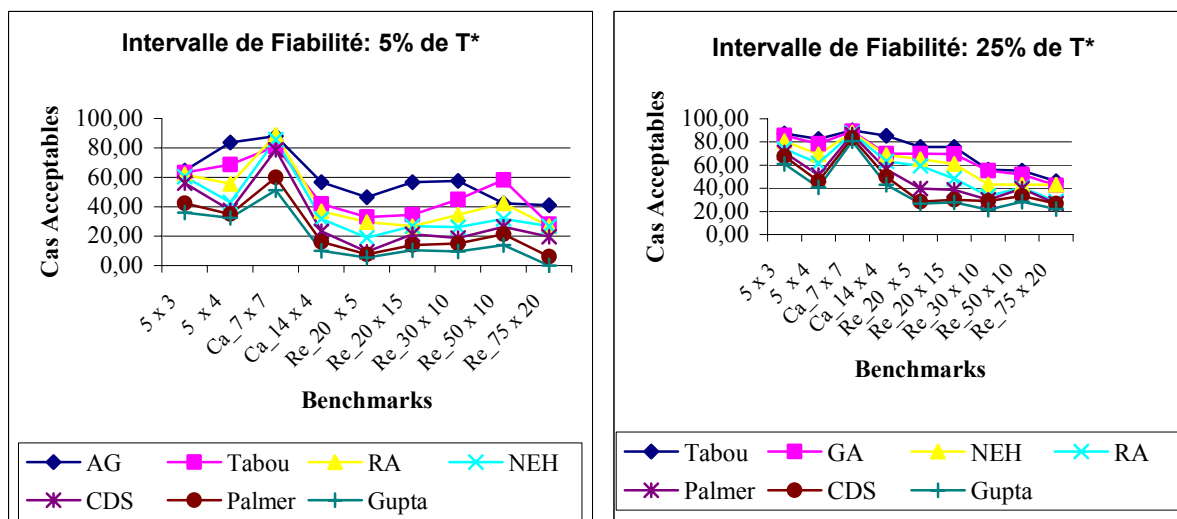


Figure 4.19: Distribution des cas acceptables par méthode et IF pour l'AGI.

Là encore le constat reste le même pour l'échantillon AGI. La robustesse des méthodes est liée au protocole de perturbation, donc niveau et classe de perturbations, et à la taille des problèmes étudiés. Et donc la tendance générale reste une tendance vers la détérioration des résultats qui s'intensifie avec l'intensification des niveaux et classes de perturbation.

Même si les résultats de l'intervalle de fiabilité fort sont meilleurs que ceux de l'intervalle de fiabilité faible, la tendance générale reste la même.

En conclusion de cette dernière partie, nous ne pouvons confirmer l'émergence d'une méthode robuste, parmi celles retenues pour l'étude. La robustesse des méthodes reste liée au protocole de perturbation d'une part et à la taille des problèmes d'autre part. Nous ne pouvons que noter que la tendance générale est la même pour les deux échantillons (AGS et AGI) et que même pour les petits benchmarks nous n'avons pu conclure à la robustesse d'une méthode.

## 6. Conclusion

Dans le domaine de la production, la génération d'un ordonnancement conjoint de production et de maintenance, performant dans une situation statique, n'est pas un gage de sécurité dans une situation dynamique. Car un ordonnancement optimal, pour les données d'une situation certaine à posteriori, ne l'est forcément pas dans une situation incertaine en présence de perturbations. Cet inconvénient majeur a motivé les recherches sur la stabilité des performances des ordonnancements sujets aux perturbations ainsi que les techniques d'amélioration de ces performances.



Nous nous sommes intéressés dans ce chapitre à l'étude de la contribution de la maintenance à la robustesse des ordonnancements conjoint de production et de maintenance. Pour cela, nous avons utilisés trois échantillons d'ordonnement différents générés par les algorithmes génétiques, dans des ateliers de type *flowshop* de permutation : un échantillon d'ordonnements de production uniquement (AGP), et deux échantillons d'ordonnements conjoints production/maintenance, le premier généré par la stratégie séquentielle (AGS), et le second par la stratégie intégrée (AGI). D'autre part, nous avons défini un protocole de perturbation, avec des classes et des niveaux de perturbations, pour étudier la variation des performances de ces ordonnancements. Nous avons étudié les méthodes d'ordonnements conjoints du chapitre 3 et utilisé quatre mesures, pour étudier l'influence de ces mesures sur la robustesse des ordonnancements.

Pour asseoir notre étude, nous nous somme posé trois questions : la réponse à la question concernant la contribution de la maintenance à la robustesse des ordonnancements conjoints est positive, car d'après les résultats obtenus nous pouvons affirmer que l'insertion de la maintenance dans les ordonnancements de production améliore la sensibilité de ces derniers face aux perturbations, et donc ils sont plus robustes. Nous avons constaté par ailleurs l'influence de la stratégie de génération (Séquentielle ou Intégrée) sur la robustesse des ordonnancements perturbés, car dans le cas de l'utilisation d'intervalles de fiabilité fort, les ordonnancements générés par la stratégie intégrée sont plus robustes que ceux générés par la stratégie séquentielle, et que c'est l'inverse dans le cas de l'utilisation d'intervalles de fiabilité faible. La réponse à la question relative à l'aspect mesures pouvant améliorer la robustesse des ordonnancements conjoints, la réponse est partiellement négative. Lors des simulations, nous n'avons trouvé aucune mesure qui a amélioré la robustesse des ordonnancements perturbés globalement. Cependant, en faisant la distinction, lors des simulations, entre les résultats obtenus par les benchmarks de petite et de grande taille, la réponse est restée négative pour les benchmarks de grande taille, par contre pour les benchmarks de petite taille, la moyenne des temps à vide et l'encours ont donné de bons résultats. Comme il n'y a pas eu de mesure émergente, la phase de validation n'a pas été appliquée.

Enfin concernant l'aspect robustesse des méthodes, les méthodes étudiées restent totalement dépendantes des niveaux et classes de perturbations d'une part et de la taille des benchmarks d'autre part.



---

# **CONCLUSION GENERALE**

---



---

## *Conclusion Générale*

---

Dans cette thèse, nous nous sommes intéressé à l'aspect maintenance industrielle qui joue aujourd'hui un rôle déterminant dans la conduite de la production industrielle. Elle a acquis dans l'entreprise manufacturière une position clef. Tandis que la main d'œuvre de fabrication a perdu de son importance, les coûts de maintenance des équipements industriels se sont accrus considérablement, ce qui se reflète dans les budgets.

Que ce soit dans le domaine de la gestion de production ou dans celui du pilotage d'un atelier de production, l'ordonnancement est un problème difficile qui recouvre souvent des enjeux économiques de première importance. Cette importance devient cruciale lorsqu'il touche deux fonctions aussi importantes qu'antagonistes au sein de l'entreprise que le sont la production et la maintenance.

Notre contribution touche deux aspects. D'une part l'adaptation des heuristiques pour l'ordonnancement de production au problème de l'ordonnancement conjoint de la production et de la maintenance préventive systématique, dans un atelier de production de type *flowshop* de permutation. Le but est d'optimiser une « *fonction objectif compromis* » entre les critères d'optimisation de la production et ceux de la maintenance. Ce problème étant NP-complet, pour le résoudre, nous nous sommes intéressé aux heuristiques des approches constructive, itérative et évolutive, qui ont prouvé leurs aptitudes de résolution pour un nombre important de problèmes d'optimisation. Nous avons abordé ce problème selon les deux stratégies d'ordonnancement conjoint les plus intéressantes : la stratégie intégrée et la stratégie séquentielle.

Dans le cas de la stratégie séquentielle nous avons développé deux types d'heuristiques pour l'insertion des tâches de maintenance : des heuristiques pour l'insertion sur une machine, puis sur plusieurs machine. Nous avons utilisé deux heuristiques pour l'insertion sur une machine : l'heuristique HB développée par Kaabi [KAA04] dans sa thèse et nous avons proposé l'heuristique de recherche en profondeur (HRP). Par ailleurs nous avons proposé trois heuristiques pour l'insertion sur plusieurs machines : l'heuristique HN qui est une adaptation de l'heuristique HB pour le cas de plusieurs machines. L'heuristique HA qui consiste à insérer les tâches de maintenance de la première machine jusqu'à la dernière, et l'heuristique HD qui consiste à insérer les tâches de maintenance de la dernière machine à la première.

Pour la stratégie intégrée, notre contribution à ce niveau est de deux types : une première contribution pour les heuristiques dédiées au *flowshop*, qui consiste à définir un ordre de précédente entre toutes les tâches, quelles soient de production ou de maintenance. La seconde concerne la recherche Tabou et les AG, à travers la définition d'un paramétrage adéquat pour la stratégie intégrée. Nous avons proposé, dans le cas de la recherche Tabou, de nouvelles stratégies pour la génération d'une solution initiale conjointe, ainsi que pour l'exploration du voisinage. Pour les AG, comme nous manipulons une séquence conjointe de tâches de production et de maintenance, le croisement et la mutation peuvent se faire aussi bien sur la production que sur la maintenance. Nous avons, donc, proposé de nouveaux opérateurs de croisement et de mutation sur la maintenance. Les opérateurs de croisement et de mutation sur la production restent les opérateurs classiques de la méthode.

Pour cette première partie de notre travail, nous avons conclu, sur la base des tests que nous avons effectués, que comme pour l'ordonnancement de production, les heuristiques de l'approche évolutive semblent plus performantes que celles de l'approche itérative et l'approche constructive. De plus, la stratégie intégrée optimise mieux la solution que la stratégie séquentielle. Ceci s'explique par le fait que pour la stratégie séquentielle deux étapes sont nécessaires pour résoudre le problème : une première étape pour obtenir un ordonnancement de production puis une seconde lors de l'insertion de la maintenance. Alors que pour la stratégie intégrée un seul passage est suffisant pour obtenir un ordonnancement conjoint production/maintenance. C'est ce qui explique entre autre qu'un bon ordonnancement de production, ne le reste pas systématiquement après insertion de la maintenance, dans le cas de la résolution par la stratégie séquentielle.

Une plate forme informatique a été développée pour les travaux de cette partie de la thèse. Elle regroupe toutes les méthodes présentées dans le troisième chapitre, par approche de résolution pour l'ordonnancement conjoint production/maintenance. Pour la production, une bibliothèque de benchmarks de Taillard [TAI93] est disponible. Pour ce qui est de la maintenance, un générateur de tâches de maintenance a été développé pour la phase insertion de la maintenance ou son intégration directe, pour la génération des ordonnancements conjoints. Nous pensons que de plus amples tests ainsi qu'un générateur de tâches de maintenance non pas aléatoire mais en adéquation avec les benchmarks de production confirmerait ce premier travail dans le domaine.

Le cas des problèmes d'ordonnancements statiques reste loin de la réalité industrielle. En effet, l'ordonnement précis des tâches de maintenance et de production déterminé dans ce cadre est inévitablement perturbé par des aléas de production. Qu'en est-il lorsqu'une politique de maintenance préventive est appliquée. La deuxième contribution concerne donc l'étude de la contribution de la maintenance à la robustesse des ordonnancements conjoints production/maintenance, nous sommes arrivés à deux résultats essentiels. Le premier concerne l'effet de la maintenance sur la robustesse des ordonnancements. Nous avons prouvé que l'utilisation de la maintenance dans la génération des ordonnancements de production permet d'améliorer les performances des ordonnancements en présence de perturbations. La deuxième conclusion concerne l'effet du choix de la méthode de génération sur la robustesse des ordonnancements. Nous sommes arrivés au résultat suivant : pour les ordonnancements possédants des intervalles de fiabilité faibles (5% de  $T^*$ ), la génération des ordonnancement conjoints production/maintenance semble meilleure, par contre pour ceux avec un intervalle de fiabilité fort (25% de  $T^*$ ), la génération par la stratégie intégrée est la mieux adaptées.

Lors des simulations, nous n'avons trouvons aucune mesure qui améliore la robustesse des ordonnancements perturbés globalement. Cependant, en faisant la distinction, lors des simulations, entre les résultats obtenus par les benchmarks de petite et de grande taille, pour les benchmarks de petite taille, la moyenne des temps à vide et l'encours ont donné de bons résultats. Enfin pour l'aspect méthodes d'ordonnancements conjoints robustes, les méthodes étudiées restent totalement dépendantes des niveaux et classes de perturbations d'une part et de la taille des benchmarks d'autre part.

Pour ce qui est des perspectives possibles pour ce travail, elles sont à envisager selon les deux axes développés dans cette thèse. Pour ce qui est de l'axe ordonnancement conjoint, plusieurs améliorations peuvent être apportées, que ce soit au niveau des méthodes implémentées, qu'au niveau des contraintes considérées :

- La première concerne l'extension de la plate forme existante à d'autres méthodes telles que les colonies de fourmis et la recherche dispersée, dans le but d'optimiser au mieux les performances des ordonnancements conjoints. Le développement d'un générateur de maintenance plus en adéquation avec les benchmarks de production, est un préalable à cette étape ;
- Comme les méthodes développées manipulent beaucoup de paramètres, une étude peut être menée pour déterminer le paramétrage le plus adéquat, compte tenu du problème à résoudre ;

→ Enfin une optimisation multicritère peut être envisagée pour l'ordonnancement conjoint production/maintenance compte tenu du fait que la fonction objectif que nous avons utilisé prend en compte les contraintes de production et de maintenance. Lors de notre étude, nous avons linéarisé cette fonction.

Pour le second axe de cette thèse, les perspectives sont nombreuses et diverses, compte tenu de la complexité, d'une part à définir de manière explicite la robustesse des ordonnancements. Et d'autre part, à mettre en place des protocoles clairs et précis pour pouvoir qualifier un ordonnancement de robuste. La richesse et la diversité bibliographique dans ce domaine dénote de l'intérêt que suscite actuellement cet aspect dans l'ordonnancement de production. Il est indéniable que la robustesse est la solution aux problèmes inhérents à l'instabilité des ordonnancements dans leurs phase exécution avec toutes les répercussions économique et financière que cela peut engendrer.

Le travail entrepris dans cette thèse avait pour objectif, essentiellement, d'ouvrir de nouvelles perspectives de coopération entre les deux fonctions de base de toute entreprise manufacturière dans le domaine de l'ordonnancement : **la production et la maintenance**. Les deux stratégies d'ordonnancement conjoint que nous proposons sont, vu le contexte industriel actuel et futur, temporellement successifs. En effet, il est très difficile dans le contexte actuel des entreprises industrielles, sauf refonte complète de l'organisation qui verrait les services production et maintenance fusionner en une seule entité, d'opter pour une politique d'ordonnancement intégré. La stratégie séquentielle est actuellement, la plus adaptée (la production représentera une contrainte pour la planification de la maintenance). Par contre, l'entreprise du futur proche est une entreprise dont l'organisation verra les services production et maintenance fusionner en une seule entité vu les contraintes de plus en plus complexes auxquelles elle doit faire face. Dans cette optique, la stratégie intégrée sera la plus adéquate. Cela se fera en profitant des périodes d'inactivité des machines pour planifier le maximum de tâches de maintenance et minimiser ainsi les arrêts de production pour maintenance. Dans les cas de figures, nous ne pourrons avoir la prétention de résoudre tous les conflits mais seulement de les ramener à un seuil acceptable qui perturberait le moins possible aussi bien la production que la maintenance.



---

**REFERENCES**  
**BIBLIOGRAPHIQUES**

---



---

## *Références Bibliographiques*

---

- [AFNOR] AFNOR, Recueil de normes françaises X 06, X 50, X 60, AFNOR.
- [ABD95] Abdelnour G., Duddek R.A., & Smith M.L., "Effect of maintenance policies on just-in-time production system", *International Journal of Production Research*, VOL. 33, pp. 565-585, 1995
- [AGG01] Aggoune R., "Minimising the makespan for the flowshop scheduling problem with availability constraints", ORP'2001, Paris (France), September 26-29, 2001.
- [AGG02] Aggoune R., *Ordonnancement d'ateliers sous contraintes de disponibilité des machines*. Thèse de doctorat soutenue à l'Université de Metz, France, 2002.
- [AHM98] Ahmad A. & Kothari D.P., "A review of recent advances in generator maintenance scheduling", *Electric machines and power systems*, VOL. 26(4), pp. 373-387, 1998.
- [AKT99] Akturk M.S. & Gorgulu E., "Theory and methodology, match up scheduling under a machine breakdown", *European Journal of Operational Research*, VOL. 112, pp. 81-97, 1999.
- [ALC02] Alcaide D., Rodriguez-Gonzalez A. & Sicilia J., "An approach to solve the minimum expected makespan Flowshop problem subject to breakdowns" *European Journal of Operational Research*, VOL 140, pp. 384-398, 2002.
- [ALO02] Aloulou M.A., *Structure flexible d'ordonnancements à performances contrôlées pour le pilotage d'atelier en présence de perturbations*, Thèse de Doctorat de l'Institut National Polytechnique de Lorraine, Nancy - France, 2002.
- [AMO99] Amodeo L., *Contribution à la simplification et à la commande des réseaux de Pétri stochastiques. Application aux systèmes de production*, Thèse de doctorat en Productique soutenue à l' INPG, Grenoble (France), 1999.
- [ANI99] Anily S., Glass C.A. & Hassin R., "Scheduling maintenance services to three machines", *Annals of Operations Research*, VOL. 86, pp.375-391, 1999.
- [ART97] Artigues C., *Ordonnancements en temps réel d'ateliers avec temps de préparation des ressources*, Thèse de doctorat soutenue à l'université Paul Sabatier de Toulouse (France), 1997.
- [ART02] Artigues C., Briand C., Portmann M.C. & Roubellat F., *Pilotage des systèmes de production, chapitre Pilotage d'atelier basé sur un ordonnancement flexible*, Edition Hermès, Kieffer J.P. & Pujo P., 2002.
- [ASH96] Ashayeri J., Teelen A. & Selen W., "A production and maintenance planning model for the process industry", *International Journal of Production Research*, VOL. 34(12), p.3311-3326, 1996.
- [BAI94] Baillet P., *Contribution à l'amélioration de la réactivité dans les systèmes de production notamment par la mise en oeuvre de concepts de décentralisation des fonctions de décision*, Thèse de doctorat soutenue à l'université d'Aix Marseille III, France, 1994.
- [BAK74] Baker K.R., *Introduction to sequencing and scheduling*, John Willy & Sons. New York, 1974.

- [BAL02] Balasubramanian J. & Grossmann I.E., "A novel branch and bound algorithm for scheduling flowshop plants with uncertain processing time", *Computers and Chemical Engineering*, VOL. 26, pp. 41-57, 2002.
- [BAR60] Barlow R. & Hunter L., "Optimal preventive maintenance policies", *Operations Research*, VOL. 8, pp.90-100, 1960.
- [BAV02] Bavier G., Les techniques d'ordonnancement, disponible sur <http://www.peda.ac-martinique.fr/ecogest/ressources/cours/ordo/Techordo.pdf>
- [BEN01a] Benbouzid F., Nader F., Varnier C. & Zerhouni N., "Un SIAD pour la génération du planning de maintenance préventive", 3<sup>e</sup> Conférence Francophone de Modélisation et de SIMulation MOSIM'01 Troyes (France), VOL. 2, pp. 939-946, 2001.
- [BEN02] Benbouzid F., Varnier C. & Zerhouni N., "Un SIAD pour l'élaboration du planning de maintenance préventive", *Revue internationale d'Ingénierie des Systèmes de Production, Mécanique (ISPM)*, VOL. 6, pp. II.42-II.53, Décembre 2002.
- [BEN03a] Benbouzid F., Bessadi Y., Guebli S., Varnier C. & Zerhouni N., "Résolution du problème de l'ordonnancement conjoint maintenance/production par la stratégie séquentielle", 4<sup>e</sup> Conférence Francophone de Modélisation et de SIMulation MOSIM'03 Toulouse (France), 2003.
- [BEN03b] Benbouzid F., Varnier C. & Zerhouni N., "Resolution of joint maintenance/production scheduling by sequential and integrated strategies", 7th International Work Conference on Artificial and Natural Neural Networks IWANN2003, 3-6 June, Spain, Proceedings LNCS 2687, pp. 782-789, 2003.
- [BEN03c] Benbouzid F., Varnier C. & Zerhouni N., "Une heuristique de recherche par voisinage pour l'ordonnancement conjoint maintenance/production", 3<sup>ième</sup> Conférence Internationale en Conception et Productique Intégrées CPI'2003, 22-24 Octobre 2003, Maroc.
- [BEN04a] Benbouzid F. & Varnier C., "Robust joint Production and Maintenance scheduling with processing time uncertainty", 9th International Workshop on project management and scheduling PMS2004, April 26-28, 2004, Nancy, France.
- [BEN04b] Benbouzid F., Varnier C. & Zerhouni N., "Resolution of joint maintenance/production scheduling with GA", 9th International Workshop on project management and scheduling PMS2004, April 26-28, 2004, Nancy, France.
- [BEN05a] Benbouzid F., Varnier C. & Zerhouni N., "Etude de l'impact de la maintenance sur la robustesse des ordonnancements conjoints production/maintenance", 6e Congrès international de génie industriel GI2005 sur CD Rom, 10 pages, 7-10 juin 2005 – Besançon (France)
- [BEN05b] Benbouzid F., Varnier C. & Zerhouni N., "Résolution du problème de l'ordonnancement conjoint production/maintenance par les stratégies séquentielle, intégrée et hybride", soumis à la revue JESA – Journal Européen des Systèmes Automatisés -.
- [BEN98] Ben-Daya M. & Makhdoum M., "Integrated production and quality model under various maintenance policies", *Journal of the Operational Research Society*, VOL. 49(8), pp.840-853, 1998.
- [BEN01] M.Bennour, C.Bloch & N.Zerhouni, "Modélisation intégrée des activités de maintenance et de production", 3<sup>e</sup> Conférence Francophone de Modélisation et de SIMulation MOSIM'01 Troyes (France), VOL. 2 pp.805-810, 2001.

- [BEM02] Bembla M. Ordonnancement conjoint production et maintenance : Critère et heuristique de résolution. Mémoire de DEA, U.F.R des Sciences et Techniques de l'Université de Franche Comté, 2002.
- [BIL05] Billaut J.C., Moukrim A. & Sanlaville E., Flexibilité et robustesse en ordonnancement, Chapitre 1, 6 et 7, Hermès Sciences Publications, 2005.
- [BLA96] Blazewicz J., Ecker K., Pesch E. Schmidt G. & Weglarz J., Scheduling computer and manufacturing processes, Springer-Verlag, 1996.
- [BOI90] Boitel D. & Hazard C, Guide de la maintenance, Edition Nathan, 1990.
- [BOU88] Boulenger A., Vers le zéro panne avec la maintenance conditionnelle, Guides de l'utilisateur, AFNOR, Paris, 1988.
- [BOU95] Boukas E.K., Zhang Q. & Yin G., "Robust production and maintenance planning in stochastic manufacturing system", IEEE Transactions on Automatic Control, VOL. 40(6), pp.1098-1102, 1995.
- [BOY86] Boyer L., Poirée M. & Salin E., Précis d'organisation et de gestion de la production, Les Editions d'Organisation, 1986.
- [BRA96] Brandolese M., Fransi M. & Pozzetti A., "Production and maintenance integrated planning", International Journal of Production Research, VOL. 34(7), pp.2059-2075, 1996.
- [BRA00] Branke J. & Mattfeld D.C., "Anticipation in dynamic optimization: The scheduling case", Proceeding of sixth Parallel Problem Solving from Nature, LNCS VOL. 1917, pp. 253-262, Springer, 2000.
- [BRA98] Branke J., "Creating robust solutions by means of evolutionary algorithms", Proceeding of fifth Parallel Problem Solving from Nature, LNCS VOL. 1498, pp. 119-128, Springer, 1998.
- [BRU98] Brucker P., Scheduling algorithms, Spriger-Verlag, 1998.
- [BUZ90] Buzacott J.A., & Shanthikumar J.G, Stochastic models of manufacturing systems, Prentice-Hall, Englewood Cliff, NJ, 1990
- [CAM70] Campbell H.G, Dudek R.R & Smith M.L, "A heuristic algorithm for n job m machine sequencing problem", Management Science, VOL. 16, pp. 630-670, 1970.
- [CAR78] Carlier J, "Ordonnements a contraintes disjonctives", R.A.I.R.O. Recherche operationelle/Operations Research, VOL.12, pp.333-351, 1978.
- [CAR88] Carlier J. & Chretienne P., Problèmes d'ordonnancement : Modélisation, Complexité, Algorithmes. Etude et recherche en informatique (ERI), Edition MASSON, 1988.
- [CAR91] De Carvalho S., Modèles stochastiques appliqués à l'optimisation de la performance et de la sûreté de fonctionnement des systèmes de production, Thèse de Doctorat de l'université Paul Sabatier. Toulouse (France), novembre 1991.
- [CAU94] CAUX C., Pierreval H. & Portmann M.C., "Les algorithmes génétiques et leur application aux problèmes d'ordonnancement", Congrès d'ordonnancement, Toulouse (France), Juin 1994.
- [CHA96] Charon I., Germa A. & Hurdy O., Méthodes d'optimisation combinatoires, Edition MASSON, 1996.
- [CHA97] Chareonsuk C., Nagarur N. & Tabucanon M.T., "A multicriteria approach to the selection of preventive maintenance intervals", International Journal of Production Economics, VOL. 49, pp. 55-64, 1997.
- [CHE02] Cherfi Z., La qualité – Démarche, méthodes et outils, Edition Hermès Sciences, Chap. 4 : l'ingénierie robuste pp. 115-154, 2002

- [CHE97] Chen W. & Muraki M., "A fussy evaluation of schedule robustness under processing time variation in batch plant", *Journal of Chemical Engineering of Japan*, pp. 260-267, 1997.
- [CHE97a] Chen W., Fuchino T. & Muraki M., "A modification module for robust schedule generation system of batch plants", *Sckiiyu Gakkaiçshi*, VOL. 40(5), pp. 378-389, 1997.
- [CHE97] Cheung K.L., & Hausmann W.H, "Joint optimization of preventive maintenance and safety stocks in an unreliable production environment", *Naval Research Logistics*, VOL. 44, pp. 257-272, 1997.
- [CHU95] Chu C., *Ordonnancement de la production: approches théoriques nouvelles*. Thèse de doctorat soutenue à l'Université de Metz, France, 1995.
- [CHU96] Chu. C & Proth.J.M *L'ordonnancement et ses applications*. Sciences de l'ingénieur. Edition MASSON, 1996.
- [CHU97] Chung-Yee L. "Minimising the makespan in two machines flowshop scheduling with availability constraints", *Operation research letters* VOL. 20, pp.129-139, 1997.
- [CHU99] Chung-Yee L., "Theory and methodology two machines flowshop scheduling with availability constraints", *Operation research letters* VOL. 41, pp.420-429, 1999.
- [CHU00] Chung-Yee L. & Zhi-Long C., "Scheduling jobs and maintenance activities on parallel machines", *Naval Research Logistics* VOL. 47, pp.145-165, 2000.
- [COS95] Costa D., *Méthodes de résolution constructives, séquentielle et évolutive pour des problèmes d'affectation sous contraintes*, Thèse de doctorat, Département de mathématique, Ecole Polytechnique Fédérale de Lausanne, Suisse, 1995.
- [COU00] Coudert T., *Apport des systèmes multi-agents pour la négociation en ordonnancement : application aux fonctions production et maintenance*. Thèse de doctorat soutenue à l'ENI de Tarbes (France), 2000.
- [COU01] Coudert T., Grabot B. & Archimède B., "Logique floue et système multi-agents pour un ordonnancement coopératif production/maintenance", 3<sup>e</sup> Conférence Francophone de Modélisation et de SIMulation MOSIM'01 Troyes (France), VOL. 1, pp.485-490, 2001.
- [COW02] Cowling P. I., & Johansson M., "Using real time information for effective dynamic scheduling", *European Journal of Operational Research*, VOL. 139(2), pp. 230-244, 2002.
- [DAL92] Dallery Y., & Gershwin S.B., "Manufacturing flow line systems: a review of models and analytical results" *Queueing Systems*, VOL. 12, pp. 3-94, 1992.
- [DAN77] Dannenbring D.G., "An evaluation of Flowshop sequencing heuristics", *Management Science*, VOL. 23, pp. 1174-1182, 1977.
- [DAV91] Davis L., *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, New York, 1991.
- [DAV00] Davenport A.J. & Beck J.C., "A survey of techniques for scheduling with uncertainty", Papier non publié disponible sur <http://eil.utoronto.ca/profiles/chris/chris.papers.html>, 2000.
- [DAV01] Davenport A.J., Gefflot C.&Beck J.C., "Slack-based techniques for building robust Schedules", *European Conference on Planning - ECP01*, Toledo, 2001.
- [DED95] Dedopoulos I.T. & Shah N., "Optimal short-term scheduling of maintenance and production for multipurpose plants", *Industrial & engineering chemistry research*, VOL. 34(1), pp. 192-201, 1995.

- [DEM77] Demmou R., Etude de familles remarquables d'ordonnements en vue d'une aide à la décision. Thèse de doctorat soutenue à l'Université Paul Sabatier de Toulouse, France, 1977.
- [DEN99] Deniaud S., Contribution à l'élaboration d'un modèle technico-économique de maintenance préventive : Cas de lignes de fabrication industrielles, Thèse de Doctorat soutenue à ENI de Belfort (France), 1999.
- [DEN99a] Deniaud S., Zehouni N. & Morel F., "Planification de la maintenance préventive d'une ligne de fabrication de GEC Alstom", APII-JESA VOL. 33(3), pp.227-250, 1999.
- [DOR92] Dorigo M. Learning and Natural Algorithms (in Italian). PhD thesis, DEI, Politecnico di Milano, Italy, 1992.
- [DOR98] Dorne R. & Hao J.K., "A new genetic Local Search Algorithm for graph coloring", Lecture Notes in Computer Science, VOL. 1498, pp. 745-754, Springer Verlag 1998.
- [DOU84] Doumeingts G., Méthode GRAI : méthode de conception des systèmes en productique. Thèse d'état soutenue à l'université de Bordeaux I (France), 1984.
- [DUM01] Dumont L. & Morel L., "Modélisation des coûts de maintenance : Application sur l'installation d'expérimentation AIRIX", 3<sup>e</sup> Conférence Francophone de Modélisation et de SIMulation MOSIM'01 Troyes (France), VOL. 3, pp.29-33, 2001.
- [ESP98] Espinouse M.-L., Flowshop et extensions : chevauchement des tâches, indisponibilité des machines et système de transport, Thèse de l'Université Joseph Fourier, Grenoble (France), 1998
- [ESP01] Espinouse M.L., Formanowicz P. & Penz B., "Complexity results and approximation algorithms for the two machines no-wait flowshop with limited machine availability", Journal of the operational research society, VOL. 52, pp. 116-121, 2001.
- [ESQ99] Esquirol P. & Lopez P., L'ordonnement, Edition Economica, 1999.
- [FLE94] Fleurent C., Algorithmes génétiques hybrides pour l'optimisation combinatoire. Thèse du département d'informatique et de recherche opérationnelle de l'université de Montréal, Canada, 1994.
- [FRE82] Frensh S., Sequencing and Scheduling: An introduction to the mathematics of the jobshop. Edition Wiley John, 1982.
- [GAO95] Gao H., Building robust schedules using temporal protection an empirical study of constraint based scheduling under machine failure uncertainty. Master's thesis, Department of Industrial Engineering, University of Toronto, 1995.
- [GAR79] Garey R.M. & Johnson S.M., Computers and Intractability, A guide to the Theory of NP-Completeness, W.H. Freeman and CO., San Francisco, 1979.
- [GHA00] Gharbi A. & Kenne J., "Production and corrective maintenance planning of FMS through simulation optimization", 4<sup>th</sup> international conference on engineering design and automation, Orlando, Florida, USA, 2000.
- [GIA88] Giard N., Gestion de la production. Edition Economica, 1988.
- [GIR00] Girard B., Guide pratique du responsable de maintenance, T1. Edition WEKA, 2000.
- [GIT94] Gits C.W., "Structuring maintenance control systems". International Journal of Operations and Production Management, VOL.14 (17), pp.5-17, 1994.

- [GLO77] Glover F., "Heuristics for integer programming using surrogate constraints", *Decision Sciences*, VOL. 8, pp. 156-166, 1977.
- [GLO86] Glover F., "Future paths for integer programming and links to artificial intelligence", *Computers and Operations Research*, VOL. 13, pp.533-549, 1986.
- [GLO89] Glover F., "Taboo search: Part I", *ORSA Journal of Computing*, VOL. 1, pp.190-206, 1989.
- [GLO90] Glover F., "Taboo search: Part II", *ORSA Journal of Computing*, VOL. 2, 1990.
- [GOL89] Goldberg D.E., *Genetic algorithms in search, Optimisation and Machine Learning*. Addison-Wesley, Mass., 1989.
- [GOP01] Gopalagrishnan M., Mohan S. & He Z., "A taboo search heuristic for preventive maintenance scheduling", *Computers and Engineering*, VOL. 40, pp. 149-160, 2001.
- [GOT93] GOTHIA, "Les problèmes d'ordonnement", *Recherche Opérationnelle/Operations Research (RAIRO-RO)*, VOL. 27(1), pp.77-150, 1993.
- [GOT02] Groupe Flexibilité du Gotha, *Flexibilité et Robustesse en ordonnancement*, Disponible sur <http://www.roadef.org/If/flexordo.ps>, 2002.
- [GRA69] Graham R.L., "Bounds on multiprocessing timing anomalies", *SIAM journal on Applied Mathematics*, pp. 416-429, 1969.
- [GRA79] Graham R.L., Lawler E.L., Lenstra J.K., & Rinooy Kan A.H.G., "Optimisation and approximation in deterministic sequencing and scheduling: a survey". *Annals of Discrete Mathematics*, VOL. 5, pp. 287-326, 1979.
- [GRA99] Graves G. & Lee C.-Y., "Scheduling maintenance and semi-resumable jobs on single machine", *Naval Research Logistics*, VOL. 46, pp.845-862, 1999.
- [GRE87] Grefenstette J.J., "Incorporating problem specific knowledge into genetic algorithm", *Genetic Algorithm and Simulated Annealing*, Davis L. Edition, Morgan Kaufmann Publishers, pp. 42-60, 1987.
- [GUP71] Gupta J.N.D., & Dudek R.A., "Optimality criteria for flowshop schedule", *AIIE Transactions* VOL. 3, pp.199-205, 1971.
- [HAL04] Hall N.G. & Posner M.E., "Sensitivity analysis for scheduling problems", *Journal of scheduling*, VOL. 7(1), pp. 49-83, 2004
- [HAN86] Hansen P., "The steepest ascent mildest descent heuristic for combinatorial programming", In *Congress on Numerical Methods in Combinatorial Optimisation*, 1986. Capri, Italy.
- [HAR95] Haran G. *Méthode P.E.R.T : Gestion et ordonnancement des projets par la méthode du chemin critique*. Edition EYROLLES 1995.
- [HOL75] Holland J. *Adaptation in natural and artifical systems*. (Cambrige, Mass : MIT press, 1975.
- [HUB81] Huber P. J., *Robust Statistics*, John Wiley & Sons, 1981.
- [JEN99] Jensen M.T. & Hansen T.K., "Robust solution to jobshop problems", *Proceeding of the Congress of Evolutionary Computation*, VOL. 2, pp. 1138-1144, 1999.
- [JEN00] Jensen M.T., "Neighbourhood based robustness applied to tardiness and total flowtime jobshops", *Proceeding of sixth Parallel Problem Solving from Nature*, LNCS VOL. 1917, pp. 283-292, Springer, 2000.
- [JEN01] Jensen M.T., "Improving robustness and flexibility of tardiness and total flowtime jobshop using robustness measures", *Applied Soft Computing*, VOL. 1, pp. 35-52, 2001.



- [JEN01a] Jensen M.T., Robust and flexible scheduling with evolutionary computation. Thèse de doctorat soutenue à la faculté des Sciences de l'Université de Aarhus, Danemark, Octobre 2001.
- [JOH54] Johnson S.M., "Optimal Two and Three-Stage Production Schedules with Setup Times included", *Naval Research Logistics Quarterly*, VOL. 1(1), 1954.
- [KAA02] Kaabi J., Varnier C. & Zerhouni N., "Heuristics for scheduling maintenance and production on a single machine", *IEEE Conference on Systems, Man and Cybernetics*. October 6-9, 2002 Hammamet, Tunisia.
- [KAA04] Kaabi J., Contribution à l'ordonnancement des activités de maintenance dans les systèmes de production. Thèse de doctorat soutenue à l'Université de Franche Comté, France, 2004.
- [KAC85] Kackar R. N., "Off-line Quality control, Parameter design and the Taguchi Method, *Journal of Quality Technology*", VOL.17 (4), pp. 176-188, 1985.
- [KAF01] Kaffel H., La maintenance distribuée : Concepts, Evaluation et Mise en œuvre. Thèse PhD, département de génie mécanique, Faculté des sciences et de génie, Université de Laval, Québec, Canada, 2001.
- [KIR83] Kirkpatrick S., Gelatt C.D.Jr & Vecchi M.P., "Optimisation by simulated annealing", *Science* VOL. 220, 1983.
- [KOB97] Kobbacy K.A.H., Fawzi B.B., Percy D.F. & Ascher H.E., "A full history proportional hazards model for preventive maintenance scheduling", *Quality and reliability engineering international*, VOL. 13, pp. 187-198, 1997.
- [KOL93] Kolski C. & Millot P., "Problems in telemaintenance and decision aid criteria for telemaintenance system design", *International Journal of Industrial Ergonomics*, VOL.11 (2), pp. 99-106, 1993.
- [KOU97] Kouvelis P. & Yu G., Robust discrete optimization and its applications. Edition Kluwer Academic, 1997.
- [LAS93] Lash S., Genetic algorithm for weighted tardiness scheduling on parallel machines. Technical report n° 93.01, Department of Industrial Engineering and Management Sciences, North-western University, Evanston, Illinois, USA, 1993.
- [LAU96] Laugier A., Allahwerdi N., Baudin J., Gaffney P., Grimson W., Groth T. & Schilders L., "Remote instrument telemaintenance", *Computer Methods and Programs in Biomedicine*, VOL. 50 (2), pp. 187-194, Elsevier Science Ireland Ltd Shannon Ireland, 1996.
- [LAV96] Lavina Y. & Loubère J. M., Maintenance et Travaux Neufs, Les règles de la sous-traitance. Les Éditions D'organisation, Paris, France, 1996.
- [LAV98] Lavina Y., La maintenance: fiabilité, maintenabilité, sécurité. Edition Masson, 1998.
- [LAW97] Lawrence S.R. & Sewell E.C., "Heuristic, optimal, static and dynamic schedules when processing times are uncertain", *Journal of Operations Management*, VOL. 15, pp. 71-82, 1997.
- [LEE00] Lee C.-Y & Chen Z.-L., "Scheduling jobs and maintenance activities on parallel machines", *Naval Research Logistics*, VOL. 47, pp.145-165, 2000.
- [LEO94] Leon V.J., Wu S.D. & Storer R.H., "Robustness measures and robust scheduling for job shops", *IIE Transactions*, VOL. 26(5), pp. 32-43, 1994.
- [LOP01] Lopez P. & Roubellat F., Ordonnancement de la production. Chapitres 1 & 2. Edition Hermès Science, 2001.
- [LYO92] Lyonnais P., Maintenance mathématique et méthode. Troisième édition, technique et édition Lavoisier, France, 1992.

- [MAL93] Malcolm M., "Plant Maintenance. A New Direction", Process and Control engineering, PACE, VOL. 46 (9), pp. 20-22, September 1993.
- [MCA65] McCALL J.J., "Maintenance policies for stochastically failing equipment: a survey", Management Science, VOL. 11(5), pp.493-524, 1965.
- [MEH99] Mehta S.V. & Uzsoy R., "Predicte scheduling of a single machine subjects to breakdowns", International Journal of Computer Integrated Manufacturing, VOL. 12, pp. 15-28, 1999.
- [MIG95] Mignon D.J., Honkomp S.J.& Reklaitis G.V., "A framework for investigating schedule robustness under uncertainty", Computers and Chemical Engineering, VOL.21, Suppl., pp. S615-S620, 1995.
- [MON96] Monchy F., La fonction maintenance. Edition Masson, 1996.
- [MOU97] Moubray J., Reliability Centred Maintenance. International Press, 1997.
- [MUH88] Mühlenbein H., Gorges-Schleuter M. & Krämer O., "Evolution algorithms in combinatorial optimization", Parallel Computing, VOL. 7, pp. 95-85, 1988
- [NAK87] Nakajima S., La maintenance productive totale (TPM), nouvelle vague de la production industrielle. Afnor Gestion, 1987.
- [NAK89] Nakajima S., La maintenance productive Totale : Mise en œuvre. AFNOR, 1989.
- [NAK86] Nakagawa T., "Periodic and sequential preventive maintenance policies, Journal of Applied Probability, VOL. 23, pp.536-542, 1986.
- [NAW83] Nawaz N., ENSCORE E.Jr. & HAM I., "A heuristic algorithm for the m machine, n job flowshop sequencing problem", Omega, VOL. 11, 1983.
- [NOY99] Noyes D., Pérès F., "Conduite intégrée de production et de maintenance dans les systèmes industriels", Troisième Congrès International de Génie Industriel, Montréal (Canada), 26-28 Mai, 1999.
- [OUA01] Ouali M.-S, Rezg N. & Xie X., "Maintenance préventive et optimisation des flux d'une ligne de fabrication", 3<sup>e</sup> Conférence Francophone de Modélisation et de SIMulation MOSIM'01 Troyes (France), VOL. 2, pp.791-796, 2001.
- [PAL65] Palmer, D.S., "Sequencing Jobs Through A Multi-Stage Process in the Minimum Total Time –A Quick Method of Obtaining A Near Optimum", Operations Research Quarterly, VOL. 6, pp. 101-107, 1965.
- [PAP76] Papadimitriou C.H., The complexity of combinatorial optimization problems, PhD thesis, Princeton, 1976.
- [PAP82] Papadimitriou C.H. & Steiglitz K., Combinatorial Optimization: Algorithms and Complexity. Prentice-Hall, 1982.
- [PAP95] Le Pape C., Artificial Intelligence in Reactive Scheduling, chapter Experiments with a distributed architecture for predictive scheduling and execution monitoring. Chapman et Hall, 1995.
- [PAZ94] Paz N.M. & Leigh W., "Maintenance scheduling: issues, results and research needs", International Journal of Operations & Production Management, VOL. 14(8), pp.47-69, 1994.
- [PEL97] Pellegrin C., Fondements de la décision de maintenance, Edition Economica, 1997.
- [PEN01] Penz B., Rapine C. & Tristram D., "Sensitivity analysis of scheduling algorithms", European Journal of Operational Research, VOL. 134, pp. 606-615, 2001.
- [PER97] Pérès F., Caillaud E. & Noyes D., "La maintenance dans l'évaluation des performances des systèmes de production : une approche par la simulation", Deuxième congrès international Franco-Québécois de Génie Industrielle, Albi, 3-5 septembre 1997.

- [PIE76] Pierskalla W. & Voelker J., "A survey of maintenance models: the control and surveillance of deteriorating systems", *Naval Research Logistics Quarterly*, VOL. 23, pp.353-388, 1976.
- [PIN95] Pinedo M., *Scheduling: Theory, Algorithms, and Systems*. Edition Prentice Hall International, series in industrial and systems engineering, 1995.
- [PIS96] Pistikopoulos E.N., Thomaidis T.V., Melin A. & Ierapetritov M.G., "Flexibility, reliability and maintenance considerations in batch plant design under uncertainty", *Computers and Chemical Engineering*, VOL. 20, Suppl., pp. S1209-1214, 1996.
- [POM01] Pomerol J.C., "Scenario development and practical decision making under uncertainty, *Decision Support Systems*", VOL. 31, pp. 197-204, 2001.
- [PON01] Ponnambalam S.G., Aravindan P. & Chandrasekaran S. "Constructive and improvement flowshop scheduling heuristics: an extensive evaluation", *Production planning and control*, VOL. 12(4), pp. 335-344, 2001.
- [POR01] Portmann M.C. & Vignier A., *Algoritmes génétiques et ordonnancement. Ordonnement de la production*. Edition Hèrmes Sciences, 2001.
- [POR96] Portmann M.C., "Genetic algorithms and scheduling: a state of the art and some propositions", *Workshop on Production Planning and Control*, pp. I-XIV, Mons, 1996.
- [QI99] Qi X., Chen T. & Tu F., "Scheduling the maintenance on single machine", *Journal of the Operational Research Society*, VOL. 50(10), pp.1071-1078, 1999.
- [REE95] Reeves C.R., "A genetic algorithm for flowshop sequencing", *Computer Operational Researchs* VOL. 22, pp. 5-13, 1995.
- [RET90] Retour D., Bouche M. & Plauchu V., "Où va la maintenance industrielle", *Problèmes Économiques*, VOL. 2 (159), pp. 7-13, 1990.
- [RIC96] Richet D. & Gabriel M., *Maintenance basée sur la fiabilité*. Edition Masson, 1996.
- [RIS96] Rishel T.D. & Christy D.P., "Incorporating maintenance activities into production planning; Integration at the master schedule versus material requirement level", *International Journal of Production Research*, VOL. 34(2), pp.421-446, 1996
- [ROB88] Roboam M., *Modèles de référence et intégration des méthodes d'analyse pour la conception des systèmes de production*. Thèse de doctorat soutenue à l'université de Bordeaux (France), 1988.
- [ROY97] Roy B., *Un chaînon manquant en RO-AD : les conclusions robustes*, Cahier du Laboratoire d'Analyse et de Modélisation de Systèmes pour l'Aide à la Décision, N° 144, Université de Paris Dauphine, 1997.
- [ROY04] Roy B., "Robustesse de quoi et vis-à-vis de quoi mais aussi robustesse pourquoi en aide en décision?" *Proceeding of the 56th meeting of the European Working Group "Multiple Criteria Decision Aiding, Coimbra, Portugal, 2002"*.
- [SAF99] Safaai D., Sigeru O., Hiroshi O., Shaharudin K. & Pathiah A.S., "Ship maintenance scheduling by genetic algorithm and constraint based reasoning", *European Journal of Operational Research*, VOL. 112, pp. 489-502, 1999.
- [SAN95] Sanmarti E., Espuna A. & Puigjaner L., "Effects of equipment failure uncertainty in batch production scheduling", *Computer Chemical Engineering*, VOL. 19, Suppl., pp. S565-S570, 1995.

- [SAN96] Sanmarti E., Puigjaner L., Huercio A. & Espuna A., "A combined scheduling/reactive scheduling strategy to minimize the effects of process operations uncertainty in batch plant", *Computers Chemical Engineering*, VOL. 20, pp. 1263-1268, 1996.
- [SAN97] Sanmarti E., Espuna A. & Puigjaner L., "Batch production and preventive maintenance scheduling under equipment failure uncertainty", *Computer Chemical Engineering*, VOL. 21(10), pp.1157-1168, 1996.
- [SAS98] Sassine C., *Intégration des politiques de maintenance dans les systèmes de production manufacturiers*. Thèse de doctorat soutenue à l'INP de Grenoble (France), 1998.
- [SHA99a] Shafaei R. & Brun P., Workshop scheduling practical (inaccurate) data. Part 1: An investigation of robustness of scheduling rules in dynamic jobshop environment using a rolling time horizon approach. *International Journal of Production Research*, VOL. 37(17), pp. 3913-3925, 1999.
- [SHA99b] Shafaei R. & Brun P., "Workshop scheduling practical (inaccurate) data. Part 2: An investigation of robustness of scheduling rules in dynamic and stochastic environment", *International Journal of Production Research*, VOL. 37(18), pp. 4105-4117, 1999.
- [SHE00] Sherwin D., "A review of overall models for maintenance management", *Journal of Quality in Maintenance Engineering*, Vol. 6 (3), pp. 138-164, 2000
- [SWA99] Swanson L., "The impact of new production technologies on the maintenance function: an empirical study", *International Journal of Production Research*, Vol. 37(4), pp. 849-869, 1999.
- [TAI93] Taillard E., "Benchmarks for basic scheduling problems", *European Journal of Operational Research* VOL.64, pp. 278-285, 1993. Disponible sur <http://ina.eivd.ch/Collaborateurs/etd/default.htm>
- [TAI01] Taillard E.D, Gambardella L.M., Gendreau M., & Potvin J.Y. "Adaptive memory programming: a unified view of metaheuristics". *European Journal of Operational Research*, VOL 135(1), pp. 1-16, 2001.
- [TKI99] T'Kindt V., *Etude des problèmes d'ordonnancement multi-critères*. Thèse de doctorat soutenue à l'Université François Rabelais, Tours, France, 1999.
- [TAL95] Talal M.A & Jay Y., "Refinery units maintenance scheduling using integer programming", *Applied Mathematical Modelling*, VOL. 19(9), pp. 543-549, 1995.
- [TAR96] Tarondeau J.C., *La gestion de production*. Paris : Presses Universitaires de France, - Que sais-je ? n° 3115, Septembre 1996.
- [TSA01] Tsai Y.T., Wang K.S. & Teng H.Y., "Optimizing preventive maintenance for mechanical components using genetic algorithms", *Reliability Engineering and System Safety*, VOL. 74, pp. 89-97, 2001.
- [VAL89] Valdez-Flores C. & Feldman R., "A survey of preventive maintenance models for stochastically deteriorating single-unit systems", *International Journal of Naval Research Logistics* VOL. 36(4), pp.419-446, 1989.
- [VER88] Verdol P., *Térotechnologie : interprétation économique d'un nouveau mode d'organisation intégrée de la disponibilité des équipements industriels*, thèse de doctorat soutenue à l'université Lumière Lyon II (France), 1988.
- [VIG99] Vignier A., Billaut J.C. & Proust C., "les problèmes d'ordonnancement de type flow-shop hybride : état de l'art", *RAIRO Recherche Opérationnelle*, VOL. 33(2), pp.117-183, 1999.

- [VIN00] Vineyard M., Amoako-Gyampah K. & Meredith J. R., "An evaluation of maintenance policies for flexible manufacturing systems", *International Journal of Operations and Production Management*, Vol. 20(4), pp. 409-426, 2000.
- [WAN96] Wang H. & Pham H., "Optimal maintenance policies for several imperfect repair models", *International Journal of Systems Science*, VOL. 27(6), pp.543-549, 1996.
- [WEI99] Weinstein L. & Chung C.-H., "Integrated maintenance and production decisions in a hierarchical production planning environment", *Computer and Operations Research*, VOL. 26, pp.1059-1074, 1999.
- [WHE96] Wheaton, R., "Reliability-Based Maintenance Requires Mill Culture Change", *Pulp and Paper*, pp. 53-61, July 1996.
- [WHI98] Whisterkamp T. A, "Evaluating the maintenance process", IIE solution, pp. 22-27, December 1998.
- [WID01] Widmer M., Hertz A. & Costa D., *Les métaheuristiques. Ordonnement de la production*. Edition Hermès Sciences, 2001.
- [WID01a] Widmer M., "Les métaheuristiques : des outils performants pour les problèmes industriels", 3<sup>ème</sup> Conférence Francophone de Modélisation et de SIMulation (MOSIM'01), Troyes (France), 25-27 avril 2001, VOL. 1, pp.13-21, 2001.
- [WU99] Wu S.D., Byeon E. & Storer R.H., "A graph theoretic decomposition of jobshop scheduling problem to achieve scheduling robustness", *Operation Research*, VOL. 47(1), pp. 113-124, 1999.
- [XIE93] Xie X.L., "Performance analysis of a transfer line with unreliable machines and finite buffers", *IIE Transactions*, VOL. 25(1), pp. 99-108, 1993.
- [ZIE90] Ziebelin D., *Raisonnement sur les contraintes : Application en ordonnancement*. Thèse de doctorat en informatique soutenue à l'université Joseph Fourier de Grenoble (France), 1990.
- [ZWI96] Zwingelstein G., *La maintenance basée sur la fiabilité, Guide pratique d'application de la RCM. Série Diagnostic et Maintenance*, AFNOR, 1996.

---

# **ANNEXES**

---



---

## *Annexes*

### *Les heuristiques dédiés au Flowshop*

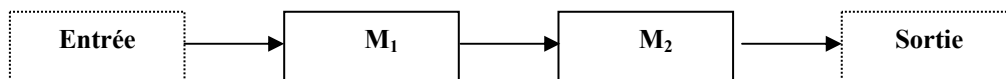
---

De nombreuses heuristiques ont été proposées pour résoudre le problème général du *flowshop* ; la grande majorité d'entre elles utilisent des transformations mathématiques pour se ramener au problème de Johnson à deux machines.

On se propose dans ce qui suit de donner intégralement l'algorithme de Johnson (cas où  $m=2$ ), et de citer les différentes heuristiques pour le cas de plus de 2 machines.

#### *L'algorithme de Johnson*

L'algorithme de Johnson permet de trouver une séquence optimale pour le problème à deux machines qui se présente comme suit : Un ensemble de  $n$  tâches doit être exécuté sur deux machines  $M_1$  et  $M_2$ . Il n'y a pas de place de stockage devant les machines. Chaque tâche doit passer d'abord dans la première machine, puis dans la deuxième machine. Le but est d'exécuter toutes les tâches en un minimum de temps.



En 1954, Johnson [JOH54] édicta une règle permettant de trouver une séquence optimale.

#### *Règle de Johnson.*

La tâche  $i$  précède la tâche  $j$  dans une séquence optimale si :  $\text{Min} \{t_{i1}, t_{j2}\} \leq \text{Min} \{t_{j1}, t_{i2}\}$

Ainsi, suivant cette règle, le début de la séquence optimale est composé des tâches ayant des durées d'exécution plus faible sur la première machine, et en fin d'ordonnancement celles ayant des durées d'exécution plus faible sur la seconde machine.

Johnson proposa ensuite un algorithme basé sur cette règle.



---

**Algorithme de Johnson**


---

**Début****Étape 1 :** constituer les groupes de tâches

$$U = \{i / P_{i1} < P_{i2}\} \text{ trouver Min } \{t_{i1}, t_{i2}\}.$$

$$V = \{i / P_{i1} \geq P_{i2}\}.$$

**Étape 2 :** Ordonner U selon l'ordre SPT sur la première machine ( $P_{i1}$ ) et V selon l'ordre LPT sur la seconde machine ( $P_{i2}$ ).**Étape 3 :** la séquence optimale est donnée par les tâches ordonnées dans U enchaînées par celles ordonnées dans V**Fin**

L'algorithme LPT (Longest Processing Time) : Les tâches sont ordonnées selon l'ordre décroissant de leurs durées opératoires.

L'algorithme SPT (Shortest Processing Time) : Les tâches sont ordonnées selon l'ordre croissant de leurs durées opératoires.

Exemple numérique :

Considérons l'ensemble des cinq tâches décrites par le tableau suivant :

	T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	T <sub>4</sub>	T <sub>5</sub>
Machine M <sub>1</sub>	3	5	1	6	7
Machine M <sub>2</sub>	6	2	2	6	5

$$U = \{T_1, T_3\} \xrightarrow{\text{SPT}} U = \{T_3, T_1\}$$

$$V = \{T_2, T_4, T_5\} \xrightarrow{\text{LPT}} V = \{T_4, T_5, T_2\}$$

En utilisant l'algorithme de Johnson, nous obtenons la séquence optimale suivante :

(T<sub>3</sub>, T<sub>1</sub>, T<sub>4</sub>, T<sub>5</sub>, T<sub>2</sub>).

**L'heuristique de Palmer**

En 1965, Palmer [PAL65] a proposé un classement d'indice décroissant pour ordonner les tâches sur les machines en fonction des temps d'exécution. L'idée est de donner une priorité aux tâches ayant des temps d'exécution croissants dans leurs gammes opératoires.

---

## Algorithme de Palmer

---

### Début

**Étape 1 :** pour chaque tâche  $i$  ( $1 \leq i \leq n$ ) calculer la valeur de l'indice  $S_i$  tel que :

$$S_i = (m-1) * P_{i,m} + (m-3) * P_{i,m-1} + (m-5) * P_{i,m-2} + \dots - (m-3) * P_{i,2} - (m-1) * P_{i,1}$$

Avec  $P_{ij}$  le temps opératoire de la tâche  $i$  sur la machine  $j$ .

**Étape 2 :** la séquence est déterminée en classant les tâches par ordre décroissant des indices.

### Fin

---

### L'heuristique CDS

En 1970, Campbell, Dudek et Smith [CAM70] ont proposé une méthode généralisant l'algorithme de Johnson. Son principe est de générer à partir du problème à  $m$  machines  $m-1$  problèmes fictifs à deux machines équivalentes.

---

### Heuristique CDS

---

#### Début

**Étape1 :** on ne considère que la première et la dernière machine (machine  $m$ ), soit pour un travail  $i$  :  $P^1_{i1} = p_{i1}$  ;  $P^1_{i2} = P_{im}$ . Tel que :

$P_{ij}$  : durée d'exécution du travail  $i$  sur la machine  $j$  et  $m$  le nombre de machines.

$P^1_{i1}$  : durée d'exécution du travail  $i$  sur la première machine équivalente.

**Étape2 :** les deux machines équivalentes sont respectivement constituées des deux premières et des deux dernières machines, sans tenir compte des machines intermédiaires. De manière générale à l'étape  $j$ , on a les durées suivantes sur les deux machines équivalentes :

$$P^j_{i1} = \sum_{k=1}^j P_{ik} \quad , \quad P^j_{i2} = \sum_{k=m-j+1}^m P_{ik} \quad j = 1, \dots, m-1.$$

#### Fin

---

$P_{ij}$  : durée d'exécution du travail  $i$  sur la machine  $j$  et  $m$  le nombre de machines.

$P^j_{i1}$  : durée d'exécution du travail  $i$  sur la  $j^{\text{ième}}$  machine équivalente.

Pour chacun de ces problèmes fictifs, on détermine la séquence optimale par la règle de Johnson (voir problème de Johnson). Conserver la meilleure solution parmi les  $m-1$ .

### L'heuristique de Gupta

En 1971, Gupta [GUP71] a présenté une autre heuristique très similaire à celle de Palmer. Il s'est aperçu que la règle de Johnson donne une séquence optimale pour le problème à 3 machines en classant les tâches par ordre décroissant des indices  $S_i$ .

---

**Heuristique Gupta pour un problème à trois machines (m=3)**


---

**Début**

**Etape 1.** Pour chaque tâche  $i$  ( $1 \leq i \leq 3$ ), calculer la valeur de l'indice  $S_i$  tel que :

$S_i = e_i / \text{Min}(t_{i1} + t_{i2}, t_{i2} + t_{i3})$  avec  $e_i = 1$  si  $t_{i1} \leq t_{i3}$ , -1 sinon.

**Etape 2 :** déterminer la séquence en classant les tâches par ordre décroissant des indices

$S_i$

**Fin**

---



---

**Heuristique de Gupta (m>3)**


---

**Début**

**Etape 1.** Pour chaque tâche  $i$  ( $1 \leq i \leq n$ ), calculer la valeur de l'indice  $S_i$  tel que :

$S_i = e_i / \text{Min}(t_{ik} + t_{i,k+1} \mid 1 \leq k \leq m-1)$  avec  $e_i = 1$  si  $t_{i1} < t_{im}$ , -1 sinon.

Tel que  $t_{ij}$  est le temps d'exécution de la tâche  $i$  sur la machine  $j$

**Etape 2 :** déterminer la séquence en classant les tâches par ordre décroissant des indices

$S_i$

**Fin**

---

**L'heuristique RA**

En 1977, DANNEBRING [DAN77] a essayé de combiner les avantages de l'heuristique de Palmer et celle de Campbell et al [CAM70], en proposant une variante de l'heuristique CDS. Cette méthode est appelée procédure à accès rapide (Rapide Access procédure : RA). Il a proposé deux méthodes de post optimisation : La procédure à accès rapide avec recherche dans un voisinage proche (Rapide Access with Close order Search (RACS)), et la procédure à accès rapide avec recherche étendue (Rapide Access with Extensive Search (RAES)). Il a défini un voisin comme étant une nouvelle séquence obtenue par simple permutation de deux tâches successives.

---

**L'algorithme RAES**


---

**Début**

Soit MST le tableau dans laquelle est enregistrée la meilleure séquence trouvée.

**Etape1 :** Trouver une séquence en utilisant l'heuristique RA.

**Etape2 :** Mémoriser cette séquence dans le tableau MST.

**Etape3 :** Trouver la meilleure séquence parmi tous les voisins.

**Si** la valeur de la fonction objectif est inférieur à celle de MST

**Alors** aller à l'étape 2.

**Sinon** STOP : le tableau MST fournit la solution de l'heuristique RAES

**Fin**

---

### ***L'heuristique NEH***

En 1983, NAWAZ, ENSCORE Jr. & HAM [NAW83] ont proposé un algorithme basé sur l'hypothèse qu'un lot ayant un temps total d'exécution élevé est prioritaire (le lot est positionner en priorité dans un ordonnancement partiel) par rapport à une tâche plus faible dans le cas de la minimisation du makespan. Nous avons adapté cette heuristique pour le cas de la minimisation de la somme des retards en privilégiant les tâches ayant la valeur  $Ne_h_i$ . Le calcul de la valeur de  $Ne_h_i$  et la détermination de l'ordre  $ORD$  se fait selon la fonction objectif, tel que les tâches les plus prioritaires sont placées au début de la liste  $L$ .

#### ***Si la fonction objectif est le Makespan:***

Les tâches ayant un temps total d'exécution élevé sont prioritaires par rapport aux tâches ayant un temps total d'exécution plus faible. Donc les tâches de production seront ordonnées par ordre décroissant de  $Ne_h_i$  (temps total d'exécution) ou  $Ne_h_i = \sum_{j=1}^m P_{ij}$

#### ***Si la fonction objectif est la somme pondérée des retards:***

Les tâches ayant des pénalités de retard élevées sont prioritaires par rapport aux tâches ayant des pénalités de retards plus faibles.

Une tâche  $P_i$  ayant une durée  $(d_i - (r_i + \sum_{j=1}^m P_{ij}))$  faible est prioritaire par rapport à une autre tâche  $P_k$  ayant une durée  $(d_k - (r_k + \sum_{j=1}^m P_{kj}))$  plus élevée. Cela veut dire que la tâche  $P_i$  qui se termine à une date très proche de sa date de fin au plus tard  $d_i$  si elle commence à sa date de début au plus tôt  $r_i$  est prioritaire par rapport à une autre tâches  $P_k$  qui se termine à une date plus loin de sa date de fin au plus tard  $d_k$  si elle commence à sa date de début au plus tôt  $r_k$ .

Donc les tâches de production seront ordonnées par ordre croissant de la valeur :

$$Ne_h_i = (1/w_i) * (d_i - (r_i + \sum_{j=1}^m P_{ij}))$$

En appliquant cet algorithme, on génère la séquence finale de manière constructive, en ajoutant à chaque étape une nouvelle tâche et en trouvant ainsi, à chaque itération, la meilleure solution partielle. L'adaptation de cet algorithme est la suivante:

---

**Algorithme NEH**


---

**Début**

**Etape 1 :** Pour chaque tâche  $P_i$  ( $i \in [1..N]$ ,  $N$  : nombre de tâche), calculer  $Neh_i = \sum_{j=1}^m P_{ij}$ .

**Etape 2 :** Ordonner dans une liste  $L$  les tâches par ordre de la valeur  $Neh_i = \sum_{j=1}^m P_{ij}$ .

**Etape 3 :** Prendre les deux premières tâches de la liste  $L$  et trouver la meilleure séquence pour ces deux tâches en calculant les deux ordonnancements possibles. Ne plus changer les positions relatives de ces deux tâches lors des prochaines étapes de l'heuristique ;  
 $i := 3$ .

**Etape 4 :** Prendre la tâche se trouvant en  $i^{\text{ème}}$  position dans la liste  $L$  et trouver la meilleure séquence en plaçant cette tâche dans les  $i$  positions possibles parmi les tâches déjà placées

**Etape 5 :** Si ( $i < N$ ) alors  $i := i+1$  ; aller à l'étape 4. Sinon

**STOP :** la séquence trouvée est la solution de l'heuristique NEH.

**Fin****Exemple numérique**

Considérons cinq tâches  $\{T_1, T_2, \dots, T_5\}$  qui doivent s'exécuter sur sept machines.

Les durées opératoires de ces tâches sur les machines sont illustrées dans le tableau suivant :

	M <sub>1</sub>	M <sub>2</sub>	M <sub>3</sub>	M <sub>4</sub>	M <sub>5</sub>	M <sub>6</sub>	M <sub>7</sub>
T <sub>1</sub>	2	1	3	5	6	4	5
T <sub>2</sub>	2	1	3	4	7	1	1
T <sub>3</sub>	3	5	6	8	3	2	4
T <sub>4</sub>	5	4	3	2	1	2	3
T <sub>5</sub>	3	1	4	1	3	4	1

L'heuristique de Palmer :  $S_i = \sum_{k=0}^6 (7 - (2 * k + 1)) * P_{i(7-k)}$

$S_1 = 36$  ;  $S_2 = 2$  ;  $S_3 = -12$  ;  $S_4 = -24$  ;  $S_5 = -2$ .

Ordre décroissant des indices :  $S_1, S_2, S_5, S_3, S_4$ .

ORD=  $\{T_1, T_2, T_5, T_3, T_4\}$ .  $C_{\max} = 41$

**L'heuristique CDS :**

J = 1

$$U = \{T_1, T_3\} \xrightarrow{\text{SPT}} U = \{T_1, T_3\}$$

$$V = \{T_2, T_4, T_5\} \xrightarrow{\text{LPT}} V = \{T_2, T_5, T_4\}$$

ORD = {T<sub>1</sub>, T<sub>3</sub>, T<sub>2</sub>, T<sub>5</sub>, T<sub>4</sub>}. C<sub>max</sub> = 47

J = 2

$$U = \{T_1, T_5\} \xrightarrow{\text{SPT}} U = \{T_1, T_5\}$$

$$V = \{T_2, T_3, T_4\} \xrightarrow{\text{LPT}} V = \{T_3, T_4, T_2\}$$

ORD = {T<sub>1</sub>, T<sub>5</sub>, T<sub>3</sub>, T<sub>4</sub>, T<sub>2</sub>}. C<sub>max</sub> = 42

	P <sub>i1</sub> <sup>2</sup>	P <sub>i2</sub> <sup>2</sup>
T1	3	9
T2	3	2
T3	8	6
T4	9	5
T5	4	5

J = 3

$$U = \{T_1, T_2\} \xrightarrow{\text{SPT}} U = \{T_1, T_2\}$$

$$V = \{T_3, T_4, T_5\} \xrightarrow{\text{LPT}} V = \{T_3, T_5, T_4\}$$

ORD = {T<sub>1</sub>, T<sub>2</sub>, T<sub>3</sub>, T<sub>5</sub>, T<sub>4</sub>}.

C<sub>max</sub> = 41

	P <sub>i1</sub> <sup>3</sup>	P <sub>i2</sub> <sup>3</sup>
T1	6	15
T2	6	9
T3	14	9
T4	12	6
T5	8	8

J = 4

$$U = \{T1, T2\} \xrightarrow{\text{SPT}} U = \{T2, T1\}$$

$$V = \{T3, T4, T5\} \xrightarrow{\text{LPT}} V = \{T3, T5, T4\}$$

ORD = {T2, T1, T3, T5, T4}.

$C_{\max} = 41$

	$P_{i1}^4$	$P_{i2}^4$
T1	11	20
T2	10	13
T3	22	17
T4	14	8
T5	9	9

J = 5

$$U = \{T1, T5\} \xrightarrow{\text{SPT}} U = \{T5, T1\}$$

$$V = \{T3, T4, T2\} \xrightarrow{\text{LPT}} V = \{T3, T2, T4\}$$

ORD = {T5, T1, T3, T2, T4}.

$C_{\max} = 44$

	$P_{i1}^5$	$P_{i2}^5$
T1	17	23
T2	17	16
T3	25	23
T4	15	11
T5	12	13

J = 6

$$U = \{T1, T3\} \xrightarrow{\text{SPT}} U = \{T1, T3\}$$

$$V = \{T2, T4, T5\} \xrightarrow{\text{LPT}} V = \{T2, T4, T5\}$$

ORD = {T1, T3, T2, T4, T5}

$C_{\max} = 44$

	$P_{i1}^6$	$P_{i2}^6$
T1	21	24
T2	18	17
T3	27	28
T4	17	15
T5	16	14

La meilleure solution est  $C_{\max} = 41$ , la séquence est ORD = {T2 T1 T3 T5 T4}.

**L'heuristique Gupta :**

$S_k = e_k / \text{Min} (P_{ki} + P_{k,i+1} \mid 1 \leq i \leq m-1)$  avec  $e_k = 1$  si  $P_{k1} < P_{km}$ , -1 sinon

$$S_1 = e_1 / \min \{t_{1k} + t_{1,k+1} \mid 1 \leq k \leq 6\}$$

$$e_1 = 1, S_1 = 1/3$$

$$S_2 = e_2 \min \{t_{1k} + t_{1,k+1} \mid 1 \leq k \leq 6\}; e_2 = -1; S_2 = -1/2$$

$$S_3 = e_3 \min \{t_{1k} + t_{1,k+1} \mid 1 \leq k \leq 6\}; e_3 = 1; S_3 = 1/5$$

$$S_4 = e_4 \min \{t_{1k} + t_{1,k+1} \mid 1 \leq k \leq 6\}; e_4 = -1; S_4 = -1/3$$

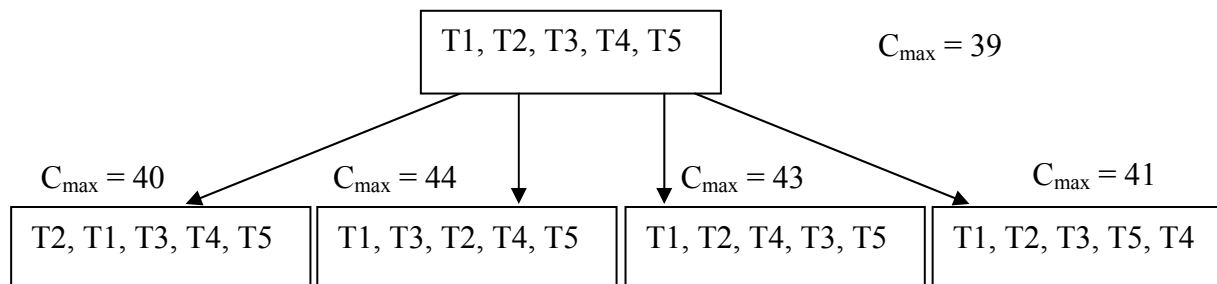
$$S_5 = e_5 \min \{t_{1k} + t_{1,k+1} \mid 1 \leq k \leq 6\}; e_5 = -1; S_5 = -1/5$$

Ordre décroissant des indices  $S_1, S_3, S_5, S_4, S_2$ ; ORD = {T1, T3, T5, T4, T2}.

$$C_{\max} = 40$$

**L'heuristique RA :**

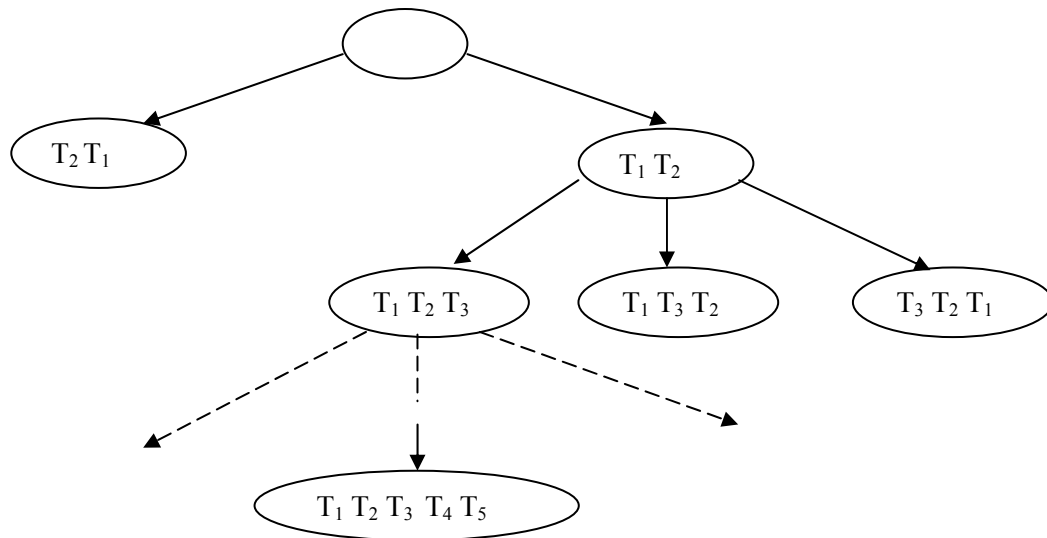
Soit l'ordre suivant {T1, T2, T3, T4, T5}, en permutant à chaque étape deux tâches successives.



$C_{\max 1} = 39$ , en permutant deux tâches successives, on obtient les voisins S1, S2, S3 et S4.

D'où  $C_{\max} = 39$ . ORD = {T1, T2, T3, T4, T5}



*L'heuristique NEH*

La séquence fournit par l'heuristique NEH est  $\{T_1, T_2, T_3, T_4, T_5\}$   $C_{\max} = 39$



---

# RESUME

---

**Résumé.** La maintenance et la production sont deux fonctions qui agissent sur les mêmes ressources. Cependant l'ordonnancement de leurs activités respectives est indépendant, et ne tient pas compte de cette contrainte. Les ressources (machines) sont toujours considérées comme disponibles à tout moment ou éventuellement durant certaines fenêtres de temps. Dès lors la planification de la maintenance n'est jamais prioritaire sur la production, pour effectuer des interventions préventives. Les travaux de cette thèse proposent quelques éléments de réponse au problème de la planification commune et intégrée des tâches de maintenance et de production, avec comme objectif le respect des contraintes intrinsèques au problème. L'objectif de notre travail est double. D'une part démontrer la nécessité de développer des heuristiques d'ordonnancement conjoint production/maintenance pour atteindre l'objectif d'optimisation de la sûreté de fonctionnement du système de production. Dans ce contexte nous avons proposé l'adaptation d'un certain nombre d'heuristiques des approches constructive, itérative et évolutive pour le cas de l'ordonnancement conjoint production/maintenance dans un atelier de type *flowshop* de permutation. D'autre part, nous avons étudié la contribution de la maintenance à la robustesse de ces ordonnancements conjoints. Le modèle proposé a pour objectif d'inscrire la génération d'ordonnements conjoints comme une démarche proactive, et de démontrer que l'intégration de la maintenance lors de la génération des ordonnancements de production est à mettre au profit de la robustesse de ces derniers.

**Mots clés :** Ordonnancement, Production, Maintenance, Heuristiques, Robustesse.

**Abstract.** Maintenance and the production are two functions which act on the same resources. However the scheduling of their respective activities is independent, and does not take into account this constraint. The resources (machines) are always regarded as being constantly or during certain time windows. Consequently, the planning of maintenance has never priority on the production, to carry out preventive interventions. In this thesis, we propose some solutions to the problem of common and integrated maintenance and production planning tasks, with the objective to respect the intrinsic constraints of the problem. The objective of our work is double. On the one hand, we want to show the need for developing joint scheduling production/maintenance heuristic, to optimise production system reliability. In this context, we propose the adaptation of some heuristics approaches: constructive, iterative and evolutionary in the case of joint scheduling production/maintenance in a permutation flowshop. On the other hand, we studied the maintenance contribution to the robustness of these joint scheduling. The suggested model has a main goal: to register the generation of joint scheduling as a proactive approach. The integration of maintenance tasks during the generation of production scheduling of production, is to be put at the profit of their robustness.

**Key Words:** Scheduling, Production, Maintenance, Heuristic, Robustness.

