

# Ordonnancement cyclique multi-produits des lignes de traitement de surface : Méthodes exactes et approchées

Adnen El Amraoui

▶ **To cite this version:**

Adnen El Amraoui. Ordonnancement cyclique multi-produits des lignes de traitement de surface : Méthodes exactes et approchées. Autre. Université de Technologie de Belfort-Montbeliard; Université de Tunis El Manar, 2011. Français. NNT : 2011BELF0164 . tel-00703221

**HAL Id: tel-00703221**

**https://theses.hal.science/tel-00703221**

Submitted on 1 Jun 2012

**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.

N° d'ordre : 164 Année : 2011

UNIVERSITE DE TECHNOLOGIE DE BELFORT-MONTBELIARD
ECOLE DOCTORALE SCIENCES PHYSIQUES POUR L'INGENIEUR ET MICROTECHNIQUES

UNIVERSITE DE TUNIS EL MANAR
ECOLE NATIONALE D'INGENIEURS DE TUNIS

# THESE

présentée par

**Adnen EL AMRAOUI**

Pour l'obtention du grade de

## DOCTEUR

En Automatique

# Ordonnancement Cyclique Multi-Produits des Lignes de Traitement de Surface : Méthodes Exacte et Approchées

Soutenue le 12 juillet 2011 devant le jury d'Examen composé de :

| | | | | |
|---|---|---|---|---|
| M[r] | KACEM I. | Professeur | UPVM, Metz, France | Président et Rapporteur |
| M[r] | ARTIBA A. | Professeur | ISTV, Valenciennes, France | Rapporteur |
| M[r] | BENHADJ BRAIEK N. | Professeur | EPT, Tunis, Tunisie | Rapporteur |
| M[r] | ELLOUZE N. | Professeur | ENIT, Tunis, Tunisie | Examinateur |
| M[r] | EL MOUDNI A. | Professeur | UTBM, Belfort, France | Directeur |
| M[r] | BENREJEB M. | Professeur | ENIT, Tunis, Tunisie | Directeur |
| M[me] | MANIER M.-A. | Maître de Conf. | UTBM, Belfort, France | Co-directeur |

Thèse préparée au Laboratoire Systèmes et Transports de l'Université de Technologie de Belfort-Montbéliard
et à l'Unité de Recherche LARA-Automatique de l'Ecole Nationale d'Ingénieurs de Tunis

# *Avant-propos*

*Nous tenons à exprimer tout d'abord nos remerciements à Monsieur Imed KACEM, Professeur et à l'Université de Paul Verlaine à Metz (UPVM), d'avoir accepté de rapporter notre travail et de nous avoir fait le grand honneur d'accepter de présider le Jury d'Examen. Qu'il trouve ici l'expression de notre profond respect.*

*Nous exprimons notre profonde gratitude à Madame Marie-Ange MANIER, Maitre de Conférence (HDR) à l'Université de Technologie Belfort-Montbéliard, à Monsieur Abdellah EL MOUDNI, Professeur à l'Université de Technologie Belfort-Montbéliard, et à Monsieur Mohamed BENREJEB, Professeur à l'Ecole Nationale de Tunis (ENIT) et Directeur de l'Unité de Recherche UR-LARA à l'ENIT pour nous avoir guidé et soutenu durant toute l'élaboration de ces travaux de recherches.*

*Notre profonde gratitude va à Messieurs Abdelhakim ARTIBA, Professeur et à l'Institut des Sciences et Techniques de Valenciennes – ISTV et Naceur BENHADJ BRAIEK, Professeur à l'Ecole Polytechnique de Tunis (EPT), pour avoir bien voulu rapporter sur nos travaux de thèse. Nous leur adressons nos sincères remerciements.*

*Nous tenons à remercier Monsieur Noureddine ELLOUZE, Professeur à l'Ecole Nationale d'Ingénieurs de Tunis, d'avoir accepté d'être membre du jury. Nous lui adressons nos vifs remerciements.*

*Finalement, nous adressons un grand merci à tous les chercheurs du Laboratoire Systèmes et Transports de l'UTBM et de L'Unité de Recherche LARA Automatique de l'ENIT et toute autre personne qui, par leur soutien, par leur amicale présence et par leur sympathie, ont rendu ce travail agréable. Nous leur exprimons ici toute notre gratitude.*

*A mes très chers parents*

*A ma sœur, mon frère et mon beau frère*

*A mes deux ravissantes nièces*

*A toute ma famille*

*A mes professeurs et mes amis*

# Table des matières

**Introduction générale** ..... **1**

**Chapitre I:** Problème d'ordonnancement : définitions, caractérisation, ligne de traitement de surface et état de l'art

I. 1 - Introduction ..... 5

I. 2 - Problèmes d'ordonnancement ..... 6
I. 2. 1 - Définition générale ..... 6
I. 2. 2 - Eléments d'un problème d'ordonnancement ..... 7
a - Les tâches ..... 7
b - Les ressources ..... 7
c - Les contraintes ..... 8
d - Les critères ..... 8
I. 2. 3 - Les ateliers ..... 9
a - Flow-shop ..... 9
b - Job-shop ..... 9
c - Open-shop ..... 10
d - Les ateliers dotés d'une ou plusieurs ressources de transport ..... 10
I. 2. 4 - Complexité des problèmes d'ordonnancement ..... 12
a - La complexité méthodologique ..... 13
b - La complexité problématique ..... 13

I. 3 - Les ateliers de traitement de surface ..... 14
I. 3. 1 - Présentation Générale ..... 14
I. 3. 2 - Ressources de traitement ..... 15
a - Cuves de traitement ..... 15
b - Bains chimiques ..... 16
c - Postes de chargement et de déchargement ..... 16
d - Opérateurs ..... 17
I. 3. 3 - Ressources de manutention ..... 18
a. Porteurs ..... 18
b. Robots ..... 19
I. 3. 4 - Lignes complexes ..... 21
a - Cuves multi-fonctions ..... 21
b - Cuves multi-bacs ..... 22
c - Ligne multi-robots ..... 23
I. 3. 5 - Impact sur l'environnement ..... 23
a - L'immersion ..... 24
b - Le rinçage ..... 24
c - L'égouttage ..... 25

I. 4 - Ordonnancement des lignes de traitement de surface ..... 25
I. 4.1 - Définition et spécificités du HSP ..... 25
I. 4. 2 - Problèmes prédictifs ..... 29

I. 4. 3 - Problèmes dynamiques ..... 31

I. 5 - Notation..... 32

I. 6 - Etat de l'art sur le "Cyclic Hoist Scheduling Problem" ..... 34
I. 6. 1 - Synthèse de la recherche en HSP ..... 34
I. 6. 2 - CHSP mono-gamme ..... 36
a. Lignes mono-robot et mono-bac ..... 36
b. Lignes mono-robot et multi-bacs ..... 38
c. Lignes multi-robots et mono-bac ..... 39
d. Lignes multi-robots et multi-bacs ..... 39
e. Problèmes connexes au HSP ..... 40
I. 6. 3 - CHSP multi-gammes ..... 42
a. Séquences opératoires et fenêtres temporelles identiques ..... 43
b. Séquences opératoires identiques et fenêtres temporelles différentes ..... 43
c. Séquences opératoires et fenêtres temporelles différentes ..... 48

I. 7 - Bilan..... 51

**Chapitre II:** Formulation, optimisation et résolution du problème CHSP multi-produits par la Programmation Linéaire Mixte

II. 1 - Introduction ..... 54

II. 2 - Formalisation du problème de base 2-cyclique hétérogène ..... 55
II. 2. 1 - Le problème ..... 55
II. 2. 2 - Notations ..... 57
II. 2. 3 - Analyse du problème ..... 59
a. Temps de cycle ..... 59
b. Durée d'une opération de trempe ..... 59
c. Dates d'exécution des opérations ..... 60
d. Description d'une séquence cyclique ..... 62
II. 2. 4 - Etude de faisabilité ..... 65

II. 3 - Modélisation linéaire du CHSP ..... 67
II. 3. 1 - Notion et notations ..... 67
II. 3. 2 - Modèle linéaire (*S1*) du problème de base 2-cyclique hétérogène ..... 69
II. 3. 3 - Complexité du modèle ..... 71
II. 3. 4 - Procédure de résolution ..... 72
II. 3. 5 - Exemple illustratif ..... 74
II. 3. 6 - Résultats ..... 76
a. Description des benchmarks ..... 76
b. Validation du modèle *(S1)* ..... 76

II. 4 - Etude de la structure et de la flexibilité d'une solution ..... 79
II. 4. 1 - Structure d'une solution ..... 79
II. 4. 2 - Flexibilité opératoire d'une solution ..... 83
a. Domaine de validité d'une séquence cyclique ..... 83
b. Impact des fenêtres temporelles sur les performances d'une solution ..... 85

II. 4. 3 - Flexibilité de transport d'une solution ..... 86
a. Modèle linéaire pour un problème connexe au CHSP : (*S2*) ..... 87
b. Complexité du modèle *(S2)* ..... 88
c. Exemple illustratif ..... 88
d. Résultats ..... 90

II. 5 - Extensions du modèle 2-cyclique hétérogène aux lignes complexes ..... 92
II. 5. 1 - Extension 1 : Ligne à cuves multi-bacs (*S3*) ..... 92
II. 5. 2 - Extension 2 : Ligne à cuves multi-fonctions (*S4*) ..... 97
II. 5. 3 - Extension 3 : Ligne à postes de chargement et déchargement associés (*S5*) ..... 100

II. 6 - Extension du modèle 2-périodique aux ordonnancements r-cycliques hétérogènes ..... 102
II. 6. 1 - Définition d'un r-cycle ..... 102
II. 6. 2 - Modèle linéaire du problème r-cyclique multi-produits: (*S6*) ..... 103
II. 6. 3 - Exemple illustratif ..... 105
II. 6. 4 - Résultats ..... 107

II. 7 - Bilan…… ..... 107

**Chapitre III:** Optimisation, modélisation et résolution du problème CHSP multi-produits par des méthodes approchées : Heuristiques et Algorithmes Génétiques

III. 1 - Introduction ..... 110

*Partie 1 : Approche heuristique* ..... 111
III. 2 - Définitions ..... 111

III. 3 - Heuristiques dans le domaine du CHSP ..... 112

III. 4 - Algorithme de liste pour le CHSP multi-produits ..... 115
III. 4. 1 - Analyse préliminaire ..... 115
III. 4. 2 - Heuristique OCMPEST ..... 115
III. 4. 3 - Illustration ..... 119

*Partie 2 : Approche métaheuristique* ..... 121
III. 5 - Principe général de fonctionnement des algorithmes génétiques (AG) ..... 122

III. 6 - Etat de l'art des CHSP résolus par des algorithmes génétiques : ..... 124

III. 7 - Algorithme génétique pour le CHSP multi-produits ..... 126
III. 7. 1 - Représentation d'une solution ..... 126
III. 7. 2 - Génération de la population initiale ..... 128
III. 7. 3 - Opérateur de croisement ..... 129
III. 7. 4 - Opérateur de mutation ..... 129
III. 7. 5 - Sélection ..... 130
III. 7. 6 - Approche polynomiale pour le temps de cycle ..... 130
III. 7. 7 - Faisabilité d'un chromosome ..... 133
III. 7. 8 - Méthode d'insertion ..... 134
III. 7. 9 - Notre démarche de résolution ..... 135

III. 7. 10 - Illustration de la démarche de calcul sur un chromosome ..... 136

*Partie 3 : Résultats, comparaison et bilan* ..... 140
III. 8 - Résultats des simulations ..... 140

III. 9 - Influence du panachage sur la durée du cycle..... 144
III. 9. 1 - Description de la démarche de résolution ..... 144
III. 9. 2 - Résultats des simulations ..... 146

III. 10 - Bilan…..... 146

**Conclusion générale ..... 148**

**Bibliographie ……151**

**Annexes…….. ..... 161**

A.1. Notion d'optimisation combinatoire..... 161

A.2. Notation dédiée au HSP..... 163

A.3. Typologie arborescente des travaux de recherche mentionnés dans notre état de l'art ..... 167

A.4. Classification d'une partie des travaux de recherches mentionnés dans notre état de l'art ..... 171

A.5. Caractéristiques et données des benchmarks 2-cyclique multi-produits hétérogènes ..... 173

A.6. Résultats de la PLM sur les benchmarks 2-cyclique multi-produits hétérogènes de l'annexe 5 ..... 185

A.7. Notre modèle linéaire mono-produit ..... 193

A.8. Modèle référence de Phillips ..... 195

A.9. Données et résultats des benchmarks 2-cyclique hétérogènes du tableau II.9 ..... 197

A.10. Résultats 4-périodique vs 2-périodique de la PLM des benchmarks multi-produits hétérogènes de l'annexe 5 ..... 203

A.11. Composante des algorithmes génétiques..... 207

A.12. Comparaison entre la PLM, l'heuristique OCMPEST et EST sur les benchmarks 2-cyclique multi-produits hétérogènes de l'annexe 4..... 215

A.13. Comparaison entre PLM, OCMPEST et AG sur des exemples 2-périodique..... 223

A.14. Données des benchmarks multi-produits hétérogènes utilisées pour tester la procédure de panachage ..... 227

A.15. Données des benchmarks multi-produits hétérogènes à 20, 30, 40 et 50 cuves de traitements..... 229

# Table des Figures

I.1. Modèle linaire d'une séquence opératoire …….....9
I.2. Modèle de trois cheminements de séquences opératoires dans un atelier de type job-shop…….....10
I.3. Taxonomie ……12
I.4. Exemple d'une ligne de traitement de surface ……...……15
I.5. Différents types d'implantations physiques de chaîne ……..16
I.6. Configuration associée des postes de chargement et de déchargement ……16
I.7. Configuration dissociée des postes de chargement et de déchargement ……...17
I.8. Schéma d'une chaîne complexe faisant intervenir des opérateurs aux postes d'entrée et de sortie ……17
I.9. Représentation graphique d'un créneau ……...19
I.10. Représentation graphique d'une opération de transport……...19
I.11. Gamme de traitement utilisant une cuve multi-fonctions ……20
I.12. Gamme de traitement utilisant une cuve multi-bacs ……21
I.13. Tronçon de ligne en I muni de deux robots ……22
I.14. Schéma synoptique des entrées/sorties d'une ligne de traitement de surface ……23
I.15. Commande cyclique d'une gamme comportant 3 traitements ……29
I.16. Typologie du cas cyclique ……...33
I.17. Arbre multi-produits CHSP ……41
I.18. Procédure de la construction d'un arbre de recherche pour déterminer un ordonnancement 2-cyclique à 2m+2 opération de transport ……...44
I.19. Exemple de définition des configurations ……...48
I.20. Principe de fonctionnement de l'algorithme ……49
II.1. Extensions proposées du modèle de base élaboré ……54
II.2. Illustration du problème à traiter ……..55
II.3. Durées des traitements ……..59
II.4. Analyse du processus de transport du point de vue produit ……60
II.5. Analyse du processus de transport de point de vue robot ……60
II.6. Les étapes du déroulement de la séquence S ……63

II.7. Diagramme de Gantt d'un ordonnancement 2-cyclique ……...64
II.8. Différence entre cuve et stage pour un problème 2-cyclique hétérogène ……....67
II.9. Procédure de recherche des ordonnancements réalisables ……...72
II.10. Méthode de résolution ……73
II.11. Diagramme de Gantt de la solution optimale ……74
II.12. Taux de réduction moyenne du temps de cycle en fonction de la fenêtre de temps et la vitesse du robot ……77
II.13. Taux de réduction moyenne du temps de cycle en fonction de la taille de la ligne de traitement ……78
II.14. Perturbation maximale tolérée pour une fenêtre de temps $[a_i, b_i]$ ……..79
II.15. Graphe de l'étude de la robustesse d'une séquence cyclique ……81
II.16. Flexibilité des durées trempe pour le premier produit ……83
II.17. Flexibilité des durées trempe pour le deuxième produit ……83
II.18. Flexibilité totale de la solution cyclique ……83
II.19. Diagramme de Gantt de la meilleure solution obtenue par le modèle (*S1*) ……89
II.20. Diagramme de Gantt de la meilleure solution obtenue par le modèle ……...89
II.21. Exemple d'une cuve multi-bacs ……93
II.22. Diagramme de Gantt de la meilleure solution trouvée ……...96
II.23. Diagramme de Gantt de la meilleure solution trouvée ……...99
II.24. Diagramme de Gant de la meilleure solution obtenue ……100
II.25. Diagramme de Gantt de la meilleure solution obtenue ……105
III.1. Meilleure solution obtenue pour des durées opératoires bornées ……113
III.2. Meilleure solution obtenue pour des durées opératoires fixées ……113
III.3. Diagramme de Gantt de la solution obtenue par l'heuristique OCMPEST ……..119
III.4. Architecture générale d'un algorithme génétique ……123
III.5. Ordonnancement cyclique faisable des opérations de transport ……...126
III.6. Illustration de l'opérateur de croisement utilisé ……128
III.7. Illustration de l'opérateur de mutation utilisé ……...129
III.8. Représentation nodale d'un mouvement de transport (gène) ……....129
III.9. Modélisation de la contrainte du respect de la durée minimale de trempe ……...130
III.10. Modélisation de la contrainte du déplacement à vide du robot ……...130
III.11. Illustration de la longueur $Z_{k,k}$ ……130
III.12. Diagramme de Gantt de la séquence cyclique SM ……..136
III.13. Schéma du graphe orienté de la séquence cyclique SM ……137

III.14. Ecart Moyen (%) de la solution de l'heuristique EST par rapport à la durée moyenne optimale d'un 2-cycle ……..……142
III.15. Ecart Moyen (%) de la solution de l'heuristique OCMPEST par rapport à la durée moyenne optimale d'un 2-cycle ……142
III.16. Ecart Moyen (%) entre les solutions des heuristiques EST et OCMPEST ……143
III.17. Schéma synoptique du principe de fonctionnement de l'algorithme ……..144

# Liste des Tableaux

I.1. Exemple de gamme pour une ligne complexe de traitement de surface ……18
I.2. Comparaison entre le HSP et des méthodes d'ordonnancement usuelles ou de problèmes similaires ……..27
I.3. Publications relatives aux différentes classes du HSP ……...34
II.1. Nombre de contraintes et de variables pour 2 types de produits ……..71
II.2. Exemple de gammes opératoires partiellement identiques ……..73
II.3. Durées opératoires effectives pour chaque type de produit ……..74
II.4. Ecart moyen entre un cycle 2-périodique hétérogène et les 2 cycles 1-périodique homogènes ……...77
II.5. Influence de le fenêtre du temps sur la durée du cycle ……85
II.6. Données de l'exemple ……..88
II.7. Meilleure solution obtenue par le modèle ……88
II.8. Meilleure solution obtenue par le modèle (*S2*) ……88
II.9. Résultats des simulations ……..90
II.10. Fenêtres de temps de l'exemple ……95
II.11. Meilleure solution trouvée par le modèle (S3) ……...…95
II.12. Meilleure solution trouvée par le modèle (S1) pour l'exemple illustratif ……...…96
II.13. Donnée de l'exemple illustratif ……...…98
II.14. Meilleure solution trouvée par le modèle (S1) pour l'exemple illustratif ……..99
II.15. Données de l'exemple illustratif ……...100
II.16. Données de l'exemple illustratif ……..105
III.1. Données de l'exemple ……...……113
III.2. Données de l'exemple ……...118
III.3. Exemple de gammes opératoires partiellement identiques ……...135
III.4. Durées opératoires effectives pour chaque type de produit ……..138
III.5 : Résultats des simulations des benchmarks de [Mateo, 2001] et autres ……...140
III.6. Comparaison entre les deux configurations $C_{A\text{-}B\text{-}C}$ et $C_{A\text{-}C\text{-}B}$ ……...……145
III.7. Comparaison entre les différentes méthodes proposées ……146

# Introduction générale

Pour pouvoir évoluer dans un espace mondial très concurrentiel, les entreprises doivent respecter les délais et assurer à la fois une bonne qualité et une diversité de leurs produits. Dans cette optique, les entreprises doivent satisfaire une liste d'exigences accrues portant sur une bonne gestion de leurs ressources matérielles et financières.

Ainsi, l'évolution des entreprises passe par la remise en cause des fondements actuels de pratiques classiques d'organisation d'atelier. Nous nous intéressons plus particulièrement à l'optimisation de la production à travers une bonne gestion des ressources potentielles disponibles. Cette dernière est fortement affectée par l'ordonnancement des tâches sur les machines. Pour résoudre un tel problème, nous devons procéder en deux étapes. La première consiste à identifier et à modéliser le système en décrivant notamment l'ensemble des contraintes auxquelles il est soumis, dans le but d'optimiser un ou plusieurs critères. La seconde étape consiste à déterminer la méthode la plus adéquate pour résoudre le problème considéré.

Le travail présenté porte sur le problème d'ordonnancement de lignes de traitement de surface. Dans ce type d'atelier, les ressources de transport revêtent une importance capitale. En effet, ce sont elles qui constituent les éléments critiques dont dépendent les performances globales du système, et non pas les moyens de production à proprement parler. Ainsi, et bien que les tâches de manutention n'apportent pas de valeur ajoutée aux produits manufacturés, les ressources associées doivent être prises en compte. En effet, dans ce type d'atelier les temps de transports des produits ne peuvent plus être négligés devant les durées opératoires. Et c'est une des caractéristiques du problème d'ordonnancement du système de manutention des lignes de traitement de surface, plus connu sous la dénomination HSP, abréviation du *Hoist Scheduling Problem*.

Le traitement de surface constitue souvent une étape obligatoire au cours du processus de fabrication d'un grand nombre de produits de formes et de tailles diverses. Ce procédé consiste à déposer sur ces derniers une couche de métal pour les recouvrir pour des raisons de

résistance à l'usure, à la corrosion, ou tout simplement pour des propriétés esthétiques. Par rapport aux divers ateliers existants, y compris avec ressources de transport, le HSP est un problème atypique car particulièrement contraint, notamment à cause des aspects chimiques des processus opératoires mis en œuvre.

L'automatisation des lignes de traitement de surface nous incite à chercher un ordonnancement optimisé des opérations de transport du ou des robot(s), ayant pour fonction d'assurer le transport des produits entre les cuves de traitements. Les objectifs à atteindre peuvent être le respect des délais de fabrication (compétitivité), ou encore la maximisation de l'utilisation des ressources (rentabilité).

Les approches d'optimisation classiques utilisées pour résoudre ce type de problème sont pour la plupart issues de la Recherche Opérationnelle et limitées à la résolution des cas les plus simples. D'après la classification des *Hoist Scheduling Problems* établie en 2003 par Manier et Bloch, de nombreuses recherches ont déjà été entreprises, tant sur les aspects statiques que dynamiques, et la majorité porte sur le cas mono-produit. Toutefois, le problème est encore loin d'avoir été entièrement résolu de manière satisfaisante, c'est-à-dire en conformité avec les problèmes concrets trouvés dans l'industrie. En outre, de nombreuses variantes du HSP restent à explorer.

Notre contribution porte en particulier sur le fonctionnement cyclique multi-produits de ces systèmes de production. La commande cyclique est adaptée aux productions en grandes séries, vu l'avantage qu'elle fournit en réduisant la combinatoire de résolution, et sous réserve que les ratios de produits soient connus à l'avance. En effet, ce type de commande nous permet d'éviter d'ordonnancer la totalité des opérations et de nous limiter à un motif de taille réduite qui va se répéter autant de fois que nécessaire pour réaliser la commande totale. L'approche multi-produits présente l'avantage de rendre la solution cyclique plus flexible. L'optimisation de la production pour un tel type d'atelier consiste à rechercher le meilleur ordonnancement des tâches à accomplir en un temps raisonnable.

Dans le premier chapitre de ce mémoire, nous proposerons, d'abord, une vue d'ensemble sur les problèmes d'ordonnancement d'ateliers classiques (flow shop, job shop, …). Ensuite, nous décrirons les lignes de traitement de surface, ainsi que les différentes configurations qu'elles peuvent prendre et nous définirons les contraintes d'ordonnancement spécifiques à ce système

de production. Enfin, nous présenterons un état de l'art sur le problème cyclique mono et multi-produits et nous situerons notre démarche par rapport à la problématique d'ensemble.

Dans le second chapitre, nous présenterons notre première méthode de résolution, basée sur un modèle linéaire et une méthode de résolution arborescente de type séparation et évaluation. Nous proposerons des modélisations pour différentes extensions physiques de la ligne et nous fournirons des exemples illustratifs et des résultats sur des benchmarks.

Dans le troisième chapitre, nous nous appuyons en partie sur des recherches préalables sur le cas cyclique mono-produit avec des heuristiques et des métaheuristiques de type algorithmes évolutionnaires. Nous proposerons tout d'abord une heuristique basée sur un algorithme de liste. L'originalité d'une telle approche réside dans le fait que le degré du cycle n'est pas fixé au préalable. Ensuite, nous présenterons une modélisation du problème HSP cyclique multi-produits par les algorithmes génétiques. Des résultats seront fournis, à partir desquels nous analyserons les performances des diverses méthodes élaborées. Enfin, nous proposerons quelques perspectives pour ce travail.

# Chapitre I

# Problème d'ordonnancement : définitions, caractérisation, ligne de traitement de surface et état de l'art

## I. 1 - Introduction

L'ordonnancement est un domaine d'application de la recherche opérationnelle et une problématique de la gestion de la production qui vise à diminuer les coûts de production et les délais de livraisons. La résolution de ce problème d'optimisation permet aux entreprises d'être plus efficaces et par conséquent, plus compétitifs sur le marché. Résoudre un tel problème consiste à organiser l'exécution de tâches, c'est-à-dire, à déterminer leurs dates de démarrage et à leurs attribuer les ressources matérielles nécessaires, en respectant les contraintes du problème et en optimisant un ou plusieurs objectifs (ou critères) préalablement définis [Pinedo, 1955].

Un ordonnancement se décompose en deux parties, pouvant revêtir une importance variable, mais toujours présentes dans un atelier [Saad, 2007]:

- ✓ l'ordonnancement prédictif qui consiste à prévoir a priori un certain nombre de décisions en fonction de données prévisionnelles et d'un modèle de l'atelier ;
- ✓ l'ordonnancement réactif qui consiste à adapter les décisions prévues en fonction de l'état courant du système et des déviations entre la réalité et le modèle théorique.

Ce type de problèmes est caractérisé par sa nature fortement combinatoire, ce qui justifie l'inexistence, à ce jour, de méthodes de résolution à la fois générale et de faible complexité algorithmique. Ceci laisse à dire, que l'étude des problèmes d'ordonnancement est d'un intérêt théorique toujours renouvelé pour les chercheurs.

Dans ce travail, nous nous intéressons à un problème d'ordonnancement prédictif cyclique connu dans la littérature sous l'appellation *Cyclic Hoist Scheduling Problem (CHSP)*. Ce type de problème peut être classé dans la catégorie des problèmes d'ateliers de type flow-shop robotisé sans attente et sans interruption [Lopez et Roubellat, 2008].

L'objet de ce chapitre consiste à introduire les notions et les concepts de base qui permettent de décrire les problèmes d'ordonnancement. L'accent est mis sur un cas particulier de lignes

de production dotées d'une ressource de transport, et notamment les lignes de traitement de surface. Notre travail se focalise plus particulièrement sur le problème d'ordonnancement cyclique de telles lignes. Ainsi, nous rappelons les caractéristiques de certains problèmes d'ordonnancement, dans une première partie. Puis, le système physique des lignes de traitement de surface ainsi que leurs systèmes de pilotage sont présentés dans une seconde partie. Enfin, dans une troisième partie, un état de l'art sur le sujet est exposé.

# I. 2 - Problèmes d'ordonnancement

## I. 2. 1 - Définition générale

Vu l'intérêt que présentent les problèmes d'ordonnancement dans tous les secteurs d'activité depuis l'informatique [Blazewicz et al, 1996] jusqu'à l'industrie manufacturière [Pinedo, 1955], de nombreux travaux de recherche se sont intéressés et continuent de s'intéresser à de tels problèmes ([Backer, 1974], [Brucker, 1998], [Carlier et Chretienne, 1988], [Conway et al, 1967], [Lopez et Esquirol, 1999]).

Le principal objectif recherché par une entreprise est d'optimiser l'utilisation de ses ressources afin d'assurer sa pérennité, son développement et sa compétitivité. La satisfaction du client au moindre coût ne peut être atteinte que par la minimisation des dépenses et l'optimisation du coût de production, et ceci à travers une bonne prévision des demandes du client.

La résolution d'un tel problème consiste notamment à ordonnancer, i.e. programmer dans le temps, l'exécution des tâches en leur attribuant les ressources nécessaires matérielles ou humaines de manière à satisfaire un ou plusieurs critères préalablement définis, tout en respectant les contraintes de réalisation [Gotha, 1993].

Un problème d'ordonnancement peut-être décomposé en trois sous-problèmes principaux [Carlier et Chretienne, 1988]:

- ✓ l'affectation : dès lors qu'une tâche peut être réalisée sur plusieurs ressources, il est nécessaire de choisir quelle(s) ressource(s) effectuera cette tâche ;
- ✓ le séquencement : précise pour chaque ressource l'ordre d'exécution des tâches qui lui sont affectées ;
- ✓ la sélection d'une solution parmi les ordonnancements possibles (ou parmi ceux générés), suivant un critère d'évaluation donné : le critère du temps total de fabrication des produits (makespan: $C_{max}$) est le plus couramment utilisé. Toutefois, il existe bien

d'autres critères à optimiser tels que la minimisation des encours ou encore la maximisation du taux d'occupation des ressources ou le temps de cycle.

Selon le problème d'ordonnancement à traiter, l'affectation et/ou le séquencement peuvent être présents ou non et traités consécutivement ou conjointement.

## I. 2. 2 - Eléments d'un problème d'ordonnancement

Parmi les problèmes d'ordonnancement d'atelier, les notions fondamentales rencontrées sont les ordres de fabrication (produits ou jobs), les ressources, les contraintes et les critères d'optimisation.

### a - Les tâches

Chaque ordre de fabrication est caractérisé par :

- ✓ un ensemble de tâches ou opérations à exécuter conformément à une séquence opératoire. Si cette séquence est linéaire, les opérations sont liées par un ordre total (une tâche doit être terminée avant que la suivante dans la séquence ne commence). Les opérations de l'ensemble des ordres de fabrication constituent les éléments à ordonnancer. ;
- ✓ une date de disponibilité (début au plus tôt) et une date de fin au plus tard [Boukef, 2009].

Chaque tâche est caractérisée principalement par un temps d'exécution appelé durée opératoire. Trois types de tâches peuvent être distingués :

- ✓ les tâches morcelables ou interruptibles, qui peuvent être exécutées par morceaux sur plusieurs fois. On parle alors de préemption ;
- ✓ les tâches non morcelables ou indivisibles, qui doivent être exécutées en une seule fois et ne peuvent être interrompues ;
- ✓ les tâches préemptives répétées, qui peuvent être interrompues, mais à la reprise, elles doivent être recommencées depuis le début.

### b - Les ressources

Une ressource est l'ensemble des moyens techniques ou financiers ou humains nécessaires à l'exécution des tâches [Gargouri et al, 1999]. Deux types de ressources sont à distinguer :

- ✓ les ressources renouvelables, qui redeviennent disponibles aussitôt libérées. Dans cette classe de ressources on peut distinguer les ressources disjonctives et les ressources cumulatives. Le premier type ne peut autoriser qu'une tâche à la fois. Par contre le second autorise l'exécution d'une ou plusieurs tâches à la fois mais généralement en nombre limité (machines, véhicules, opérateurs, etc) ;

- ✓ les ressources consommables, qui ne sont plus disponibles une fois allouées à une tâche (budget, matière première, etc) [Mesghouni, 1999].

**c - Les contraintes**

Généralement, un projet peut-être soumis à deux types de contraintes :

- ✓ les contraintes potentielles : représentent des exigences au niveau des variables temporelles d'ordonnancement. Incluant les contraintes de gamme ou de précédence entre les tâches et les dates limites d'ordonnancement à ne pas dépasser dans certains cas (fenêtres de temps, date de début au plus tard, date de fin au plus tard) [Gargouri, 2003] ;
- ✓ les contraintes de ressources : représentent des exigences imposées par les ressources allouées, comme la capacité limitée d'une ressource à accueillir des produits en même temps. On distingue alors la sous-classe des contraintes disjonctives, où une seule ressource est partagée par plusieurs tâches à la fois et la sous classe des contraintes cumulatives, où plusieurs ressources sont partagées par plusieurs tâches [Saad, 2007].

**d - Les critères**

Un critère est une fonction à optimiser traduisant l'objectif à atteindre. La définition de ce critère est essentielle pour l'évaluation et le choix d'une solution parmi les ordonnancements satisfaisant les contraintes auxquelles est soumis notre problème.

Un ou plusieurs critères peuvent être envisagés pour l'évaluation d'une solution. Si l'utilisation de plusieurs critères conduit à l'obtention des mêmes solutions optimales, on dit que ces critères sont équivalents et vis versa [Carlier et Chrétienne, 1988].

Deux types de critères sont à distinguer :

- ✓ les critères réguliers varient d'une façon monotone en fonction des dates de fin d'exécution des tâches. On peut citer à titre d'exemple, le maximum des dates d'achèvement des actions ou encore la moyenne des dates d'achèvement des actions ;
- ✓ les critères irréguliers, contrairement aux précédents, ne sont pas fonction monotone des dates de fin d'exécution des tâches. A titre d'exemple, la minimisation du coût de stockage des matières premières ou l'optimisation des changements d'outillage [Gargouri, 2003].

La satisfaction de plus d'un critère à la fois, peut s'avérer un problème délicat dans la mesure où certains critères peuvent être contradictoires. [Boukef, 2009]

## I. 2. 3 - Les ateliers

Une des classifications, la plus répandue, des problèmes d'ordonnancement se base sur le type d'atelier utilisé. Le type d'un atelier est généralement défini par le type et le nombre de machines qui le constituent ainsi que l'ordre de passage d'un produit sur les machines pendant son processus de fabrication. Une première classification distingue les problèmes dits à une machine de ceux à plusieurs machines.

Dans notre étude, nous nous intéressons aux problèmes d'ordonnancement d'atelier à ressources multiples [Ghedira, 2006]. Quatre types d'ateliers sont à distinguer: flow-shop, job-shop, open-shop auxquels nous ajoutons les ateliers dotés d'une ressource de transport. Ces quatre types d'ateliers seront détaillés par la suite.

### a - Flow-shop

Egalement appelés ateliers à cheminement unique. Dans ce type d'atelier, les différents types de produits à traiter sur la ligne de production possèdent souvent une même séquence opératoire linéaire. Ce type de séquence opératoire, modélisé par la figure I.1, se caractérise par le fait que chaque tâche possède au plus un successeur et au plus un prédécesseur.

Dans ce type d'atelier, deux sortes de productions sont à distinguer : la production continue et la production discrète. La première distingue les raffineries et les papeteries où la production est caractérisée par sa fluidité et où on interdit le stockage. Quant au deuxième type, il est applicable à la production des produits de grande consommation fabriqués à la chaîne comme dans l'industrie automobile.

Dans certains ateliers, des machines dupliquées peuvent exister et fonctionner en parallèle, dans ce cas on parle d'un atelier de type flow-shop hybride.

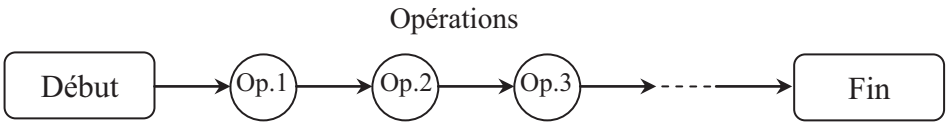



**Figure I.1.** Modèle linaire d'une séquence opératoire

### b - Job-shop

Dans ce type d'atelier, aussi appelés ateliers à cheminement multiples, chaque classe de produits possède sa propre gamme opératoire. Ainsi, les produits peuvent suivre des chemins différents pendant leurs processus de fabrication. Le flow-shop peut-être considéré comme un cas particulier du job-shop. La figure I.2 modélise le cheminement des produits dans un atelier de type job-shop.

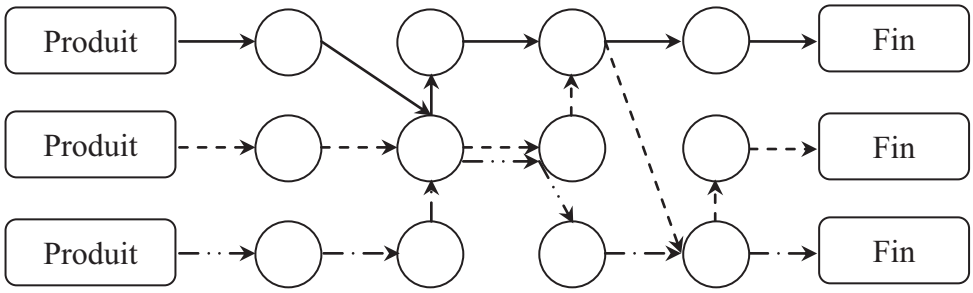



**Figure I.2.** Modèle de trois cheminements de séquences opératoires dans un atelier de type job-shop

Comme extension de cette classe d'ateliers on trouve le problème du job shop flexible, défini comme l'atelier où toute opération élémentaire peut être traitée par un pool de machines polyvalentes avec des durées opératoires différentes ou égales. La flexibilité est dite totale si toutes les machines de l'atelier sont polyvalentes, sinon elle est dite partielle [Kacem, 2003].
La flexibilité dans les problèmes d'ordonnancement engendre l'apparition d'un double problème d'optimisation combinatoire, faisant apparaître, en plus du problème classique de séquencement des opérations sur l'ensemble des ressources, un problème d'affectation des opérations aux ressources [Mesghouni, 1999].

**c - Open-shop**

Dans ce cas, l'ordonnancement, n'est pas aussi contraignant que dans les ateliers précédents. En effet, les gammes opératoires ne sont pas à priori spécifiées. On se contente juste de donner les opérations à réaliser mais aucune indication sur l'ordre des opérations n'est fournie. Ceci augmente par ailleurs la combinatoire du problème. L'ordonnancement consiste alors à fixer dans une première étape la gamme opératoire de chaque produit et puis à ordonner l'ensemble des produits, dans une seconde étape. Contrairement aux autres modèles d'ateliers, l'open-shop n'est pas couramment utilisé dans les productions manufacturières, mais se rencontre dans différents domaines d'application [Boukef, 2009].

**d - Les ateliers dotés d'une ou plusieurs ressources de transport**

Ces ateliers sont contraints par un transport assuré par un ou plusieurs robots de capacité limitée. On distingue trois classes de systèmes :

- ✓ Les lignes de traitement de surface. Le problème d'ordonnancement associé est plus connu dans la littérature sous l'appellation *Hoist Scheduling Problem* (HSP) : dans ce type de système, chaque produit doit subir une succession d'opérations de trempe dans un ensemble de cuves contenant des solutions chimiques. Les produits sont montés sur

des porteurs (ou supports) pour faciliter leur transport qui est assuré par un ou plusieurs robots de manutention. Le HSP présente deux particularités. La première concerne les ressources de transport. En effet, les robots assurant le transport des porteurs sont montés sur un rail unique au dessus des cuves et peuvent entrer en collision. Ainsi, il faut assurer un bon ordonnancement des mouvements de transport des robots afin d'éviter les collisions qui peuvent survenir. La deuxième particularité de ce type de problème, c'est que les durées de trempe (ou encore les durées opératoires) ne sont pas fixes mais plutôt comprises entre une borne minimale et une borne maximale (ou encore fenêtre de temps).

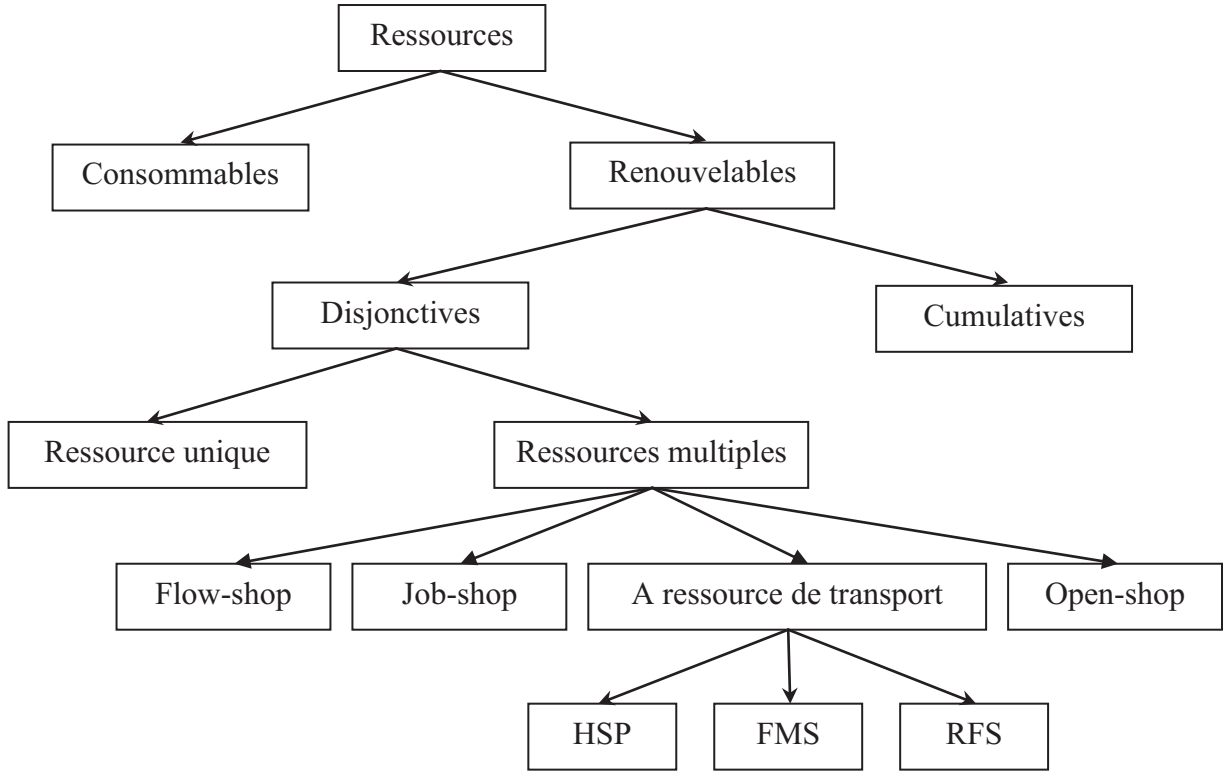
- ✓ Les cellules robotisées (robotic cell) représentent une sous-classe du problème *Single Flexible Machine*. Dans ce type d'atelier le transport des pièces entre les différentes machines de la cellule de production est assuré au moyen d'un seul robot de manutention. Un ou plusieurs espaces de stockage intermédiaires peuvent être situés en entrée ou en sortie de chaque unité de production. Ce problème peut être considéré comme un problème connexe du HSP où on fait appel à une seule ressource de transport pour assurer le déplacement des produits ayant tous la même gamme opératoire [Crama, 1997], [Brauner-Vettier, 2005], [Brauner et al, 2005].

- ✓ Les systèmes de Production Flexible Manufacturière (PFM), aussi appelé *Flexible Manufacturing Systems* (FMS). La flexibilité de ce type d'atelier est due aux gammes opératoires et aux productions de petites et moyennes séries. Cette caractéristique est due également aux machines, capables d'assurer plusieurs types d'opérations à la fois [Dupas, 2004]. Dans ce type d'atelier, les produits sont montées sur des porte-pièces ou encore palettes, et leur transport est assuré par une ou plusieurs ressources de transport (par exemple des véhicules autonomes ou *Automated Guide Vehicle* (AGV)) [Brauner et al, 2005]**.**

Ces différents ateliers se caractérisent notamment par le nombre, le type et la nature des ressources de transport et de d'infrastructure du réseau de transport : par exemple, pour les lignes de traitement de surface, les robots circulent sur un rail unique au dessus des cuves ; dans les FMS, on trouve souvent des réseaux maillés où les ressources de transport doivent partager des tronçons communs avec des sens de déplacement unidirectionnels ou bidirectionnels. Dans tous les cas, le partage d'un même réseau par plusieurs véhicules

implique des risques de collision qu'il faut arbitrer et correspondent à des contraintes additionnelles du problème d'ordonnancement associé [Crama, 1997].

La taxonomie que nous avons élaborée dans la figure I.3 constitue une ébauche de classification des problèmes d'ateliers. D'autres critères peuvent être intégrés dans la classification, tels que le type d'ordonnancement recherché (réactif ou prédictif). Il convient de noter que nous avons conservé la dénomination anglaise, la plus répandue dans le domaine, pour certains types d'ateliers.




HSP : Hoist Scheduling Problem
FMS : Flexible Manufacturing System
RFS : Robotic Flexible System

**Figure I.3.** Taxonomie

## I. 2. 4 - Complexité des problèmes d'ordonnancement

Les problèmes d'ordonnancement d'ateliers sont pour la plupart des problèmes combinatoires (voir définitions en annexe1) extrêmement difficiles, d'où la difficulté de résoudre de tels problèmes moyennant des algorithmes universels efficaces.

Ainsi pour évaluer les performances d'un algorithme, il faut évaluer son coût en termes de temps d'exécution (nombres d'opérations exécutées par l'algorithme) et/ou en taille mémoire

définissant la taille nécessaire pour stocker les différentes structures de données pour l'exécution. Ces deux concepts sont appelés la complexité en temps et en espace de l'algorithme. Cette complexité algorithmique permet de mesurer les performances d'un algorithme et de le comparer avec d'autres algorithmes réalisant les mêmes fonctionnalités.
On distingue deux types de complexité [Garey et Johnson, 1979].

**a - La complexité méthodologique**

Elle exprime une fonction du nombre d'opérations élémentaires de calcul effectuées par la méthode ou par l'algorithme de résolution en fonction du nombre des données du problème traité [Saad, 2007].

**b - La complexité problématique**

Cette notion est liée à la difficulté du problème à résoudre et au nombre d'opérations élémentaires qu'un algorithme déterministe doit effectuer pour trouver l'optimum en fonction de la taille du problème. Selon son degré de complexité, un problème peut appartenir à l'une des quatre classes suivantes [Sakarovitch, 1984].

- ✓ les problèmes les plus difficiles : ce sont les problèmes pour lesquels il n'existe aucune méthode de résolution. Ils sont également dits indécidables ;
- ✓ les problèmes de la classe P : un problème de décision est dit polynomial s'il existe un algorithme de complexité polynômiale pour sa résolution. Mais, ne pas connaître un tel algorithme ne signifie pas forcement qu'il n'existe pas ;
- ✓ les problèmes de la classe NP : ils sont dits NP-difficiles. Ces problèmes ne peuvent pas a priori être résolus en un temps polynomial que par des méthodes approchées (appelées également heuristiques). Au cours de leur exécution, ces algorithmes font des choix dont l'optimalité n'est pas démontrable.

Le problème d'ordonnancement d'un atelier de type flow-shop ainsi que le problème d'ordonnancement de type job-shop sont parmi les problèmes d'optimisation combinatoire les plus difficiles à résoudre. Il en va de même pour les problèmes avec ressources de transport qui sont en majorité des problèmes NP-difficiles. En particulier, il a été prouvé que dans le cas des lignes de traitement de surface, le problème d'ordonnancement associé (HSP) fait partie de la classe des problèmes NP-difficiles, même dans le cas d'un seul type de produit et un seul robot de manutention [Lei et Wang, 1989][b].
Dans la pratique, on se contentera souvent d'utiliser les notations de Landau pour mesurer le coût d'un algorithme [Shoup, 2009].

***Rappel :*** On note $U_n = \mathrm{O}(V_n)$ s'il existe une constante $k_1 > 0$ telle que :

$$\forall\, n \in \mathbb{N},\ U_n \leq k_1 V_n$$

Intuitivement, cette notation traduit une majoration du coût : dire que $U_n = \mathrm{O}(n^2)$, c'est dire que $U_n$ va croître au plus proportionnellement au carré de $n$ (mais sa croissance peut être en fait linéaire). De manière implicite, les informaticiens utilisent cette notation pour mesurer le pire des cas : dire que $U_n = \mathrm{O}(n^2)$ ne veut pas dire que le coût sera toujours quadratique, mais que dans le pire des cas, il le sera.

Pour traduire plus précisément l'idée que deux suites ont le même ordre de grandeur, on introduit une nouvelle notation :

***Notation :*** on note : $U_n = \Theta(V_n)$, lorsqu'il existe deux constantes $k_1 > 0$ et $k_2 > 0$ telles que :

$$\forall n \in \mathbb{N}, k_1 V_n \leq U_n \leq k_2 V_n$$

On peut alors classer les algorithmes suivant leur complexité $c_n$, qui sera qualifiée de :

- logarithmique lorsque $c_n = \Theta(log\ n)$ ;
- linéaire lorsque $c_n = \Theta(n)$ ;
- quasi-linéaire lorsque $c_n = \Theta(n.log\ n)$ ;
- quadratique lorsque $c_n = \Theta(n^2)$ ;
- polynomiale lorsque $c_n = \Theta(n^k)$ avec $k > 0$ ;
- exponentielle lorsqu'il existe $\lambda > 1$ tel que $\lambda^n = \Theta(c_n)$.

Dans la première partie de ce chapitre, nous avons présenté des définitions et des généralités sur les problèmes d'ordonnancement. Dans la suite nous nous concentrons sur une classe particulière de problèmes d'ordonnancement d'atelier : celle associée aux ateliers de traitement de surface.

# I. 3 - Les ateliers de traitement de surface

## I. 3. 1 - Présentation Générale

Au cours d'un processus de montage ou de production, il est souvent nécessaire de modifier l'état surfacique des pièces. Pour cela des opérations de traitement de surface sont assurées en plus de l'usinage et du montage. Ces opérations consistent à immerger des pièces dans une succession de bains chimiques et/ou électrolytiques. Les lignes de production associées sont

généralement constituées d'un ou plusieurs postes d'entrée et de sortie, d'un ensemble de cuves, souvent disposées en ligne, permettant de réaliser les traitements souhaités : attaque acide, rinçage, dépôt de métal (cuivre, argent ou or). Les pièces à traiter sont montées par lots sur des supports appelés porteurs et le transport de ces derniers entre les cuves de traitement est assuré par un système de manutention, composé de palans ou de robots (hoists) (figure I.4).

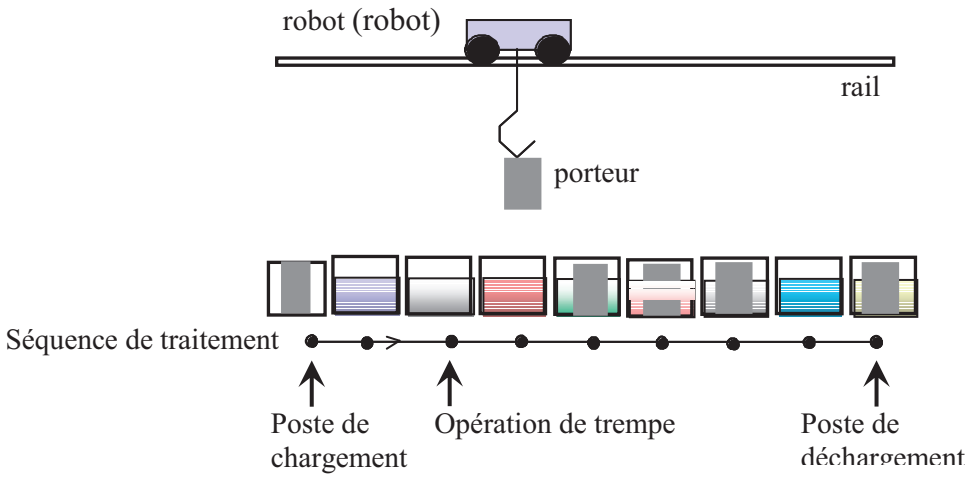



**Figure I.4.** Exemple d'une ligne de traitement de surface [Bloch, 1999]

Les règles de sécurité et d'hygiène sont assurées par des systèmes ne figurant pas sur la figure I.4 tels que les systèmes de ventilation et de lavage de fumées, de traitement des eaux.
Un atelier de traitement de surface comporte deux types de ressources que nous détaillons dans la suite : des ressources de traitement et des ressources de manutention [Manier-Lacoste, 1994] et [Bloch et al, 2008].

## I. 3. 2 - Ressources de traitement

### a - Cuves de traitement

Elles contiennent des bains chimiques et reçoivent les produits à traiter. Ces cuves sont fabriquées dans des matériaux neutres par rapport aux bains et aux produits qu'elles contiennent. Elles sont généralement implantées dans l'atelier suivant une forme d'I, d'U, d'O ou de H (figure I.5) [Manier-Lacoste, 1994]. La configuration linéaire (en I), ou encore dite configuration de base, est la plus étudiée en littérature. Elle est caractérisée par l'absence d'un système de transfert entre les différentes parties de la ligne [Mangione, 2003].

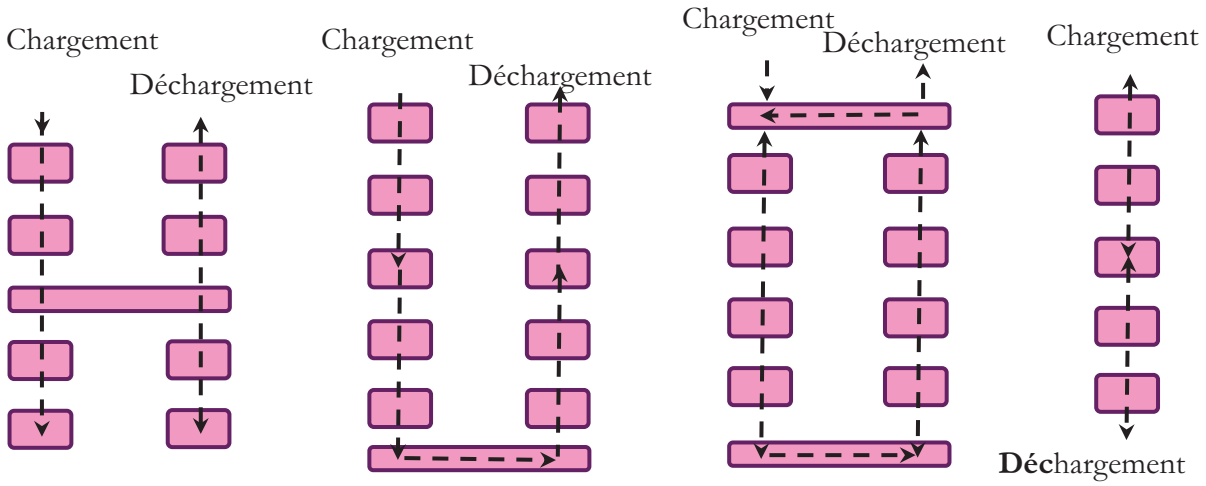



**Figure I.5.** Différents types d'implantations physiques de chaîne

**b - Bains chimiques**

Ils sont les ressources consommables d'une telle production. Ce sont des solutions chimiques utilisées pour l'application d'un dépôt de matière sur les produits. Ce dépôt de matière peut être protecteur comme le chromage ou décoratif comme pour la dorure, l'argenture ou encore le nickelage. La pollution d'un bain est fonction de la nature des bains voisins et de la gamme opératoire puisque chaque produit est susceptible de s'égoutter en cours de transport.

**c - Postes de chargement et de déchargement**

Les postes de chargement sont des cuves fictives d'entrée des pièces brutes sur la ligne de traitement provenant des ateliers en amont tandis que les postes de déchargement sont les postes de sorties des pièces finies. Ces postes de chargement et de déchargement ont la particularité d'avoir une capacité infinie la plupart du temps. Il est possible parfois que l'entrée et la sortie des pièces s'effectuent par le même poste, dans ce cas on parle de postes de chargement et de déchargement associées (figure I.6). Dans le cas contraire, les deux postes sont dits dissociés (figure I.7).

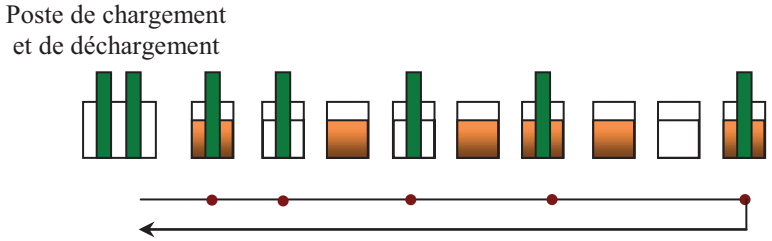



**Figure I.6.** Configuration associée des postes de chargement et de déchargement

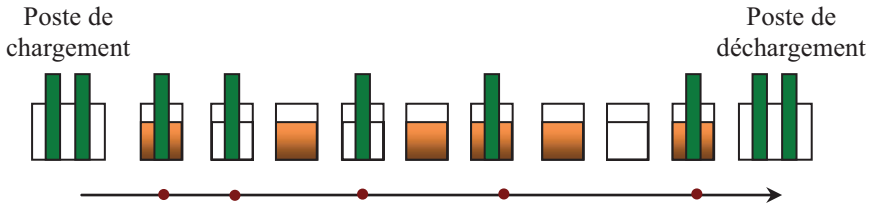



**Figure I.7.** Configuration dissociée des postes de chargement et de déchargement

### d - Opérateurs

Un ou plusieurs opérateurs sont affectés aux postes de chargement et de déchargement quand le processus de gestion de ces postes n'est pas automatisé. Par ailleurs, bien que la présence d'un ou plusieurs opérateurs induise des contraintes de ressources supplémentaires, ils ne sont pas considérés la plupart du temps comme une ressource limitative du système (figure I.8). Toutefois, la disponibilité d'un seul opérateur induit une configuration associée des postes de chargement et de déchargement. Dans ce cas les opérations de chargement et de déchargement ne peuvent pas être effectuées simultanément. En fait, l'hypothèse la plus simple consiste à faire abstraction des contraintes induites par la présence d'opérateurs. Dans ce cas, les opérations de chargement et de déchargement sont assimilées à des opérations fictives de trempe dont les durées opératoires sont considérées fixes ou bornées. Cette hypothèse est la plus couramment utilisée en littérature.

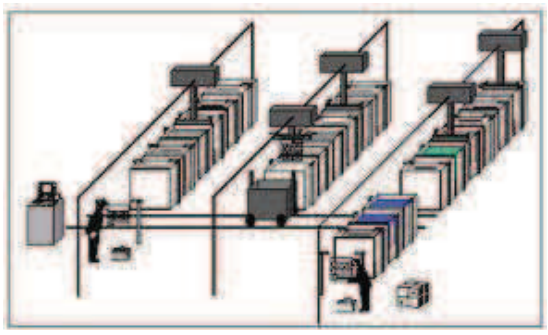


**Figure I.8.** Schéma d'une chaîne complexe faisant intervenir des opérateurs aux postes d'entrée et de sortie [Bloch, 1999]

## I. 3. 3 - Ressources de manutention

### a. Porteurs

Ils se présentent sous différentes formes en fonction de la taille des pièces à produire. On distingue les paniers qui sont utilisés généralement pour les pièces de petites tailles et traitées

en grande quantité ou en vrac (vis, boulons, …) ainsi que les cadres ou les barres sur lesquels les pièces sont vissés ou accroché (cartes de circuits imprimés, …). Le traitement d'une unité est généralement confondu avec celui de son porteur. Dans certains cas, et bien que la gamme opératoire soit fonction du type de produit à traiter. Celle-ci pourrait présenter des opérations spécifiques applicables uniquement au porteur. A titre d'exemple, des opérations de nettoyage pourront être appliquées uniquement au porteur après le déchargement des pièces, comme le montre le tableau I.1. [Manier-Lacoste, 1994]

**Tableau I.1.** Exemple de gamme pour une ligne complexe de traitement de surface

| N° de la cuve | Traitement | Nb. Bacs | Durée de trempe mini (s) | Durée de trempe maxi (s) | Durée d'égouttage |
|---|---|---|---|---|---|
| 1 | chargement | 1 | 90 | ∞ | 0 |
| 6 | dégraissage | 1 | 180 | ∞ | 10 |
| 7 | rinçage | 1 | 30 | ∞ | 0 |
| 8 | rinçage jets | 1 | 30 | 120 | 0 |
| 12 | micro attaque | 1 | 29 | 35 | 10 |
| 13 | rinçage | 1 | 30 | ∞ | 5 |
| 14 | rinçage jets | 1 | 30 | 90 | 0 |
| 20 | activation | 1 | 45 | 90 | 10 |
| 21 | Cu électrolytique | 10 | 2754 | 3000 | 10 |
| 28 | rinçage | 1 | 30 | ∞ | 5 |
| 29 | rinçage jets | 1 | 30 | 90 | 0 |
| 19 | activation | 1 | 45 | 90 | 10 |
| 16 | Sn-Pb électrolytique | 3 | 720 | 900 | 5 |
| 15 | rinçage | 1 | 30 | ∞ | 5 |
| 14 | rinçage jets | 1 | 30 | 120 | 0 |
| 2 | séchage (étuve) | 4 | 1080 | 1200 | 0 |
| 1 | déchargement | | 60 | ∞ | 0 |
| 9 | démétallisation des outillages | 2 | 420 | 600 | 5 |
| 11 | rinçage jets | 1 | 60 | ∞ | 15 |

**b. Robots**

C'est le système de manutention qui assure le transport de chaque porteur d'une cuve à une autre conformément à la gamme opératoire de celui-ci. Le ou les robots utilisés possèdent généralement les mêmes caractéristiques (vitesse, encombrement,..) et se déplacent sur le même rail se trouvant au dessus de la ligne de traitement. Dans la plupart des installations, chaque robot est muni d'un préhenseur unique lui interdisant de transporter simultanément plusieurs porteurs.

Dans l'industrie, on définit le mouvement de transport d'un porteur d'une cuve à une autre par le terme *créneau* [Manier-Lacoste, 1994]. Un créneau peut être décomposé en quatre étapes élémentaires (figure I.9) :

(1) déplacement à vide du robot depuis la cuve au dessus de laquelle il se trouve jusqu'à celle dans laquelle le porteur à transporter termine une opération de trempe.

(2) descente du robot, saisie du porteur ensuite levée et attente pour permettre au porteur de s'égoutter au-dessus de la cuve de traitement.

(3) transport du porteur considéré à la prochaine cuve conformément à la gamme opératoire du produit en cours de traitement.

(4) stabilisation du robot au dessus de la cuve de traitement, ensuite descente, libération du porteur et remontée à vide.

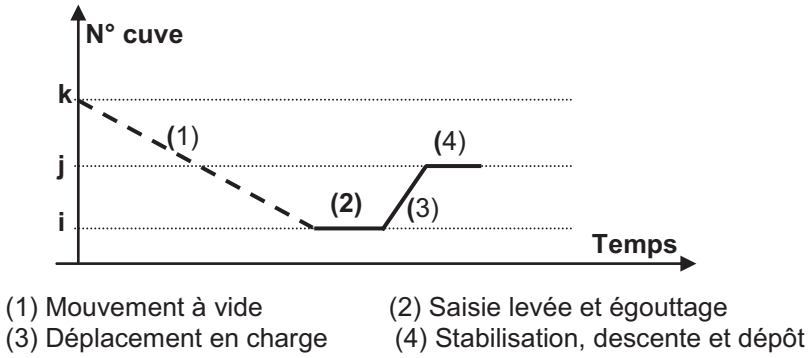



**Figure I.9.** Représentation graphique d'un créneau [Manier et Lamrous, 2008]

La figure I.10 modélise les différents mouvements autorisés pour le robot. Les flèches en continu correspondent au mouvement en charge du robot tandis que les flèches en pointillés correspondent aux déplacements à vide du robot.

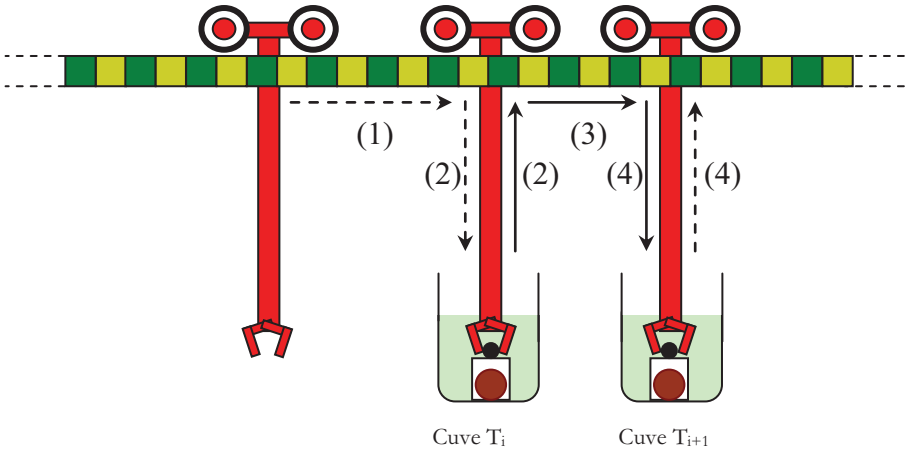



**Figure I.10.** Représentation graphique d'une opération de transport

Généralement, les mouvements de déplacement en charge du robot sont plus lents que les mouvements à vide. De plus, les durées d'immobilisation et d'égouttage sont des données du problème. Elles sont définies par des ingénieurs chimistes et des designers afin de garantir un niveau de qualité acceptable de pureté pour les bains de traitements (en limitant la pollution par les produits passant et s'égouttant au-dessus des cuves).

## I. 3. 4 - Lignes complexes

Dans le cas des lignes les plus simples ou ayant une configuration dite de base, les cuves sont disposées sur une seule ligne et les postes de chargement et de déchargement sont situés aux extrémités. Cette configuration linéaire, dite en I, est la plus étudiée en littérature. Toutefois, pour des raisons d'optimisation, il est intéressant d'étudier d'autres types de lignes avec des cuves de capacité supérieure à un (dites cuves multi-bacs), ou avec des cuves utilisées à plusieurs reprises durant le processus de traitement d'un produit (cuves multi-fonctions). Dans certains cas, les opérations de transport des porteurs peuvent être assurées par plusieurs robots de manutention. Toutes ces configurations correspondent à des lignes de traitement de surface dites complexes [Liu et al, 2002].

### a - Cuves multi-fonctions

Dans une gamme opératoire, une opération de trempe peut-être exécutée plus d'une fois pendant le processus de traitement, à l'exemple de l'opération de rinçage. Ainsi, la cuve destinée à cette opération pourrait-être dupliquée. Toutefois, pour des raisons d'économie de ressources, les deux traitements peuvent se faire dans la même cuve ; dans ce cas, cette dernière est dite multi-fonctions.

L'utilisation de cuves multi-fonctions permet également de réduire l'encombrement de la ligne. Ainsi, on définit une cuve multi-fonctions, comme une cuve dans laquelle plusieurs opérations d'une même gamme opératoire sont réalisées (figure I.11). En raison de la fréquence de visite élevée de ce type de cuves, des goulots d'étranglement sur la ligne de traitement peuvent être engendrés (cuve n°14 du tableau I.2).

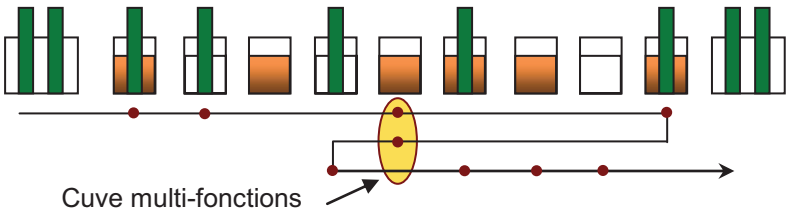



**Figure I.11.** Gamme de traitement utilisant une cuve multi-fonctions

Les opérations de traitement réalisées dans une cuve multi-fonctions ne différent que par leurs durées de trempe ou plus précisément par leurs fenêtres de temps.

Habituellement, deux traitements consécutifs ne sont jamais réalisés dans le même bain ou dans la même cuve : ça n'a pas de sens de sortir un produit d'un bain de traitement pour le remettre dans le même bain juste après quelques instants. Il est à noter que l'utilisation de telles cuves impose aux porteurs de se déplacer dans les deux sens de la ligne.

Dans la littérature, ce type de ressources induit une contrainte dite de recirculation [Pinedo, 1955]. Cette contrainte de recirculation a été introduite en particulier pour l'étude des problèmes d'ordonnancement d'atelier de type job-shop réentrant, où les tâches à réaliser peuvent être traitées plusieurs fois sur la même machine [Gupta et Sivakumar, 2006].

**b - Cuves multi-bacs**

Dans certaines cuves, les durées opératoires sont beaucoup plus longues que pour les autres durées d'immersion. Dans ce cas et afin d'éviter les goulots d'étranglement on utilise des cuves multi-bacs. [Lamothe et Delmas, 1997], [Hanen et Munier, 1993], [Xu et Huang, 2004], [Zhou et Ling, 2003]. Une telle cuve est capable d'accueillir en même temps un certain nombre de porteurs ne dépassant pas le nombre de bacs qu'elle contient. Les bacs d'une même cuve contiennent bien évidemment le même traitement chimique et chaque bac ne peut contenir qu'un porteur à la fois (figure I.12).

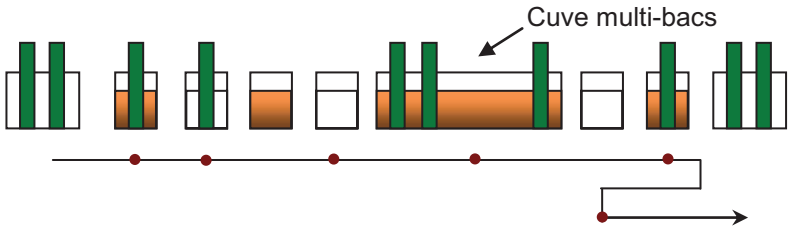



**Figure I.12.** Gamme de traitement utilisant une cuve multi-bacs

Généralement, le nombre de cuves multi-bacs sur une ligne de traitement de surface est assez faible et ne dépasse pas l'ordre 3 (exemple : cuves d'étuve). Toutefois, le nombre de bacs de telles cuves peut atteindre les dix bacs. Le tableau I.1 présente un exemple de gamme utilisant des cuves multi-bacs.

En conclusion, une cuve mono-bac est une cuve de capacité unitaire (ressource disjonctive) alors qu'une cuve multi-bacs est une cuve qui peut accueillir simultanément plusieurs porteurs mais dont le nombre ne doit pas dépasser sa capacité maximale (ressource cumulative).

**c - Ligne multi-robots**

Dès que le nombre de cuves dépasse la dizaine, le robot devient la ressource critique de la ligne. Et afin d'augmenter la productivité, il est possible de disposer plusieurs robots qui vont se partager les tâches (figure I.13). Etant tous disposés sur le même rail, plusieurs problèmes de gestion apparaissent. Ces problèmes se résument dans l'affectation des opérations de transport aux robots et la gestion des risques de collisions. [Lamothe et al, 1996], [Leung et Levner, 2006]

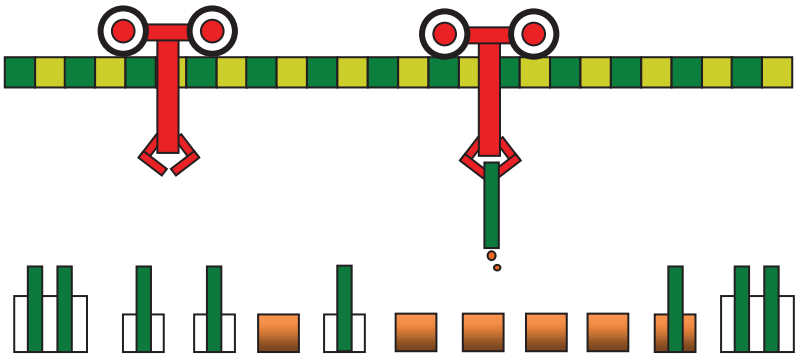


**Figure I.13.** Tronçon de ligne en I muni de deux robots

## I. 3. 5 - Impact sur l'environnement

Sur une ligne de traitement de surface, les cuves contiennent à la fois des bains chimiques ou électrolytiques et de l'eau qui va servir principalement pour le nettoyage des produits et des porteurs, qui seront utilisés par la suite au cours du processus de traitement du reste des produits. Ces opérations de nettoyage sont à l'origine d'une très grande quantité de rejets polluants dans l'eau.

Pour des raisons environnementales, des réglementations strictes ont été mises en place afin d'inciter les entreprises de traitement de surface à mettre en place des dispositifs pour le traitement des rejets en eaux usées. Ces dispositifs se résument dans la création de stations d'épuration servant à diminuer les déchets toxiques aqueux. Durant un processus de production par campagne et pour de très grandes quantités, il arrive que ces stations soient saturées, ce qui peut entrainer l'arrêt immédiat de la ligne de production. Ainsi, il s'avère parfois intéressant de s'intéresser à l'impact environnemental sur la ligne de production [Subaï et al, 2005]. La figure I.14 modélise les entrées/sorties d'une ligne de traitement de surface ainsi que les rejets toxiques émis au cours du processus de production (gaz, eaux polluées, …). Ces rejets polluants sont occasionnés principalement par les opérations d'immersion, d'égouttage et de rinçage. Le choix d'une valeur particulière pour les durées

d'immersion, de rinçage ou d'égouttage n'est pas alors sans conséquence. Il s'avère donc intéressant de trouver un compromis entre ces trois délais afin de maximiser la productivité tout en minimisant l'impact des effluents émis sur l'environnement [Subaï et al, 2005].

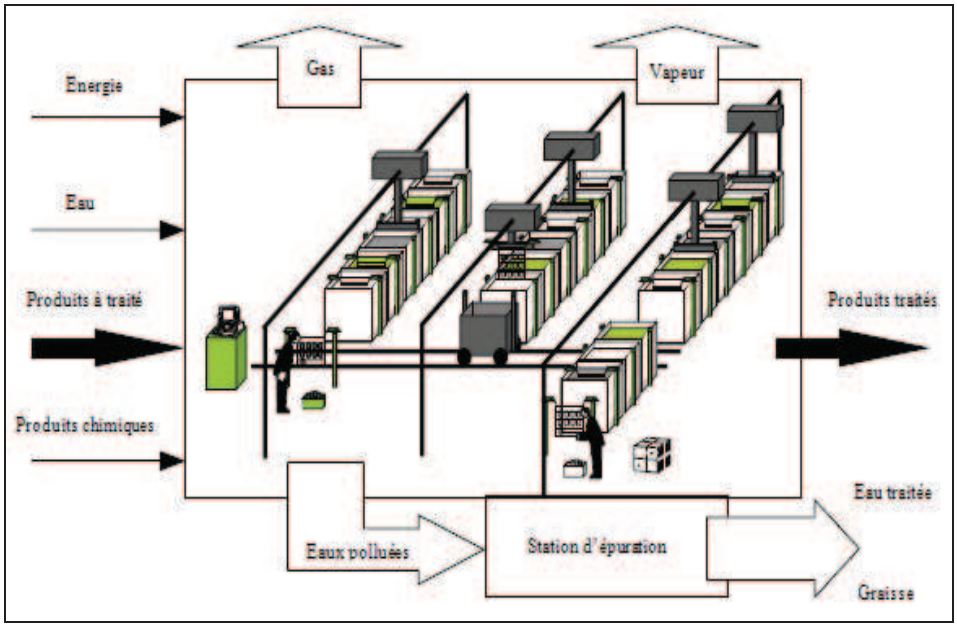



**Figure I.14.** Schéma synoptique des entrées/sorties d'une ligne de traitement de surface [Subaï et al, 2003]

**a - L'immersion**

La durée de cette opération est liée directement à la concentration des bains chimiques. Cette concentration est gérée par les chimistes et fonction de la durée de production souhaitée. Ainsi, plus la concentration des bains est élevée, plus la durée d'immersion est courte et par conséquent, plus la pollution des bains de rinçage en aval des trempes (se trouvant à la station d'épuration) est affectée. Toutefois, cette concentration n'est pas stationnaire mais plutôt variable (décroissante) en fonction du temps pour des raisons d'évaporation, dilution et autres.

**b - Le rinçage**

Cette opération assure le nettoyage des pièces, afin d'arrêter la réaction du produit actif du traitement précédent et diminue la pollution des cuves suivantes. La pollution des eaux de rinçage dépend essentiellement de :

- ✓ la concentration des produits du traitement amont,
- ✓ la durée de rinçage,
- ✓ l'entraînement, défini par la quantité de produit entrainée d'un bac de traitement ou de rinçage au suivant par unité de temps ou de surface. Il est directement lié à l'égouttage

mais dépend d'autres facteurs tels que la viscosité du produit entraîné, le type de porteur, la forme et la nature des pièces, leur nombre, ...

Plus le produit actif en aval est concentré, ou plus la durée de rinçage est importante, plus la quantité dispersée dans la cuve de rinçage est élevée d'où l'augmentation de la quantité de rejets pollués, évacués par vidanges. Néanmoins, une durée de rinçage trop faible risque d'affecter les cuves des traitements suivants ou de détériorer la qualité des pièces traitées.

**c - L'égouttage**

Il permet de diminuer l'effet de l'entraînement des pièces et de leur porteur. Plus le temps d'égouttage est élevé, plus la valeur de l'entraînement est faible. Mais ceci augmente le temps de cycle et peut même affecter la qualité des pièces (des effets d'oxydation peuvent apparaître).

# I. 4 - Ordonnancement des lignes de traitement de surface

## I. 4.1 - Définition et spécificités du HSP

Conformément au paragraphe I.2.2., nous définissons au préalable les éléments constitutifs d'un problème d'ordonnancement d'atelier de traitement de surface. Sur une ligne de traitement de surface, les tâches à ordonnancer sont les opérations de trempe et les opérations de transport des porteurs. Les principales ressources sont les cuves et les robots, telles que définies précédemment. Un tel atelier peut (dans une première approximation) être considéré comme un atelier de type flow-shop ou job-shop avec des contraintes classiques et d'autres spécifiques énoncées ci-après :

- ✓ chaque produit à traiter possède une gamme de traitement spécifiant la liste et l'ordre des cuves dans lesquelles il doit être traité ;
- ✓ la gamme spécifie également les durées de trempe dans chaque cuve, non sous forme d'une durée fixe ou minimale, mais en termes de durées de séjour minimale et maximale ;
- ✓ toutes les pièces entrent sur la ligne de traitement par la même cuve (cuve de chargement) et la quittent par une même cuve (cuve de déchargement) ;
- ✓ les ressources de traitement et de transport sont de capacité limitée (égale à 1 pour les robots) ;
- ✓ pour déplacer un produit d'une cuve à une autre cuve de traitement, la cuve de destination ne devrait pas être chargée à sa capacité maximale. Elle doit contenir au

moins un emplacement ou bac inoccupé. En effet, il n'y a pas de cuve dédiée au stockage intermédiaire sur la ligne qui permettrait de pallier ce problème de capacité ;

- ✓ les mouvements des produits entre les cuves sont assurés par un ou plusieurs robots et les durées de déplacement sont du même ordre de grandeur que les durées de trempe et ne peuvent donc pas être négligées ;
- ✓ pour limiter les risques de détérioration des produits (par oxydation), un robot ne doit pas effectuer de pause pendant un mouvement en charge, en dehors des temps d'égouttage et de stabilisation prédéfinis ;
- ✓ entre deux déplacements en charge effectués successivement par un même robot, ce dernier doit avoir le temps de se déplacer à vide entre la cuve destination du premier transport et la cuve origine du second. Le robot doit donc ainsi arriver à cette dernière cuve à temps pour en retirer le porteur avant que le temps de trempe de celui-ci ne dépasse sa borne maximale. Notons que si les deux transports concernent le même porteur, le temps à vide est nul ; le robot se contente de rester immobile au dessus de la cuve associée jusqu'à ca que l'opération de trempe soit achevée ;
- ✓ si plusieurs robots sont présents sur une ligne de traitement, il faut éviter les conflits occasionnés par un croisement entre les robots ;
- ✓ si les produits sont fixés sur des porteurs durant leurs processus de traitement, le problème de gestion de ces supports peut être considéré.

L'objectif principal de l'automatisation des lignes de traitement de surfaces consiste à maximiser la productivité en utilisant les ressources disponibles (cuves, bains et robots) au maximum de leur capacité. La satisfaction du client au moindre coût se traduit par la production de la quantité de produits nécessaires en garantissant une bonne qualité de ses derniers : la tenue et l'épaisseur des dépôts.

Le problème d'ordonnancement des lignes de traitement de surface, plus connu en littérature sous l'appellation *Hoist Scheduling Problem* (HSP) consiste à trouver un ordre et des dates d'exécution de l'ensemble des opérations, en respectant les contraintes précédentes et en optimisant un ou plusieurs critères (le plus souvent traduisant la maximisation de la productivité). Le processus de traitement s'effectue de manière continue, c'est-à-dire sans interruption et sans stock intermédiaire ; de plus aucune attente en charge n'est autorisée pour le robot. En conséquence, la fin d'une opération de trempe correspond au début d'une opération de transport et vice versa. Il suffit donc d'ordonnancer soit les opérations de trempe, soit les opérations de transport puisque les unes se déduisent des autres. Une description

détaillée des problèmes d'ordonnancement des ateliers de traitement de surface est présentée dans [Bloch et al, 2008].

Le HSP est un problème particulièrement contraint, différent des autres problèmes d'ateliers classiques qui ne considèrent souvent qu'un seul sous-ensemble de ces contraintes. C'est en particulier un problème sans attente et sans stock où les durées opératoires sont bornées par valeurs inférieure et supérieure. Toutefois, plusieurs travaux approchant le problème du HSP ont été menés et on peut citer comme référence les travaux de Hosein et Ross, portant sur la résolution d'un problème de job shop dans un environnement galvanoplastique [Hosein et Ross, 1975]. Dans cette approche, les auteurs ne considèrent qu'un sous ensemble de contraintes relatives aux ressources et négligent complètement les contraintes sur les ressources (risques de collision et respect du temps de déplacement). D'autres chercheurs comme Lieberman [Lieberman et Turksen, 1981], [Lieberman et Turksen, 1982] et Matsuo [Matsuo et al, 1991] se sont intéressés aux problèmes des lignes à ressources de transport partageant le même rail (connu sous le nom du Crane Scheduling Problem) et se sont penchés sur les contraintes de collision entre robots. Toutefois, dans la plupart de ces travaux les auteurs considèrent que les durées opératoires sont fixées et non bornées et que les durées de transport en charge sont négligeables par rapport aux durée opératoires.
Dans le tableau I.2, nous illustrons une comparaison, tirée de [Shapiro, 1985], entre notre problème : le *Hoist Scheduling Problem*, le *Crane Scheduling Problem* [Danganzo, 1989] et les deux problèmes d'ateliers classiques, le *job shop* et le *flow shop*. Notons par ailleurs qu'une analogie a été présentée dans [Manier, 2006], entre le HSP et un problème de tournées sélectives avec fenêtres de temps (noté m-STSPTW).

**Tableau I.2.** Comparaison entre le HSP et des méthodes d'ordonnancement usuelles ou de problèmes similaires [Manier, 94]

| *Paramètre* | *Flow Shop & Job Shop* | *Crane Scheduling* | *Hoist Scheduling Problem (HSP)* |
|---|---|---|---|
| Tâches ou/et Travaux | nombre fini de jobs | nombre fixé d'opérations | nombre de produits ou porteurs très grand (peut être fini), nombre fini d'opérations |
| Machines | nombre fini de machines | | nombre fini de cuves |
| Temps de déplacement entre deux machines (ou deux cuves) | $\approx 0$<br><br>généralement pas d'agent de transport ou négligé vis-à-vis des durées opératoires | $\approx 0$ | $\neq 0$ |
| But | | ordonner les opérations pour éviter les collisions | |
| Temps opératoires | fixes | $\geq$ borne inférieure | inclus dans un intervalle de temps (borne inf sup) |
| Pause (attente) | stock tampon = file d'attente entre deux postes | autorités sur le poste | interdite sur les postes (pas de stock tampon) |
| Dates de début des opérations | | connues à l'avance | retrait d'un porteur d'une cuve, fonction de sa date de dépôt dans cette cuve (fonction de l'ordonnancement) → non connu d'avance |

En plus du Crane Scheduling Problem, il existe d'autres problèmes connexes au HSP, faisant intervenir une ou plusieurs ressources de transport durant leurs processus de production. Et plus particulièrement, on distingue deux classes de problèmes d'ordonnancement liées à d'autres ateliers avec ressources de transport. Il s'agit du *Robotic Scheduling Problem* et du *Flexible Manufacturing System Scheduling Problem*. La première classe est issue des cellules de production automatisées où plusieurs machines sont servies par un seul robot qui réalise les opérations de transport des pièces entre les machines depuis un point d'entrée jusqu'à un point de sortie [Crama et al, 2000]. Le Robotic Flow Shop (RFS) peut être vu comme un cas particulier du HSP dans lequel il y a un seul robot et tous les jobs ont le même cheminement

sur les machines. Un objectif classique consiste à trouver une séquence de déplacements du robot de manière à maximiser le débit pour un nombre infini de pièces entrantes. Ce problème est dans le cas général NP-difficile. Quant à la deuxième classe, liée aux ateliers de Production Manufacturière Flexible, elle se caractérise d'une part, par une certaine souplesse relative aux gammes de production et concerne d'autre part, des productions de petites et moyennes séries [Gentina et al, 2001]. Ce type de production est organisé autour d'un système de transport pouvant véhiculer plusieurs types de pièces sur un porte-pièce dédié aussi appelé palette, reliant des machines capables de réaliser plusieurs types d'opérations. La notion d'encours, caractéristique des systèmes de production, qui est assimilable sous certaines hypothèses simplificatrices au nombre de palettes dans le système, revêt une importance particulière dans ce type de problématique. Il s'agit d'un problème complexe composé lui-même de plusieurs problèmes NPdifficile. Toutefois par rapport au HSP, les problèmes relatifs à ces ateliers ne considèrent que des durées opératoires fixes ou minimales, autorisant des pauses en charge des robots (car pas de contraintes chimiques), etc.

Il existe divers types de configurations possibles des problèmes d'ordonnancement des lignes de traitements de surface. En effet, le mode de pilotage de ce type de lignes est fortement lié au type de production à exécuter. Cette production dépend entre autres du nombre de produits à traiter et du temps disponible entre la commande et la production. On distingue deux grandes classes de problème que nous détaillons dans les deux paragraphes suivants [Bloch et al, 1996][b], [Bloch, 1999].

### I. 4. 2 - Problèmes prédictifs

Ils regroupent les problèmes d'ordonnancement cycliques et les autres problèmes, prédictifs mais non cycliques. Généralement, on fait appel au premier type de pilotage principalement dans le cas d'une production de masse. Dans ce cas, les robots sont programmés pour répéter une séquence de mouvement fixée autant de fois que nécessaire pour exécuter la totalité de la production. Cette séquence de mouvements est appelée : *cycle*.

A chaque début de cycle, l'état du système reste inchangé, entre autres, la position des robots, le nombre et l'emplacement des porteurs ainsi que les temps écoulés dans chaque cuve (durées opératoires). Le temps total nécessaire pour l'exécution d'une telle séquence est appelée *longueur de cycle* ou *période*. La figure I.15 représente l'exécution par un robot de deux séquences de mouvements successives, dans le cas d'une gamme à trois traitements.

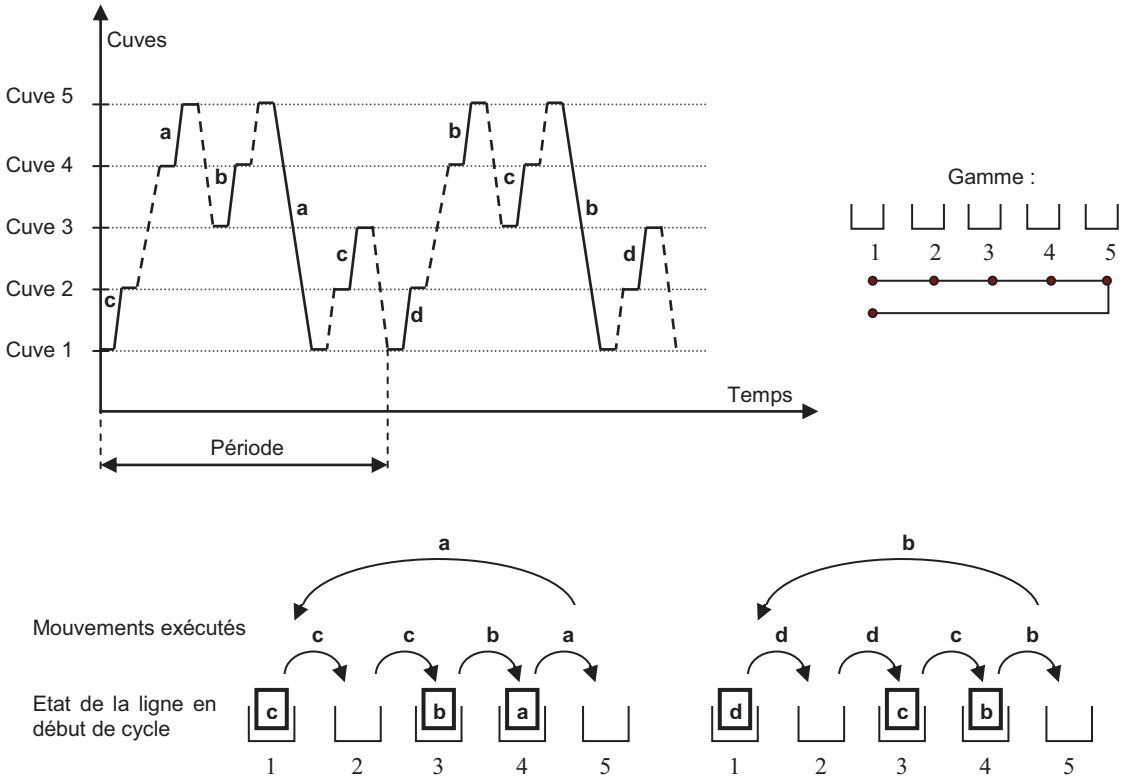



**Figure I.15.** Commande cyclique d'une gamme comportant 3 traitements

Dans le contexte multi-produits, ce type de commande peut s'avérer avantageux dans le cas où les produits traités ont la même gamme opératoire ou que leurs gammes ne différent que par la durée d'un ou deux traitements chimiques. C'est le cas par exemple des produits de même type mais de tailles différentes. Dans ce cas la surface traitée est réduite et par conséquent la durée opératoire d'immersion est diminuée.

La commande cyclique du problème d'ordonnancement des lignes de traitement de surface est connue dans la littérature sous l'appellation du *CHSP* abréviation du *Cyclic Hoist Scheduling Problem*. La solution recherchée pour un tel problème est une séquence faisable et répétitive des mouvements du robot. Généralement, le critère d'optimisation considéré pour une telle séquence est la minimisation de la durée du cycle ce qui revient également à la maximisation de la production.

Pour un tel type d'ordonnancement on définit un nombre *n,* ou *degré du cycle*. Ce nombre correspond au nombre de produits entrant et quittant le cycle durant une période.

Un autre type de problème prédictif, mais non cyclique, a été présenté par Varnier [Varnier, 1996]. Il vise à limiter la perte de productivité générée par un changement de cycle induit par une nouvelle production. Ce changement de gammes opératoires entraine un nouveau problème d'optimisation lié à la période transitoire entre plusieurs productions mono-produit.

Le critère à minimiser dans ce cas est la durée de la phase transitoire. Pour résoudre le problème, l'auteur fait appel à la Programmation Logique sous Contraintes et propose une procédure de recherche par séparation et évaluation permettant d'arbitrer les contraintes disjonctives liées au partage des ressources et limitées par la conservation des ordres partiels des cycles, données du problème.

Pour remédier au problème transitoire, dans le cas d'une telle production, une solution faisable, mais non optimisée, consiste à vider complètement la chaîne avant de lancer un nouveau type de produits. [Ghedira, 2006].

### I. 4. 3 - Problèmes dynamiques

On fait appel à un tel type de production, quand la ligne est dédiée à traiter simultanément plusieurs types de produits sur un horizon relativement court. Dans ce cas, les responsables doivent disposer d'un système de pilotage flexible. Deux types de situations sont alors à considérer : le fonctionnement normal de la ligne et sa perturbation par l'arrivée d'un lot prioritaire. Concernant le premier cas, les gammes opératoires des produits à traiter ainsi que l'ordre de passage des produits sur la ligne et les ratios des pièces à traiter sont connus à l'avance ; en plus la qualité des traitements est l'objectif primordial. Par conséquent, dès l'arrivée d'un nouveau porteur en entrée de la ligne, un ordonnancement des opérations de transport associées doit être déterminé, de sorte que toutes les contraintes soient vérifiées (aucune relaxation de contrainte n'est autorisée). Généralement, le lancement de ce porteur est réalisé le plus tôt possible, et le critère à minimiser est la date de fin de tous les travaux. On parle alors de HSP dynamique [Lamothe, 1996]. Quant au second cas, tout est imprévisible ; n'importe quel type de produit peut entrer sur la ligne à tout instant. L'aspect réactif de la commande du système lui permet de traiter en priorité et dès que possible la production de certains produits considérés comme prioritaires. Ainsi le système de pilotage utilisé doit être capable de s'adapter à tout changement de gamme. Dans ce cas, la priorité est donnée aux objectifs de productivité et de flexibilité (en termes de diversité des travaux traités). L'affectation des tâches de transport aux robots s'effectue plutôt en temps réel, ce qui pourrait entrainer la violation de la contrainte de fenêtre de temps et donc une qualité moindre de certains produits. On parle de *HSP réactif* ou *temps réel* [Bloch et al, 1997]. La principale caractéristique qui différencie ces deux classes de problèmes est l'horizon de décision considéré. En effet, le DHSP correspond à une gestion par travail alors que le RHSP relève d'une gestion à la tâche [Bloch et al, 2008].

### I. 4. 4 - Autres problèmes traités

Des chercheurs ont étudié des problèmes de conception de lignes (implantation ou dimensionnement des ressources), en plus du problème d'ordonnancement. Ainsi, certains se sont intéressés à la recherche de la disposition optimale des cuves ([Grunder et al, 1997], [Fleury et al, 2001]), ou encore à la recherche du nombre de bacs nécessaires pour améliorer la production et remédier aux problèmes de goulots d'étranglement [Liu et al, 2002]. Egalement, [Manier et Lamrous, 2006] cherchent à minimiser de la durée du cycle, tout en modifiant le design de la ligne de production considérée et ceci en rajoutant des ressources de transport.

## I. 5 - Notation

Afin de définir un cadre de description systématique permettant de spécifier précisément le problème traité et de le situer parmi les différentes classes de HSP, une notation ainsi qu'une typologie ont été proposées par Manier et Bloch dans [Manier et Bloch, 2003].
La notation proposée repose sur quatre champs, faisant apparaitre quatre facteurs principaux :

Le type de HSP |les paramètres physiques |les paramètres logiques |le critère à optimiser

$$\alpha \mid \beta \mid \delta \mid \gamma$$

- ✓ le premier champ $\alpha$ = XHSP correspond au type de problème étudié : cyclique, prédictif, dynamique ou encore réactif ;
- ✓ Le second champ $\beta$ spécifie certaines caractéristiques du système physique à travers onze paramètres : nombre de lignes, type, nombre et spécificités des ressources de traitement et de transport ;
- ✓ Le troisième champ $\delta$ traduit les caractéristiques du système logique (liées aux produits, donc aux gammes opératoires) ;
- ✓ Le quatrième champ $\gamma \in \{C_{max}, T_{min}, \ldots\}$ correspond aux critères à optimiser ; en d'autres termes, l'objectif à atteindre. Pour le HSP, cet objectif peut être :
  - Minimisation la période d'un cycle (T ou $T_{min}$) pour le cas d'un ordonnancement périodique,
  - Minimisation la date de fin de production ($C_{max}$),
  - Minimisation des encours,
  - Maximisation de l'utilisation des ressources,

- Minimisation du nombre de robots utilisés sur une ligne et un horizon donné.

Dans le cas d'une approche multi-critères, le champ $\gamma$ peut contenir autant de paramètres que de critères à optimiser.

La notation complète prend la forme suivante :

$$\text{XHSP} \mid \text{nl, ntransfer, synchro, } (\text{mh, mt, ct})_{\text{i=1 to nl}} \text{ / nc, circ, ret, empty / load-unload } \mid \text{nparts / nps, nop, clean, recrc } \mid \text{criterion}$$

Une description plus détaillée de cette notation dédiée aux problèmes d'ordonnancement des ateliers de traitement de surface est présentée dans l'annexe 2 de ce mémoire.

Cette typologie a été présentée sous forme arborescente dans [Manier et Bloch, 2003], et chaque type de problèmes XHSP est arboré sur l'une des figures de l'annexe 3. Chaque feuille d'un arbre correspond à une instanciation d'un sous-ensemble des champs de la notation associée. Cette décomposition arborescente nous a permis de distinguer entre les cas les plus traités en littérature et ceux encore peu explorés jusqu'à ce jour. Ainsi, l'ordonnancement 1-cyclique mono-produit pour le problème d'ordonnancement des lignes de traitement de surface est l'ordonnancement le plus étudié en littérature. L'arborescente suivante recense une très grande partie des travaux portant sur le CHSP en utilisant la classification proposée par Manier et Bloch en 2003 et dédiée au problème du HSP. Toutefois, quelques chercheurs (grisé dans la figure I.16) se sont intéressés à l'étude du problème cyclique multi-produits, objet de notre étude. Dans un tel problème, une ligne de production peut traiter simultanément deux ou plusieurs produits sans nécessiter de vider au préalable la ligne de traitement. Une partie de ces travaux, portant plus précisément sur le problème CHSP, sera exposée et détaillée dans la section état de l'art (section I.6).

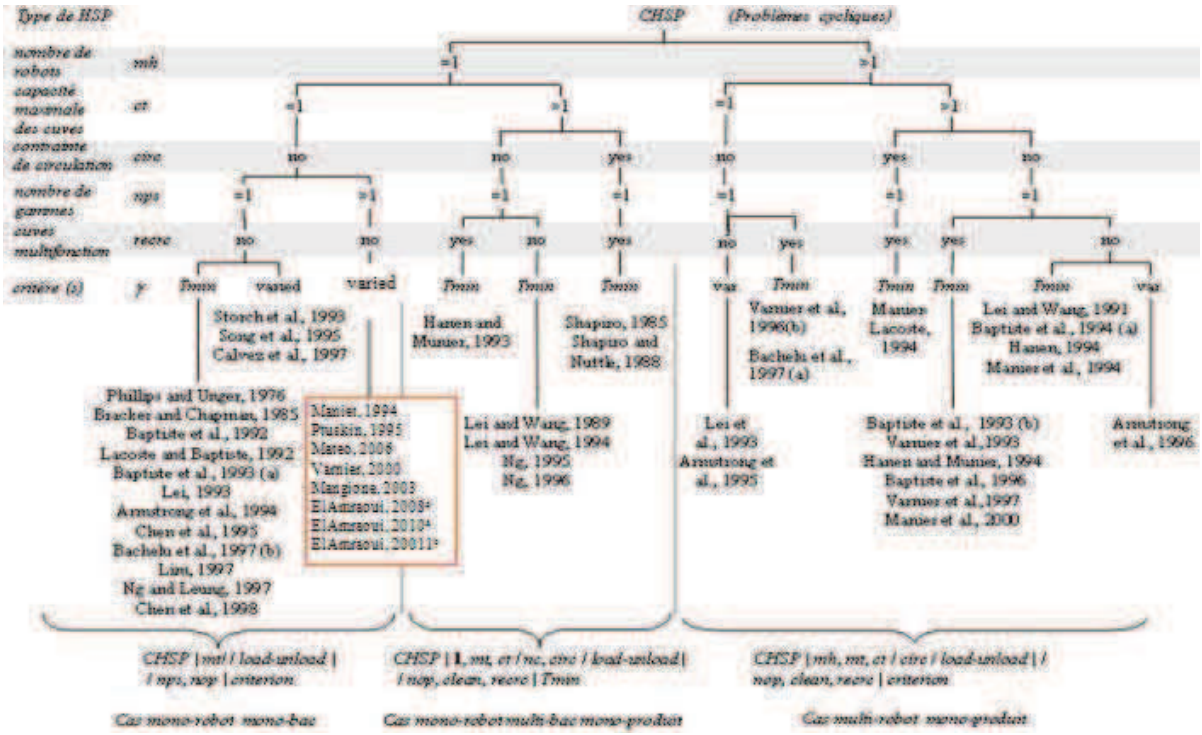



**Figure I.16.** Typologie du cas cyclique

# I. 6 - Etat de l'art sur le "Cyclic Hoist Scheduling Problem"

## I. 6. 1 - Synthèse de la recherche en HSP

Depuis une quarantaine d'années et plus particulièrement depuis l'article fondateur de Philips et Unger [Phillips et Unger, 1976] de très nombreux travaux de recherche se sont intéressés à la gestion des lignes automatisées de traitement de surface. La richesse de la littérature dédiée aux problèmes de HSP ainsi que la diversité des cas traités a suscité un classement de ces divers travaux de recherche. C'est pour cette raison que Manier et Bloch [Manier et Bloch, 2003] ont proposé à la fois, une adaptation et une extension de la notation usuelle utilisée pour caractériser les problèmes d'ordonnancement d'atelier [Graham et al, 1979].

Le tableau I.3 fournit une synthèse des principales publications. Ce tableau distingue les cas les plus traités, notamment les problèmes cycliques mono-produit, des thèmes qui n'ont que peu ou même pas, été abordés en littérature. La majorité de ces travaux traitent le cas cyclique mono-gamme pour une configuration de ligne dite simple (en I, mono-robot et avec cuves mono-bac). Une classification de la majorité de ces travaux en utilisant la notation proposée par Manier et Bloch est en annexe 4.

**Tableau I.3.** Publications relatives aux différentes classes du HSP (Complété à partir de [Bloch et al, 2008])

| | |
|---|---|
| **CHSP** | **Mono-robot, mono-bac et mono-gamme :** [Phillips et Unger, 1976], [Lei et al, 1993], [Lei, 1993], [Armstrong et al, 1994], [Song et al, 1995], [Ng et Leung, 1997], [Lim, 1997], [Chen et al, 1998], [Kats et Levner, 1997], [Storch et al, 1993], [Chen et Su, 1994], [Chen et al, 1994], [Levner et al, 1995], [Calvez et al, 1997], [Backer, 1974], [Bracker et Chapman, 1985]. |
| | **Mono-robot, mono-bac et multi-gammes :** [El Amraoui et al, 2008][a], [Ptuskin, 1995],[Varnier et Jeunehomme, 2000], [Mateo et Companys, 2006], [Manier-Lacoste, 1994], [Mangione, 2003], [Mateo, 2001], [Mateo et al, 2000]. |
| | **Mono-robot, multi-bacs et mono-gamme :** [Shapiro et Nuttle, 1988], [Lei et Wang, 1989][a], [Hanen et Munier, 1993], [Lei et Wang, 1994], [Liu et al, 2002], [Zhou et Ling, 2003], [Xu et Huang, 2004], [Kuntay et al, 2006], [Ng, 1995], [El Amraoui et al, 2009][a]. |
| | **Multi-robots, mono-bac et mono-gamme :** [Lei et Wang, 1991], [Lei et al, 1993], [Yang et al, 2001], [Manier et Lamrous, 2008], [Zhou et Liu, 2008], [Zhou et Li, 2009]. |
| | **Multi-robots, multi-bacs et mono-gamme :** [Lei, 1993], [Hanen, 1994], [Manier et Baptiste, 1994], [Armstrong et al, 1996], [Baptiste et al, 1996], [Riera et Yorke-Smith, 2002], [Varnier et al, 1997], [Manier et al, 2000], [Zhou et Li, 2009]. |
| **PHSP** | **Ordonnancement de phases transitoires :** [Varnier, 1996]. |
| | **Ordonnancement des travaux en entrée de ligne :** [Caux et Pierreval, 1997], [Caux et al, 1995][a], [Caux, 1993], [Caux et al, 1992], [Fleury, 1995], [Fleury et al, 2001] , [Fleury, 1993], [Lacomme, 1998], [Boch et al, 1999], [Bloch, 1999]. |
| **DHSP** | **Mono-robot et mono-bac :** [Yih, 1994], [Cheng et Smith, 1995], [Ge et Yih, 1995], [Fargier et Lamothe, 2001], [Hindi et Fleszar, 2004], [Paul et al, 2007], [Lamothe et al, 1994], [Lamothe et al, 1995], [Bloch et al, 1996][a], [Bloch et al, 1996][b]. |
| | **Mono-robot et multi-bacs :** [Spacek et al, 1999]. |
| | **Multi-robots :** [Lamothe, 1996]. |
| **RHSP** | **Mono-robot et mono-bac :** [Yih et al, 1993]. |
| | **Multi-robots :** [Thesen et Lei, 1990, Jégou et al, 2006]. |

Le problème d'ordonnancement cyclique a été très largement étudié vu l'intérêt qu'il présente en réduisant considérablement l'aspect combinatoire du problème par rapport au problème global. En effet, en faisant appel à la commande cyclique, on n'est plus obligé d'ordonnancer la totalité des tâches mais plutôt, on se limite à l'étude d'une séquence finie de tâches. Cette séquence est appelée cycle et sera répétée autant de fois que nécessaire afin de satisfaire la demande initiale.

Vu que nos travaux concernent la recherche d'un ordonnancement cyclique multi-produits et vu le nombre élevé des publications portant sur le HSP, nous nous limiterons dans notre état de l'art aux publications portant sur les problèmes cycliques. Pour plus de détails, le lecteur peut se reporter à [Manier et Bloch, 2003] ou [Bloch et al, 2008] ou [Ji et Wan, 2001].
Dans [Manier et Bloch, 2003], Bloch et Manier ont défini une typologie caractérisant l'ensemble des thématiques propres au HSP et elles ont recensé les différents travaux par rapport à cette typologie. En se référant à cette typologie, on peut placer une première partie de nos travaux dans la branche mono-robot, mono-bac et multi-gammes et puis une seconde partie dans la branche mono-robot, multi-bacs et multi-gammes.

## I. 6. 2 - CHSP mono-gamme

Dans ce paragraphe, nous présentons quelques approches de modélisation et de résolution ayant été proposées dans la littérature pour la résolution du problème CHSP mono-gamme (mono-produit). Nous les distinguons en utilisant la classification mentionnée dans le tableau I.3.

### a. Lignes mono-robot et mono-bac

Philips et Unger [Phillips et Unger, 1976] sont les premiers à s'être intéressés à la recherche d'une solution optimale cyclique, en 1976. Les auteurs ont traité un exemple industriel à 13 cuves, dont les postes de chargement et de déchargement sont associés. Depuis, ce cas est considéré comme un exemple de référence. Pour résoudre le problème 1-cyclique, les auteurs font appel à la programmation entière mixte et à une technique de type séparation et évaluation. Les variables du modèle sont : les dates de retrait des porteurs des cuves (où $t_i$ est la date de retrait du porteur de la cuve $i$), le temps de cycle à optimiser $T$ et des variables booléennes $y_{i,j}$ définissant l'ordre des mouvements de transport du robot. Toutefois, les auteurs n'atteignent pas la solution optimale vu qu'ils fixent au préalable le nombre de porteurs pendant un cycle. Ils ont proposé également une extension de leur modèle pour traiter le cas des cuves multi-fonctions.

Pour résoudre le même problème, Bracker et Chapman [Bracker et Chapman, 1985] proposent un algorithme pour générer l'ensemble des ordonnancements possibles pour le robot et utilisent un algorithme d'optimisation de type simplexe pour retrouver la solution optimale. Pour illustrer leur méthode, les auteurs traitent le cas d'une ligne simple à six cuves.

Armstrong, Lei et Wang se sont fixés comme objectif de recherche, l'amélioration de l'efficacité des travaux antérieurs, plus précisément, ils se sont orientés vers l'amélioration des règles d'évaluation utilisées dans les recherches arborescentes [Armstrong et al, 1994], [Armstrong et al, 1992]. Ils ont repris les méthodes de recherche arborescente par séparation et évaluation de Shapiro et Nuttle, mais au lieu de résoudre à chaque nœud un problème linéaire, ils utilisent un paramètre appelé *Minimum Time Span* (MTS). Ce dernier permet non seulement de fournir une borne inférieure du temps nécessaire pour accomplir une séquence donnée de mouvements, mais aussi de déterminer des cycles non-optimaux.

Lim applique une approche métaheuristique pour la résolution d'un problème d'ordonnancement de lignes de traitement de surface dans un atelier de circuits imprimés [Lim, 1997]. Cette approche consiste à faire appel aux algorithmes génétiques comme méthode de résolution. Pour ce faire, l'auteur propose un codage direct des mouvements de transport en charge du robot. La première population est générée d'une façon aléatoire. Les populations suivantes sont construites en appliquant une fonction croisement de type LOX (Linear Order Crossover) et une fonction mutation classique : échange aléatoire de l'ordre d'exécution de deux mouvements de transport dans une séquence. Les meilleurs individus générés sont sélectionnés pour créer la nouvelle population (sélection élitiste). Cette approche a été appliquée sur le benchmark de référence de Philips et Unger [Phillips et Unger, 1976]. En faisant varier la taille des populations ainsi que le nombre de générations, l'algorithme a montré une convergence plus rapide vers la solution optimale que dans le cas de Philips et Unger [Phillips et Unger, 1976]. Toutefois, en considérant un processus de génération aléatoire et des opérateurs non dédiés, et sachant que la complexité du problème traité réduit considérablement le nombre de solutions faisables, on peut s'étonner que la solution optimale soit effectivement atteinte.

Chen, Chu et Proth [Chen et al, 1998] utilisent une hybridation de deux algorithmes de recherche arborescente de type séparation et évaluation. Au départ, les auteurs définissent les distributions initiales de porteurs dans les cuves, en utilisant un paramètre $K_{max}$ : défini comme le nombre maximum de porteurs présents sur la ligne pendant un cycle. Le premier algorithme vise à rechercher les ordonnancements possibles des mouvements du robot, en fonction des distributions initiales. Quant au second algorithme, il sert à rechercher les ordonnancements optimaux. La taille du premier arbre de recherche est limitée par le paramètre $K_{max}$. Cette méthode s'avère très intéressante au vu des résultats présentés.

**b. Lignes mono-robot et multi-bacs**

Shapiro et Nuttle [Shapiro, 1985], [Shapiro et Nuttle, 1988] ont repris le problème de référence de Phillips et Unger, à la seule différence qu'ils ont considéré que les postes de chargement et de déchargement étaient séparés. Ils sont également les premiers à avoir traité le cas des lignes à cuves multi-bacs. Dans une première étape, ces auteurs utilisent une nouvelle notion de vecteur définissant les ordres des opérations de retrait des porteurs. La solution optimale est ensuite recherchée par une méthode de type séparation et évaluation. Ces auteurs sont les premiers à avoir défini la notion de cuves multi-bacs (duplicated tank) et la notion de circulation des porteurs. Cette dernière notion est traduite par le fait que lorsque le nombre de porteurs présents sur la ligne de traitement est constant et limité, chaque fois qu'un porteur sort de la ligne, il est immédiatement rechargé et réintroduit au cycle suivant.
Les auteurs ont réussi à atteindre l'optimum pour tous les benchmarks qu'ils ont traités. Toutefois, le temps de simulation reste assez élevé. Ainsi, cette approche semble ne pas être adaptée aux problèmes de grandes tailles.

Lei et Wang [Lei et Wang, 1989][a] se sont intéressés au problème n-périodique pour des lignes simples avec des postes de chargement et de déchargement dissociés. Pour la recherche de solutions, les auteurs ont défini une fonction ordonnancement $\Phi$, où $\Phi(i,j) = k$ signifie qu'un numéro d'ordre, $k$, a été affecté à l'opération de retrait du porteur $j$ de la cuve $i$. Pour résoudre le problème en globalité, les auteurs sont contraints de coupler l'énumération des ordonnancements satisfaisant les contraintes de la fonction $\Phi$ à une méthode de recherche de type séparation et évaluation. Cette approche a été complétée en considérant les cuves-multi-bacs ainsi qu'une configuration associée des postes de chargement et de déchargement. Ces auteurs ont réussi à prouver que le problème cyclique est NP-complet, même dans le cas d'une ligne simple [Lei et Wang, 1989][b].

Zhou et Ling [Zhou et Ling, 2003] ont proposé une modification du modèle de Philips et Unger [Phillips et Unger, 1976] pour remédier au problème des goulots d'étranglement. En effet, certaines durées de trempe peuvent être très importantes par rapport à d'autres, ce qui peut créer une faiblesse au niveau de la cadence du flux de production. Le principe de cette approche consiste à remplacer la cuve du goulot d'étranglement par une cuve multi-bacs ou par plusieurs cuves. Une telle modification au niveau de la ligne nécessite la prise en compte des nouveaux temps de trempe qui sont différents du rapport de la durée opératoire par le nombre de cuves utilisées à ce poste. Ainsi, les auteurs modifient les bornes de fenêtres de

temps (pour ces cuves, ils utilisent une moyenne obtenue en divisant les valeurs de ces bornes par le nombre de bacs). Notons que l'utilisation de *m* bacs au niveau d'une cuve de traitement traduit le fait que celle-ci traite un produit tous les *m* cycles.

**c. Lignes multi-robots et mono-bac**

Dans le cadre du fonctionnement de lignes à deux robots, Lei a proposé une approche heuristique dans [Lei et Wang, 1991], qu'elle a baptisée *Minimum Common Cycle* (MCC) ou "plus petit cycle commun". Cette méthode a été appliquée pour la recherche de la solution 1-cyclique optimale. Cette heuristique recherche une partition optimale de la ligne en deux zones de cuves, une cuve limitrophe aux deux zones servant de cuve pivot. Toutes les cuves d'une même zone sont affectées à un même robot, et la recherche de la solution consiste en la résolution de deux sous-problèmes pour chaque robot. L'heuristique permet de trouver le plus petit cycle commun à ces deux problèmes. Cette approche impose à la gamme un ordre bien spécifique des opérations, suivant le sens d'implantation des cuves.

Manier et Lamrous proposent une approche évolutionnaire pour résoudre à la fois un problème de dimensionnement des ressources de transport et d'ordonnancement cyclique mono-produit (CHSP) [Manier et Lamrous, 2006]. Les auteurs présentent un modèle étendu pour traiter le problème multi-robots. Ils considèrent les mouvements de transport à vide au lieu des mouvements en charge pour coder la séquence cyclique recherchée. Un même formalisme est alors obtenu pour représenter des solutions à 1, 2…ou *mh* robots. Dans leur approche, les auteurs gèrent les mouvements de transport des différents robots présents sur la ligne et les contraintes établies permettent d'aboutir à une solution faisable en faisant des restrictions sur les conflits de collisions pouvant être occasionnés.

**d. Lignes multi-robots et multi-bacs**

Baptiste, Manier et Varnier utilisent la Programmation Logique sous Contraintes (PLC) pour aborder ce problème [Varnier, 1996], [Manier, 94], [Baptiste et al, 1995]. L'intérêt d'une telle méthode est de permettre une formalisation simple des problèmes complexes, tels que le HSP cyclique, qui intègrent des contraintes nombreuses et de natures diverses, notamment de type partage des ressources qui induisent des disjonctions de fenêtres temporelles. En incluant un module de propagation des contraintes, ainsi qu'un module de retour arrière, la programmation logique sous contraintes facilite nettement l'implémentation des méthodes de recherche arborescente. Le moteur développé détermine progressivement l'ordonnancement en arbitrant les contraintes disjonctives des mouvements du robot. Il calcule les dates de début

$t_i$ des opérations à effectuer, ainsi que du temps de cycle $T_{cycle}$ que l'on cherche à minimiser. L'inconvénient est le temps de calcul qui s'est avéré élevé. Pour remédier à ce problème, les auteurs ont proposé une amélioration de la recherche arborescente [Baptiste et al, 1994]. Cette amélioration consiste à associer à chaque nœud de l'arbre une famille de disjonctions. Ainsi, l'arbitrage est réalisé sur cette famille de disjonctions et non plus sur une seule disjonction.
Dans [Manier-Lacoste, 1994], on trouve de nombreuses extensions de ce modèle de base incluant les cuves multi-fonctions, multi-bacs, diverses configurations des postes de chargement et de déchargement ainsi que la circulation et la non-circulation des porteurs. On y trouve également une extension aux problèmes de lignes multi-robots (aussi dans [Baptiste et al, 1996] et [Manier et al, 2000]). Cette approche consiste à affecter aux robots non pas des cuves, mais des tâches de transport, en équilibrant si possible leur charge de travail. Les zones d'évolution de ces robots peuvent alors être continues ou non, disjointes ou non, ce qui a poussé les auteurs à prendre en compte les risques de collisions entre les robots. Pour résoudre ce problème et assurer une bonne gestion de ces ressources de transport, les auteurs ont ajouté au modèle de base un ensemble de contraintes disjonctives, décrivant toutes les configurations conflictuelles dans lesquelles les robots sont susceptibles de se trouver.

Che et Chu proposent une méthode pour la résolution du problème de goulots d'étranglement [Che et Chu, 2005]. Les auteurs présentent une approche de résolution du problème en un temps polynomial. La démarche est la suivante : chercher tout d'abord le temps de cycle optimal pour une séquence donnée. A partir de cette valeur trouvée, calculer le nombre optimal des cuves dupliquées pour chaque opération. Cette procédure semble intéressante mais l'optimalité de la solution n'est pas garantie.

**e. Problèmes connexes au HSP**

Liu et al ont proposé des extensions du modèle de base de Philips et Unger incluant les cuves multi-fonctions, multi-bacs et diverses configurations des postes de chargement [Liu et al, 2002]. Le modèle linéaire défini a été implémenté à l'aide du solveur de programmation linéaire Cplex. L'auteur introduit dans son modèle une variable $w_i$, traduisant le temps d'attente en charge du robot au dessus de la cuve de destination. Ce modèle permet, dans certains cas, de réduire considérablement le temps de cycle et de rendre certaines séquences de mouvements faisables. Toutefois, ce modèle omet une contrainte stricte du problème HSP, imposant l'interdiction des temps d'attente en charge et d'interruption ainsi que des stocks intermédiaires entre les cuves de traitements. Bien que cette étude soit voisine de celle d'un

CHSP, il est intéressant de la présenter ici, puisque sans ce temps d'attente ce modèle peut être considéré comme une extension du modèle de base de Philips et Unger [Phillips et Unger, 1976].

Song, Zabinski et Storch [Song et al, 1993] ont développé un modèle analytique utilisant la programmation entière mixte. Pour ce modèle, l'hypothèse de recherche d'un ordonnancement 1-cyclique n'est pas prise en compte. Ils ont développé une heuristique baptisée *Earliest Starting Time* (EST) et inspirée des travaux de Yih [Yih, 1990], [Yih et Thesen, 1989], [Yih et Chiu, 1993], [Yih et al, 1993], qui consiste à introduire sur la ligne les porteurs un à un, sans modifier les ordres préalablement déterminés. Toutefois, leurs dates d'entrée peuvent varier en raison des éventuels conflits entre mouvements. La solution choisie est celle permettant de traiter le maximum de porteurs dans un cycle. Cette approche présente deux restrictions : la durée opératoire des traitements est supposée fixe et les temps de déplacement à vide du robot sont négligés. Ces mêmes auteurs ont présenté dans [Song et al, 1995] un algorithme tenant compte des temps bornés des durées opératoires. Pour cela, ils ont défini dix règles heuristiques définissant le niveau de priorité des porteurs présents sur la ligne, affectant ainsi des tâches au robot. Certaines de ces heuristiques permettent de faire varier les dates d'entrées des porteurs sur la ligne, et comme précédemment, la solution retenue est celle qui maximise le nombre de porteurs traités sur un cycle.

Pour réduire la complexité de résolution du problème considéré, certains travaux concernent l'étude de la faisabilité des séquences cycliques en relaxant une des contraintes spécifiques au HSP : celle relative aux durées de trempes bornées et non fixées ([Kats et Levner, 1997], [Jiang et Liu, 2007]). De ce fait ils ne traitent pas un problème de type HSP au sens strict du terme, mais un problème connexe de type *Cyclic Robotic Scheduling*. Et par conséquent, atteindre la solution optimale pour un problème de type CHSP pourrait ne pas se réaliser. Le caractère flexible de la solution cyclique, conféré par les fenêtres de temps, sera étudié dans le chapitre II.

Dans la suite de notre état de l'art, nous nous intéressons aux approches élaborées pour la modélisation et/ou la résolution du problème CHSP multi-produits ou multi-gammes, faisant plus particulièrement l'objet de notre travail.

## I. 6. 3 - CHSP multi-gammes

Pour présenter l'état de la recherche sur le CHSP multi-gammes, nous faisons au préalable un zoom sur la partie grisée de la typologie arborescente donnée en figure. Ce nouvel arbre (figure I.17) complète le précédent, en reprenant les mêmes paramètres distinctifs, même si ceux-ci ne sont pas forcément utilisés aux mêmes niveaux du graphe.

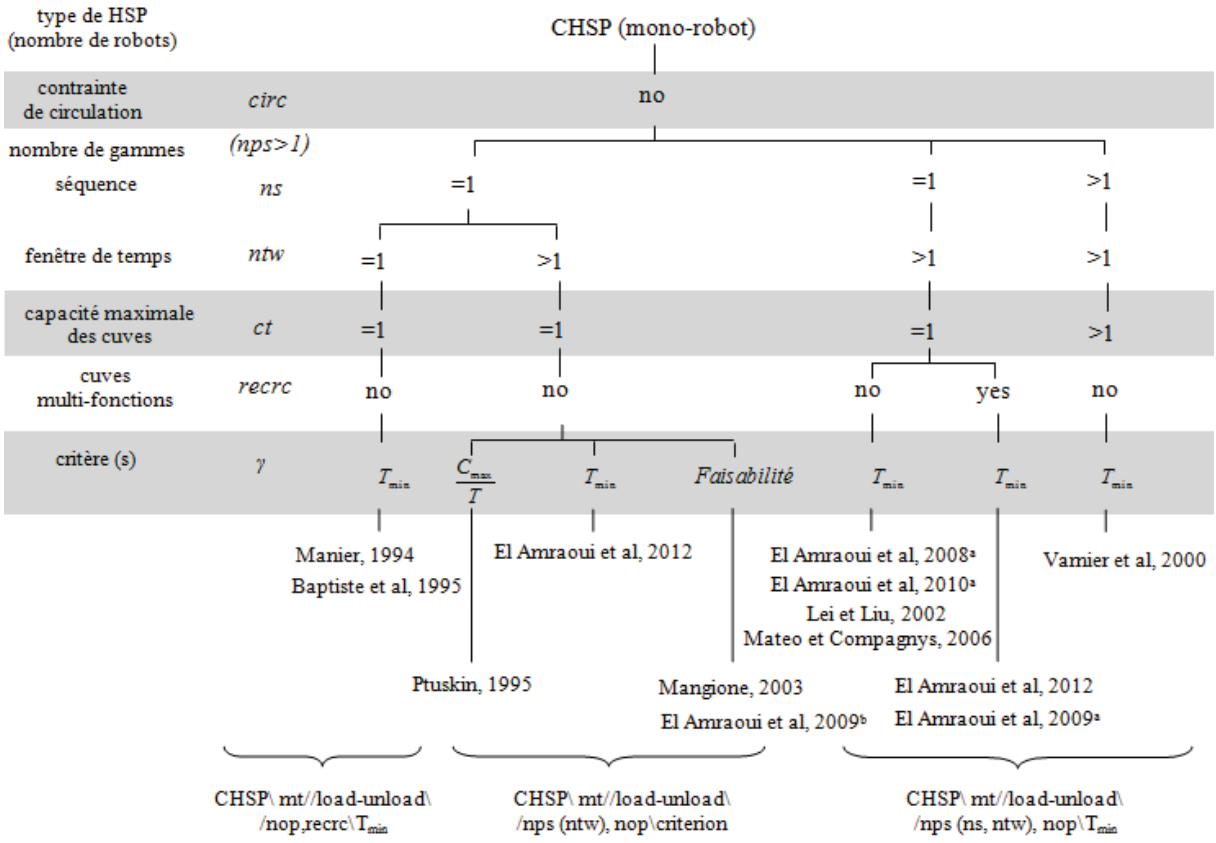



**Figure I.17.** Arbre multi-produits CHSP

Selon cette classification, on distingue trois cas principaux, en se basant sur le paramètre gamme opératoire (nps selon la notation de [Manier et Bloch, 2003]) :

- ✓ les problèmes mono-produits, où tous les produits suivent la même gamme, c'est-à-dire même séquence opératoire et mêmes fenêtres temporelles pour les durées de trempe. Toutefois, on autorise les produits à avoir des temps de trempe réels différents dans une cuve donnée, à condition de respecter ces fenêtres ;
- ✓ les problèmes multi-gammes, où les produits suivent également la même séquence opératoire, mais avec des fenêtres temporelles fonction des produits. C'est le cas de lignes traitant des produits proches (familles). On reste dans un contexte de type flow shop comme dans le cas précédent si on considère des lignes simples ;

- ✓ les problèmes multi-gammes, où séquences et fenêtres temporelles varient selon les produits.

Dans la notation du HSP, le premier cas se traduit par un paramètre *nps* prenant la valeur par défaut « ensemble vide » (mais indiqué =1 dans l'arborescence). Mais dans les deux derniers cas, *nps*>1. Pour distinguer ces situations dans notre arborescence, nous décomposons *nps* en deux sous-champs : *ns* (number of sequences) pour le nombre de séquences opératoires, *ntw* (number of time windows) pour le nombre de fenêtres temporelles différentes relatives aux temps de trempe dans chaque cuve). Ceci nous conduit à caractériser nos trois cas par le couple (*ns, ntw*), prenant respectivement les valeurs : (=1, =1), ((=1, >1), (>1, >1). Toutefois, nous ne modifions pas pour autant la notation initiale des problèmes de HSP.

**a. Séquences opératoires et fenêtres temporelles identiques**

Manier et Baptiste proposent une extension de leur modèle de base pour aborder le problème cyclique multi-produits [Manier-Lacoste, 1994], [Manier et Baptiste, 1994]. Les auteurs considèrent le cas où les produits sont tous identiques et tolèrent que les durées opératoires soient différentes dans la même cuve de traitement. Ils utilisent la Programmation Logique sous Contraintes pour résoudre le problème. Ce problème est une extension n-cyclique du cas mono-produit 1-cyclique. Mais, il reste relativement simple à résoudre vu qu'il concerne en fait une production mono-gamme.

**b. Séquences opératoires identiques et fenêtres temporelles différentes**

Ptuskin est le premier à avoir abordé le problème cyclique multi-produits [Ptuskin, 1995]. Il définit la notion de panachage à l'entrée de la ligne. Ptuskin considère que la séquence ainsi que les ratios des produits à l'entrée sont connus à l'avance. A titre d'exemple pour des produits A, B et C il définit un ordre d'entrée des produits : A-B-C-B. Un tel motif est produit tous les quatre unités de cycle *T*. En effet, l'auteur cherche une séquence n-cyclique, où *n* types de produits sont traités simultanément sur la ligne, qui minimise *T* (durée de cycle commune à tous les produits).

La méthode proposée par Ptuskin consiste à subdiviser le problème multi-produits en *n* problèmes mono-produit et à adapter une période commune à tous ces problèmes comme solution du problème multi-produits considéré. L'auteur fait appel à la logique floue pour déterminer la qualité des traitements effectués, c'est-à-dire pour vérifier le respect des bornes minimales et maximales des fenêtres de temps.

Dans sa thèse de doctorat [Mangione, 2003], Mangione traite le problème cyclique multi-produits. L'auteur cherche à trouver la séquence de mouvements du robot qui minimise la durée du cycle. Cette minimisation peut être traduite comme une maximisation du taux de production. Dans le chapitre III de son mémoire de doctorat, l'auteur propose une analogie entre les problèmes du flow-shop robotisé sans attente et le problème du CHSP. Pour cela, il considère le cas d'une ligne à deux cuves et applique l'algorithme de Gilmore et Gomory pour obtenir une solution réalisable pour le cas multi-produits. Cette solution, et bien qu'elle soit réalisable, n'est pas forcement optimale. L'exemple traité par Mangione est très simple à résoudre et ainsi les temps de simulation ne semblent pas être très significatifs.

Les recherches suivantes traitent de cas s'approchant le plus de ceux que nous avons étudiés. Ceci explique que nous les détaillons plus particulièrement.

Lei et Liu présentent d'abord dans [Lei et Liu, 2001] une analyse formelle du problème 2-cyclique hétérogène, portant sur la faisabilité d'une séquence cyclique. Ensuite, ils modélisent le problème cyclique par un graphe orienté paramétré. Leur objectif est de trouver un cycle hamiltonien (cycle passant par tous les sommets une et une seule fois) dans le graphe, minimisant le paramètre inconnu (la durée du cycle).Pour résoudre le problème, les auteurs proposent une méthode de type séparation et évaluation, faisant appel à l'analyse formelle déjà établie. La racine de leur arborescence représente les mouvements de chargement des deux produits. Et l'arbre est construit progressivement en rajoutant un mouvement de transport pour chaque produit, à chaque niveau (figure I.18). Toutes les combinaisons possibles, à ce rang, sont générées par des permutations. Les feuilles de l'arbre correspondent aux séquences cycliques. La procédure d'exploration élaborée est réalisée en profondeur d'abord. Si une solution optimale est trouvée pour un sous-problème, elle est réalisable, mais pas nécessairement optimale, pour le problème global. Comme elle est réalisable, elle peut être utilisée pour éliminer toute sa descendance : si la borne inférieure (durée du cycle de la séquence partielle) d'un nœud dépasse la valeur d'une solution déjà connue, alors on peut affirmer que la solution optimale globale ne peut être contenue dans le sous-ensemble de solutions représenté par ce nœud. La recherche continue jusqu'à ce que tous les nœuds soient explorés ou éliminés.

Toutefois, la fonction objectif est trop simple, ce qui entraine l'exclusion des solutions partielles à des moment avancés. Cela aura pour effet de rallonger encore les temps d'exécution de ces algorithmes, bien que le test de faisabilité des solutions partielles au cours de la construction de l'arbre pourrait entrainer l'élimination d'un bon nombre de nœuds. En

effet, la nature très contraignante du CHSP et plus particulièrement du cas multi-produits réduit très considérablement le nombre des solutions faisables.

Ces auteurs se sont limités dans leurs travaux aux cas des lignes simples, mono-robot et au cas particulier où les deux produits possèdent la même séquence opératoire. Avec l'approche qu'ils proposent, ils sont parvenus à résoudre le problème équivalent au 2-cyclique hétérogène à 9 cuves de traitement, avec des cuves de chargement et de déchargement dissociées, en un temps de simulation n'excédant pas les 20 minutes.

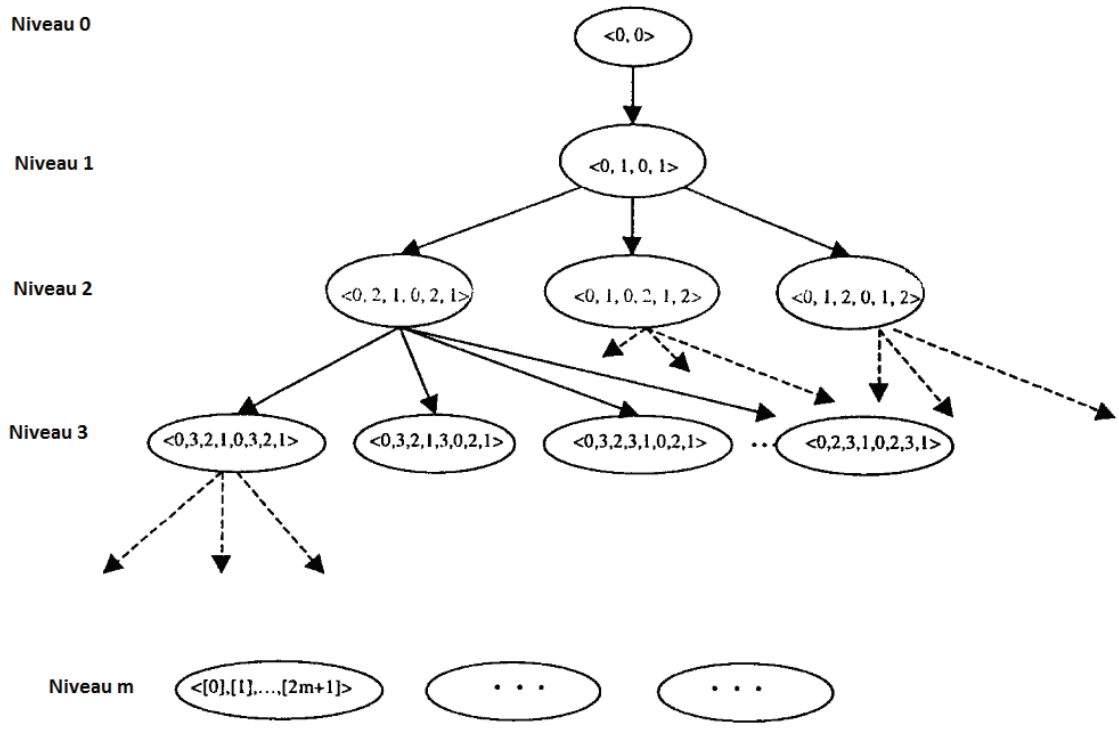



**Figure I.18.** Procédure de la construction d'un arbre de recherche pour déterminer un ordonnancement 2-cyclique à 2m+2 opération de transport [Lei et Liu, 2001]

Mateo et Companys traitent le cas multi-produits pour des lignes de traitement simples, c'est à dire avec des cuves mono-bac et mono-fonction. Ils cherchent une solution 2-cyclique sous l'hypothèse de 2 types de produits différents, mais de même séquence opératoire. Manier a également traité le même problème dans [Manier-Lacoste, 1994]. Toutefois, elle s'est limitée au cas où tous les produits traités ont les mêmes fenêtres de temps. En contrepartie son modèle est *n*-cyclique (avec $n \geq 1$).

Mateo et Companys. proposent une procédure de type séparation et évaluation pour résoudre le problème [Mateo et Companys, 2006]. Leur modèle 2-cyclique hétérogène peut-être appliqué pour un problème 2-cyclique homogène. Dans ce modèle, le premier mouvement

d'un robot correspond au début de l'extraction d'un produit de la cuve de chargement alors que le dernier mouvement correspond au dépôt du produit dans la cuve de déchargement. Les auteurs définissent une nouvelle notion appelée stage. Cette notion sert principalement à distinguer les mouvements de transport du produit de type 1 des mouvements du produit de type 2. A titre illustratif un produit de type 1 passant successivement par les cuves 0, 1, 2, ..k, .. p passe par la séquence de stages: $S_0$, $S_2$, $S_4$, …, $S_{2k}$, .., $S_{2p}$. Et pour la même séquence opératoire, un produit de type 2 passe successivement par les stages $S_1$, $S_3$, $S_5$, …, $S_{2k+1}$, …$S_{2p+1}$.

Pour mieux expliquer cette notion de stages , on peut dire que les auteurs considèrent le cas d'une ligne en I où toutes les cuves sont des cuves multi-bacs (2 bacs par cuve donc une cuve de capacité 2). Et les auteurs interdisent aux cuves plus d'un produit à la fois. En d'autres termes, deux stages quelconques $S_{2i}$ et $S_{2i+1}$ correspondent aux deux bacs virtuels de la cuve i.

Paramètres du modèle

Les auteurs définissent les variables suivantes :

- $C$ : temps de cycle,
- $a_k$ : borne inférieure de la k$^{ème}$ opération de trempe,
- $b_k$ : borne supérieure de la k$^{ème}$ opération de trempe,
- $h_{[k]}$ : k$^{ème}$ opération de transport en charge de la séquence cyclique de transport,
- $f_k$ : temps de transport en charge du robot entre les stages $S_k$ et $S_{k+2}$,
- $e_{k,k'}$: temps de transport à vide du robot entre les stages $S_k$ et $S_{k'}$,
- $H = (0, h_{[1]}, h_{[2]}, ..., h_{[p+1]})$ : vecteur mouvements du robot.

Les variables $f_k$ et $e_{k,k'}$ ne dépendent pas du type de produit à traiter.

Les auteurs définissent l'*extra time* comme étant la différence entre le temps de saisie du produit et le temps d'arrivée du robot au dessus de la cuve. Ils définissent, également, *le temps de process*, comme étant la différence entre le temps maximum de traitement $b_k$ et le temps minimum de traitement $a_k$.

Modèle

Mateo et Companys cherchent à minimiser la durée totale du temps de cycle tout en respectant les contraintes suivantes :

- contraintes sur la durée de trempe

$a_k \le t_{[j]} - (t_{[i]} + f_{k-2}) \le b_k$, k=2, 3, …, p+1 si j>i.

$a_k \le t_{[j]} + C - (t_{[i]} + f_{k-2}) \le b_k$, k=2, 3, …, p+1 si j>i.

avec $h_{[i]} = S_{k-2}$ et $h_{[j]} = S_k$ : deux mouvements consécutifs du même produit.

- contraintes liées à la ressource de transport

$$t_{[k+1]} \ge (t_{[k]} + f_{[k]} + e_{[k]+2,[k+1]}),$$

$$t_0 + C \ge t_{[p]} + f_{[p]} + e_{[p]+2,0},$$

- contraintes de cohérence

les auteurs se sont limités à l'étude de quatre cas de transport qu'ils ont détaillés dans [Mateo et al, 2000]

*cas 1*: produit de type 1 dans le stage *k+2* de la cuve *i+1*,

*cas 2* : produit de type 2 dans le stage *k+3* de la cuve *i+1*,

*cas 3* : la cuve i+1 est vide et le prochain produit à arriver est de type 1,

*cas 4* : la cuve i+1 est vide et le prochain produit à arriver est de type 2.

Procédure de résolution du problème

L'approche proposée par Mateo, est une approche qui traite le cas 2-cyclique, multi-produits, et qui consiste à ordonner les mouvements du robot pour deux produits différents, afin de minimiser le temps de cycle *C*.

La procédure de séparation, proposée par les auteurs consiste à insérer à chaque nœud de l'arborescence une cuve, ce qui revient à insérer les deux mouvements de transport en charge de chaque type de produit, relatifs à cette cuve. Ainsi, l'arbre proposé serait construit comme suit : à un niveau *r* de l'arbre et au nœud *(i+1)* la séquence cyclique de mouvement à ce niveau, définie par $IP_{[i][r+1]}$, peut être générée en insérant au vecteur $IP_{[i][r]}$ les deux mouvements de traitement relatifs à la cuve suivante.

Concernant la procédure d'évaluation du temps de cycle, pour le choix de la solution envisagée, l'auteur définit une borne minimale d'évaluation du temps de cycle donnée par la formule suivante :

$$LB\left(H_{[i][r]}\right) = C\left(H_{[i][r]}\right) + \max\left\{\left[\sum_{h_j \in U}(f_j + e_{j,j+1}) - \sum_{h_k \in I}(w_k)\right], 0\right\}$$

Où :

$H_{[i][r]}$ : est le i$^{ème}$ nœud du niveau r de l'arbre de recherche, définissant l'ordre partiel des mouvements de transport du robot insérés à ce niveau.

$C(H_{[i][r]})$ : est le temps de cycle associé à la séquence $H_{[i][r]}$.

$W_k$ : est le temps d'attente du robot au niveau de la cuve k.

Cette approche qui traite le cas cyclique multi-produits a prouvé que le traitement d'un cas 2-périodique fournit un temps de cycle meilleur que dans le cas d'une production 1-périodique lors du traitement de deux produits.

Dans le problème qu'ils considèrent, la durée de séjour et plus particulièrement la fenêtre de temps de chaque produit dans une cuve de traitement est différente des autres produits. Toutefois, Mateo et Companys se sont limités aux cas des lignes simples, mon-robot et au cas particulier où les deux produits possèdent la même séquence opératoire.

**c. Séquences opératoires et fenêtres temporelles différentes**

Varnier et Jeunehomme se sont intéressés au problème CHSP multi-produits dans un cas de fonctionnement transitoire [Varnier et Jeunehomme, 2000]. La raison principale qui les a incités à considérer les régimes transitoires est la nature même des problèmes étudiés qui sont des régimes permanents finis pour lesquels la durée du transitoire est loin d'être négligeable.

L'objectif des recherches menées par Varnier et Jeunehomme est de trouver un fonctionnement cyclique de la ligne permettant d'effectuer le traitement de produits de gammes différentes, en garantissant productivité et qualité.

Les auteurs définissent par trajectoire, le chemin correspondant au traitement que le porteur suit pendant un cycle. Ainsi, si *n* porteurs sont présents sur la ligne, il y aura *n* trajectoires pendant un cycle. Le chemin du porteur introduit sur la ligne au début du cycle correspond à la première trajectoire. Tandis que la seconde correspond au chemin du porteur introduit sur la ligne au début du cycle précédent. Ces trajectoires, dans le cadre d'un fonctionnement multi-produits, peuvent être suivies par des produits de gammes différentes.

Dans leur approche, les auteurs proposent une liste *L* de produits (de gammes différentes) et cherchent pour une telle liste un ordonnancement tel qu'à chaque date de début de cycle, le traitement d'un produit quelconque de *L* pourrait être lancé. Il s'avère ainsi nécessaire de prendre en compte toutes les configurations que le système est susceptible de rencontrer dans ce mode de fonctionnement multi-produits (figure I.19). A titre illustratif, si le nombre de produits *np* vaut 2 (produits A et B) et le nombre de trajectoires *n* est de 2, il faudra considérer les 4 configurations suivantes : $C_{A,A}$ ; $C_{B,A}$ ; $C_{A,B}$ ; $C_{B,B}$.

Les auteurs proposent une approche de type séparation et évaluation, leur permettant d'obtenir un ordonnancement cyclique pour les divers produits traités sur la ligne. Dans cette optique, ils proposent de mixer les ordonnancements cycliques mono-produit et considèrent toutes les combinaisons possibles. La résolution du problème a été effectuée par la Programmation Logique sous Contraintes (PLC).

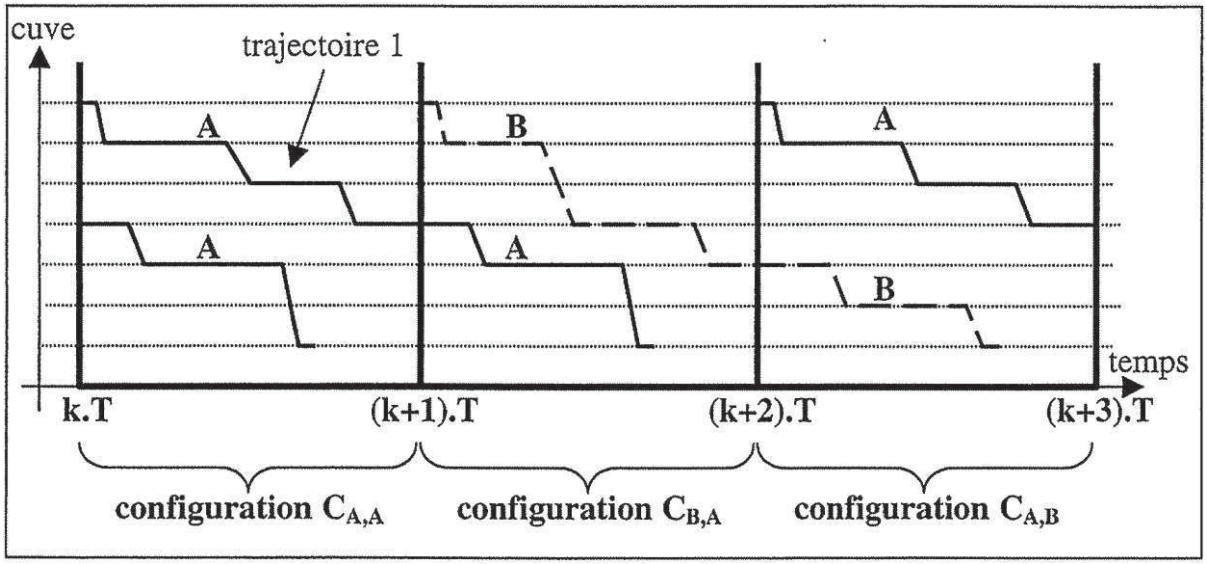



**Figure I.19.** Exemple de définition des configurations

Démarche de résolution

La démarche développée par les auteurs consiste tout d'abord, en la recherche d'une solution mono-produit pour chaque élément de la liste L. Ces solutions sont ensuite couplées de manière à obtenir l'ensemble des couplages $S_{i1, \ldots, ip}$ où les *ip* désignent le numéro de la solution mono-produit du produit *p*.

Pour chaque couplage $S_{i1, \ldots, ip}$ des configurations sont considérées et pour chacune d'entre elles, un ensemble d'ordonnancements admissibles des $R_i^k$ est recherché.

L'algorithme de type séparation et évaluation couplé avec un algorithme de type simplexe permettant de considérer l'ensemble des contraintes propres à chaque configuration et de déterminer la solution optimale si elle existe. La figure I.20 modélise le principe de fonctionnement de l'algorithme développé [Jeunehomme, 1999].

Cette approche, à l'exemple de celle de Ptuskin, traite le problème CHSP multi-produits. Toutefois, elle se distingue de celle de [Ptuskin, 1995] par le fait que le produit entrant en

traitement à chaque début de cycle est choisi aléatoirement et qu'aucune contrainte sur le choix des gammes n'est imposée.

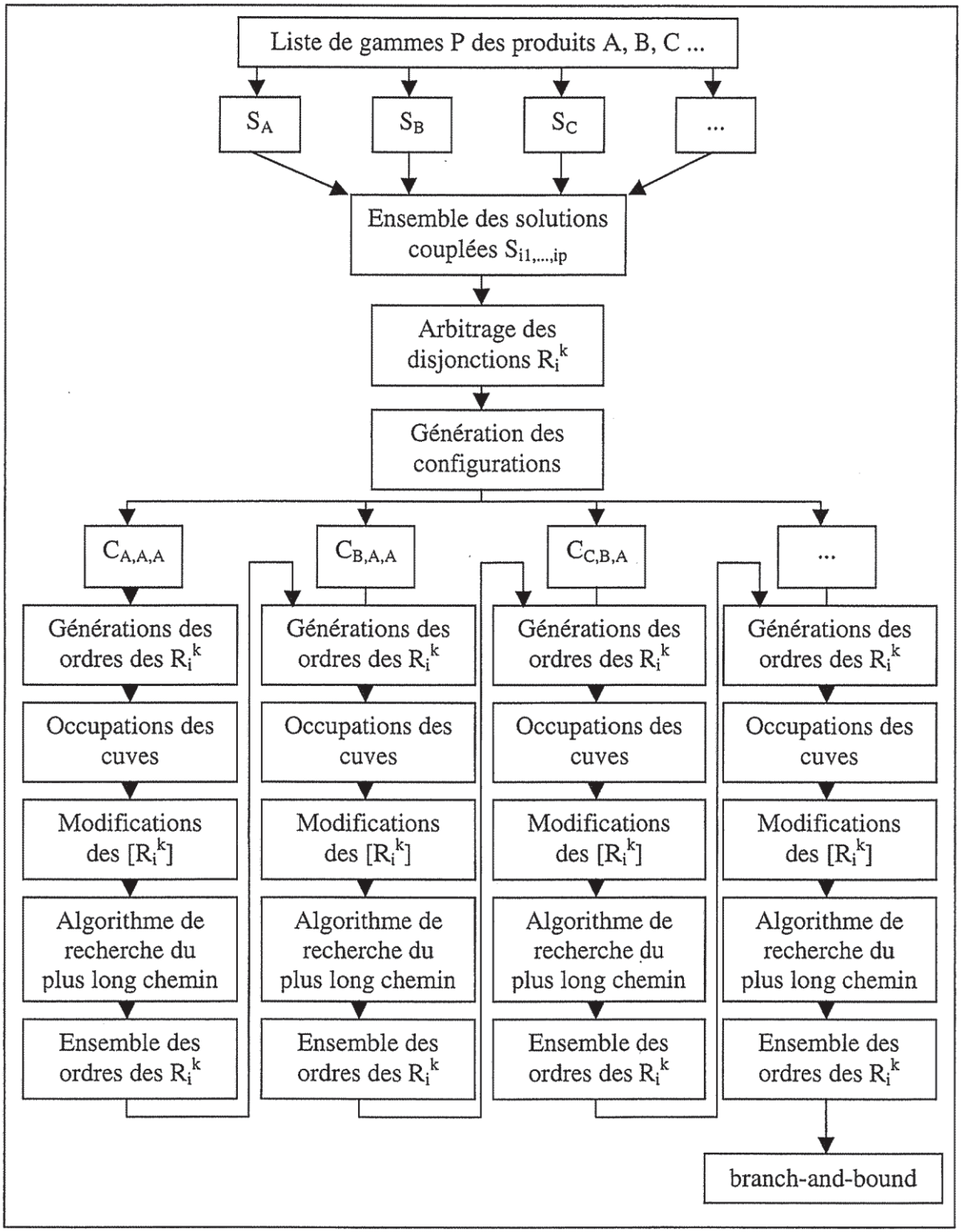



**Figure I.20.** Principe de fonctionnement de l'algorithme [Jeunehomme, 1999]

## I. 7 - Bilan

Le HSP est un problème très spécifique par rapport aux autres problèmes d'ordonnancement classiques d'ateliers (job shop, flow shop, ..). Cette spécificité est traduite par un ensemble de contraintes strictes auxquelles est soumis notre problème d'ordonnancement. Ces contraintes spécifiques entrainent une croissance exponentielle de sa complexité.

De nombreux problèmes de type HSP demeurent encore irrésolus et peu de chercheurs semblent s'être intéressés au cas des lignes multi-produits et/ou multi-robots. Exceptés quelques travaux de recherche à l'exemple de [Lei et Wang, 1991], [Lei et Wang, 1989][b] et Manier [Manier-Lacoste, 1994], [Manier et Lamrous, 2006], qui se sont intéressés à l'étude des problèmes multi-robots ou Ptuskin [Ptuskin, 1995], Varnier [Varnier et Jeunehomme, 2000] et Mateo [Mateo et Companys, 2006] qui ont proposé diverses approches pour la résolution de problèmes multi-produits, la plupart des auteurs ayant étudié le CHSP, se sont limités à la modélisation du cas mono-robot et mono-produit. Ceux qui se sont intéressés au problème mono-produit ont proposé des extensions multiples de leurs modèles incluant diverses configurations des postes de chargement et de déchargement. Contrairement aux autres, ceux qui se sont intéressés au cas multi-produits se sont limités à l'étude des lignes simples avec une configuration dissociée des postes de chargement et de déchargement. La majorité des travaux proposent des approches de modélisation et de résolution exactes basées sur des algorithmes de type séparation et évaluation, à l'exception de Lim [Lim, 1997] et Manier et Lamrous [Manier et Lamrous, 2006] qui ont développé des approches évolutives.

La suite de ce mémoire propose de nouvelles approches de résolution dédiées aux problèmes cycliques multi-produits. Le chapitre II est consacré tout d'abord, à la modélisation et la résolution linéaire du problème 2-cyclique multi-produits pour des lignes simples en I. Ce modèle est caractérisé par son caractère évolutif. Nous en proposons donc ensuite des extensions pour traiter des lignes plus complexes (multi-bacs, multi-fonctions) et diverses configurations du poste de chargement et de déchargement (associés, dissociés). Après avoir intégré la gestion des contraintes introduites par les caractéristiques des différents types de produits à traiter, une généralisation du modèle 2-cyclique multi-produits est élaborée pour la recherche de solutions optimales $n$-cyclique multi-produits ($n \geq 2$).

Une amélioration du modèle général traitant le problème $n$-cyclique multi-produits pour le rendre plus proche de la réalité industrielle est élaboré et les résultats obtenus pour certains exemples semblent être très prometteurs.

Le chapitre III décrit deux méthodes approchées pour la résolution du problème *n*-cyclique multi-produits afin de remédier aux problèmes de résolution avec des méthodes exactes pour des lignes de grandes tailles. La première méthode proposée est une heuristique, basée sur un algorithme de liste. La seconde méthode est une méthode approchée classique, qui consiste à appliquer la méthode des algorithmes génétiques pour la modélisation et la résolution du problème généralisé 2-cyclique.

Dans ces deux chapitres, nous comparons à partir d'exemples de référence les résultats obtenus avec ceux des recherches antérieures et avec ceux obtenus des différentes méthodes proposées. Nous rapportons et nous analysons aussi les résultats obtenus sur d'autres problèmes comprenant différents niveaux de complexité.

En conclusion de ce mémoire, un bilan global de notre démarche dans lequel nous montrerons les diverses avantages des outils utilisés, et analysons également leurs faiblesses pour la résolution du HSP.

# Chapitre II

# Formulation, optimisation et résolution du problème CHSP multi-produits par la Programmation Linéaire Mixte

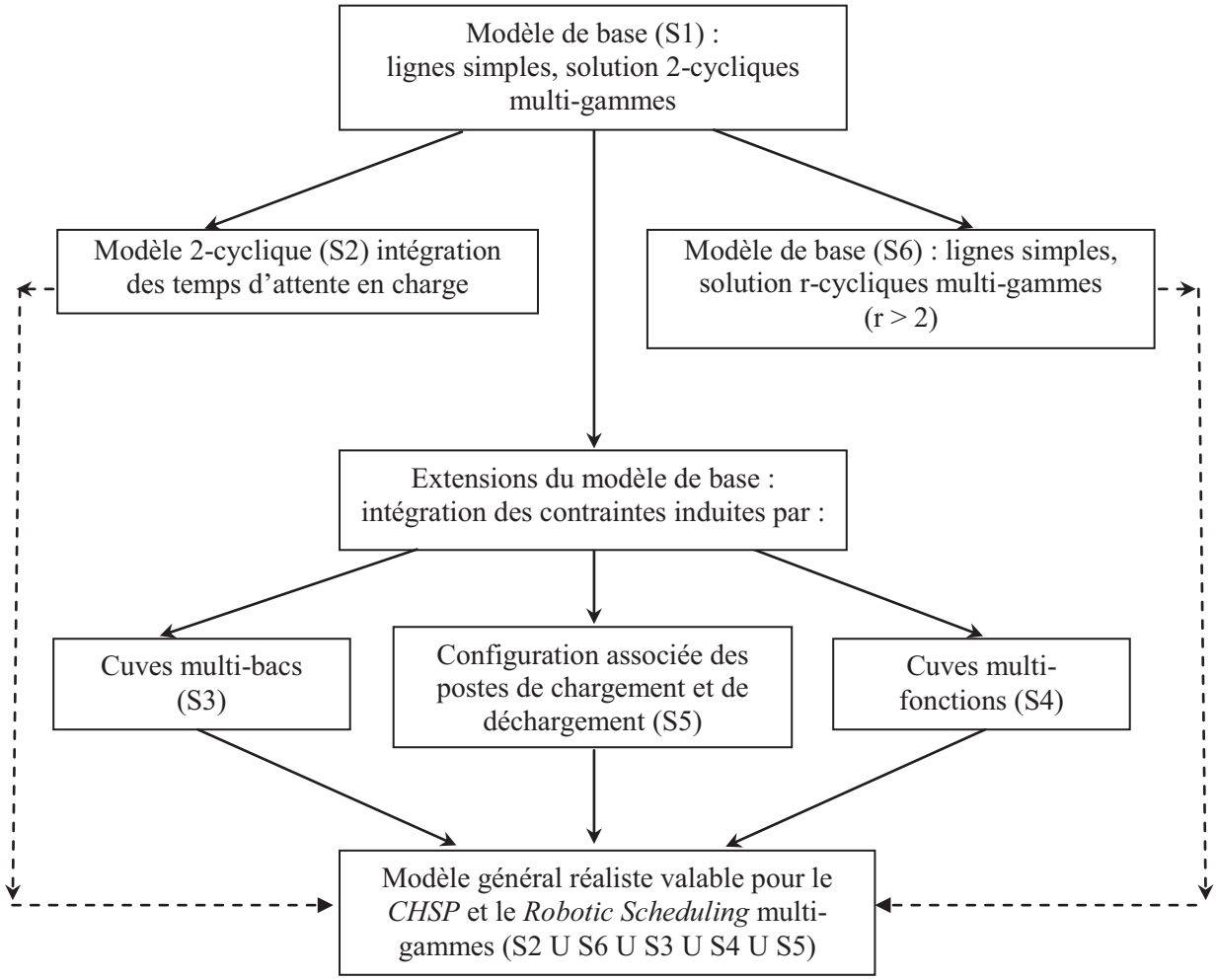
## II. 1 - Introduction

Notre propos est ici de présenter une nouvelle approche de modélisation et de résolution du problème d'ordonnancement cyclique multi-gammes des ateliers de traitement de surface. Pour cela, on considère une liste de *np* produits ($np \geq 2$) : possédant la même séquence opératoire et ne différent que par les fenêtres de temps (par exemple, cas des produits de même type mais de tailles différentes). Pour ce genre de production, nous chercherons un ordonnancement cyclique tel qu'au cours de chaque cycle, *np* différents types de produits entrent et *np* types soient finis. L'objectif est de trouver un fonctionnement *np-cyclique* de la ligne permettant d'effectuer le traitement simultané de ces *np* types de produits, tout en garantissant productivité et qualité. L'ordre d'entrée des produits sur la ligne n'est pas fixé au préalable et le nombre *np* de produits est soumis à des ratios prédéfinis.

Cette approche tire son originalité du fait que c'est la première qui s'intéresse au cas r-cyclique multi-gammes ; problème très peu abordé dans le cas multi-produits : seuls Mateo et Companys [Mateo et Companys, 2006] et Lei et Liu [Lei et Liu, 2001] ont abordé un problème similaire : celui d'un *2*-cycle hétérogène adapté aux lignes simples.

Dans un premier temps, nous avons commencé par modéliser et résoudre ce même cas simplifié, ce qui nous a permis de valider notre démarche et notre modèle en comparant nos résultats avec ceux de la littérature. Dans un deuxième temps, nous avons successivement élaboré diverses extensions de ce modèle que nous appelons modèle de base. Les premières extensions permettent de résoudre des instances pour des lignes dites complexes, en termes de type de ressources de traitement et de configuration de postes de chargement et déchargement. La dernière extension généralise le modèle de base au cas des solutions r-cycliques ($r \geq 2$).

La figure II. 1. illustre la démarche suivie depuis le modèle de base jusqu'au modèle général multi-gammes.




Si : système de contraintes associé au modèle ou à l'extension considéré(e).
Si U Sj : système de contraintes constitué des contraintes de Si et des contraintes de Sj.

**Figure II.1.** Extensions proposées du modèle de base élaboré

# II. 2 - Formalisation du problème de base 2-cyclique hétérogène

## II. 2. 1 - Le problème

Le problème que nous avons tout d'abord modélisé est une extension du problème de base mono-produit, pour lequel un ordonnancement optimal 1-périodique est cherché. Les lignes considérées sont caractérisées par :

- ✓ une implantation linéaire des cuves ;
- ✓ la présence d'un seul robot de manutention ;
- ✓ l'existence exclusive de cuves mono-bac et mono-fonction ;

- ✓ une configuration dissociée des postes de chargement et de déchargement. Cette configuration est la moins contraignante car elle exclut les conflits qui peuvent exister entre les opérations de chargement et de déchargement dans le cas d'une configuration associée de ces postes. De ce fait, elle a été adoptée par les chercheurs ayant abordé le CHSP multi-produits ([Mateo et Companys, 2006], [Lei et Liu, 2001], [Varnier et Jeunehomme, 2000]).

De plus, nous adoptons les conventions suivantes :

- ✓ les opérations de chargement et de déchargement correspondent respectivement au premier et au dernier traitement de la gamme opératoire ;
- ✓ l'indexation des cuves suit celle des traitements de la gamme : conformément à la notation du paragraphe II.2.1., le $i^{ème}$ traitement d'une gamme aura lieu dans la cuve $c=i-1$ ($i$=1 à $nc+2$), $nc$ étant le nombre de cuves (figure II.2) ;
- ✓ l'entrée d'un produit sur la ligne correspond à l'instant où il commence son chargement sur un porteur ;
- ✓ la sortie d'un produit de la ligne correspond à son arrivée au poste de déchargement. La durée de l'opération de déchargement n'est donc pas prise en compte.

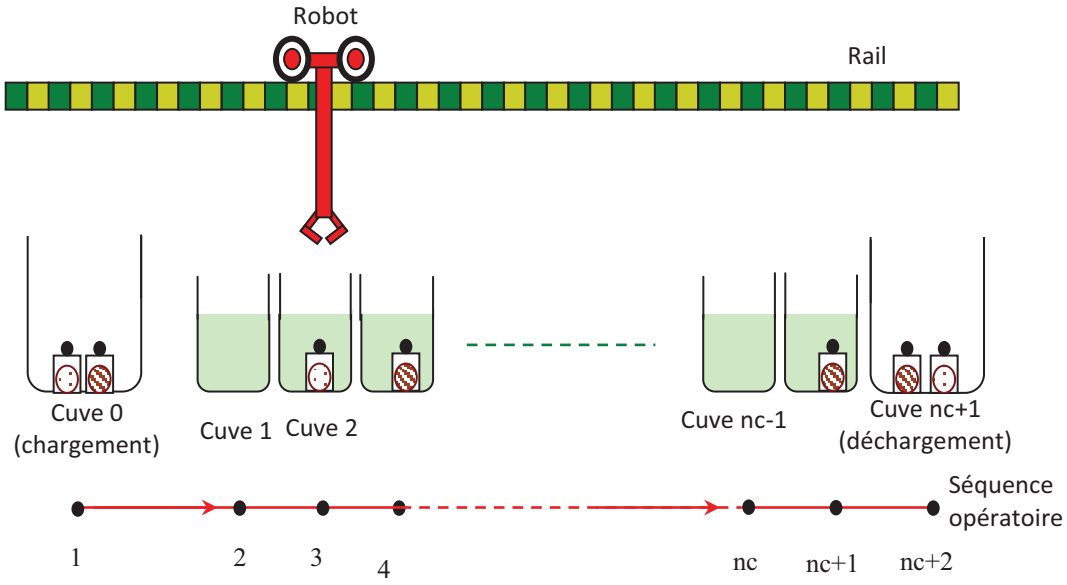



**Figure II.2.** Illustration du problème à traiter

Nous rappelons également les principales contraintes du HSP, dans les conditions énoncées :

(C1) la durée opératoire de chaque traitement doit être comprise entre une borne minimale et une borne maximale (sauf pour les cuves fictives de chargement et déchargement dont la capacité est supposée illimitée),

(C2) chaque cuve de la ligne a une capacité unitaire et ne peut recevoir qu'un produit à la fois,

(C3) un robot ne peut pas effectuer deux opérations de transport en charge simultanément,

(C4) un robot doit avoir au moins le temps de se déplacer à vide entre deux opérations de transport en charge : entre la cuve où il vient de déposer un porteur et la cuve d'où il doit retirer le prochain.

Pour le type d'ateliers considéré, nous cherchons l'ordonnancement optimal 2-périodique pour deux types de produits, ayant la même séquence opératoire mais avec des limites de durées opératoires différentes. Dans un ordonnancement 2-périodique multi-produits, chaque opération est réalisée exactement deux fois : une fois pour chaque type de produit. Les produits sont à traiter en proportions équivalentes (50% pour le produit de type 1 et 50% pour le produit de type 2). Le critère que nous cherchons à optimiser est la durée totale du cycle 2-périodique pour l'ensemble des produits des deux types. La faisabilité et l'optimalité de la durée de ce cycle sont à assurer. Le problème d'ordonnancement 2-cyclique hétérogène que nous étudions s'exprime sous la forme :

**CHSP | mt / / diss | nparts / nps, mt+2 | $T_{min}$**

## II. 2. 2 - Notations

Les notations utilisées pour la description, l'analyse mathématique ainsi que la modélisation du problème abordé sont définies ci-après :

Les paramètres du problème :

| | |
|---|---|
| nc | le nombre de cuves de traitement. Les postes de chargement et de déchargement sont considérées comme des cuves fictives et respectivement notées cuve 0 et cuve (*nc+1*) |
| np | le nombre de porteurs (lots de produits) considéré, c'est-à-dire le nombre de porteurs se trouvant simultanément dans le système ($1 \leq np \leq np_{max}$, $np_{max}$ étant une constante représentant le nombre maximal possible). np peut être soit une donnée fixée a priori, soit une variable |
| $nt^j$ | le nombre de traitements du produit j ($1 \leq nt^j \leq nc$) |
| $O^T_{(i,j,c)}$ | la $i^{ème}$ opération de trempe du produit j dans la cuve c ($1 \leq i \leq nt^j$, $1 \leq j \leq np$ et $1 \leq c \leq nc$) |

| | |
|---|---|
| $a_i^j$ | durée minimale de la $i^{ème}$ opération de trempe du produit j ($1 \leq i \leq nt^j$ et $1 \leq j \leq np$) |
| $b_i^j$ | durée maximale de la $i^{ème}$ opération de trempe du produit j ($1 \leq i \leq nt^j$ et $1 \leq j \leq np$) |
| $O^H_{(i,j,c)}$ ou $(j, i)^t$ | $i^{ème}$ opération de transport assurée par le robot, du produit de type j de la cuve de départ (c) correspondant au traitement i jusqu'à la cuve d'arrivée correspondant au traitement i+1 ($1 \leq i \leq nt^j$, $1 \leq c \leq nc$ et $1 \leq j \leq np$) |
| $c_{i,k}$ | durée du déplacement à vide du robot de la cuve i à la cuve k ($0 \leq i, k \leq nc+1$) |
| S | séquence cyclique des mouvements de transport du robot |
| $d_i$ | durée de déplacement en charge de la cuve i à la cuve i+1 ($0 \leq i \leq nc$) |
| $e_i$ | temps d'attente à vide du robot au dessus de la cuve i (si le robot arrive à cette cuve avant la fin d'un traitement) ($0 \leq i \leq nc$) |

Les variables de décision :

| | |
|---|---|
| $o^j_i$ | durée de la $i^{ème}$ opération de trempe du produit j ($1 \leq i \leq nt^j$ et $1 \leq j \leq np$) |
| $t_i^j$ | date de début de l'opération de transfert $O^H_{(i,j,c)}$ ($0 \leq i \leq nt^j$ et $1 \leq j \leq np$) |
| T | temps de cycle |
| $DO^T_{(i,j,c)}$ | date de début de $O^T_{(i,j,c)}$ ($1 \leq i \leq nt^j$, $1 \leq c \leq nc$ et $1 \leq j \leq np$) |
| $FO^T_{(i,j,c)}$ | date de fin de $O^T_{(i,j,c)}$ ($1 \leq i \leq nt^j$, $1 \leq c \leq nc$ et $1 \leq j \leq np$) |
| $DO^H_{(i,j,c)}$ | date de début de $O^H_{(i,j,c)}$ ($1 \leq i \leq nt^j$, $1 \leq c \leq nc$ et $1 \leq j \leq np$) |
| $FO^H_{(i,j,c)}$ | date de fin de $O^H_{(i,j,c)}$ ($1 \leq i \leq nt^j$, $1 \leq c \leq nc$ et $1 \leq j \leq np$) |
| $rang(O^H_{(i,j,c)}) = k$ ou $[r] = k$ | k est le numéro de la cuve déchargée en $i^{ème}$ position dans la séquence cyclique S ($0 \leq k \leq nc$, $1 \leq j \leq np$, $0 \leq c \leq nc$ et $0 \leq r \leq n-1$), n étant une constante représentant le nombre total des mouvements de transport dans la séquence cyclique S), $n=(nt^j+1).np$ |
| $y^{j,k}_{r,s}$ | $y^{j,k}_{r,s} = \begin{cases} 1 & \text{si } rang(O^H_{(r,j,c)}) < rang(O^H_{(s,k,c')}) \text{ dans la séquence } S \\ 0 & \text{sinon} \end{cases}$<br>comme première simplification, on désigne par $y^j_{r,s}$ la variable $y^{j,j}_{r,s}$ |

Un ordonnancement cyclique est défini par la durée réelle ou exacte des opérations de trempe $o_i^j$, les dates de début des opérations de transport $t_i^j$, ainsi que la durée de cycle $T$ qu'on cherche à minimiser. Le nombre total $N$ de variables à déterminer est donc :

| | |
|---|---|
| $N = 1 + \sum_{j=1}^{np} (2.nt^{j} - 1)$ | (II.1) |

Toutefois, comme le processus de traitement se déroule d'une façon continue, c'est-à-dire sans attente, sans interruption et sans stock intermédiaire, il est bien évident, qu'on peut déduire les durées $o_i^j$ des dates $t_i^j$. Ainsi les inconnues du problème se limitent aux dates $t_i^j$ et la durée du temps de cycle. Le nombre total $N$ de variables se réduit par conséquent à :

| | |
|---|---|
| $N = 1 + \sum_{j=1}^{np} (nt^{j} - 1)$ | (II.2) |

## II. 2. 3 - Analyse du problème

### a. Temps de cycle

La production étant cyclique, l'état de la ligne à la fin de chaque cycle correspond à son état au début du cycle suivant. Le temps de cycle global à minimiser correspond à l'instant du retour de la ligne à son état initial c'est-à-dire après que le robot ait effectué tous les déplacements à vide et en charge, toutes les attentes à vide et qu'il soit retourné à sa cuve de départ. Dans ce cas, le temps de cycle est donné par :

| | |
|---|---|
| $T = \sum_{p=0}^{n-1} (e_{[p]} + d_{[p]} + c_{[p],[p+1]})$ | (II.3) |

### b. Durée d'une opération de trempe

Afin de pouvoir calculer la durée d'une opération de trempe $i$ d'un produit $j$, il faut identifier si celle-ci commence et se termine au cours d'un même cycle ($1^{er}$ cas) ou si elle commence dans un cycle et se termine au cycle suivant ($2^{ème}$ cas). Dans le second cas, on parlera de chevauchement de cycle par le traitement considéré (figure II.3).

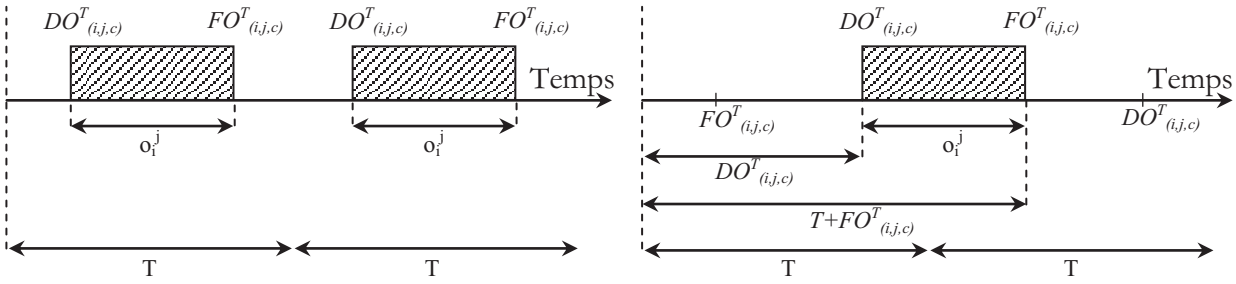



**Figure II.3.** Durées des traitements

Si le 1$^{er}$ cas est vérifié alors $o_i^j$ s'écrit :

| | | | |
|---|---|---|---|
| $o_i^j = FO^T_{(i,j,c)} - DO^T_{(i,j,c)}$ | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^j$ | (II.4) |

Par contre, si c'est le 2$^{nd}$ cas qui est vérifié alors $o_i^j$ s'écrit :

| | | | |
|---|---|---|---|
| $o_i^j = T + FO^T_{(i,j,c)} - DO^T_{(i,j,c)}$ | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^j$ | (II.5) |

La durée de trempe est bornée. Elle est comprise entre des valeurs minimale et maximale. Au-delà de ces bornes, la pièce est considérée défectueuse :

| | | | |
|---|---|---|---|
| $a_i^j \le o_i^j \le b_i^j$ | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^j$ | (II.6) |

Les relations (II.4), (II.5) et (II.6) permettent d'écrire :

| | | | |
|---|---|---|---|
| $a_i^j \le T.y_{i,i+1}{}^j + FO^T_{(i,j,c)} - DO^T_{(i,j,c)} \le b_i^j$ | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^j$ | (II.7) |

### c. Dates d'exécution des opérations

L'exécution des opérations de transport étant successive, la date de début d'une opération *i* dépend des opérations qui la précédent dans la séquence cyclique *S*. En d'autres termes, elle dépend des activités précédentes du robot. De cette manière, la date de début d'une opération de transport $O^H_{(i,j,c)}$ peut-être calculée ainsi :

| | | |
|---|---|---|
| $DO^H_{(i,j,c)} = \sum_{p=0}^{rang(O^H_i)} (e_{[p]} + d_{[p]} + c_{[p],[p+1]})$ | $\forall i = 1,...,nt^j$ | (II.8) |

Durant le processus de traitement, aucune attente et aucun stock intermédiaire ne sont tolérés. Ainsi, en analysant les mouvements de transport du point de vue produit (figure II.4) et du point de vue robot (figure II.5), on peut conclure les assertions suivantes (1- 4) :

*Du point de vue produit*

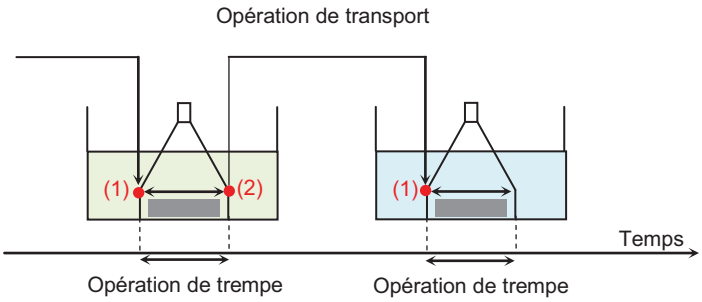



**Figure II.4.** Analyse du processus de transport du point de vue produit

(1) la fin d'une opération de transport correspond au début d'une opération de trempe.

(2) la fin d'une opération de trempe correspond à l'instant de début d'une opération de transport.

*Du point de vue robot*

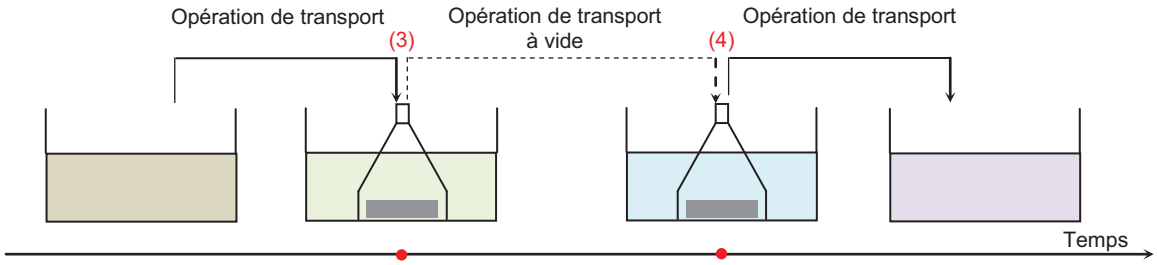



**Figure II.5.** Analyse du processus de transport de point de vue robot

(3) la fin d'une opération de transport correspond à l'instant de début d'un mouvement de transport à vide.

(4) la fin d'une opération de déplacement à vide intervient avant le début de l'opération de transport suivante dans le cycle, si deux opérations successives de transport ne concernent pas le même porteur. Autrement dit, le robot a le droit d'effectuer des pauses à vide.

Ces assertions sont traduites par les équations suivantes :

| | | | | |
|---|---|---|---|---|
| $DO^{H}_{(i+1,j,c)} = FO^{T}_{(i,j,c)}$ | | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^{j}$ | (II.9) |

| | | | | |
|---|---|---|---|---|
| $FO^{T}_{(i,j,c)} = DO^{H}_{(i,j,c)}$ | | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^{j}$ | (II.10) |

| | | | | |
|---|---|---|---|---|
| $FO^{H}_{(i,j,c)} - DO^{H}_{(i',j',c')} = c_{c+1,c'} + e_{c'}$ | $y^{j,j'}_{i,i'} = 1$ | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^{j}$ | (II.11) |

| | | | | |
|---|---|---|---|---|
| $FO^{H}_{(i',j',c')} - DO^{H}_{(i,j,c)} = c_{c'+1,c} + e_{c}$ | $y^{j,j'}_{i,i'} = 0$ | $\forall j = 1,...,np$ | $\forall i = 1,...,nt^{j}$ | (II.12) |

Ainsi, comme nous l'avons déjà évoqué au chapitre I, trouver le meilleur ordonnancement qui minimise la durée totale du cycle revient à trouver le meilleur ordonnancement des opérations de trempe ou des mouvements de transport en charge, ou encore, l'ordonnancement des mouvements à vide du robot.

### d. Description d'une séquence cyclique

Nous définissons le mouvement $m_i$ du robot comme le mouvement de transport en charge du produit de la cuve *i* à la cuve (i+1) (i=0 à *nc)*. De plus, on note par $(m_i, j_k)$ le i$^{ème}$ mouvement de transport du produit de type *j*, où $0 \leq i \leq nc$ et $j_k = \{j_A$ ou $j_B\}$. Conformément à la notation du paragraphe II.2.2, cela revient à la correspondance : $O^{H}_{(i,j,i-1)}=(m_i,j)$.

On rappelle que la ressource de transport est à capacité unitaire, c'est-à-dire que le robot ne peut pas transporter plus d'un porteur à la fois. Par conséquent, durant la période d'un cycle donné, chaque mouvement de transport, pour chaque type de produit, est exécuté une et une seule fois. Donc pour l'ensemble des produits, chaque mouvement ($m_i$, ●) est exécuté deux fois pendant une séquence 2-cyclique des mouvements de transport en charge. Ainsi, (*nc*+1) mouvements de transport en charge sont assurés pour chaque type de produit, ce qui donne *2(nc+1)* transports assurés par le robot.

En utilisant ces notations, une séquence cyclique *S* donnée peut-être représentée par :

$$S = \left\langle (m_{[0]}, j_{[0]}), (m_{[1]}, j_{[1]}), (m_{[2]}, j_{[2]}), ..., (m_{[2.nc+1]}, j_{[2.nc+1]}) \right\rangle$$

avec : $m_{[i]}=m_k$ et $j_{[i]}=j_A$, signifie que le $i^{ème}$ mouvement de transport de la séquence cyclique est le $k^{ème}$ mouvement de transport de la gamme opératoire d'un produit de type A. Ce mouvement correspond au déplacement en charge du produit A de la cuve *k* à la cuve *k+1*. Sans perte de généralités, la première opération de transport de la séquence cyclique peut-être

fixée au préalable, et soit ainsi, la première opération de transport du premier produit choisie comme première opération de la séquence $S$ (où $(m_{[0]}, j_{[0]}) = (m_0, j_A)$).

**Exemple 1.**

Soit l'exemple d'une ligne à 2 cuves de traitements ($nc$=2) et soit A et B, les deux types de produits à traiter. On considère la séquence cyclique suivante :

$$S = \left\langle (m_{[0]}, j_{[0]}), (m_{[1]}, j_{[1]}), (m_{[2]}, j_{[2]}), (m_{[3]}, j_{[3]}), (m_{[4]}, j_{[4]}), (m_{[5]}, j_{[5]}) \right\rangle$$

$$S = \left\langle (m_0, j_A), (m_2, j_B), (m_1, j_A), (m_0, j_B), (m_2, j_A), (m_1, j_B) \right\rangle$$

Nous considérons un état initial de la ligne illustré par le premier schéma de la figure II.6 où un produit de type B est présent dans la cuve 2 en début de cycle. Les étapes de la séquence cyclique S à partir de cet état initial sont illustrées par les schémas suivants sur cette même figure. Notons que cette séquence autorise le chevauchement de cycles (un produit qui commence un traitement dans une cuve au cours d'un cycle donné peut le terminer au cycle suivant). On y fait apparaître les mouvements en charge exécutés par le robot, pour les deux types de produits (job *A* et job *B*). Dans ce cas, le robot visite les cuves dans l'ordre 0-2-1-0-2-1. Notons que l'état de la ligne à la fin d'un cycle correspond à son état à son début, entre autres la position du robot, et l'emplacement des produits dans les cuves, hors postes de chargement et déchargement. Le temps total nécessaire pour l'exécution d'une telle séquence est appelé longueur de cycle ou période.

Le choix d'une approche de modélisation dépend de la nature du problème traité. Quel que soit ce choix, une description préalable du problème est nécessaire, suivie d'une analyse mathématique et logique précise de ses hypothèses. Cette analyse nous permet de définir un ensemble de lemmes et corollaires garantissant la faisabilité et/ou l'optimalité de la solution.

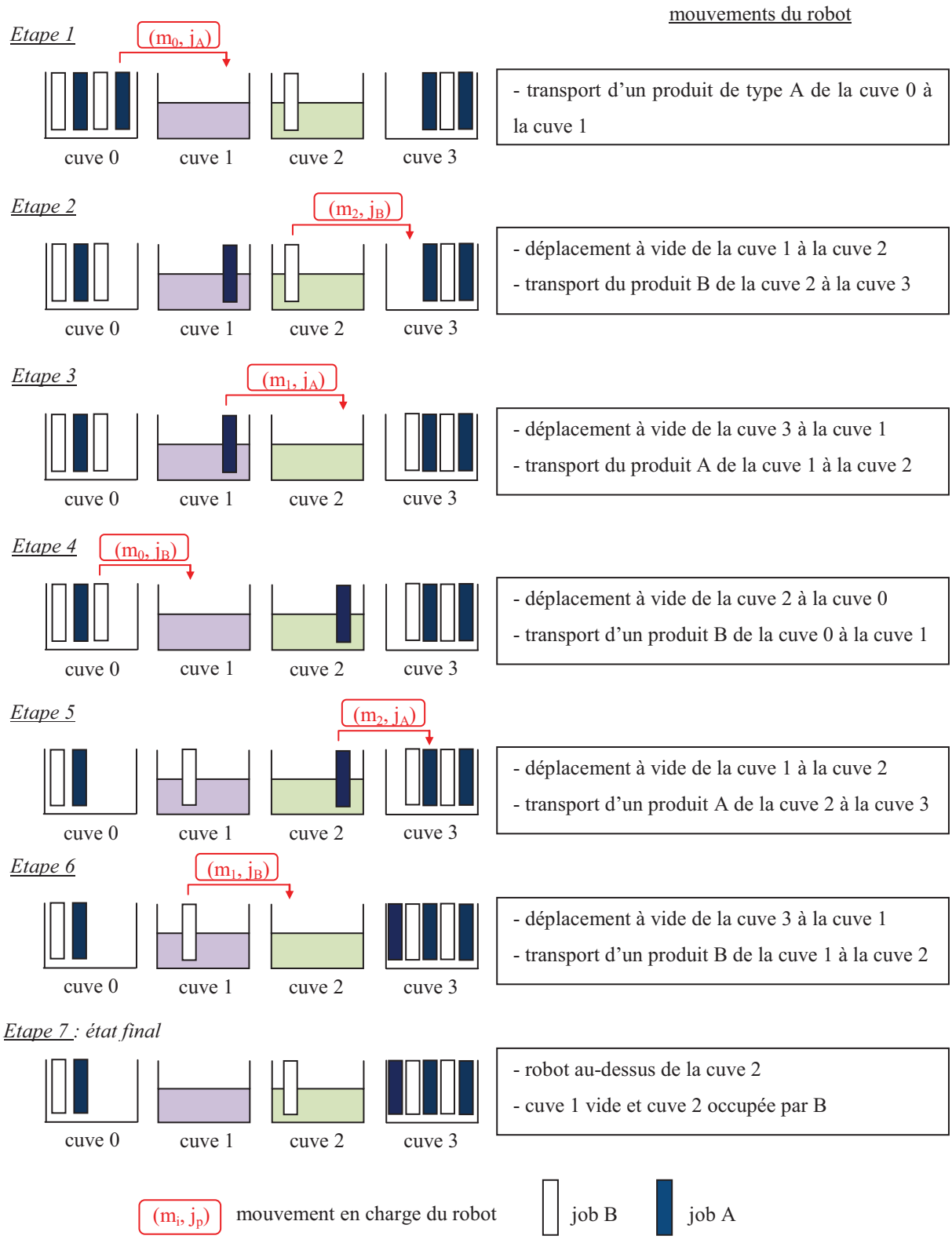



**Figure II.6.** Les étapes du déroulement de la séquence S

## II. 2. 4 - Etude de faisabilité

Les contraintes pouvant occasionner la non-faisabilité d'une séquence cyclique mono-produit restent valables pour un ordonnancement 2-cyclique hétérogène. Toutefois, d'autres conflits sont possibles et peuvent engendrer la violation des contraintes du problème.

Dans ce sens, nous pouvons visualiser, sur le diagramme de Gantt de la figure II.7, les différents conflits possibles, qu'ils soient liés au robot (1) ou aux cuves (2). En effet, dans le cas (1) le robot doit en même temps déplacer le produit 1 de la cuve 2 à la cuve 3, et le produit 2 de la cuve 4 à la cuve 5. Ceci est impossible dans le cas des lignes à une seule ressource de transport. Quant au conflit (2), il présente une violation de l'hypothèse de mono-capacité des ressources (contrainte (C2)) où chaque cuve ne peut contenir qu'un seul produit à la fois. Ainsi, dans le cas (2) la cuve 2 est déjà occupée par un produit 1, quand le robot vient déplacer un produit 2 de la cuve 2 à la cuve 3.

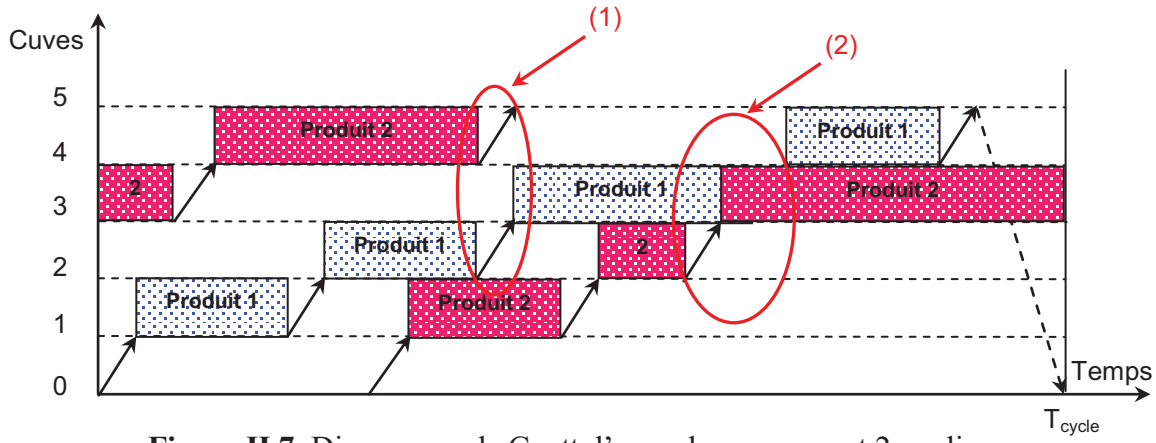



**Figure II.7.** Diagramme de Gantt d'un ordonnancement 2-cyclique

L'étude de faisabilité réalisée ici représente une analyse complémentaire à celle élaborée au paragraphe II.2.3 Elle s'exprime sous forme de lemmes et corollaire, qui définissent des conditions de faisabilité d'un ordonnancement 2-cyclique hétérogène du point de vue du type de produit.

***Lemme 1.***

Un ordonnancement cyclique est faisable si et seulement si la séquence cyclique *S* satisfait la condition suivante : entre chaque paire de mouvements $((m_i, \bullet), (m_i, \bullet))$, il existe un et un seul mouvement $(m_{i+1}, \bullet)$.

***Preuve.***

Pour démontrer ceci, on procède par un raisonnement par l'absurde.

On suppose qu'entre deux mouvements ($m_i$, ●) il existe tout d'abord plus d'un mouvement ($m_{i+1}$, ●) dans ce cas il existe deux mouvements ($m_{i+1}$, ●) et la faisabilité de la séquence dépendra du nombre de mouvements ($m_{i+2}$, ●) existant entre les mouvements ($m_{i+1}$, ●) et ainsi l'hypothèse reste récurrente.

On suppose qu'entre deux mouvements ($m_i$, ●) aucun mouvement ($m_{i+1}$, ●) n'existe entre les deux mouvements ($m_i$, ●) dans ce cas la cuve (i+1) sera occupée par deux produits en même temps, or les cuves sont de capacité unitaire et donc la séquence n'est pas faisable.

Ce qui est absurde.

Traiter simultanément deux types de produits peut entrainer également des contraintes supplémentaires au niveau des contraintes d'occupation de cuves. En effet, chaque séquence cyclique *S* doit respecter une deuxième règle d'ordre liée à l'hétérogénéité de l'ordonnancement. Cette règle se résume dans le *lemme 2*.

***Lemme 2.***

Un ordonnancement cyclique multi-produits (hétérogène) est faisable si et seulement si, la séquence cyclique *S* satisfait la condition suivante : entre chaque paire de mouvements (($m_i$, ●), ($m_i$, ●)), il existe *un et un seul* mouvement ($m_{i+1}$, ●) pour le même type de produit que le premier mouvement ($m_i$, ●) de la séquence.

***Preuve.***

On raisonne par l'absurde pour démontrer ce *lemme 2*.

On suppose le cas d'une séquence cyclique faisable où entre deux mouvements ($m_i$, *A*) et ($m_i$, *B*) il existe un seul mouvement ($m_{i+1}$, *B*). L'exécution du mouvement ($m_{i+1}$, *B*) par le robot signifie qu'il existe un produit de type *B* à la cuve i, or l'opération ($m_i$, *A*) exécutée au préalable se traduit par l'occupation de la cuve i par un produit de type A et comme les cuves sont considérées à capacités unitaires alors c'est absurde.

En conclusion et d'après le *lemme 2*, la faisabilité d'un ordonnancement 2-cyclique hétérogène dépend de l'ordre des mouvements ($m_i$, ●) et ($m_{i+1}$, ●) des deux types de produits présents sur la ligne. Par conséquent et d'après le *corollaire 1*, un nombre fini d'ordres

partiels des mouvements ($m_i$, ●) et ($m_{i+1}$, ●) est autorisé pour un ordonnancement 2-cyclique multi-produits.

***Corollaire 1.***

Suite à ce que nous avons démontré au *lemme 2*, nous pouvons affirmer que dans le cas d'un ordonnancement 2-cyclique hétérogène, il existe exactement quatre ordres partiels des mouvements ($m_i$, ●) et ($m_{i+1}$, ●) assurant la faisabilité de la séquence cyclique *S*.

Cet ordre partiel est défini par le type de pièces présentes sur la ligne au début du cycle. Ainsi, en considérant les deux cuves i et (i+1), on peut valider les seuls quatre ordres partiels faisables suivants :

*Cas 1* : la cuve *i* est occupée par un produit de type $j_A$, la cuve i+1 est vide au début du cycle et la première opération de transport de l'ordre partiel est ($m_i$, $j_A$) ; dans ce cas, le seul ordre partiel faisable est : $\langle (m_i, j_A), (m_{i+1}, j_A), (m_i, j_B), (m_{i+1}, j_B) \rangle$ ;

*Cas 2* : les cuve *i* est occupée par un produit de type $j_B$, la cuve i+1 est vide au début du cycle et la première opération de transport de l'ordre partiel est ($m_i$, $j_B$); dans ce cas, le seul ordre partiel faisable est : $\langle (m_i, j_B), (m_{i+1}, j_B), (m_i, j_A), (m_{i+1}, j_A) \rangle$ ;

*Cas 3* : la cuve *i+1* est déjà occupée par un produit de type $j_A$ au début du cycle et la première opération de transport de l'ordre partiel est ($m_{i+1}$, $j_A$); dans ce cas, le seul ordre partiel faisable est : $\langle (m_{i+1}, j_A), (m_i, j_B), (m_{i+1}, j_B), (m_i, j_A) \rangle$ ;

*Cas 4* : la cuve *i+1* est déjà occupée par un produit de type $j_B$ au début du cycle et la première opération de transport de l'ordre partiel est ($m_{i+1}$, $j_B$); dans ce cas, le seul ordre partiel faisable est : $\langle (m_{i+1}, j_B), (m_i, j_A), (m_{i+1}, j_A), (m_i, j_B) \rangle$ ;

Cette analyse de faisabilité de la séquence cyclique nous sera utile pour définir les contraintes disjonctives dans la modélisation linéaire que nous proposons dans le paragraphe suivant.

# II. 3 - Modélisation linéaire du CHSP

## II. 3. 1 - Notion et notations

Dans un but de simplification du modèle, nous avons fait appel à une nouvelle indexation inspirée de la notion de *stage* introduite par Mateo et Companys dans [Mateo et Companys, 2006]. Avec cette nouvelle indexation, chaque opération de transport ($m_i$, ●) sera réduite à un

seul paramètre *k,* définissant à la fois le numéro de l'opération de transport (i) et le type de produit à déplacer (*A* ou *B*). Dans cette optique, à chaque opération ($m_i$, ●) on fait correspondre un nouveau paramètre *k* tel que :

- si un mouvement *k* est pair alors c'est un mouvement de transport d'un produit de type *A* de la cuve *i* à la cuve (*i+1*) ou du stage (*2i*) au stage (*2i+2*), (*k=2i, ∀i = 0,…nc*).
- si un mouvement *k* est impair alors c'est un mouvement de transport d'un produit de type *B* de la cuve *i* à la cuve (*i+1*) ou du stage *(2i+1)* au stage *(2i+3)* (*k=2i+1, ∀i = 0,…nc*). (figure II.8)

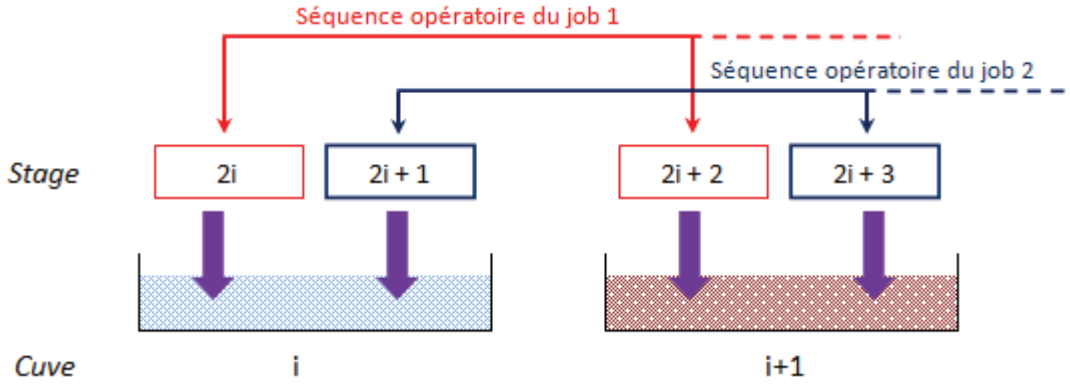



**Figure II.8.** Différence entre cuve et stage pour un problème 2-cyclique hétérogène

Compte tenu de l'introduction de cette notion de stage, nous complétons ou modifions les notations définies au paragraphe II.2.2, et nous utiliserons les nouvelles notations pour la description et la mise en équations du problème abordé :

| | |
|---|---|
| n | nombre total des mouvements de transport dans la séquence cyclique S<br>n = 2nc + 2 |
| $s_i$ | numéro de la cuve du traitement correspondant au stage i (i = $s_i$) et relation avec n |
| $o_i$ | durée réelle de l'opération de trempe au stage i |
| $a_i$ | durée minimale de l'opération de trempe au stage i |
| $b_i$ | durée maximale de l'opération de trempe au stage i |
| $O^H_i$ | opération de transfert d'un produit de la cuve du traitement du stage i à la cuve suivante |
| $d_i$ | durée du déplacement en charge à partir du stage i (durée de l'opération $O^H_i$) |
| $c_{si,sj}$ | durée du déplacement à vide du stage i au stage j |

| | |
|---|---|
| $t_i$ | date de début d'exécution de l'opération $O^H_i$, donc date de retrait du premier porteur de la cuve du stage i. $t_i$ correspond aussi à la fin de l'opération $O^T_i$ |
| M | un entier très grand représentant $+\infty$ |

## II. 3. 2 - Modèle linéaire (*S1*) du problème de base 2-cyclique hétérogène

En nous basant sur la nature du CHSP, nous nous proposons de développer une modélisation linéaire mono-objectif du problème multi-produits, minimisant le temps de cycle. En étudiant la description donnée par Manier [Manier et Baptiste, 1994] et Lamothe [Lamothe, 1996], on remarque que parmi les modèles linéaires proposés dans la littérature, deux modèles sont utilisés pour résoudre le problème de base (voir chapitre I) :

- Le modèle dit *par porteur,* qui traite de l'ensemble des opérations relatives aux porteurs pouvant se trouver simultanément dans le système ; [Manier et Baptiste, 1994], [Shapiro et Nuttle, 1988], [Lei et Wang, 1989][a].
- Le modèle dit *par période,* qui exprime les contraintes liant les opérations intervenant au cours d'un cycle donné de période T ; [Phillips et Unger, 1976], [Lei, 1993], [Lei et Wang, 1994], [Armstrong et al, 1994], [Lamothe, 1996].

Pour des raisons de simplifications et afin de tenir compte des différents types de produits présents sur la ligne, nous avons opté pour un modèle par porteur que nous décrivons ci-après [El Amraoui et al, 2010][a].

- **Fonction objectif :** le seul critère qu'on vise à optimiser est la durée du cycle *T*.

| | |
|---|---|
| Minimiser T | (II.13) |

- **Contraintes du problème:**

Contraintes sur les cuves : elles régissent la durée d'occupation de chaque cuve par chaque type de produit et bornent les durées réelles de trempe dans les cuves.

| | | |
|---|---|---|
| $d_i + a_{i+2} - M(1 - y_{i,i+2}) \leq t_{i+2} - t_i \leq d_i + b_{i+2} + M(1 - y_{i,i+2})$ | $\forall i = 0,...,n-3$ | (II.14) |

| | | |
|---|---|---|
| $d_i + a_{i+2} - My_{i,i+2} \leq t_{i+2} - t_i + T \leq d_i + b_{i+2} + My_{i,i+2}$ | $\forall i = 0,...,n-3$ | (II.15) |

Contraintes sur le robot : elles permettent de garantir à la ressource de transport de se déplacer à vide entre deux mouvements de transport quelconques, quelque soit l'ordre d'exécution de ces deux mouvements (contraintes disjonctives).

| | | | |
|---|---|---|---|
| $t_j - t_i \geq d_i + c_{s_{i+2},s_j} - M(1-y_{i,j})$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (II.16) |

| | | | |
|---|---|---|---|
| $t_i - t_j \geq d_j + c_{s_{j+2},s_i} - My_{i,j}$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (II.17) |

Bornes inférieures : ces bornes nous permettent de déduire des valeurs minimales à respecter par les variables $t_i$ et $T$. En effet, la date de début de l'un des mouvements en charge de la séquence cyclique doit être au moins égale à la somme du temps de déplacement en charge du premier produit de la cuve de chargement à la cuve 1 et du temps de déplacement à vide du robot de la cuve 1 jusqu'à la cuve de chargement du prochain traitement (II.18). Quant à la durée du cycle, elle est tout le temps supérieure ou égale à la date de départ d'un mouvement en charge auquel on rajoute la durée du déplacement en charge et le déplacement à vide depuis la cuve de destination jusqu'à la cuve 0 : position initial du robot au début du cycle (II.19).

| | | |
|---|---|---|
| $t_i \geq d_0 + c_{s_2,s_i}$ | $\forall i = 1,...,n-1$ | (II.18) |

| | | |
|---|---|---|
| $T - t_i \geq d_i + c_{s_{i+2},s_0}$ | $\forall i = 1,...,n-1$ | (II.19) |

Contraintes de faisabilité : ces deux contraintes assurent la faisabilité de la séquence cyclique. En effet, la première contrainte (II.20) garantit la condition définie dans le *lemme 1* : durant une séquence 2-cyclique, si deux opérations de trempe, appartenant à deux différents types de produits, se partagent la même cuve, alors au maximum une des deux tâches peut commencer dans un cycle et prendre fin dans le cycle suivant. Quant à la deuxième contrainte (II.21), elle assure la faisabilité de la séquence de point de vue type de produits (*lemme 2*).

| | | | |
|---|---|---|---|
| $y_{i,i+2} + y_{i+1,i+3} \geq 1$ | Si $(s_i = s_{i+1})$ | $\forall i = 2,...,n-4$ | (II.20) |

| | | | |
|---|---|---|---|
| $y_{i+2,i+1} + y_{i+3,i} \geq 3 - y_{i,i+2} - y_{i+1,i+3}$ | Si $(s_i = s_{i+1})$ | $\forall i = 2,...,n-4$ | (II.21) |

Contraintes binaires: elles permettent de définir la matrice binaire $y_{i,j}$ traduisant l'ordre d'exécution des opérations de transport en charge au cours d'un cycle.

| | | | |
|---|---|---|---|
| $y_{i,j} \in \{0, 1\}$ | | $\forall i, j = 0,...,n-1$ | (II.22) |

Contraintes de cohérence : elles sont utilisées pour bien définir la relation liant les variables $y_{i,j}$ et $y_{j,i}$, où $y_{i,j} = (1-y_{j,i})$ et $y_{j,i} = (1-y_{i,j})$.

| | | | |
|---|---|---|---|
| $(1-y_{i,j}) \geq y_{j,i}$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (II.23) |

| | | | |
|---|---|---|---|
| $y_{j,i} \geq (1-y_{i,j})$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (II.24) |

Contraintes sur les variables: elles définissent les variables positives et sont redondantes avec les équations II.25 et II.26, mais sont représentées juste à titre indicatif.

| | | |
|---|---|---|
| $t_i \geq 0$ | $\forall i = 0,...,n-1$ | (II.25) |

| | |
|---|---|
| $T \geq 0$ | (II.26) |

Contraintes d'initialisation : dans la contrainte II.27, la date du premier mouvement de transport du premier produit est choisie comme date initiale du cycle considéré, fixée à zéro dans la contrainte II.28.

| | | |
|---|---|---|
| $y_{0,i} = 1$ | $\forall i = 1,...,n-1$ | (II.27) |

| | |
|---|---|
| $t_0 = 0$ | (II.28) |

## II. 3. 3 - Complexité du modèle

Dans le modèle linéaire mixte que nous proposons, la séquence cyclique *S* recherchée, est déterminée par l'ordre et les dates d'exécution des mouvements de transport du robot (valeur des variables binaires $y_{i,j}$ et des variables $t_k$), et par la durée du cycle *T*. Ainsi, le nombre total *N* de variables est :

| | |
|---|---|
| $N = 2(nc+1)+1+\frac{2nc(2nc+1)}{2} = n+1+\frac{(n-1)(n-2)}{2}$ | (II.29) |

Dans le modèle proposé, le nombre de contraintes est donné par :

| | |
|---|---|
| $Nc = 2(n-2)(n+2)+(3n-8)$ | (II.30) |

Le nombre total de variables est donc $N = \theta(\frac{n^2}{2})$ et le nombre total de contraintes du modèle linéaire est $Nc = \theta(2n^2)$. Le tableau II.1 donne la valeur de ces nombres pour diverses tailles d'instance.

Il est important de noter que la complexité du problème traduit la durée de résolution du système linéaire. Cette durée est, comme on peut le constater, dépendante du nombre de contraintes disjonctives à arbitrer et par conséquent à la fois du nombre de solutions du problème et de la taille de la séquence cyclique. Toutefois, la taille de cette séquence dépend directement des produits à traiter (nombre, gammes opératoires, …) et des caractéristiques physiques de la ligne (nombre de cuves, …).

**Tableau II.1.** Nombre de contraintes et de variables pour 2 types de produits

| *Nb. de traitements par produit (n/2)* | *Nc* | *N* |
|---|---|---|
| 5 | 214 | 47 |
| 6 | 308 | 68 |
| 7 | 418 | 93 |
| 8 | 544 | 122 |
| 9 | 686 | 155 |
| 10 | 844 | 192 |

## II. 3. 4 - Procédure de résolution

La résolution du modèle se déroule en deux étapes :

- Etape 1 : rechercher un ordonnancement réalisable sous la forme d'une séquence cyclique des mouvements de transport assurés par le robot. Cette séquence doit respecter toutes les contraintes du système.
- Etape 2 : rechercher la solution optimale minimisant la durée du cycle pour l'ordonnancement déterminé à l'étape 1.

**Etape 1** : Traite les disjonctions par une procédure de type séparation et évaluation (*branch-and-bound)*.

La première étape indiquée consiste à construire progressivement un ordonnancement réalisable. La construction de la solution fait appel à une procédure de type séparation et évaluation. A chaque niveau de l'arbre, est vérifiée la consistance du système de contraintes. En effet, à chaque nœud de l'arbre un couple d'opérations de transport, aléatoirement choisi, est ordonné. Ceci revient à instancier les variables $y_{i,j}$ du modèle; et à vérifier sa consistance grâce à la propagation des contraintes, illustrée par la figure II.9. Si le système s'avère consistant alors, dans ce cas, un nouveau couple de mouvements peut être ordonné, c'est-à-dire qu'un nouveau nœud fils peut être considéré. Sinon, un *backtrack* (retour arrière) est autorisé et l'ordre inverse des mouvements est considéré. Une fois que tous les couples de mouvements (ou d'opérations) sont ordonnés, c'est à dire une fois que toutes les disjonctions sont arbitrées, on aboutit à un système réduit d'équations définissant une séquence cyclique faisable. La résolution de ce système réduit permet de déterminer la ou les solutions associées à cette séquence de mouvements. Chaque solution trouvée est caractérisée par ses dates de début de mouvements en charge (dont les durées effectives de trempe sont déduites) et la durée de son cycle (*T*).

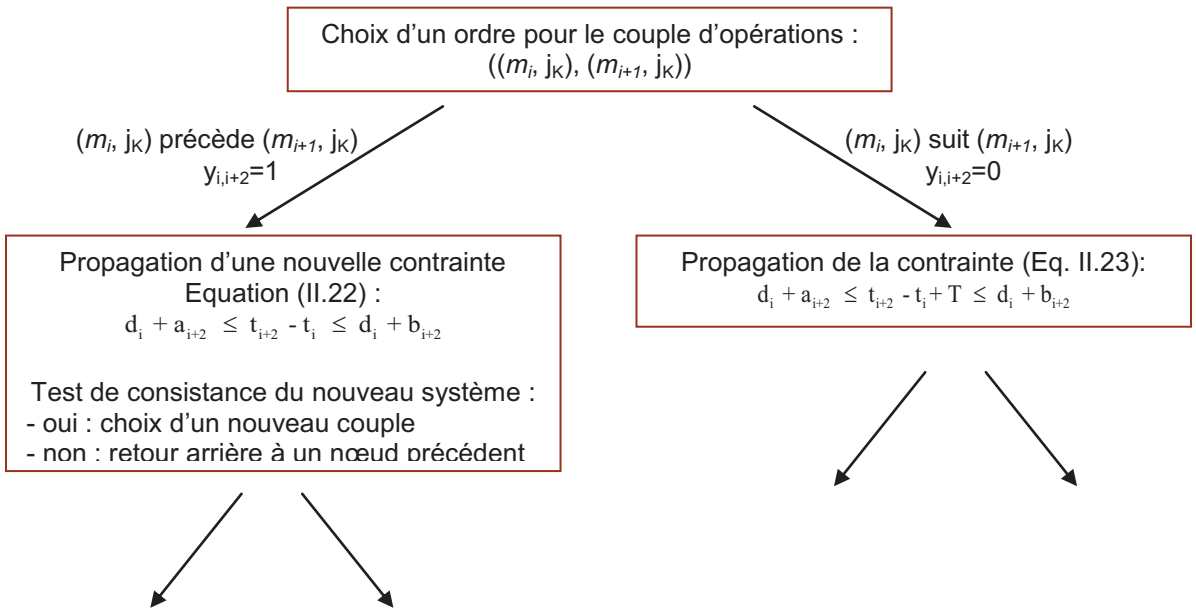



**Figure II.9.** Procédure de recherche des ordonnancements réalisables

**Etape 2** : rechercher le meilleur ordonnancement réalisable relatif à la solution optimale.

A partir de la dernière feuille atteinte de l'arbre et jusqu'à ce que l'arbre de recherche ait été entièrement exploré, des *backtrack*s sont autorisés et les procédures de l'étape 1 sont exécutées afin d'aboutir à la solution optimale (figure II.10).

Ainsi, suite à ces deux étapes, nous arrivons à résoudre d'une façon optimale notre système d'équations (*S1*) et nous aboutissons finalement à l'ordonnancement recherché.

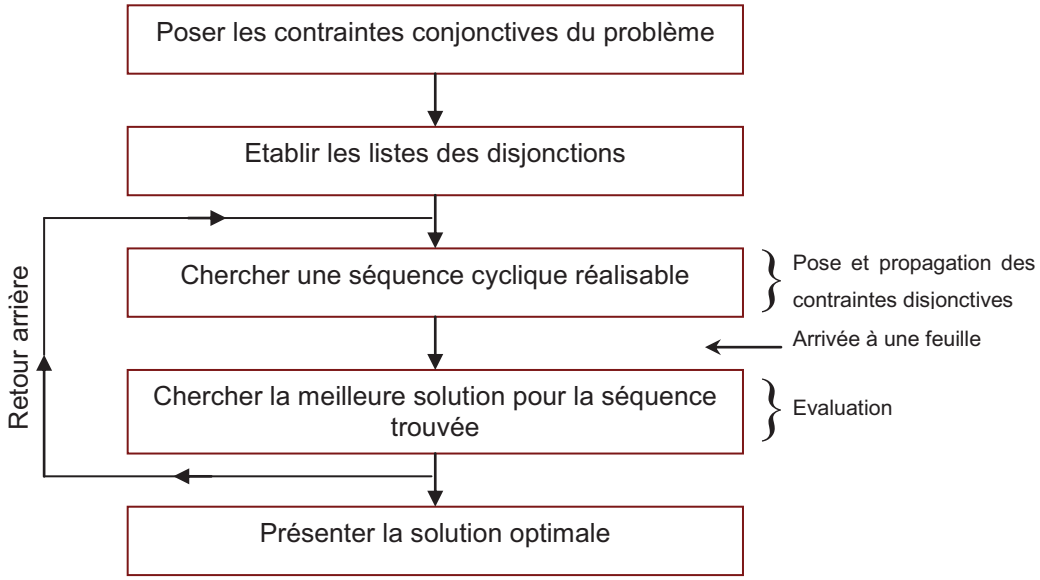



**Figure II.10.** Méthode de résolution

## II. 3. 5 - Exemple illustratif

Pour illustrer le modèle élaboré, considérons l'exemple du tableau II.2., relatif à deux types de produits sur une ligne à 3 cuves non incluant les postes de chargement et de déchargement (cuves 0 et 4 considérées dissociées). La durée de déplacement du robot en charge (respectivement à vide) entre deux cuves successives est de 10 u.t. (respectivement 5 u.t.).

**Tableau II.2.** Exemple de gammes opératoires partiellement identiques [Mateo, 2001]

| Type de produit | Fenêtre de temps | Cuve i | | | |
|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 |
| Job 1 | $[a_{2i}, b_{2i}]$ | [0 - +∞] | [80 - 96] | [41 - 46] | [61 - 66] |
| Job 2 | $[a_{2i+1}, b_{2i+1}]$ | [0 - +∞] | [76 - 91] | [21 - 26] | [30 - 66] |

En appliquant le modèle linéaire mixte proposé à cet exemple, nous trouvons la solution optimale de la littérature qui minimise la durée totale du cycle hétérogène, et ce en un temps de simulation très court (au plus quelques secondes). Cette séquence cyclique est représentée par le diagramme de Gantt de la figure II.11. Pour cette solution, nous obtenons :

- la durée de cycle minimale : T = 272 u.t. (unité de temps)
- la séquence cyclique des mouvements :

$$S = \langle (m_0, j_1), (m_2, j_2), (m_3, j_2), (m_1, j_1), (m_2, j_1), (m_0, j_2), (m_3, j_1), (m_1, j_2) \rangle$$

- le vecteur des dates de début des opérations de transport :

$$X = \langle t_{[0]}=0, t_{[1]}=166, t_{[2]}=90, t_{[3]}=252, t_{[4]}=141, t_{[5]}=15, t_{[6]}=212, t_{[7]}=55 \rangle$$

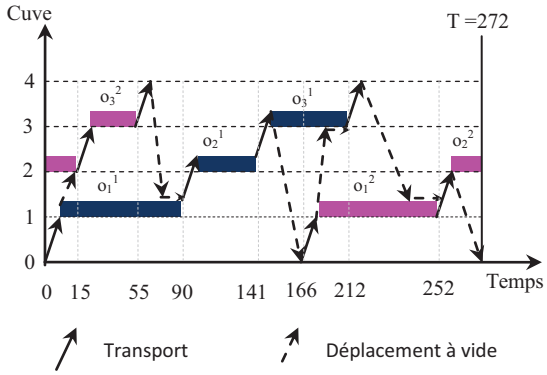



**Figure II.11.** Diagramme de Gantt de la solution optimale

Ces résultats, permettent de retrouver facilement les durées opératoires effectivement réalisées durant le processus de traitement du tableau II.3, obtenu par application de la formule suivante :

| | | |
|---|---|---|
| $o_i = t_i - t_{i-2} - d_{s_{i-2}}$ | $\forall i = 2,...,n-3$ | (II.31) |

**Tableau II.3.** Durées opératoires effectives pour chaque type de produit

| Type de produit | Cuve i | | |
|---|---|---|---|
| | 1 | 2 | 3 |
| Job 1 | 80 | 41 | 61 |
| Job 2 | 76 | 25 | 30 |

Pour cet exemple, les durées opératoires sont toutes égales à la borne minimale de la fenêtre de temps, exceptée la durée opératoire du second produit dans la seconde cuve de traitement.

Nous remarquons également, que pour assurer l'optimalité de la solution, le robot est resté immobile au dessus de la cuve 2, durant toute la durée du traitement du produit 1 dans cette cuve. Ceci s'explique par une durée opératoire dans cette cuve faible par rapport aux autres durées de trempe, et du même ordre de grandeur que les durées de déplacement du robot.

## II. 3. 6 - Résultats

### a. Description des benchmarks

Dans la littérature, il existe très peu de benchmarks sur le *HSP*, et encore moins sur le *CHSP* multi-produits. Nous n'avons validé notre programme linéaire mixte que pour les instances proposées dans [Mateo, 2001]. Seules instances disponibles traitant du CHSP multi-produits suivant la même séquence opératoire mais avec des bornes de durées différentes.

Ces données sont générées de façon aléatoire. Les bornes minimales des fenêtres de temps ($a_i$) sont générées entre 20 et 80 par une loi de distribution uniforme. Les bornes maximales sont générées en fonction des valeurs des $a_i$ et de la largeur de la fenêtre de temps. Mateo définit trois classes de Fenêtre de temps : Large (FL), $b_i$= [$2a_i$, $10a_i$]; Moyenne (FM), $b_i$= [$1.5a_i$ $2a_i$] ; et des Etroite (FE), $b_i$=U[$1.2a_i$, $1.5a_i$]. A chaque fenêtre, il fait correspondre trois vitesses du Robot : Rapide (RR), avec $d_i$=$1.5c_{i,i+1}$; Moyenne (RM), avec $d_i$=$2c_{i,i+1}$; et Lente (RL),avec $d_i$=$3c_{i,i+1}$. Les temps de déplacement à vide $c_{i,i+1}$ sont générés de façon aléatoire entre 5 et 10 u.t.

En combinant par paires les fenêtres de temps ((FE/FM) et (FE/FL)) aux trois types de vitesses du robot, six séries d'instances sont générées et pour chaque série, cinq benchmarks sont considérés. Ces instances sont générées pour des lignes de traitements à 5, 6, 7, 8, 9 et 10 cuves de traitements. Ainsi, le nombre total de benchmarks produits est de 180 benchmarks. Les données complètes de ces instances sont détaillées en annexe 5.

### b. Validation du modèle *(S1)*

Toutes ces instances ont été résolues à l'aide de notre programme linéaire en nombres entiers, implémenté avec le solveur Cplex 10.1 de l'éditeur ILOG, sur un processeur Intel® Core™2 Duo de fréquence 2 GHZ. Pour chacune d'elles, nous avons atteint l'optimum en un temps de simulation ne dépassant pas les 15 minutes pour des lignes à 10 cuves de traitements. Les résultats de ces instances sont reportés dans l'annexe 6 de ce mémoire. Elles nous ont permis de vérifier et valider le modèle de base élaboré.

En annexe 6, on trouve également le temps de cycle du premier type de produits ($T_1$) ainsi que le temps de cycle du second type de produits ($T_2$) dans un cas de fonctionnement mono-produit. Pour déterminer les variables $T_1$ et $T_2$, nous avons élaboré un modèle linéaire mono-produit (donné en annexe 7), en nous inspirant du modèle de référence de [Phillips et Unger, 1976], (voir annexe 8) ainsi que des modèles de [Manier-Lacoste, 1994] et [Liu et al, 2002]. Compte tenu des résultats obtenus, nous avons calculé pour chaque ordonnancement 2-cyclique, le rapport de réduction *R(%)* définissant l'écart relatif entre le temps de cycle de l'ordonnancement 2-cyclique hétérogène et la somme des temps de cycle fournis par les ordonnancements 1-cycliques homogènes [Mateo et al, 2000]. Ce rapport de réduction est donné par l'équation suivante :

$$R(\%) = \frac{(T_1 + T_2) - T}{(T_1 + T_2)} \times 100 \qquad \text{(II.32)}$$

Les résultats de ces réductions sont rapportés dans le tableau II.4 et exposés dans l'histogramme des figures II.12 et II.13. Chaque réduction du tableau correspond à une moyenne des écarts obtenus pour les 5 instances générées pour chaque triplet (nombre de cuves, type de fenêtre, vitesse du robot). En fait, ces réductions peuvent être considérées comme étant en partie sous-estimées. En effet, considérer successivement deux productions mono-produit distinctes sous-entend d'attendre d'avoir vidé la ligne du premier type de produits avant de commencer le second cycle. A moins de parvenir à imbriquer les deux cycles, comme [Varnier, 1997], cela suppose une diminution du rendement lors de cette transition. Aussi, traiter différents types de produits simultanément peut s'avérer plus intéressant pour les industriels afin de leur permettre de réduire à la fois les coûts de production et les délais de livraison tout en augmentant la production et par conséquent le profit.

Les comparaisons que nous venons d'effectuer montrent l'intérêt du développement d'un modèle 2-périodique multi-produits, par rapport aux modélisations 1-périodiques généralement proposées dans la littérature.

D'après les résultats du tableau II.4, on remarque que le rapport de réduction *R(%)* varie d'une façon assez considérable atteignant une valeur moyenne de *36,56%* pour des lignes à 10 cuves de traitements, pour une fourchette de fenêtres de trempe (FE/FM) et un robot rapide. La figure II.12 montre l'évolution de la moyenne du rapport (moyenne de chaque ligne du tableau II.4) en fonction des fenêtres de temps et de la vitesse du robot. Cette valeur varie de

façon proportionnelle en fonction de la vitesse du robot. De plus, le gain est plus intéressant pour des fenêtres de type (FE/FM) que pour des fenêtres de type (FE/FL). En effet, pour des fenêtres larges, on améliore de manière plus importante le temps de cycle d'un ordonnancement 1-périodique que celui d'une solution 2-périodique, d'où un écart relatif moindre.

**Tableau II.4.** Ecart moyen entre un cycle 2-périodique hétérogène et les 2 cycles 1-périodique homogènes

| Fenêtre | Vitesse du robot | Nombre de cuves de traitements | | | | | |
|---|---|---|---|---|---|---|---|
| | | 5 | 6 | 7 | 8 | 9 | 10 |
| (FE/FM) | RL | *2,08%* | *-0,75%* | *13,61%* | *15,62%* | *30,03%* | *21,38%* |
| (FE/FM) | RM | *2,74%* | *12,11%* | *9,83%* | *23,01%* | *18,55%* | *27,58%* |
| (FE/FM) | RR | *11,14%* | *21,11%* | *16,24%* | *23,42%* | *31,92%* | *36,56%* |
| (FE/FL) | RL | *0,55%* | *4,93%* | *7,25%* | *3,20%* | *19,24%* | *13,25%* |
| (FE/FL) | RM | *4,91%* | *4,46%* | *6,99%* | *17,92%* | *22,86%* | *10,76%* |
| (FE/FL) | RR | *3,36%* | *13,66%* | *13,62%* | *7,04%* | *19,75%* | *25,71%* |

Une autre moyenne du rapport de réduction a été déterminée pour chaque taille de la ligne de traitement (moyenne par colonne du tableau II.4). Elle est représentée dans la figure II.13. Le gain obtenu varie de *4,13%* pour des lignes à 5 cuves à *23,72%* pour des lignes à 9 cuves. On remarque que ce rapport de réduction ne varie pas d'une façon linéaire en fonction du nombre de cuves. Toutefois, il confirme à nouveau qu'un ordonnancement 2-cyclique hétérogène, faisant alterner d'une façon régulière deux types de produits, peut améliorer la production d'une ligne de traitement de surface par rapport à des productions mono-gamme successives.

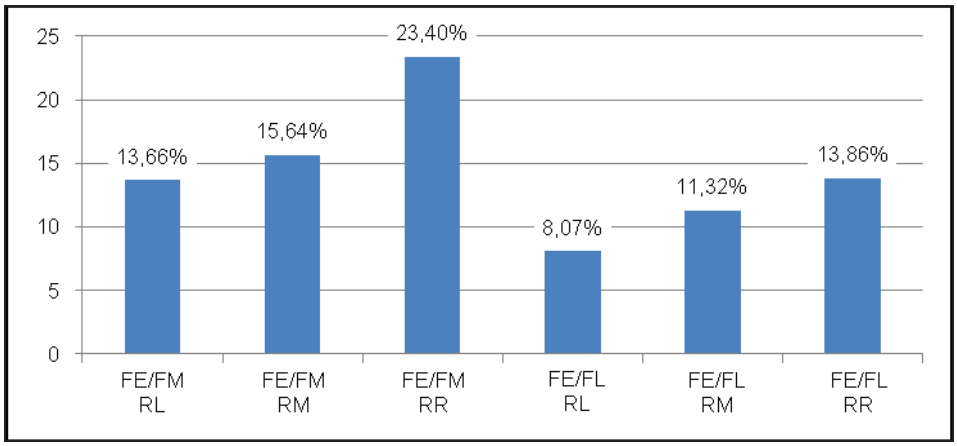



**Figure II.12.** Taux de réduction moyenne du temps de cycle en fonction de la fenêtre de temps et la vitesse du robot

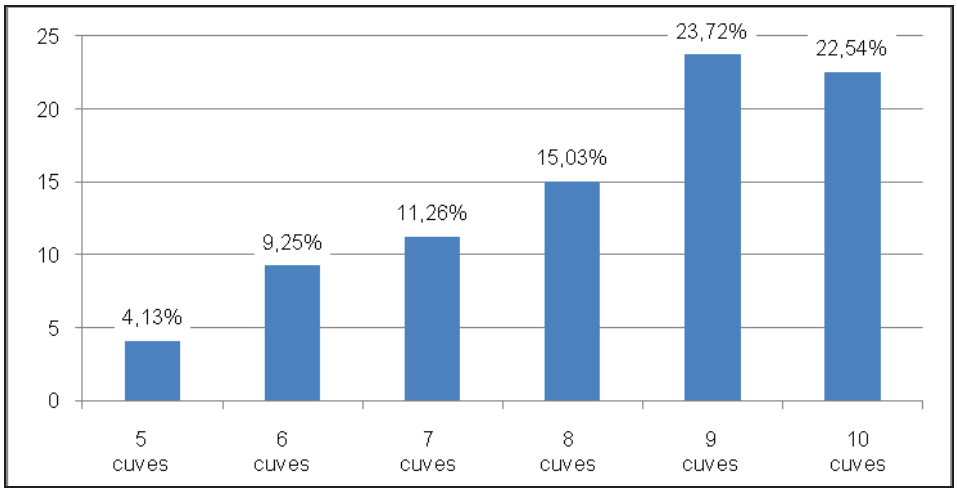



**Figure II.13.** Taux de réduction moyenne du temps de cycle en fonction de la taille de la ligne de traitement

# II. 4 - Etude de la structure et de la flexibilité d'une solution

## II. 4. 1 - Structure d'une solution

Une des caractéristiques du *Hoist Scheduling Problem* est relative aux opérations de trempe dont la durée est bornée par une durée minimale, à respecter impérativement et une durée maximale, dont le dépassement peut nuire à la qualité du produit ou l'endommager. Cette particularité, nous a incités à réfléchir sur la notion de robustesse d'une séquence cyclique. Cette notion a été déjà définie dans plusieurs disciplines. A défaut d'unicité, nous donnons ici une définition plutôt consensuelle qui a été proposée dans [Billaut et al, 2005] : "un ordonnancement est *robuste* si sa performance est peu sensible à l'incertitude des données et aux aléas". "Il faut, dans ce cas, entendre 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".

Nous nous limiterons dans notre analyse à l'étude de la meilleure solution d'un ordonnancement 2-cyclique hétérogène. Pour ce faire, nous définissons la notion de perturbation $\delta_i$ pour chaque opération de trempe. Cette notion traduit la possibilité qu'une séquence offre au niveau des variations de durées opératoires tolérables (figure II.14). Cette perturbation est comprise entre 0 et $\delta_{i\,max}$, avec :

| | |
|---|---|
| $\delta_i$ | perturbation au stage i, ($\delta_i = o_i - a_i$) |

| | |
|---|---|
| $\delta_{i\,max}$ | perturbation maximale de trempe au stage i, ($\delta_{i\,max} = b_i - a_i$) |

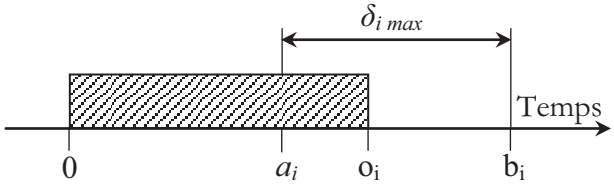



**Figure II.14.** Perturbation maximale tolérée pour une fenêtre de temps [$a_i$, $b_i$]

Nous commençons par étudier la robustesse de la solution de l'exemple illustratif proposé au paragraphe précédent (tableaux II.2 et II.3). On rappelle qu'un ordonnancement cyclique est défini par l'ensemble de trois grandeurs *S*, X et *T*, correspondant respectivement, à la séquence cyclique de mouvements, au vecteur des dates de début de ces opérations de transport et à la durée de cycle. Ainsi, la solution optimale de l'exemple illustratif est donnée par :

$$S^* = \langle (m_0, j_1), (m_2, j_2), (m_3, j_2), (m_1, j_1), (m_2, j_1), (m_0, j_2), (m_3, j_1), (m_1, j_2) \rangle$$

$$X^* = \langle t_{[0]}=0, t_{[1]}=166, t_{[2]}=90, t_{[3]}=252, t_{[4]}=141, t_{[5]}=15, t_{[6]}=212, t_{[7]}=55 \rangle$$

$T^* = 272$ u.t.

En utilisant la notion de *stage*, définie précédemment, cette séquence cyclique sera équivalente à la séquence suivante :

$$S^* = \langle 0, 5, 7, 2, 4, 1, 6, 3 \rangle$$

Pour étudier la robustesse de cette solution 2-cyclique hétérogène, on lui applique une certaine perturbation $\delta_i$ au niveau des fenêtres de temps. On suppose, dans ce cas, que la nouvelle solution optimale est donnée par (S', X', T'). En appliquant de telles perturbations à notre séquence cyclique, nous sommes parvenus à identifier cinq cas possibles.

*Cas 1* : *(S', X', T') = ($S^*$, $X^*$, $T^*$)*

En modifiant la borne minimale de la fenêtre de temps du produit 2 dans la cuve 2 par une des valeurs de l'intervalle [21 - 25] ($0 \leq \delta_5 \leq 4$), notre séquence cyclique reste inchangée. Ce qui est dû au fait que la durée de trempe effective du produit 2 dans cette cuve est de 25 u.t., soit la borne maximale.

*Cas 2* : *(S') = ($S^*$), (X') ≠ ($X^*$) et (T') = ($T^*$)*

En faisant passer la borne minimale de la durée de trempe du produit 2 dans la cuve 3 de 30 à 40 u.t., la séquence cyclique ainsi que le temps de cycle restent inchangés. Toutefois, le vecteur des temps de début des mouvements de transport subit une modification au niveau de sa dernière composante :

$$X^* = \langle t_{[0]}=0, t_{[1]}=166, t_{[2]}=90, t_{[3]}=252, t_{[4]}=141, t_{[5]}=15, t_{[6]}=212, t_{[7]}=65 \rangle$$

En effet, avec une telle perturbation, la durée de trempe du second produit dans la cuve 3 est augmentée de 10 u.t. ($\delta_7$=10 u.t.) et ainsi au lieu que le robot reste immobile au dessus de la cuvé 1 pendant 10 u.t. (durée nécessaire pour que le produit 1 termine sa première opération de trempe, comme dans le cas 1), le robot saisit le produit 1 aussitôt arrivé au dessus de la cuve 1.

*Cas 3* : *(S') = ($S^*$), (X') ≠ ($X^*$) et (T') ≠ ($T^*$)*

En faisant passer cette même borne minimale de 40 à 50 u.t. notre séquence cyclique demeure la même. Toutefois, le temps de cycle ainsi que le vecteur des temps de début des mouvements de transport sont modifiés et nous avons :

$$X^* = \langle t_{[0]}=0, t_{[1]}=176, t_{[2]}=100, t_{[3]}=262, t_{[4]}=151, t_{[5]}=15, t_{[6]}=222, t_{[7]}=75 \rangle$$

$$T^* = 282 \text{ u.t.}$$

En effet, avec une nouvelle perturbation ($\delta'_7$=10 u.t.), la durée de trempe du second produit dans la cuve 3 est augmentée de nouveau de 10 u.t. ainsi que T. Dans ce cas, aucun temps d'attente à vide n'est permis pour le robot, lui permettant d'absorber cette perturbation comme dans le cas précédent.

*Cas 4: (S') ≠ ($S^*$), (X') ≠ ($X^*$) et (T') ≠ ($T^*$)*

En considérant les données initiales de l'exemple étudié (fenêtre de temps [21, 26] pour le produit 2 dans la cuve 2) et en faisant passer la durée de trempe minimale du produit 2 dans la cuve 3 de 50 à 60 u.t. la solution cyclique est modifiée et, par conséquent, ses nouvelles grandeurs sont données par :

$$S^* = \langle 0, 7, 2, 4, 1, 6, 3, 5 \rangle$$

$$X^* = \langle t_{[0]}=0, t_{[1]}=166, t_{[2]}=90, t_{[3]}=252, t_{[4]}=141, t_{[5]}=283, t_{[6]}=212, t_{[7]}=45 \rangle$$

$$T^* = 308 \text{ u.t.}$$

Cette modification de l'ordonnancement est dû principalement au fait que les durées de trempes considérées sont relativement importantes par rapport aux durées de déplacement du robot.

*Cas 5: solution non faisable*

Si la borne minimale de la durée de trempe du produit 2 dans la cuve 3 passe de 60 à 67 u.t., le modèle linéaire ne pourra pas être résolu. Dans ce cas la borne minimale va dépasser la borne maximale de trempe et par conséquent, une nouvelle fenêtre de temps doit être redéfinie.

En conclusion, vu les cas de figure traités précédemment (figure II.15), nous pouvons affirmer qu'une séquence cyclique reste inchangée (S, X et T) tant que la perturbation imposée ne dépasse pas la valeur ($o_i$ - $a_i$). Cette assertion est traduite par le *lemme 3*.

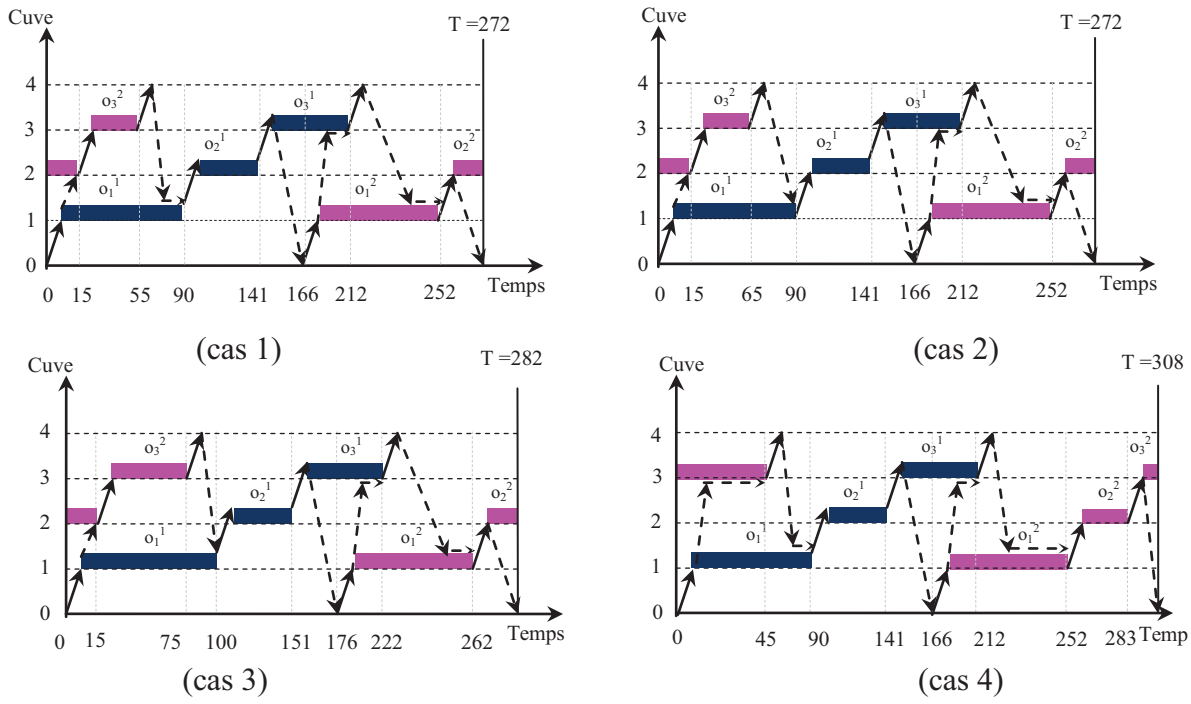



**Figure II.15.** Graphe de l'étude de la robustesse d'une séquence cyclique

***Définition :*** dans le cas d'un fonctionnement cyclique, un ordonnancement est dit *robuste* au sens large si l'ensemble de ses grandeurs (S, X, T) reste inchangé aux aléas.

***Lemme 3.***

Une solution cyclique (S, X, T) est robuste au sens large pour toute perturbation vérifiant la

condition suivante : $0 \leq \delta_i \leq o_i - a_i$. [El Amraoui et al, 2012]

***Preuve.***

Soit $a_i^{'} = a_i + \delta_i$, une nouvelle borne minimale de la durée de trempe.

Pour toute perturbation $\delta_i$ satisfaisant la condition suivante $0 \leq \delta_i \leq o_i - a_i$, la borne minimale de la fenêtre de temps reste inférieure à la durée de trempe effective ($a_i^{'} \leq o_i$) et par conséquent, l'ordonnancement reste faisable et inchangé.

Supposons que $\delta_i > o_i - a_i$; dans ce cas, la borne minimale de la fenêtre de temps devient supérieure à la durée de trempe effective, ce qui entraine la non faisabilité de la séquence cyclique. Et ainsi, une nouvelle solution doit être trouvée par le modèle linéaire.

Ce type d'analyse de la structure de la solution peut être utilisé en phase d'exploitation de la ligne, par exemple pour garantir des performances de l'ordonnancement prévisionnel face à un certain niveau de perturbation. En effet, on sait que l'usure des bains chimiques va nécessiter à termes un remplacement avec un coût important associé. Pour retarder cet investissement, il est possible de conserver le bain plus longtemps, mais cela suppose d'allonger le temps de trempe dans la cuve concernée pour une qualité équivalente des produits. En ce sens, notre étude fournit un domaine de validité de l'ordonnancement proposé, en termes d'allongement, avec une certaine garantie de performance (un niveau de robustesse).

## II. 4. 2 - Flexibilité opératoire d'une solution

### a. Domaine de validité d'une séquence cyclique

Nous nous proposons ici d'étudier la flexibilité offerte par une séquence cyclique multi-produits au niveau des variations de durées opératoires tolérables (les marges de trempe). Dans notre modèle 2-cyclique, présenté précédemment, nous modifions la fonction objectif en : maximiser *T*. Notre but est d'étudier des plages de validité possibles de chaque opération sur l'intervalle [min T, max T]. Ainsi, pour la séquence $S = \langle 0, 5, 7, 2, 4, 1, 6, 3 \rangle$, nous connaissons la valeur minimale de T correspondant à l'ordonnancement optimal. Le modèle linéaire modifié nous permet d'obtenir une valeur maximale de T à 335. La plage de validité de T pour cet exemple et sans fixer la séquence correspond donc à l'intervalle de tolérance [272, 335]. Par discrétisation de chaque valeur du temps de cycle incluse dans cet intervalle, nous calculons les durées minimales et maximales pour chaque opération de trempe. Les résultats obtenus sont reportés sur les figures II.16, II.17 et II.18 [El Amraoui et al, 2009][b].

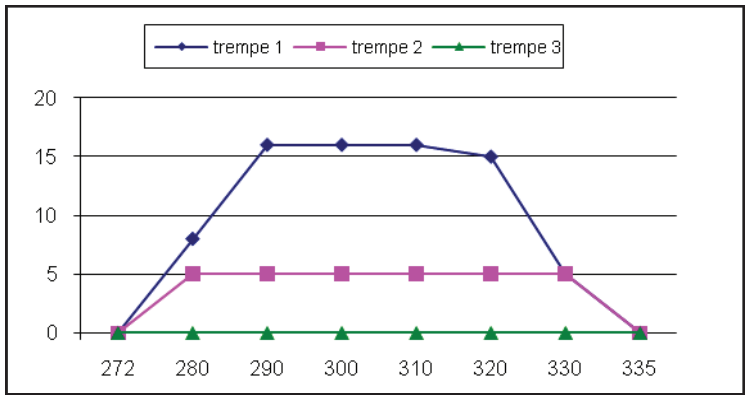



**Figure II.16.** Flexibilité des durées trempe pour le premier produit

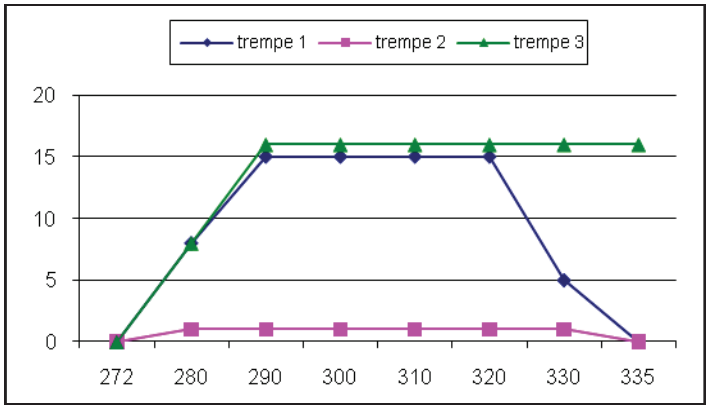



**Figure II.17.** Flexibilité des durées trempe pour le deuxième produit

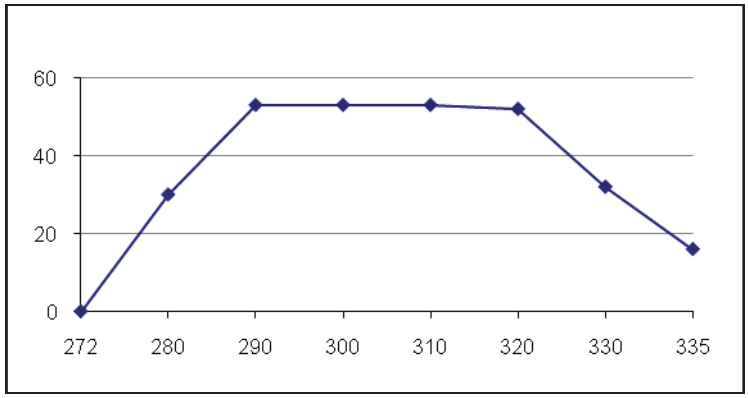



**Figure II.18.** Flexibilité totale de la solution cyclique

Ces résultats montrent que certaines durées de trempe tolèrent une flexibilité considérable face à des variations du temps de cycle. D'autres opérations sont critiques dans la mesure où aucune flexibilité n'est tolérée : c'est le cas de la troisième opération de trempe du produit 1.

De plus, sur les figures II.16 et II.17, nous observons que l'absence de flexibilité (marge $\delta_i$ nulle pour tous les traitements) correspond à la plus petite valeur possible de la période (T=272 u.t.). Ainsi, la flexibilité induite par les fenêtres temporelles au niveau des temps de trempe induit certes une augmentation de la complexité du problème à résoudre, mais permet d'obtenir de meilleures solutions cycliques.

Nous remarquons également que le phénomène de flexibilité n'est pas linéaire (figure II.18). En effet, si le temps de cycle passe de 272 à 280 u.t. la flexibilité totale tolérée pour toutes les opérations de trempe (somme des $\delta_i$) est de 30 u.t. soit le tiers de la variation du temps de cycle. Néanmoins, si le temps de cycle passe de 290 à 320 u.t., la flexibilité tolérée reste inchangée.

**b. Impact des fenêtres temporelles sur les performances d'une solution**

Dans le but de réduire la combinatoire de résolution du CHSP, certains chercheurs ont relaxé la contrainte de bornes des durées opératoires, en fixant la valeur a priori (en général à la durée minimale) [Song et al, 1993], [Kats et al, 2006], [Che et al, 2009]. De ce fait, ils ont proposé des algorithmes de résolution polynomiaux. Toutefois, cette hypothèse simplificatrice les prive de la flexibilité offerte par les fenêtres de temps et les empêche d'atteindre la solution optimale du problème. En fait, nous pouvons considérer qu'ils étudient plutôt un problème connexe au CHSP, de type Robotic Scheduling.

Pour mieux illustrer les différences au niveau des résultats obtenus pour ces deux problèmes, nous avons à nouveau résolu les instances de [Mateo, 2001], mais en modifiant la largeur des fenêtres de temps (avec borne maximale égale à la durée minimale de trempe). Chaque série de benchmarks correspond à deux largeurs de fenêtre de temps et chaque fenêtre correspond à trois vitesses de robot. Les résultats de ces simulations sont reportés dans le tableau II.5. Celui-ci fournit le temps de cycle *T'* (respectivement *T*) pour les instances dont les fenêtres de temps sont à largeurs nulles (respectivement à fenêtres de temps à largeurs non nulles), ainsi que le temps de simulation correspondant *CPU'* (respectivement *CPU*). Dans le tableau II.5 figure aussi le rapport de réduction $R_1(\%)$, que nous définissons comme étant l'écart relatif entre les temps de cycle des deux cas considérés. Il est calculé par l'équation II.33. Ces résultats traduisent bien l'impact de la largeur des fenêtres de temps sur les performances de la solution 2-cyclique.

| | |
|---|---|
| $R_1(\%) = \frac{T' - T}{T'} \times 100$ | (II.33) |

**Tableau II.5.** Influence de le fenêtre du temps sur la durée du cycle

| | | Référence | T' | CPU' | T | CPU | $R_1$(%) |
|---|---|---|---|---|---|---|---|
| Série 1 | 5 cuves | 105040905 | 719 | 123 | 319 | 110 | 55,63 |
| | | 102070205 | 711 | 112 | 333 | 118 | 53,16 |
| | | 209050705 | 767 | 125 | 377 | 116 | 50,85 |
| | | 201080405 | 730 | 163 | 494 | 163 | **32,33** |
| | | 301060405 | 1013 | 151 | 594 | 143 | 41,36 |
| | | 304090405 | 1005 | 141 | 558 | 143 | 44,48 |
| Série 2 | 6 cuves | 111041606 | 983 | 165 | 450 | 174 | 54,22 |
| | | 117071706 | 746 | 163 | 362 | 174 | 51,47 |
| | | 211051406 | 942 | 161 | 606 | 163 | 35,67 |
| | | 219081406 | 939 | 177 | 420 | 184 | 55,27 |
| | | 316061906 | 823 | 161 | 548 | 163 | 33,41 |
| | | 316091906 | 788 | 181 | 525 | 174 | 33,38 |
| Série 3 | 7 cuves | 127042807 | 1072 | 109 | 690 | 115 | 35,63 |
| | | 130072507 | 1073 | 223 | 426 | 236 | 60,30 |
| | | 226052507 | 1029 | 220 | 370 | 231 | **64,04** |
| | | 223082207 | 1048 | 169 | 579 | 175 | 44,75 |
| | | 325062907 | 1198 | 111 | 775 | 113 | 35,31 |
| | | 327092307 | 1044 | 225 | 438 | 237 | 58,05 |
| Série 4 | 8 cuves | 140043108 | 1077 | 171 | 585 | 172 | 45,69 |
| | | 134073908 | 1253 | 179 | 645 | 176 | 48,52 |
| | | 232053108 | 1073 | 86 | 676 | 76 | 37,00 |
| | | 240083808 | 983 | 288 | 410 | 278 | 58,29 |
| | | 337063608 | 1321 | 159 | 831 | 148 | 37,09 |
| | | 338093908 | 1291 | 336 | 627 | 345 | 51,43 |
| Série 5 | 9 cuves | 142044509 | 1327 | 125 | 799 | 115 | 39,79 |
| | | 142074509 | 1344 | 141 | 773 | 139 | 42,48 |
| | | 245054909 | 1240 | 91 | 793 | 93 | 36,05 |
| | | 244084409 | 1317 | 243 | 759 | 243 | 42,37 |
| | | 349064709 | 1343 | 669 | 628 | 672 | 53,24 |
| | | 348094609 | 1496 | 321 | 849 | 305 | 43,25 |
| Série 6 | 10 cuves | 159045910 | 1324 | 611 | 737 | 614 | 44,34 |
| | | 158075710 | 1553 | 179 | 910 | 184 | 41,40 |
| | | 260055210 | 1569 | 308 | 911 | 306 | 41,94 |
| | | 259085210 | 1256 | 658 | 632 | 660 | 49,68 |
| | | 359065910 | 1484 | 225 | 911 | 230 | 38,61 |
| | | 359095810 | 1631 | 771 | 821 | 778 | 49,66 |

## II. 4. 3 - Flexibilité de transport d'une solution

### a. Modèle linéaire pour un problème connexe au CHSP : (*S2*)

Compte tenu des résultats de notre analyse basée sur la flexibilité relative aux fenêtres temporelles, nous avons réfléchi sur la possibilité et l'opportunité d'identifier et d'exploiter d'autres sources de flexibilité du HSP. En dehors des cuves, la seule ressource qui autorise

une souplesse supplémentaire est la ressource de transport. Hors, en termes de déplacement du robot, on autorise déjà les temps d'attente à vide. Il ne reste donc plus que les temps en charge. Mais théoriquement, ceux-ci sont fixés a priori de manière stricte, ce qui interdit les pauses en dehors des temps d'égouttage et de stabilisation, sous peine d'entraîner une perte de qualité. Rappelons que le temps donné pour l'égouttage correspond à un compromis entre une borne minimale permettant de limiter la pollution des bains chimiques suivants et une borne maximale à ne pas dépasser pour éviter une dégradation de la qualité du produit par oxydation à l'air libre. Ce raisonnement nous amène à nous interroger sur les possibilités réelles d'autoriser des stations à l'air libre, par exemple pour allonger le temps d'égouttage, et donc limiter la pollution dans le but de retarder le remplacement des bains. Si cette possibilité existe, cela veut dire qu'on pourrait également envisager des attentes hors égouttage. En conséquence et bien que nous ayons conscience de sortir du cadre strict du HSP (car cela revient à considérer à nouveau un problème connexe de type Robotic Scheduling avec attentes), nous avons étudié l'impact que peut avoir un choix judicieux d'une durée de pause au cours d'une ou de plusieurs opérations de transport.

Dans cette démarche, nous avons modifié notre modèle 2-cyclique de base (S1), en relaxant la contrainte de pauses en charge interdites. Nous avons donc intégré une nouvelle variable $w_i$ correspondant aux temps d'attente en charge du robot, qui s'ajoutent aux temps de transport $d_i$ préalablement définis. Ces temps peuvent être considérés comme des durées d'égouttage supplémentaires ou autres temps au cours du transport d'un porteur. Nous obtenons le nouveau modèle linéaire (*S2*) suivant :

| | | |
|---|---|---|
| $d_i + a_{i+2} - M(1 - y_{i,i+2}) \leq t_{i+2} - t_i - w_i \leq d_i + b_{i+2} + M(1 - y_{i,i+2})$ | $\forall i = 0,...,n-3$ | (II.34) |

| | | |
|---|---|---|
| $d_i + a_{i+2} - My_{i,i+2} \leq t_{i+2} + T - t_i - w_i \leq d_i + b_{i+2} + My_{i,i+2}$ | $\forall i = 0,...,n-3$ | (II.35) |

| | | | |
|---|---|---|---|
| $t_j - t_i \geq d_i + w_i + c_{s_{i+2},s_j} - M(1-y_{i,j})$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (II.36) |

| | | | |
|---|---|---|---|
| $t_i - t_j \geq d_j + w_j + c_{s_{j+2},s_i} - My_{i,j}$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (II.37) |

| | | |
|---|---|---|
| $t_i \geq d_0 + w_0 + c_{s_2,s_i}$ | $\forall i = 2,...,n-1$ | (II.38) |

| | | |
|---|---|---|
| $T - t_i \geq d_i + w_i + c_{s_{i+2},s_0}$ | $\forall i = 2,...,n-1$ | (II.39) |

| | | |
|---|---|---|
| $w_i \geq 0$ | $\forall i = 0,...,n-1$ | (II.40) |

Notre modèle de base (*S1*) correspond alors à un cas particulier du modèle (*S2*) où toutes les variables $w_i$ sont fixées à 0. [El Amraoui et al, 2008][a] , [El Amraoui et al, 2008][b]

**b. Complexité du modèle *(S2)***

La séquence cyclique *S* à déterminer est définie par les dates de début des transports $t_i$, les durées d'attente en charge au dessus de la cuve de chargement $w_i$ ainsi que la durée de cycle *T*. Par conséquent, le nombre total *N'* de variables du modèle (*S2*) n'est autre que le nombre de variables du modèle de base (*S1*) auquel on rajoute le nombre de variables $w_i$.

| | |
|---|---|
| $N' = n + N = n + \left[ n + 1 + \frac{(n-1)(n-2)}{2} \right] = 2n + 1 + \frac{(n-1)(n-2)}{2}$ | (II.41) |

Dans le modèle *(S2)*, le nombre de contraintes est augmenté de (*n-1*) contraintes provenant de l'inégalité (II.40), ce qui fait passer le nombre de contraintes du nouveau modèle à :

| | |
|---|---|
| $Nc' = 2(n-2)(n+2) + (3n-8) + (n-1) = 2(n-2)(n+2) + (4n-9)$ | (II.42) |

Ainsi le nombre total de variables du modèle (*S2*) est $N' = \theta(\frac{n^2}{2})$ et le nombre total de contraintes du modèle linéaire est $Nc' = \theta(2n^2)$. En d'autres termes, la complexité du modèle raffiné (*S2*) reste du même ordre de grandeur que celle du modèle de base (*S1*).

**c. Exemple illustratif**

Soit l'exemple d'une ligne à 3 cuves avec une durée de déplacement du robot en charge (respectivement à vide) entre deux cuves successives de 10 u.t. (respectivement 5 u.t.). Les fenêtres de temps des durées opératoires sont données dans le tableau II.6. Les tableaux II.7 et II.8 donnent les solutions optimales obtenues à partir des modèles (*S1*) et (*S2*).

**Tableau II.6.** Données de l'exemple

| Type de produit | Fenêtre de temps | Cuve i | | | |
|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 |
| Job 1 | $[a_{2i}, b_{2i}]$ | [0 - +∞] | [80 - 96] | [41 - 46] | [61 - 66] |
| Job 2 | $[a_{2i+1}, b_{2i+1}]$ | [0 - +∞] | [76 - 91] | [21 - 26] | [61 - 66] |

**Tableau II.7.** Meilleure solution obtenue par le modèle (*S1*)

| **T** | $t_0$ | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ | $t_7$ |
|---|---|---|---|---|---|---|---|---|
| **308** | 0 | 166 | 90 | 252 | 141 | 283 | 212 | 51 |

**Tableau II.8.** Meilleure solution obtenue par le modèle (*S2*)

| | $t_0$ | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ | $t_7$ |
|---|---|---|---|---|---|---|---|---|
| **T** | 0 | 166 | 90 | 252 | 141 | 15 | 212 | 65 |
| **272** | $w_0$ | $w_1$ | $w_2$ | $w_3$ | $w_4$ | $w_5$ | $w_6$ | $w_7$ |
| | 0 | 0 | 0 | 0 | 0 | 14 | 0 | 0 |

En général, dans un problème de type *CHSP*, la ressource de transport n'est pas autorisée à s'immobiliser en charge durant le processus de traitement en dehors des instants d'égouttage et de stabilisation. Dans ce cas, le temps de cycle correspondant à l'exemple illustratif présenté ci-dessus est de 308 u.t. (figure II.19). Ce même temps de cycle peut être réduit à 272 u.t. dans le cas où on tolère que la ressource de transport s'arrête en charge pendant 14 u.t au-dessus de la cuve 2 ou 3 (figure II.20).

D'après les résultats obtenus pour cet exemple, nous constatons qu'avec le modèle (*S2*) nous réduisons le temps de cycle de *11,7%.* De plus, le nombre de séquences cycliques faisables augmente. Ceci est logique puisque le modèle (*S2*) relaxe une des contraintes de (*S1*). Le domaine solution correspondant à (*S2*) inclut donc celui correspondant à (*S1*) pour lequel tous les $w_i$ prennent la valeur 0.

D'après notre exemple, la solution optimale obtenue par le modèle (*S2*) est la séquence cyclique $\langle 0, 5, 2, 7, 4, 1, 6, 3 \rangle$. Cette séquence ne fait pas partie de l'ensemble des solutions faisables du modèle (*S1*). En effet, sans pause autorisée, cette séquence induirait un dépassement de la borne maximale de la durée de trempe du second type de produit dans la cuve 3. Toutefois, la solution optimale du modèle de base (*S1*) ($\langle 0, 7, 2, 4, 1, 6, 3, 5 \rangle$) fait partie de l'ensemble des solutions du modèle (*S2*).

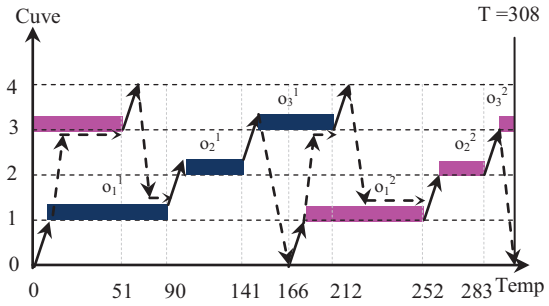



**Figure II.19.** Diagramme de Gantt de la meilleure solution obtenue par le modèle (*S1*)

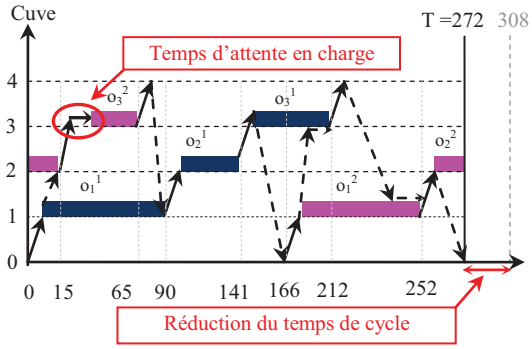



**Figure II.20.** Diagramme de Gantt de la meilleure solution obtenue par le modèle (*S2*)

En conclusion, le modèle (*S2*) ne semble certes pas correspondre dans un premier temps à un CHSP au sens strict de sa définition, mais plutôt à un problème connexe de type *Robotic Scheduling*. En effet, le fait d'autoriser des temps d'attentes en charge peut affecter la qualité des pièces produites, notamment par apparition d'effets d'oxydation des produits restés trop longtemps hors des bains chimiques [Subaï et al, 2003], [Subaï et al, 2005]. Ainsi, dans le raisonnement que nous avons tenu précédemment, le choix ou la modification d'une valeur particulière des durées d'égouttage et de stabilisation n'est pas sans conséquence. La maîtrise de ce type de choix revient en fait aux chimistes et non aux planificateurs. Ceux-ci peuvent néanmoins travailler de concert sur ce type de ligne et trouver un compromis ou de nouvelles valeurs permettant de maximiser la productivité tout en garantissant la qualité requise.

### d. Résultats

Pour évaluer l'impact des temps d'attente ajoutés au modèle (*S1*), nous considérons les instances du tableau II.9. Les données de ces dernières sont générées de façon aléatoire par une loi de distribution uniforme. Les fenêtres de temps, les durées de déplacement en charge et à vide du robot ainsi que les résultats des simulations sont reportés à l'annexe 9.

**Tableau II.9.** Résultats des simulations

| | | Résultats (*S2*) | | | | Résultats (*S1*) | | | Réductions | |
|---|---|---|---|---|---|---|---|---|---|---|
| | n° | Nœud' | T' | ∑$w_i$ | CPU' | Nœud | T | CPU | $R_2$(%) | $R_3$(%) |
| 5 cuves | 1 | 157 | 375 | 55 | 7 | 5 | 485 | 4 | 14,67 | 22,68 |
| | 2 | 219 | 568 | 30 | 5 | 21 | 573 | 4 | 05,28 | 00,87 |
| | 3 | 76 | 320 | 5 | 9 | 17 | 340 | 5 | 01,56 | 05,88 |
| 6 cuves | 4 | 588 | 380 | 14 | 6 | 245 | 405 | 3 | 03,68 | 06,17 |
| | 5 | 339 | 465 | 42 | 4 | 68 | 650 | 3 | 09,03 | 28,46 |
| | 6 | 934 | 449 | 7 | 4 | 81 | 454 | 3 | 01,56 | 01,10 |
| 7 cuves | 7 | 669 | 634 | 39 | 73 | 66 | 814 | 70 | 06,15 | 22,11 |
| | 8 | 2222 | 552 | 35 | 143 | 41 | 583 | 80 | 06,34 | 05,32 |
| | 9 | 1657 | 655 | 35 | 135 | 242 | 688 | 110 | 05,34 | 04,80 |
| 8 cuves | 10 | 4525 | 683 | 107 | 17 | 29 | 892 | 11 | 15,66 | 23,43 |
| | 11 | 4835 | 655 | 40 | 15 | 561 | 735 | 11 | 06,10 | 10,89 |
| | 12 | 7794 | 903 | 43 | 21 | 67 | 928 | 10 | 04,76 | 02,69 |
| 9 cuves | 13 | 10174 | 620 | 5 | 234 | 2421 | 705 | 134 | 00,81 | 12,06 |
| | 14 | 22734 | 710 | 20 | 403 | 1027 | 770 | 170 | 02,82 | 07,79 |
| | 15 | 17914 | 777 | 35 | 366 | 994 | 782 | 89 | 04,50 | 00,64 |
| 10 cuves | 16 | 65621 | 845 | 15 | 123 | 1728 | 870 | 12 | 01,77 | 02,87 |
| | 17 | 18218 | 765 | 15 | 45 | 555 | 775 | 11 | 01,96 | 01,29 |
| | 18 | 58738 | 895 | 80 | 124 | 1915 | 910 | 13 | 08,94 | 01,65 |

Les résultats obtenus avec le modèle (S2) (resp. (S1)) sont fournis dans le tableau II.9 en termes de :

- nombre de nœuds : Nœud' (resp. Nœud), i.e. nombre de disjonctions réalisées ;
- période de l'ordonnancement 2-cyclique hétérogène : T' (resp. T) ;
- somme des durées d'attente en charge : ∑$w_i$ (seulement pour (*S2*)) ;
- durée de simulation en secondes CPU' (resp. CPU) ;
- deux rapports de réductions en pourcentage $R_2$(%) et $R_3$(%) : $R_2$(%) est le rapport de la somme des durées des temps d'attente en charge sur la durée totale du cycle T' (équation II.43). $R_3$ est l'écart relatif entre T et T' (équation II.44).

$$R_2(\%) = \frac{\sum w_i}{T'} \times 100 \qquad \text{(II.43)}$$

$$R_3(\%) = \frac{T - T'}{T} \times 100 \qquad \text{(II.44)}$$

Les temps de simulation avec le modèle (S2) varient entre 5 secondes et une dizaine de minutes (contre 3 minutes pour (S1)). Jusqu'à 7 cuves, voire 8, le temps de résolution est quasi identique qu'avec le modèle (S1). Entre 8 et 10 cuves, ce temps peut passer du simple au sextuple suivant les instances. Cela s'explique par l'augmentation du nombre de variables du modèle et du nombre de solutions faisables.

D'après les résultats obtenus, les deux rapports de réductions $R_2$ et $R_3$, n'évoluent pas proportionnellement. Dans l'exemple n°13, en rajoutant un temps d'attente du robot en charge de 5u.t. (soit de 0.81% de la durée totale du cycle), nous sommes parvenus à réduire de 12% la durée de cycle. Ces résultats montrent l'intérêt possible de cette modification du problème, dans une certaine limite cependant, et sous réserve que les temps ajoutés ne nuisent pas à la qualité des produits. D'ailleurs, par rapport aux résultats que nous présentons, deux ordonnancements équivalents en termes de temps de cycle et de somme des temps d'attente peuvent correspondre respectivement à une solution faisable et une autre non faisable. En effet, il faut encore analyser la répartition du temps total d'attente entre les différents mouvements en charge du robot. Ainsi, un ajout de 10 secondes d'égouttage au-dessus d'une cuve peut ne pas être acceptable en termes de qualité, alors que 5 secondes au-dessus de deux cuves peuvent être tolérées.

## II. 5 - Extensions du modèle 2-cyclique hétérogène aux lignes complexes

### II. 5. 1 - Extension 1 : Ligne à cuves multi-bacs (*S3*)

Dans certaines gammes de production, certaines durées opératoires sont relativement supérieures (deux fois ou plus) à la moyenne des durées opératoires, ce qui provoque un goulot d'étranglement au niveau de la cuve de traitement correspondante. Pour remédier à ce problème et maximiser le rendement de la ligne, une solution consiste à dupliquer les cuves critiques. On parle alors de cuves multi-bacs : cuves pouvant accueillir plus d'un porteur à la fois (pas nécessairement le même type de produit). Chaque emplacement, appelé bac, d'une cuve multi-bacs, doit respecter la contrainte (C2) dans son expression générale, soit :

(C2) une cuve ne peut traiter simultanément plus de porteurs qu'elle ne contient de bacs, i.e. chaque bac ne peut recevoir qu'un porteur à la fois. [El Amraoui et al, 2009][a]

Plusieurs travaux de recherche se sont intéressés à la modélisation d'une ligne de traitement avec des cuves multi-bacs [Shapiro et Nuttle, 1988], [Manier-Lacoste, 1994], [Liu et al, 2002], [Zhou et Ling, 2003]. Toutefois, nous n'avons pas connaissance de travaux abordant la modélisation multi-produits cyclique de tels systèmes. Pour traiter ce cas, nous reprenons certaines des hypothèses logiques émises dans le cadre de fonctionnement 1-cyclique mono-produit par [Shapiro et Nuttle, 1988], puis par [Manier-Lacoste, 1994], à savoir :

- ✓ chaque porteur n'est traité que dans un seul des bacs d'une cuve donnée ;
- ✓ les porteurs sont retirés des bacs selon la loi '*premier arrivé, premier sorti*' (dans notre cas pour chaque type de produit).

Dans le cas d'une ligne à cuve multi-bacs, le temps de déplacement entre deux bacs (plus précisément pour les bacs extrêmes) d'une même cuve n'est pas négligeable. Pour modéliser le problème 2-cyclique hétérogène à cuves multi-bacs, nous proposons les notations suivantes :

| | |
|---|---|
| $q_j$ | capacité (nombre de bacs) de la cuve utilisée aux stages 2j et 2j+1, j=0, .., n-1 |
| $m_i$ | nombre de bacs alloués au stage i |
| $x_{i,k}$ | variable binaire, $x_{i,k}$ =1 si k bacs sont utilisés au stage i, i=0, .., n-1, k=1, .., $m_i$ |
| $d^k_{si}$ | durée du déplacement en charge depuis la cuve associée au stage i (par rapport au cas mono-bac, une durée moyenne constante est considérée entre deux bacs quelconques de deux cuves successives) |
| $c^k_{si,sj}$ | durée du déplacement à vide de la cuve du stage i à la cuve du stage j (là aussi, une durée moyenne constante est considérée).<br>De plus, la variable $c_{si,si}$ n'est plus nulle et elle est remplacée dans le modèle de base par la variable : $\sum_{k=1}^{m_i} c^k_{si,si} x_{i,k}$ |

Nous rappelons que notre problème consiste à produire deux différents types de produits en quantités égales. Ainsi, pendant un cycle, chaque opération de transfert est réalisée exactement deux fois ; donc deux différents types de produits entrent sur la ligne et deux en sortent. Une cuve ayant $q_j$ bacs peut recevoir au plus $q_j$ porteurs à la fois. Si les $q_{2j}$ bacs sont occupés par un produit de type 1 (au stage *2j*) et les $q_{2j+1}$ bacs sont occupés par le second type

de produit (au stage *2j+1*) (avec $q_{2j} + q_{2j+1} \leq q_j$), dans ce cas il faudra $q_{2j}$ cycles (resp. $q_{2j+1}$) à un produit de type 1 (resp. type 2) pour subir le traitement associé à cette cuve.

Ainsi les durées opératoires effectives pour un produit de type 1 et de type 2, sont données, respectivement, par les équations suivantes :

| | | | |
|---|---|---|---|
| $o_i = (t_i - t_{i-2} - d_{s_{i-2}}) + (m_{k1} - 1)T$ | $i \equiv 0$ [2] | $\forall i = 2,...,n-3$ | (II.45) |

| | | | |
|---|---|---|---|
| $o_i = (t_i - t_{i-2} - d_{s_{i-2}}) + (m_{k2} - 1)T$ | $i \equiv 1$ [2] | $\forall i = 2,...,n-3$ | (II.46) |

La figure II.21 illustre le cas d'une cuve contenant 3 bacs, traitant simultanément 3 porteurs dont l'entrée et la sortie suivent la loi FIFO (First In First Out). Le pilotage de la ligne est ici supposé mono-produit 1 cyclique. Sur cette figure, $o_i$ correspond à la durée opératoire effective et $p_i$ correspond à la durée opératoire apparente (au cours d'un cycle). En effet, sur cet exemple le temps séparant le dépôt d'un porteur dans la cuve multi-bacs et la récupération d'un autre dans un autre bac de cette cuve est de $p_i$ alors que le traitement complet du porteur 1 nécessite exactement $o_i$ unités de temps. Ainsi pour des raisons de cohérence de notations la durée opératoire apparente doit être comprise entre $a_i$-$(q_i$-$1)T$ et $b_i$-$(q_i$-$1)T$.

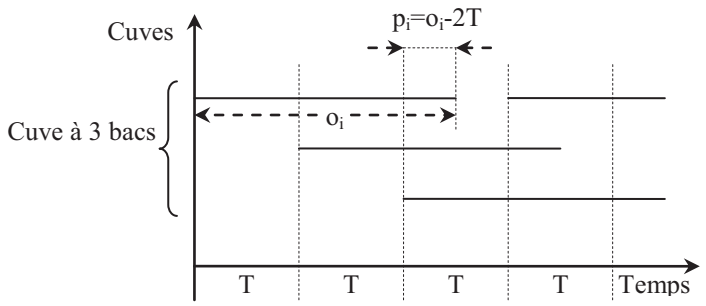



**Figure II.21.** Exemple d'une cuve multi-bacs

Dans le modèle linéaire (*S3*) que nous proposons, nous devrions théoriquement considérer un sous-problème d'affectation en plus du problème de séquencement traité dans (*S1*). En réalité, dans le but de minimiser la durée du cycle (*T*), nous cherchons à déterminer à la fois la meilleure séquence cyclique faisable et le nombre de bacs à utiliser pour chaque type de produits par cuve (défini par la matrice des $(x_{i,k})_{i=0,..,n-1,\ k=1,..,mi}$). Nous ramenons donc notre problème d'affectation à un problème de dimensionnement de la capacité de certaines cuves.

Pour aboutir au modèle (*S3*) il faut adopter le nouveau paramétrage proposé pour les durées de déplacement, remplacer les équations (II.14) et (II.15) par les équations suivantes (II.47, II.48, II.49, II.50, II.51 et II.52), et ajouter la contrainte (II.51):

| | | | |
|---|---|---|---|
| $d^{k}_{s_{i-2}} + a_i - M(1-y_{i-2,i}) - M(1-x_{i,k}) \le t_i - t_{i-2} + (k-1)T$ | $\forall i=2,...,n-1$ | $\forall k=1,...,m_i$ | (II.47) |
| $t_i - t_{i-2} + (k-1)T \le d^{k}_{s_i} + b_i + M(1-y_{i-2,i}) + M(1-x_{i,k})$ | $\forall i = 2,...,n-1$ | $\forall k = 1,...,m_i$ | (II.48) |
| $d^{k}_{s_{i-2}} + a_i - My_{i-2,i} - Mx_{i,k} \le t_i - t_{i-2} + kT$ | $\forall i = 2,...,n-1$ | $\forall k = 1,...,m_i$ | (II.49) |
| $t_i - t_{i-2} + kT \le d^{k}_{s_{i-2}} + b_i + My_{i-2,i} + Mx_{i,k}$ | $\forall i = 2,...,n-1$ | $\forall k = 1,...,m_i$ | (II.50) |
| $\sum_{k=1}^{m_i} x_{i,k} = 1$ | $\forall i = 2,...,n-1$ | $\forall k = 1,...,m_i$ | (II.51) |
| $m_{2j} + m_{2j+1} \le q_j$ | | $\forall j = 0,...,nc+1$ | (II.52) |

Les équations (II.47), (II.48), (II.49) et II.50) modélisent les contraintes de fenêtres de temps et les contraintes (II.51) et (II.52) définissent le nombre de bacs utilisés par *stage*.

L'ajout de nouvelles variables de décision $x_{i,k}$, augmente le nombre de variables du modèle (*S1*) qui devient *N″,* donné par :

$$N'' = n \cdot m_i + N \le n \cdot n + \left[ n + 1 + \frac{(n-1)(n-2)}{2} \right] = n^2 + n + 1 + \frac{(n-1)(n-2)}{2} \quad \text{(II.53)}$$

L'ajout de *n* contraintes supplémentaires (II.51) au modèle de base, modifie aussi le nombre total de contraintes qui devient *Nc″*, avec :

$$Nc'' = Nc + n \quad \text{(II.54)}$$

Ainsi le nombre total de variables du modèle (*S3*) est $N'' = \theta(\frac{3}{2}n^2)$ et le nombre total de contraintes est $Nc'' = \theta(2n^2)$. En conséquence, notre système (S3) garde le même ordre de complexité que celle du modèle de base (*S1*).

**- Exemple illustratif**

Considérons une ligne de traitement à 5 cuves, non inclus les postes de chargement et de déchargement (considérés toujours dissociés) (voir données au tableau II.10). La durée de déplacement du robot en charge (respectivement à vide) entre deux cuves successives est de 10 u.t. (respectivement 9 u.t.). Cet exemple est inspiré de la ligne Zinc [Shapiro, 1985] traitée par Manier dans [Manier-Lacoste, 1994]. La cuve 4 est une cuve multi-bacs (contenant 6 bacs).

Les résultats de simulation de cet exemple par le modèle (*S3*) sont reportés au tableau II.11 et la meilleure solution trouvée est représentée sur la figure II.22. Le temps de cycle de la meilleure solution obtenue est de 352 u.t. Pour ce problème, nous avons au préalable fixé à 3 le nombre de bacs de la cuve 4 affectés à chaque type de produits ($m_8$=$m_9$=3). Pour la solution optimale, seuls deux bacs de la cuve 4 sont utilisés pour chaque type de produit ($m_{k1}$= $m_8$=2 et $m_{k2}$= $m_9$=2), soit quatre bacs en tout sur les six disponibles. La cuve 4 n'est donc pas utilisée au maximum de sa capacité. En observant ce qui se passe au niveau de la cuve 4 (figure II.22), on constate que, de plus, seuls 3 bacs sont utilisés simultanément. Il serait donc intéressant d'envisager d'autoriser dans notre modèle l'utilisation partagée d'un même bac par des produits de types différents. Cela permettrait une économie supplémentaire sur les ressources de traitement. Notons enfin que la durée de trempe effective, dans cette cuve, pour les deux types de produit est respectivement de 400 u.t. et 463 u.t, durées déduites à partir des équations (II.45) et (II.46).

**Tableau II.10.** Fenêtres de temps de l'exemple

| Type de produit | Fenêtre de temps | Cuve i (nombre de bacs) | | | | | |
|---|---|---|---|---|---|---|---|
| | | 0 | 1 (1) | 2 (1) | 3 (1) | 4 (6) | 5 (1) |
| Job 1 | $[a_{2i}, b_{2i}]$ | [0 - +∞] | [50 - 114] | [80 - 184] | [60 - 145] | [300 - 400] | [117 - 167] |
| Job 2 | $[a_{2i+1}, b_{2i+1}]$ | [0 - +∞] | [49 - 126] | [71 - 178] | [54 - 152] | [450 - 500] | [56 - 166] |

**Tableau II.11.** Meilleure solution trouvée par le modèle (S3)

| | $t_0$ | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ |
|---|---|---|---|---|---|---|---|
| **T** | 0 | 19 | 60 | 83 | 102 | 166 | 185 |
| **352** | $t_7$ | $t_8$ | $t_9$ | $t_{10}$ | $t_{11}$ | | |
| | 204 | 250 | 269 | 288 | 327 | | |

Pour cet exemple, si on considère la cuve 4 à capacité unitaire, la meilleure solution obtenue (avec le modèle (*S1*) dans le cas de cuves toutes mono-bacs) correspond à un temps de cycle de 826 u.t. (voir tableau II.12). Ainsi, l'augmentation de la capacité de la cuve goulot a permis de réduire la durée du cycle de 43%, ce qui correspond à une amélioration notable.

Cet exemple illustre l'intérêt du modèle linéaire (*S3*) qui recherche la meilleure solution minimisant la période du cycle, tout en affectant le nombre de bacs à utiliser par cuve et par produit. Ce modèle pourrait donc être aussi utilisé dans une phase de conception du système physique, et notamment à des fins de dimensionnement de la capacité des cuves.

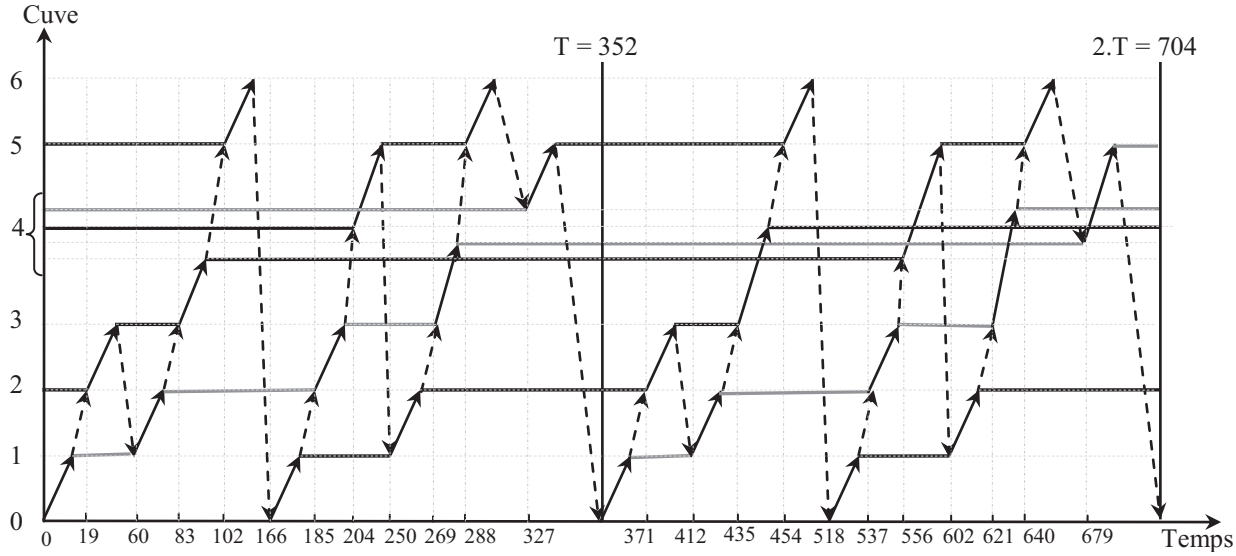



**Figure II.22.** Diagramme de Gantt de la meilleure solution trouvée

**Tableau II.12.** Meilleure solution trouvée par le modèle (S1) pour l'exemple illustratif

| | $t_0$ | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ |
|---|---|---|---|---|---|---|---|
| **T** | 0 | 336 | 124 | 472 | 280 | 611 | 435 |
| **826** | $t_7$ | $t_8$ | $t_9$ | $t_{10}$ | $t_{11}$ | | |
| | 773 | 745 | 407 | 46 | 509 | | |

## II. 5. 2 - Extension 2 : Ligne à cuves multi-fonctions (*S4*)

On parle de ligne à cuves multi-fonctions dans le cas où un produit passe plus d'une seule fois par la même cuve durant son processus de traitement. Ainsi, le nombre de traitements d'un produit n'est plus égal au nombre de cuves de la ligne ; mais il se peut qu'il le dépasse si toutes les ressources sont utilisées.

Par analogie avec les travaux réalisés dans le cas mono-produit [Manier-Lacoste, 1994], [Hanen et Munier, 1993], nous adoptons la restriction réaliste suivante : deux traitements réalisés dans une même cuve ne sont jamais consécutifs dans la séquence opératoire ; ce qui revient à dire qu'un porteur retiré d'une cuve ne peut pas y être de nouveau déposé dès le traitement suivant.

La contrainte (C2) s'applique pour toutes les cuves de la ligne, qu'elles soient mono-fonction ou multi-fonctions. Toutefois, l'expression de cette contrainte dans le modèle de base (*S1*), ne concerne que les conflits entre deux produits (de même type ou non) subissant le même traitement dans la même cuve. En plus de cela, nous devons aussi empêcher deux porteurs portant le même type de produits de se retrouver simultanément dans une cuve multi-fonctions pour y subir deux opérations différentes de la gamme. Ainsi soient i et j, deux traitements (ou *stages*) d'une gamme réalisés dans la même cuve multi-fonctions k ($s_i$=$s_j$=k, où j>i+2). On suppose également que les traitements i et i+2 ne sont pas réalisés dans la même cuve. Afin d'éviter les conflits d'occupation de cette cuve, quatre cas de séquencement sont autorisés pour les traitements i, j, i-2 et j-2 (j>i+2). Ils sont déduits par analogie avec les cas présentés au *corollaire 1* :

*Cas 1* : la cuve k est occupée au début du cycle pour le traitement *i*. Dans ce cas, le seul ordre partiel faisable pour les opérations de transport est : $\langle (i, j-2, j, i-2) \rangle$

*Cas 2* : la cuve k est occupée au début du cycle pour le traitement *j.* Dans ce cas, le seul ordre partiel faisable est : $\langle (j, i-2, i, j-2) \rangle$

*Cas 3* : la cuve k est vide au début du cycle et le traitement (*i-2)* doit être effectué en premier. Par conséquent, le seul ordre partiel faisable est : $\langle (i-2, i, j-2, j) \rangle$

*Case 4:* la cuve k est vide au début du cycle et le traitement (*j-2)* doit être effectué en premier. Par conséquent, le seul ordre partiel faisable est : $\langle (j-2, j, i-2, i) \rangle$

Nous obtenons finalement le système d'équations (II.55) et (II.56) qui, ajouté au modèle (*S1*), forme le nouveau modèle 2-cyclique hétérogène (*S4*) adapté au cas des lignes multi-fonctions.

| | | | | |
|---|---|---|---|---|
| $y_{i-2,i} + y_{j-2,j} \geq 1$ | Si ($s_i = s_j$) | $i+2 \prec j$ | $\forall i = 2,...,n-4$ | (II.55) |

| | | | | |
|---|---|---|---|---|
| $y_{i,j-2} + y_{j,i-2} \geq 3 - y_{i-2,i} - y_{j-2,j}$ | Si ($s_i = s_j$) | $i+2 \prec j$ | $\forall i = 2,...,n-4$ | (II.56) |

En passant du modèle linéaire de base (*S1*) au modèle de lignes à cuves multi-fonctions (*S4*), nous avons gardé le même nombre de variables. Toutefois, nous rappelons que le nombre total des opérations de transport est fonction du nombre de produits et de traitements. Par contre, l'ajout des contraintes (II.55) et (II.56) augmente de *2.(n-5)* le nombre de contraintes du modèle de base, faisant ainsi passer leur nombre total à *Nc'''*, avec :

| | |
|---|---|
| $Nc''' = Nc + 2(n-5)$ | (II.57) |

Ainsi le nombre total de variables du modèle (*S4*) reste $\theta(\frac{3}{2}n^2)$ et le nombre total de contraintes est $Nc''' = \theta(2n^2)$ [El Amraoui et al, 2011][b].

**Exemple illustratif**

Soit une ligne de traitement à 4 cuves, non inclus les postes de chargement (cuve 0) et de déchargement (cuve 5). La gamme de traitement ainsi que les fenêtres de temps pour chaque type de produit sont données au tableau II.13. La durée de déplacement du robot en charge (respectivement à vide) entre deux cuves successives est de 15 u.t. (respectivement 5 u.t.). Dans cet exemple, la cuve 2 est une cuve multi-fonctions où les deux types de produits passent deux fois durant leurs séquences opératoires respectives (traitements 2 et 5).

**Tableau II.13.** Donnée de l'exemple illustratif

| Traitement | 0 (fictif) | 1 | 2 | 3 | 4 | 5 | 6 (fictif) |
|---|---|---|---|---|---|---|---|
| Cuve | 0 | 1 | 2 | 3 | 4 | 2 | 5 |
| Job 1 | [0 - +∞] | [40 - 100] | [20 - 80] | [25 - 75] | [25 - 110] | [30 - 105] | [0 - +∞] |
| Job 2 | [0 - +∞] | [35 - 105] | [25 - 80] | [35 - 100] | [25 - 100] | [45 - 100] | [0 - +∞] |

Les résultats de simulation relatifs à cet exemple par le modèle (*S4*) sont reportés au tableau II.14 et la meilleure séquence 2-cyclique trouvée est représentée sur la figure II.23. D'une part, on remarque qu'au cours d'une séquence, la capacité unitaire de la cuve multi-fonctions est bien respectée. D'autre part, le robot assure sans interruption le transport des produits d'une cuve à une autre. Un seul temps d'attente à vide de 5 u.t. au niveau du premier traitement du second produit à été détecté.

**Tableau II.14.** Meilleure solution trouvée par le modèle (S1) pour l'exemple illustratif

| T | $t_0$ | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ | $t_7$ | $t_8$ | $t_9$ | $t_{10}$ | $t_{11}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **308** | 285 | 285 | 285 | 285 | 285 | 285 | 285 | 285 | 285 | 285 | 285 | 285 |

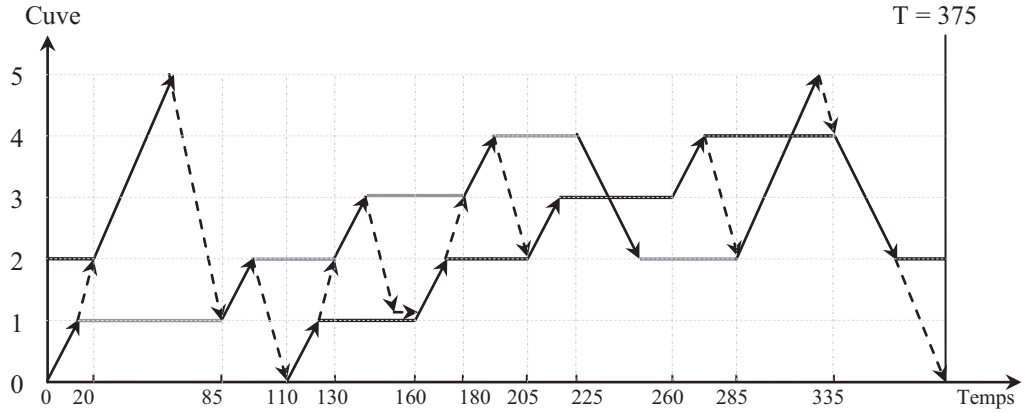



**Figure II.23.** Diagramme de Gantt de la meilleure solution trouvée

## II. 5. 3 - Extension 3 : Ligne à postes de chargement et déchargement associés (*S5*)

Dans le modèle de base (*S1*), nous avons considéré le cas le plus simple pour la configuration des postes de chargement et de déchargement afin de simplifier le modèle. On se propose, dans ce paragraphe, de modéliser une configuration associée de ces postes.

Dans une telle configuration, un seul opérateur doit réaliser les tâches de chargement et de déchargement, qui sont au nombre de quatre, à chaque cycle. En se référant aux travaux traitant le cas mono-produit [Manier-Lacoste, 1994], [Hanen et Munier, 1993], on peut considérer que l'opérateur dispose d'un seul emplacement pour réaliser les deux opérations ; celles-ci ne peuvent donc être réalisées que consécutivement, puisque l'opérateur ne peut réaliser qu'une seule des deux tâches à la fois.

Les contraintes supplémentaires (équations (II.58) à (II.61)), induites par cette configuration, garantissent que l'opérateur dispose d'un temps suffisant entre chaque opération de déchargement et de chargement. Cette durée minimale correspond à la durée $a_0$ (ou $a_1$).

$$T + t_0 - t_{n-2} \geq d_{s_{n-2}} + a_0 \qquad \text{(II.58)}$$

$$t_1 - t_{n-1} \geq d_{s_{n-1}} + a_1 - My_{1,n-1} \qquad \text{(II.59)}$$

| | |
|---|---|
| $t_1 - t_{n-2} \geq d_{s_{n-2}} + a_1 - My_{1,n-2}$ | (II.60) |

| | |
|---|---|
| $T + t_0 - t_{n-1} \geq d_{s_{n-1}} + a_0$ | (II.61) |

L'ajout de ces équations au modèle de base augmente uniquement le nombre de contraintes, qui reste, toutefois, du même ordre ($\theta(2n^2)$).

**Exemple illustratif**

Soit une ligne à trois cuves de traitement, non inclus le poste de chargement et déchargement (cuve 0). Le temps de déplacement en charge du robot entre deux cuves successives est de 15 u.t., celui à vide est de 5 u.t.. Le reste des données de l'exemple figure au tableau II.15.

**Tableau II.15.** Données de l'exemple illustratif

| Traitement | 0 (fictif) | 1 | 2 | 3 |
|---|---|---|---|---|
| Cuve | 0 | 1 | 2 | 3 |
| Job 1 | [45 - 100] | [40 - 100] | [20 - 80] | [25 - 75] |
| Job 2 | [45 - 100] | [35 - 105] | [25 - 80] | [35 - 100] |

La figure II.24 représente la meilleure solution obtenue par simulation du modèle (*S5*). Elle illustre le fait que le porteur du produit de type 1 est déchargé puis chargé à nouveau à la cuve 0 pendant 45 u.t. en tout (entre 275 u.t. et 320 u.t.). Quant au produit de type 2, son porteur va passer 95 u.t. dans la cuve 0 avant d'être remis en service (entre 100 et 195 u.t.).

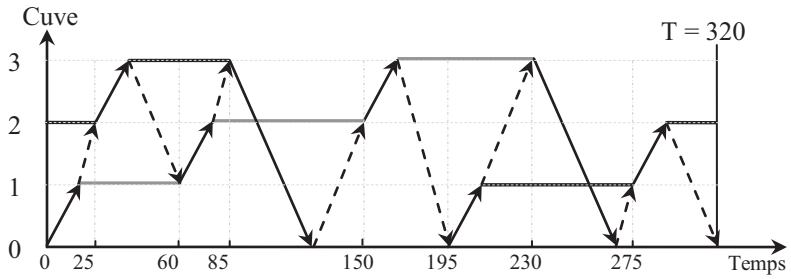



**Figure II.24.** Diagramme de Gant de la meilleure solution obtenue

# II. 6 - Extension du modèle 2-périodique aux ordonnancements r-cycliques hétérogènes

## II. 6. 1 - Définition d'un r-cycle

En général, dans le cas mono-produit, un cycle 1-périodique constitue une solution particulière du problème général puisque les produits (du même type) sont contraints à entrer sur la ligne à des intervalles de temps réguliers (égaux). Une généralisation du modèle de base 1-périodique (voir annexe 6) au modèle r-périodique ($r \geq 1$) mono-produit, a été proposée par Shapiro [Shapiro, 1985], [Shapiro et Nuttle, 1988], Lei et Wang [Lei et Wang, 1989][a] et Manier [Manier-Lacoste, 1994]. Dans ce cas, la solution recherchée correspond à un cycle **r-**périodique ($r \geq 1$) encore appelé r-cycle par Shapiro [Shapiro, 1985], ou cycle de degré r par Lei et Wang [Lei et Wang, 1989][a]. Au cours d'un tel cycle, exactement ***r*** porteurs entrent et ***r*** porteurs sortent du système, transportant tous le même type de produit. Le temps moyen séparant l'entrée de deux produits successifs sur la ligne est obtenu en divisant le temps de cycle ***T*** par le degré ***r*** du cycle. Cette durée, définie par Manier dans [Manier-Lacoste, 1994], correspond aussi au temps moyen de sortie d'un porteur.

Dans ce paragraphe, nous nous attachons à la généralisation du modèle de base 2-cyclique hétérogène en un modèle *r-cyclique hétérogène* ($r \geq 2$) (ou *r-cyclique multi-produits*), où durant un cycle, exactement $r$ différents types de produits entrent et $r$ différents types de produits sortent du système.

Dans le problème r-cyclique multi-produits que nous nous proposons de modéliser, les propriétés inhérentes au cas r-cyclique mono-produit ne sont plus conservées. En effet, les séquences opératoires sont toutes identiques, mais pas les fenêtres de temps dans une cuve donnée suivant les produits. Par conséquent, les durées opératoires effectives dans chaque cuve ne sont pas nécessairement identiques pour les différents types de produits présents sur la ligne. Toutefois, l'état de la ligne au début de chaque cycle est inchangé (cuves occupées et cuves libres, temps de trempe écoulés dans les cuves, position du robot). Dans la solution r-cyclique cherchée, les durées de trempe dans chaque cuve sont identiques pour les mêmes types de produits entrés dans le système à des dates régulières avec un pas de temps proportionnel à $T$. Ainsi, les porteurs $p$ et $(p+kr)$ ont les mêmes temps opératoires, k est un entier quelconque.

Les raisonnements que nous avons tenus précédemment pour étendre le modèle 2-cyclique de base, restent valables pour l'extension r-cyclique. Les seules différences qui apparaissent se situent au niveau de la représentation des contraintes, ainsi que sur leur nombre.

## II. 6. 2 - Modèle linéaire du problème r-cyclique multi-produits: (*S6*)

Afin de modéliser les mouvements de transport des r-premiers produits entrés sur la ligne dans un cycle r-périodique hétérogène, nous utilisons les mêmes données et variables que pour le problème 2-cyclique, hormis le degré du cycle.

Pour rappel, dans un but de simplification du modèle, nous faisons toujours appel à la nouvelle indexation ou *stage* prédéfinie précédemment, avec ici *r* stages réalisés à chaque cuve (contre 2 dans le cas 2-cyclique). Avec cette nouvelle indexation, chaque opération de transport ($m_i$, ●) sera réduite à un seul paramètre *k,* avec *k=ri+(j-1). k* définit à la fois le numéro de l'opération de transport i (*i=0,…,n-1)* et le type de produit j à déplacer (j=*1 …, r*). Si un mouvement *k* est défini par : *k=p* [*r*] (ou encore *k=rn+p*), alors il s'agit du mouvement de transport ($m_n$, $j_{p+1}$), c'est-à-dire le $(n+1)^{ième}$ mouvement de transport du produit de type (*p+1)*.

Nous proposons le modèle linéaire r-cyclique multi-produits suivant :

- la fonction objectif exprime la minimisation de la durée totale du cycle :

| | |
|---|---|
| Minimiser T | (II.62) |

- les contraintes (II.63) et (II.64) assurent qu'une durée de trempe est comprise entre une durée minimale et une durée maximale, telle que :

| | | |
|---|---|---|
| $d_{i-r} + a_i - M(1 - y_{i-r,i}) \leq t_i - t_{i-r} \leq d_{i-r} + b_i + M(1 - y_{i-r,i})$ | $\forall i = r,...,n-1$ | (II.63) |

| | | |
|---|---|---|
| $d_{i-r} + a_i - My_{i-r,i} \leq t_i - t_{i-r} + T \leq d_{i-r} + b_i + My_{i-r,i}$ | $\forall i = r,...,n-1$ | (II.64) |

- les contraintes (II.65) et (II.66) vérifient qu'un robot peut se déplacer à vide entre deux mouvements de transport quelconques :

| | | |
|---|---|---|
| $t_j - t_i \geq d_i + c_{s_{i+r},s_j} - M(1 - y_{i,j})$ | $\forall i = 0,...,n-r-1$ | (II.65) |

| | | |
|---|---|---|
| $t_i - t_j \geq d_j + c_{s_{j+r},s_i} - My_{i,j}$ | $\forall i = 0,...,n-r-1$ | (II.66) |

- les contraintes (II.67) et (II.68) assurent la faisabilité de la séquence cyclique dans le cas où deux traitements i et j quelconques se partagent la même cuve :

| | | | | |
|---|---|---|---|---|
| $y_{i-r,i} + y_{j-r,j} \geq 1$ | Si ($s_i = s_j$) | $j > (i + 2r - 1)$ | $\forall i, j = r, \ldots, n-1$ | (II.67) |

| | | | | |
|---|---|---|---|---|
| $y_{i,j-r} + y_{j,i-r} \geq 3 - y_{i-r,i} - y_{j-r,j}$ | Si ($s_i = s_j$) | $j > (i + 2r - 1)$ | $\forall i, j = r, \ldots, n-1$ | (II.68) |

- les contraintes (II.69) imposent que, pour chaque groupe de *r* traitements se partageant la cuve, au plus une d'entre elles commence dans un cycle et prend fin au cycle suivant. Cette contrainte sera démontrée par la suite :

| | | | |
|---|---|---|---|
| $\sum_{i=k.r}^{k.r+(r-1)} y_{i,i+r} \geq r - 1$ | Si ($s_i = s_{i+1} = \ldots = s_{i+r-1}$) | $\forall i = 0, \ldots, n-r-1$ | (II.69) |

- (II.70) et (II.71) définissent des bornes minimales pour les dates de début des opérations de transport et pour le temps de cycle :

| | | |
|---|---|---|
| $t_i \geq d_{s_0} + c_{s_r, s_i}$ | $\forall i = 1, \ldots, n-1$ | (II.70) |

| | | |
|---|---|---|
| $T - t_i \geq d_{s_i} + c_{s_{i+r}, s_0}$ | $\forall i, j = r, \ldots, n-1$ | (II.71) |

- les contraintes (II.72), (II.73) et (II.74) sont des contraintes d'intégrité :

| | | |
|---|---|---|
| $y_{i,j} \in \{0, 1\}$ | $\forall i, j = 0, \ldots, n-1$ | (II.72) |

| | |
|---|---|
| $T \geq 0$ | (II.73) |

| | | |
|---|---|---|
| $t_i \geq 0$ | $\forall i = 0, \ldots, n-1$ | (II.74) |

- les contraintes (II.75) et (II.76) permettent de mettre $y_{j,i}$ à 1 chaque fois que $y_{i,j}=0$ et inversement.

| | | | |
|---|---|---|---|
| $(1-y_{i,j}) \geq y_{j,i}$ | $i \prec j$ | $\forall i, j = 1, \ldots, n-1$ | (II.75) |

| | | | |
|---|---|---|---|
| $y_{j,i} \geq (1-y_{i,j})$ | $i \prec j$ | $\forall i, j = 1, \ldots, n-1$ | (II.76) |

- les contraintes (II.77) et (II.78) affectent la première opération de transport du robot et initialisent sa date de début à 0.

| | | |
|---|---|---|
| $y_{0,i} = 1$ | $\forall i = 1, \ldots, n-1$ | (II.77) |

| | |
|---|---|
| $t_0 = 0$ | (II.78) |

La contrainte (II.69) est une contrainte stricte du modèle définissant la faisabilité de la séquence cyclique. Cette contrainte est une généralisation de la contrainte (II.20) du modèle de base (*S1*). Elle traduit l'assertion suivante : si r opérations de trempe se partagent la même ressource, alors au plus une seule opération parmi celles-là peut commencer dans un cycle et se terminer dans le suivant. C'est-à-dire qu'un seul chevauchement est permis. En effet, nous autorisons les chevauchements de cycles dans nos modèles. Toutefois, ceci peut occasionner des contraintes d'occupation de cuves. Afin de remédier à ce problème, chaque séquence cyclique *S* donnée doit respecter l'assertion donnée dans le *lemme 3*.

***Lemme 3.***
Un ordonnancement r-cyclique est faisable si et seulement si, la séquence cyclique *S* satisfait la condition suivante : pour chaque r-uplet de mouvements ($m_i$, ●), au maximum un seul mouvement ($m_{i+1}$, ●) pourrait précéder tous les mouvements ($m_i$, ●).

***Preuve :*** Par l'absurde (voir preuve du *lemme 1*)

Traiter simultanément *r* produits distincts entraine des contraintes spécifiques supplémentaires. En effet, chaque séquence cyclique *S* doit tenir compte de l'hétérogénéité de l'ordonnancement. Cette règle se traduit par le *lemme 4*.

***Lemme 4.***
Un ordonnancement cyclique multi-produits (hétérogène) est faisable si et seulement si, la séquence cyclique *S* satisfait la condition suivante : pour chaque r-uplet ($m_i$, ●), s'il existe *un* ($m_{i+1}$, ●) qui précède tous les mouvements ($m_i$, ●) de la séquence S, *alors* ce mouvement a le même type de produit que le dernier mouvement ($m_i$, ●) de ce r-uplet.

***Preuve :*** Par récurrence sur r et en utilisant le *Lemme 2* (cas pour *r=2*).

## II. 6. 3 - Exemple illustratif

Afin d'illustrer le modèle (*S6*), considérons l'exemple à trois produits distincts à produire en proportions égales. Les fenêtres de temps sont données au tableau II.16.

**Tableau II.16.** Données de l'exemple illustratif

| Traitement | 0 (fictif) | 1 | 2 | 3 |
|---|---|---|---|---|
| Cuve | 0 | 1 | 2 | 3 |
| Job 1 | [0 - +∞] | [25 - 30] | [30 - 35] | [40 - 75] |
| Job 2 | [0 - +∞] | [40 - 85] | [80 - 170] | [20 - 45] |
| Job 3 | [0 - +∞] | [50 - 150] | [20 - 65] | [80 - 250] |

La solution obtenue par le modèle linéaire (*S5*), pour cet exemple, est représentée à la figure II.25. Notons qu'afin d'assurer la faisabilité de la séquence cyclique, une fois que le transport du produit 2 de la cuve 2 à la cuve 3 est assuré, le robot rejoint la cuve 1 et reste à vide 5 u.t. au-dessus de cette cuve avant d'assurer le transport du produit 1 de cette dernière à la cuve 2.
La meilleure séquence cyclique de transport en charge S retournée par Cplex au bout de quelque secondes est la suivante :

$$S = \left\langle \begin{matrix} (m_0, j_1), (m_2, j_2), (m_1, j_1), (m_3, j_2), (m_2, j_1), (m_0, j_3), (m_3, j_1), (m_1, j_3), (m_0, j_2), \\ (m_2, j_3), (m_1, j_2), (m_3, j_3) \end{matrix} \right\rangle$$

En utilisant l'indexation des stages (avec ici 3 stages par cuve), cette séquence est équivalente à la séquence suivante : $S = \langle 0, 7, 3, 10, 6, 2, 9, 5, 1, 8, 4, 11 \rangle$. Cette solution illustre le fait que, durant un cycle, deux produits (produit 1 et produit 3) sont tout d'abord chargés, puis traités et enfin déchargés. Le traitement du produit de type 2 chevauche deux cycles successifs.

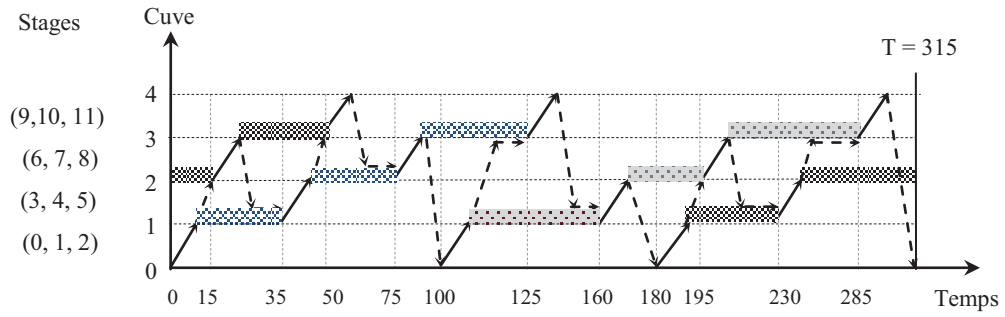



**Figure II.25.** Diagramme de Gantt de la meilleure solution obtenue

Notons, enfin, que dans la solution obtenue, l'ordre d'entrée des produits sur la ligne au cours d'un cycle est (*j1, j3, j2*), et non (*j1, j2, j3*). En effet, une des caractéristiques remarquable de notre modèle r-cyclique est qu'il ne fixe pas a priori la séquence d'entrée sur la ligne. Cela nous permet à la fois de minimiser la durée totale du cycle et de définir le meilleur ordre d'insertion des différents types de produits.

### II. 6. 4 - Résultats

Nous avons simulé notre modèle r-périodique sur les benchmarks de [Mateo, 2001] à 5 cuves (voir données en annexe 5). Le tableau des résultats obtenus est reporté en annexe 10.

Ce tableau montre qu'avec un cycle 4-périodique, nous améliorons le temps de cycle :

- jusqu'à 32% par rapport au cas de 2 productions cycliques mono-produit successives,
- jusqu'à 8,21% par rapport à une solution 2-cyclique hétérogène optimale.

Le temps de résolution varie entre 29 secondes et 24 heures. Dans ce dernier cas, cela correspond à un arrêt forcé de la simulation ; la solution optimale n'est donc pas obligatoirement trouvée, et parfois, les performances de la solution 4-cyclique sont alors moins bonnes que celles de cycles mono-produits successifs (entre 1,31 et 3,32% de dégradation pour 4 instances sur les 30 testées). L'augmentation du temps de résolution s'explique par une augmentation conséquente du nombre de variables et de contraintes compte tenu du degré du cycle.

## II. 7 - Bilan

Nous nous sommes intéressés, dans ce chapitre, à la modélisation et à l'optimisation cyclique multi-produits des lignes de traitement de surface.

Notre approche de résolution est basée sur la programmation linéaire mixte (PLM). Pour modéliser le problème, nous avons élaboré six modèles linéaires. Le premier modèle optimise la durée du cycle pour un ordonnancement 2-cyclique hétérogène. Il a été validé sur plus de 180 exemples tirés de la littérature [Mateo, 2001]. Il a été utilisé/modifié, par la suite, pour étudier la robustesse des solutions cycliques et la flexibilité accordée par les ressources de traitement et de transport. Pour les cuves, il s'agit des durées opératoires bornées. Pour le robot, nous sommes sortis du cadre strict de l'étude d'un HSP en relaxant la contrainte de non attente en charge, ce qui revient à considérer un Cyclic Robotic Scheduling Problem. Toutefois, nous justifions cet écart par l'obtention d'une nette amélioration du temps de cycle, acceptable sous réserve d'une non-dégradation de la qualité des produits.

Les troisième, quatrième et cinquième Programmes Linéaires proposés, modélisent des cas de lignes de traitement de surface complexes, pour un fonctionnement 2-cyclique multi-produits. Ces extensions intègrent respectivement des cuves de capacité non unitaire (multi-bacs), des cuves multi-fonctions, ou des postes de chargement et déchargement associés.

Le sixième Programme Linéaire généralise notre modèle de base à la recherche d'ordonnancements r-cycliques hétérogènes. En nous basant sur les résultats des simulations réalisées, nous avons notamment résolu des problèmes 4-cycliques hétérogènes. Toutefois, l'augmentation conséquente du temps de résolution (parfois plus de 24 heures sans preuve d'optimalité) montre les limites de l'utilisation de cette méthode exacte, en tout cas pour des instances de taille importante qui correspondent néanmoins plus à la réalité industrielle que les divers cas d'école traités par de nombreux chercheurs. Aussi, pour répondre au problème posé, nous avons pensé que la mise en œuvre d'une méthode approchée pouvait être pertinente pour ce type d'instances. Dans le chapitre suivant, nous présentons donc un algorithme de liste et un algorithme génétique dédié au CHSP multi-produits.

# Chapitre III

# Optimisation, modélisation et résolution du problème CHSP multi-produits par des méthodes approchées : Heuristiques et Algorithmes Génétiques

## III. 1 - Introduction

Les problèmes d'ordonnancement d'atelier sont, dans le cas général, des problèmes de type NP-difficiles ou encore de complexité ouverte. Il en va ainsi notamment pour le cas du Cyclic Hoist Scheduling Problem dès les cas les plus simples [Lei et Wang, 1989][b]. De nombreux ouvrages, à l'exemple de [Carlier, 1988] ou [Brucker et al, 2002], ont abordé l'étude de la complexité et plus particulièrement celle des problèmes d'ordonnancement. D'un point de vue opérationnel, cette complexité justifie l'utilisation de méthodes de résolution approchées pour traiter des instances de tailles importantes. Dans cette optique, nous nous proposons dans ce chapitre de développer des méthodes approchées pour la résolution du CHSP multi-produits.

Certes, les méthodes approchées ont été relativement peu développées pour la résolution de problèmes cycliques. Néanmoins, on trouve par exemple les travaux de Chrétienne (1996; 1997) et Song (1993) sur la redéfinition et l'exploitation des algorithmes de liste dans le cas des problèmes à machines identiques. Dans une démarche similaire, notre objectif est ici d'exploiter le potentiel des approches dites heuristiques et méta-heuristiques pour résoudre le CHSP multi-produits.

La première partie de ce chapitre présente un algorithme de liste que nous avons élaboré en nous basant sur les travaux de recherches de Song (1993). La seconde partie de ce chapitre est dédiée à une classe particulière de méthodes approchées de type méta-heuristique : les algorithmes génétiques, que nous utilisons pour résoudre le problème CHSP multi-produits.

# Partie 1 : Approche heuristique

## III. 2 - Définitions

Une approche heuristique est définie, dans la littérature, comme suit :

---

**Définition III.1** [Chrétien-Goni, 2002]

---

Une *heuristique* est une méthode, conçue pour un problème d'optimisation donné, qui produit une solution non nécessairement optimale lorsqu'on lui fournit une instance de problème.
*Ou* comme étant une technique de résolution incertaine dont l'usage n'est justifié que par l'efficacité antérieure dans des problèmes analogues.
*Ou encore* comme un algorithme qui fournit rapidement (en un temps polynomial) une solution réalisable, pas nécessairement optimale, pour un problème d'optimisation NP-difficile.

---

La qualité d'une heuristique s'évalue selon deux critères scientifiques. Le premier est un critère pratique ou empirique, où on implémente l'algorithme approximatif et on évalue la qualité de ses solutions par rapport aux solutions optimales. Le second est un critère mathématique où à travers une démonstration mathématique, on prouve que l'heuristique garantit des performances.
Une heuristique est dite :

- optimiste : dans le cas où la valeur qu'on cherche à maximiser (resp. minimiser) est surestimée (resp. sous-estimée) ;
- pessimiste : l'inverse de ci-dessus ;
- informative : lorsqu'une heuristique $H_1$ est plus informative (solution très proche de la solution optimale) qu'une autre $H_2$ alors dans ce cas une condition nécessaire et suffisante doit être satisfaite :

$$\left|H_1(s) - f^*(s)\right| \prec \left|H_2(s) - f^*(s)\right|$$

- dépendante (resp. indépendante) du domaine : quand elle est non applicable (resp. applicable) à n'importe quel problème d'optimisation. Dans le cas où cette heuristique est indépendante du domaine, elle sera qualifiée de métaheuristique.

## III. 3 - Heuristiques dans le domaine du CHSP

Dans la littérature des problèmes d'ordonnancement dynamiques des lignes de traitement de surface, des solutions efficaces ont développées, basées sur des méthodes heuristiques ([Yih, 1994], [Hindi et Fleszar, 2004], [Henrik et al, 2005], etc). Toutefois, peu d'heuristiques ont été mises en œuvre pour le cas cyclique mono-produit et pas du tout dans le cas cyclique multi-produits. A notre connaissance, à part quelques algorithmes de type méta-heuristiques, qui seront évoqués plus en détails dans la seconde partie du chapitre, un seul article dans la littérature est consacré au développement d'une méthode heuristique pour la résolution du CHSP. Il fut présenté par Song et al en 1993 [Song et al, 1993]. Ceux-ci proposent un modèle analytique mono-produit basé sur la programmation entière mixte qui ne fait pas l'hypothèse que l'ordonnancement recherché est 1-cyclique. Ils utilisent une heuristique nommée *Earliest Start Time Method* (ou *EST*). Cette dernière consiste à introduire les porteurs un par un sur la ligne, sans modifier l'ordonnancement relatif aux porteurs déjà considérés, mais en retardant l'entrée du nouveau produit en cas de conflit entre porteurs pour l'utilisation de l'une des ressources. L'ordonnancement calculé permet de connaitre les bornes minimale ($x_{min}$) et maximale ($x_{max}$) du temps séparant l'entrée des deux porteurs successifs (noté $x$ et appelé offset au sens défini par [Shapiro, 1985] et [Shapiro et al, 1988]). L'heuristique *EST* est alors réutilisée plusieurs fois en imposant une borne inférieure de l'offset $x$ comprise entre ($x_{min}$) et ($x_{min} + x_{max}$)/2. Initialement la borne utilisée est égale à $x_{min}$, puis elle est régulièrement incrémentée d'une valeur $\delta$ jusqu'à atteindre sa valeur maximale. La valeur retenue est celle qui, parmi toutes celles évaluées, maximise le nombre de porteurs traités en un temps donné. Cette méthode permet aux auteurs de déterminer un cycle mono-produit pour une ligne comportant quatre cuves de traitement et de poste de chargement et de déchargement dissociés. Les auteurs déterminent également un ordonnancement 4-périodique pour une configuration identique comportant cinq cuves [Song et al, 1993]. Toutefois, il faut préciser que cette approche comporte deux hypothèses simplificatrices : les durées opératoires sont supposées fixes et les temps de déplacement à vide du robot sont considérés très faibles devant les durées de trempe.

Les mêmes auteurs ont également proposé un algorithme pour ordonnancer les opérations de transport lorsque les temps opératoires sont bornés [Storch et al, 1993]. La méthode présentée est basée sur l'utilisation de 10 règles heuristiques. Certaines de ces règles sont utilisées pour déterminer l'intervalle de temps auquel l'offset $x$ doit appartenir. Puis la procédure énumère les différentes valeurs de cette variable, dont chacune est utilisée comme donnée d'entrée

dans un module de simulation. Ce dernier est basé sur l'utilisation de règles heuristiques simples qui définissent le niveau de priorité associé à chaque porteur présent sur la ligne et gèrent l'affectation des tâches au robot. La simulation permet ainsi d'évaluer l'ordonnancement des tâches de transport associé à chaque valeur de $x$. la solution retenue est celle qui maximise le nombre de pièces non défectueuses produites en un temps donné. L'article cité reste toutefois ambigu quant à la prise en compte des déplacements à vide du robot, ce qui ne nous permet pas d'affirmer avec certitude que cette hypothèse simplificatrice a été supprimée.

En conclusion, à ce jour, l'EST est la seule approche développée pour l'ordonnancement 1-cyclique (voire r-cyclique) des lignes de traitements de surface. Bien que la solution obtenue garantisse des pièces de bonne qualité et satisfasse toutes les contraintes auxquelles est soumis le problème posé, les durées effectives de trempes restent fixées au préalable et non pas bornées.

**Limites de cette approche**

Cette approche est une adaptation de l'heuristique d'insertion au plus tôt des produits à traiter, plus connue en littérature sous le nom du *Earliest Starting Time* présentée dans [Yih, 1994]. Les auteurs se sont limités à l'étude du cas mono-produit, et pour simplifier leur modèle de résolution, ils ont fixé les durées opératoires aux bornes inférieures des fenêtres de temps. De ce fait, ils ne traitent pas non plus un problème de type HSP au sens strict du terme, mais un problème connexe de type *Cyclic Robotic Scheduling*.

Considérons l'exemple de deux types de produits traités sur une ligne à cinq cuves incluant les cuves de chargement (cuve 0) et de déchargement (cuve 4) et possédant la même séquence opératoire : d'abord la cuve 1 puis la cuve 2 et enfin la cuve 3 avant d'être mis à la cuve de déchargement. Les fenêtres de temps de chaque type de produit sont données dans le tableau III.1. et peuvent varier d'un type à l'autre. Le mouvement à vide du chariot entre deux cuves successives est de 5 u.t. et de 10 u.t. pour un mouvement en charge (incluant les mouvements de stabilisation et d'égouttage).

**Tableau III.1.** Données de l'exemple

| Type de produit | Fenêtre de temps | Cuve i | | | |
|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 |
| Job 1 | $[a_{2i}, b_{2i}]$ | [0 - +∞] | [80 - 96] | [41 - 46] | [61 - 66] |
| Job 2 | $[a_{2i+1}, b_{2i+1}]$ | [0 - +∞] | [76 - 91] | [21 - 26] | [30 - 66] |

La solution optimale pour ce problème a un temps de cycle de 272 u.t. (figure III.1) tandis que la période est de 353 u.t. pour le cycle obtenu avec l'heuristique EST (figure III.2), en imposant comme durées de trempe les bornes minimales des fenêtres de temps ($b_i=a_i$). L'écart relatif par rapport à la solution optimale avec temps bornés est compris entre 20 et 25%.

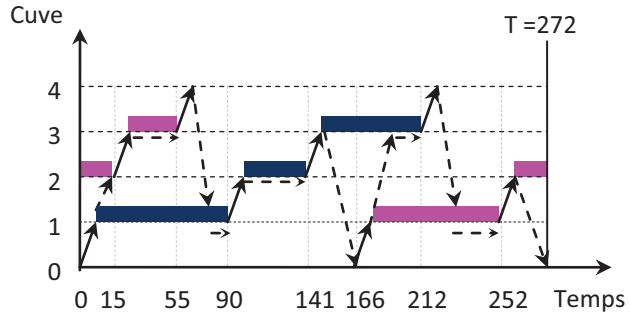



**Figure III.1.** Meilleure solution obtenue pour des durées opératoires bornées

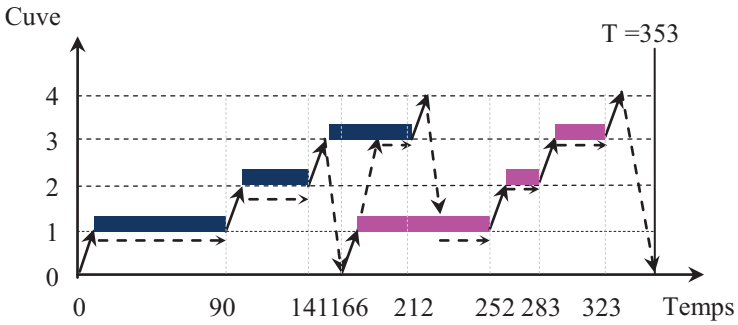



**Figure III.2.** Meilleure solution obtenue pour des durées opératoires fixées

Cet exemple illustre la performance relativement faible de l'heuristique EST par rapport à l'optimum. Ceci est dû principalement à la fois à la non remise en cause des opérations déjà séquencées (algorithme de type glouton) et au fait de fixer à priori les durées opératoires à leurs valeurs minimales (voir analyse au chapitre II, §II.4.2.b). Dans le but de résoudre le CHSP par une méthode approchée plus efficace, nous proposons une nouvelle approche de

résolution qui se base sur la flexibilité offerte par les durées opératoires bornées et non fixées. L'originalité de cette approche réside dans le fait que le degré du cycle est non fixé.

# III. 4 - Algorithme de liste pour le CHSP multi-produits

## III. 4. 1 - Analyse préliminaire

L'idée initiale que nous avions était d'adapter au cas cyclique une heuristique efficace générée pour résoudre des problèmes de HSP dynamiques. En effet, compte tenu du nombre limité de benchmarks existants pour le Dynamic HSP, plusieurs chercheurs ont utilisé des benchmarks connus en CHSP. Ils ont ainsi évalué l'efficacité de leurs modèles dynamiques en résolvant des instances déterministes (en termes d'arrivée des produits). C'est le cas par exemple de [Spacek et al, 1998] qui ont proposé une approche combinant l'algèbre des dioïdes et une règle heuristique de gestion des conflits. Pour le benchmark de [Phillips et Unger, 1976], leur algorithme a fourni une solution 3-périodique dès l'introduction du 3$^{ème}$ porteur sur la ligne.

A partir de ces premières observations et de l'analyse de l'état de l'art présenté au paragraphe précédent, nous avons orienté nos recherches vers l'élaboration d'une heuristique initialement basée sur l'heuristique la plus simple (EST), mais où plusieurs produits sont traités simultanément sur la ligne et où les durées opératoires sont bornées. En réalité, nous verrons que notre algorithme combine et adapte plusieurs heuristiques existantes.

## III. 4. 2 - Heuristique OCMPEST

Nous avons élaboré un algorithme de liste, baptisé OCMPEST (*Optimized Cyclic Multi-Parts Earliest Starting Time*), dans lequel à chaque étape $k$, une liste d'opérations candidates $L^k$ est testée ; puis une seule opération est ajoutée à cette liste selon un ordre de priorité explicité par la suite [El Amraoui et al, 2010][b].

La procédure de construction progressive que nous utilisons pour générer une séquence cyclique est une procédure par insertion et actualisation, inspirée des travaux de [Lei et Wang, 1989][a] et de [Hindi et Fleszar, 2004]. Ces auteurs utilisent dans leurs travaux une matrice spécifique pour la construction progressive de leurs ordonnancements, en choisissant à chaque étape le prochain transport à placer parmi une liste d'opération candidates. Toutefois, dans notre cas, et pour garantir une séquence périodique, nous imposons une séquence d'entrée cyclique pour les produits à traiter (par exemple A-B/A-B/…/A-B). L'introduction des produits sur la ligne est basée sur le principe d'insertion au plus tôt. Soumis à des contraintes

de disponibilité des ressources et des durées opératoires, un régime cycle permanent est établi, pour ce système de données. Ce régime va nous fournir le degré et le temps du cycle.

De manière plus précise, l'évolution de l'algorithme général mettant en oeuvre notre heuristique OCMPEST peut être décrite ainsi : après l'initialisation de la première liste des opérations candidates, la boucle principale de l'algorithme s'exécute tant que le critère d'arrêt n'est pas atteint. Le critère d'arrêt que nous utilisons est l'ordonnancement de la totalité des opérations de transports de tous les produits à insérer. Lors d'une itération de la boucle principale, la procédure de passage à l'itération *k+1* est lancée (ligne 3 de l'algorithme III.1), suivie par la procédure d'insertion de nouvelles opérations candidates dans la nouvelle liste des solutions potentielles (ligne 4). Cette dernière est ensuite triée par ordre croissant des dates de début au plus tôt des opérations de transport (ligne 5) et l'opération de rang 1 est la première à être sélectionnée comme opération candidate (ligne 6). Si la contrainte d'occupation des cuves est respectée (ligne 7) et si la contrainte de la durée opératoire est respectée (ligne 8) alors cette opération de rang 1 est sélectionnée dans la liste des opérations choisies (ligne 10). Sinon, si la cuve est libre mais que la durée de trempe dépasse la borne maximale, la valeur de ce dépassement est calculée et stockée (ligne 14). Un retour en arrière (*backtrack*) est alors autorisé pour considérer d'autres opérations candidates. Si durant le processus de retour on tombe sur l'opération de trempe du même produit et qui précède celle où on a détecté une violation de la contrainte de temps, dans ce cas le lancement de cette opération de trempe sera retardé de la valeur du dépassement (ligne 19) et on continue la résolution. Enfin, si la cuve est occupée, choisir l'opération suivante dans la liste $L^{k+1}$ (ligne 28).

**Algorithme III.1 :** *Algorithme de l'heuristique OCMPEST*

```
1  initialisation de la première liste des opérations candidates
2  tant qu'il reste des opérations de transport dans la séquence d'entrée cyclique faire
3      procédure de passage à l'étape k+1,
4      procédure d'insertion de nouvelles opérations candidates,
5      trier la liste L^{k+1} par ordre croissant des temps de début des opérations de transport,
6      choisir la première opération de la liste triée comme solution candidate,
7      si la cuve d'arrivée de la solution candidate est libre alors
8          si la durée opératoire de l'opération de transport vérifie la contrainte de
9          fenêtre de temps alors
10             choisir la première opération candidate de la liste,
11             sinon
12                 si la durée de trempe de l'opération choisie à l'étape k+1 dépasse la
13                 borne supérieure de la fenêtre de temps alors
```

```
                    procédure de calcul du dépassement
                    tant que pas de solution à l'itération k+1 faire
                        retour à la liste précédente L^k et choisir l'opération candidate
                        suivante,
                        si l' opération candidate est O_{i-1}^j alors
                            retarder la date de début de l'opération de transport O_{i-1}^j de
                            δ_{i-1}^j,
                            trier la liste des opérations candidates par ordre croissant des
                            dates de début au plus tôt,
                            reprendre la procédure de résolution,
                        fin
                    fin
                fin
        sinon   (si la cuve de destination est occupée) alors
                    choisir l'opération candidate suivante de la liste L^{k+1},
                fin
            fin
    fin
fin
```

Dans ce qui suit, nous expliquons plus en détail les différentes procédures mises en œuvre dans cet algorithme.

Dans la procédure d'initialisation (Procédure III.1), la première liste $L^0$ des opérations candidates est constituée de la première opération de transport en charge du produit 1 depuis la cuve de chargement jusqu'à la première cuve de traitement.

**Procédure III.1 :** *Initialisation*

// initialisation de la première liste des opérations candidates :

1 $L^0 \leftarrow \{O_0^1 = (1, 0)\}$ ;

La deuxième procédure (Procédure III.2) permet d'effectuer le passage de l'algorithme de l'itération *k* à l'itération suivante : *k+1*.

**Procédure III.2 :** *Passage de l'itération k à l'itération k+1*

// à chaque étape, l'itération k+1est formée ainsi :

1 tous les éléments de $L^k$ sont copiés dans $L^{k+1}$ ;

2 $O_i^j$, l'opération *i* du produit *j*, choisie à l'itération *k* est enlevée de la liste $L^{k+1}$ ;

3 la date $dt_k^p$ pour chaque opération $O_k^p$ dans la liste $L^{k+1}$est actualisée par l'expression suivante : $\max\left\{dt_k^{\,p};\left(ft_i^{\,j}+c_{i+1,k}^{\,j,p}\right)\right\}$ ;

avec $dt_k^p$ : date de début de l'opération de transport $O_k^p$,
$ft_i^j$ : date de fin de l'opération de transport $O_k^p$,
$c_{i+1,k}^{j,p}$ : durée du déplacement à vide de la cuve *i+1* à la cuve *k*.

La procédure III.3 illustre l'ordre d'exécution des opérations de transport par le robot.

**Procédure III.3 :** Insertion de nouvelles opérations candidates

// insertion de nouvelles opérations candidates selon la solution choisie à l'itération précédente $O_i^j$ :

1 si l'opération $O_i^j$ n'est pas la dernière opération du produit j alors insérer l'opération $O_{i+1}^j$ dans la liste $L^{k+1}$ et calculer $dt_{i+1}^j$ par la formule suivante:

$$dt_{i+1}^{\ j} \leftarrow \left( ft_i^{\ j} + a_{i+1}^{\ j} \right)$$

2 si l'opération $O_i^j$ n'est pas la première opération du produit j (i=0) alors insérer l'opération $O_{i+1}^j$ et la première opération du produit suivant dans la séquence cyclique d'entrée, définie précédemment ($O_0^{j+1}$) dans la liste $L^{k+1}$ et actualiser leurs dates de début respectivement par les équations suivantes :

$$dt_{i+1}^{\ j} \leftarrow \left( ft_i^{\ j} + a_{i+1}^{\ j} \right)$$

$$dt_{i+1}^{\ j} \leftarrow \left( ft_i^{\ j} + c_{i+1,0}^{\ \ j+j+1} \right)$$

avec $a_{i+1}^j$ : durée minimale de la $(i+1)^{ème}$ opération de trempe du produit *j*.

Pour minimiser la durée de cycle, et compte tenu de la flexibilité de l'ordonnancement due aux fenêtres de temps, nous proposons l'insertion des produits sur la ligne dès que possible : un nouveau produit fait donc son entrée sur la ligne si toutes les contraintes sont respectées (contraintes sur les cuves et sur la ressource de transport). Si une des contraintes est violée, l'ordonnancement réalisé est remis en question et on essaie de réparer la séquence par un retour itératif (*backtrack*).

La procédure III.4 vérifie le respect de la durée de trempe maximale à ne pas dépasser. En effet, la procédure III.3 garantit uniquement le respect de la durée de trempe minimale.

**Procédure III.4 :** *Respect des durées de trempe*

// test sur la durée de trempe effective

1 vérifier que la durée de trempe pour chaque solution candidate vérifie l'équation suivante:

$$dt_i^{\ j} \leq \left( do_i^{\ j} + b_i^{\ j} \right)$$

avec $b_{i+1}^j$ : durée maximale de la $(i+1)^{ème}$ opération de trempe du produit *j*,
$do_i^j$ : date de début de la $(i+1)^{ème}$ opération de trempe du produit *j*.

Enfin, la procédure III.5 a pour objectif d'optimiser les bactracks, en calculant le retard occasionné par la disponibilité de la ressource de transport, et en propageant ce retard en reculant du même temps la date de début des opérations de trempe antérieures du produit associé. L'objectif est d'absorber le dépassement identifié qui peut occasionner la non-conformité du produit. Soit $o_i^j$ la durée de la $i^{ème}$ opération de trempe du produit *j* qui présente un dépassement de la durée maximale de trempe tolérée $b_i^j$. Cet excédent est calculé et affecté à une variable $\delta_{i-1}^j$ (Procédure III.5).

**Procédure III.5 :** *Calcul du dépassement de la borne supérieure*

// calcul dû retard du au dépassement de la borne supérieure de la fenêtre de temps ($o_i^j > b_i^j$)
1 calculer le dépassement et l'enregistrer dans la variable $\delta_{i-1}^j$ :

$$\delta_{i-1}^{\ j} \leftarrow \left( dt_i^{\ j} - do_i^{\ j} - b_i^{\ j} \right)$$

## III. 4. 3 - Illustration

Considérons l'exemple de deux produits traités sur une ligne à deux cuves et où les postes de chargement (cuve 0) et de déchargement (cuve 3) sont dissociés. Les fenêtres de temps de chaque type de produit sont données dans le tableau III.2. Le déplacement à vide (resp. en charge) du chariot entre deux cuves successives est de 10 u.t. (resp. 20 u.t.).

**Tableau III.2.** Données de l'exemple

| Type de produit | Fenêtre de temps | Cuve i | | |
|---|---|---|---|---|
| | | 0 | 1 | 2 |
| Job 1 | $[a_{2i}, b_{2i}]$ | [0 - +∞] | [50 - 60] | [70 - 90] |
| Job 2 | $[a_{2i+1}, b_{2i+1}]$ | [0 - +∞] | [50 - 55] | [45 - 55] |

Pour cet exemple, on impose une séquence d'entrée cyclique, des produits de type 1 et 2 à traiter. Soit la séquence suivante: job1- job2- job1- job2- job1- job2- … et le quadruplet (j, c, n, $dt_i$), définissant la $i+1^{ème}$ opération de transport en charge du produit de type *j* et de rang *n* dans la séquence cyclique d'entrée de la cuve c à la cuve suivante, lancée à la date $dt_i$.
Le première liste d'initialisation contient la première opération de transport du produit de type 1 de la cuve 0 à la cuve 1. La date de début de cette opération de transport est pris comme

origine des temps, soit (1, 0, 1, 0). A la première étape de notre algorithme, la nouvelle liste des opérations candidates contient deux nouvelles opérations : (1, 1, 1, 70) et (2, 0, 2, 30). Après un tri par ordre croissant des temps de début, on optera pour la deuxième option. Toutefois, la cuve 1 est déjà occupée par le produit de type 1 et donc on devrait choisir la première opération candidate. La nouvelle liste à l'étape k=2 contient les deux options suivantes (1, 2, 1, 160) et (2, 0, 2, 110). D'après le critère de sélection adopté on choisira la solution (2, 0, 2, 110). En se basant sur le critère d'insertion, la nouvelle liste d'opérations candidates à l'étape k=3 contient trois options : (1, 2, 1, 160), (2, 1, 2, 180) et (1, 0, 3, 140). La première cuve est déjà occupé par le produit de type 2, donc on optera pour l'opération (1, 2, 1, 160). La nouvelle liste $L^4$ contient alors: (2, 1, 2, 200) et (1, 0, 3, 210). La cuve 2 est vide donc on choisit l'opération (2, 1, 2, 200). Toutefois, si on calcule la durée de trempe du produit de type 2 dans la cuve 1 (date de sortie au plus tôt moins la date d'entrée) on trouve 200-130=70 u.t., soit un dépassement de 15 u.t. du temps opératoire maximum autorisé. Par conséquent $\delta_0^2$= 15u.t (ligne 14 de l'algorithme III.1). Un backtrack dans ce cas est autorisé (ligne 16). La liste des opérations candidates à l'itération k=3 précédente contient : (1, 2, 1, 160), (2, 1, 2, 180) et (1, 0, 3, 140). Or les cuves 1 et 2 sont occupées donc un nouveau backtrack s'impose à l'étape k=2 avec $L^2$=((1, 2, 1, 160), (2, 0, 2, 110)). A ce niveau se trouve l'opération (2, 0, 2, 110) qui précède celle où on a détecté un dépassement de la durée de trempe. On retarde alors cette opération de transport de $\delta_0^2$ (= 15u.t.) (lignes : 19→20). On reprend la procédure avec une nouvelle liste $L^2$ modifiée contenant : (1, 2, 1, 160) et (2, 0, 2, 125). C'est de nouveau l'opération (2, 0, 2, 125) qui est sélectionnée et ainsi de suite jusqu'à l'ordonnancement de la totalité des opérations de transport. Le diagramme de Gantt de la solution 2-cyclique obtenue est donné à la figure III.3.

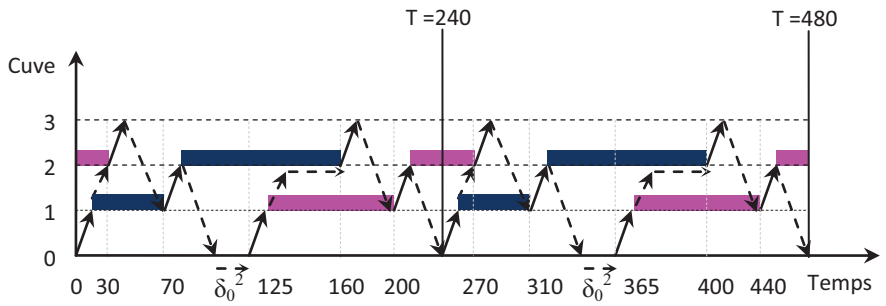



**Figure III.3.** Diagramme de Gantt de la solution obtenue par l'heuristique OCMPEST

Notons que pour cet exemple, notre démarche d'optimisation permet de rendre faisable une séquence cyclique initialement non faisable et donc de réduire le temps de cycle. En effet,

sans autoriser ce retard à vide, cette séquence serait irréalisable et le second produit ne serait introduit sur la ligne qu'une fois le premier déposé dans la cuve de déchargement.

Comme nous le verrons à la fin de ce chapitre, nous avons validé notre heuristique sur des benchmarks de la littérature. Nous avons également comparé les performances obtenues avec celles de l'heuristique de base EST, de notre méthode exacte développée au chapitre précédent, ainsi que de la méthode détaillée dans la deuxième partie de ce chapitre. En effet, bien que les résultats obtenus avec OCMPEST soient tout à fait satisfaisants, le choix d'un algorithme de liste peut se discuter comparé aux nombreuses autres méthodes approchées existantes. En ce sens, il nous a semblé important de poursuivre notre démarche en comparant notre algorithme avec une autre classe de méthodes approchées, en l'occurrence les méta-heuristiques, couramment utilisés pour résoudre des problèmes d'optimisation. D'autre part, comme nous l'avons déjà évoqué, un aspect de OCMPEST, qui peut être vu à la fois comme un avantage et une limite, est la non maîtrise du degré du cycle obtenu. C'est une autre raison pour laquelle nous avons développé un autre algorithme, mais cette fois avec degré de cycle imposé.

## Partie 2 : Approche métaheuristique

L'approche proposée dans cette partie repose sur l'utilisation d'un algorithme évolutionnaire et plus particulièrement un algorithme génétique. Nous débutons cette partie par une brève description de ce dernier type d'algorithme utile à la compréhension de la suite de ce chapitre.
Il existe de très nombreux ouvrages ou articles de synthèse traitant des algorithmes génétiques et évolutionnaires, auxquels le lecteur intéressé pourra se référer [Holland, 1975], [Goldberg, 1989][a], [Davis, 1991], [Goldberg, 1994], [Bäck, 1996], [Caux et al, 1995][b], [Portmann, 1996], [Gen et Cheng, 1997].
Cette description générale sera suivie de la présentation plus détaillée des choix faits pour développer un algorithme génétique dédié à la résolution du problème qui nous intéresse : le CHSP multi-produits. Puis nous présenterons les résultats obtenus en appliquant cette méthode de résolution sur des problèmes issus de la littérature scientifique.

## III. 5 - Principe général de fonctionnement des algorithmes génétiques (AG)

Les algorithmes génétiques font partie de la famille des algorithmes évolutifs. Ils sont basés sur le processus d'évolution génétique des organismes biologiques à travers des générations selon la théorie de l'évolution de Darwin. Ces méthodes sont basées sur un processus aléatoire de parcours pour l'exploitation de l'espace de recherche.

John Holland [Holland, 1975] est le premier à avoir développé les principes fondamentaux des AGs, utilisés par la suite, et pour la première fois, par [Golberg, 1989][a] pour la résolution des problèmes d'optimisation. Ces méthodes d'optimisations ont suscité l'intérêt de nombreux chercheurs à l'exemple de [Golberg, 1989][b], [Michalewicz et al., 1992], [Davis et al, 1993], [Mahfoud et Golberg, 1995], [Kacem et al, 2001], [Bourazza et al., 2002], [Dridi, 2004], [Deb et al., 2006], [Tangour, 2007].

Les algorithmes génétiques présentent plusieurs avantages par rapport aux algorithmes classiques d'optimisation. Ces avantages se résument ainsi :

- ✓ un critère de souplesse est conféré à la fonction à optimiser. En effet, on n'a pas besoin de se soucier de la nature de la fonction objectif à optimiser (continuité, dérivabilité, etc.) pour évaluer la fonction du coût ;
- ✓ une évaluation parallèle de plusieurs solutions candidates (population de taille *N*) au lieu de l'évaluation d'un seul point itéré dans les algorithmes classiques ;
- ✓ une diversification des solutions candidates grâce aux règles de transition probabilistes (probabilité de croisement et de mutation), nous permettant ainsi d'éviter de tomber dans des optimaux locaux et d'orienter plutôt la recherche vers un optimum global. Cette transition probabiliste n'est pas applicable dans le cas des algorithmes déterministes où la transition entre deux itérations successives est imposée par la structure et la nature de l'algorithme.

Cette approche a été validée pour des problèmes nécessitant une exploration performante et économique du point de vue temps de calcul. Cette approche est maintenant appliquée dans de nombreux domaines de l'optimisation, notamment l'ordonnancement et la logistique.

Les AGs constituent une classe de stratégies de recherche réalisant un équilibrage entre l'exploration et l'exploitation. Ils représentent des procédures faisant appel à un choix aléatoire comme outil pour guider l'exploration dans l'espace des paramètres codés

[Mesghouni, 1999]. Ce sont des algorithmes itératifs de recherche qui visent à optimiser une fonction prédéfinie, appelée fonction *coût* ou fonction *fitness*.

Les algorithmes génétiques emploient un vocabulaire emprunté à la fois à la biologie et à l'informatique. Les AGs traitent un ensemble d'*individus* appelé *population*. Chaque individu possède deux représentations, nommées *génotype* et *phénotype*. La première correspond à l'ensemble des allèles d'un individu portés par l'ADN d'une cellule vivante en biologie, traduite par une représentation codée d'une solution potentielle sous la forme d'un *chromosome* pour les AGs. Tandis que la seconde représentation, traduit l'aspect physique et physiologique observable de l'individu obtenu à partir de son *génotype*, ce qui correspond à la solution potentielle du problème à optimiser en utilisant la formulation originale du problème.

Un *chromosome* est composé par une succession linéaire de *gènes* et chaque *gène* peut prendre plusieurs valeurs appelées *allèles*.

Quelque soit la problématique considérée, l'architecture générale d'un algorithme génétique comporte six composantes principales (figure III.4) :

1. un codage approprié des solutions,
2. une génération procédurale de la population initiale (nommée génération initiale), ce choix conditionnera la rapidité de convergence des AGs vers l'optimum,
3. une fonction à optimiser appelée fonction d'évaluation ou encore *fitness*,
4. un mécanisme de sélection des individus candidats à un éventuel couplage,
5. une fonction de génération de nouvelles solutions au moyen de deux opérateurs génétiques :

   - opérateur de croisement : manipule les gènes des parents afin d'aboutir à des chromosomes meilleurs ou différents, appelés fils.

   - opérateur de mutation : modifie les valeurs des gênes de chromosomes afin de garantir l'exploration de l'espace.
6. un paramétrage de dimensionnement définissant la taille de la population, le critère d'arrêt, la probabilité d'application des opérateurs génétiques [Caux, 1995][b].

Pour plus de détails sur ces différentes composantes, le lecteur pourra se reporter à l'annexe 11.

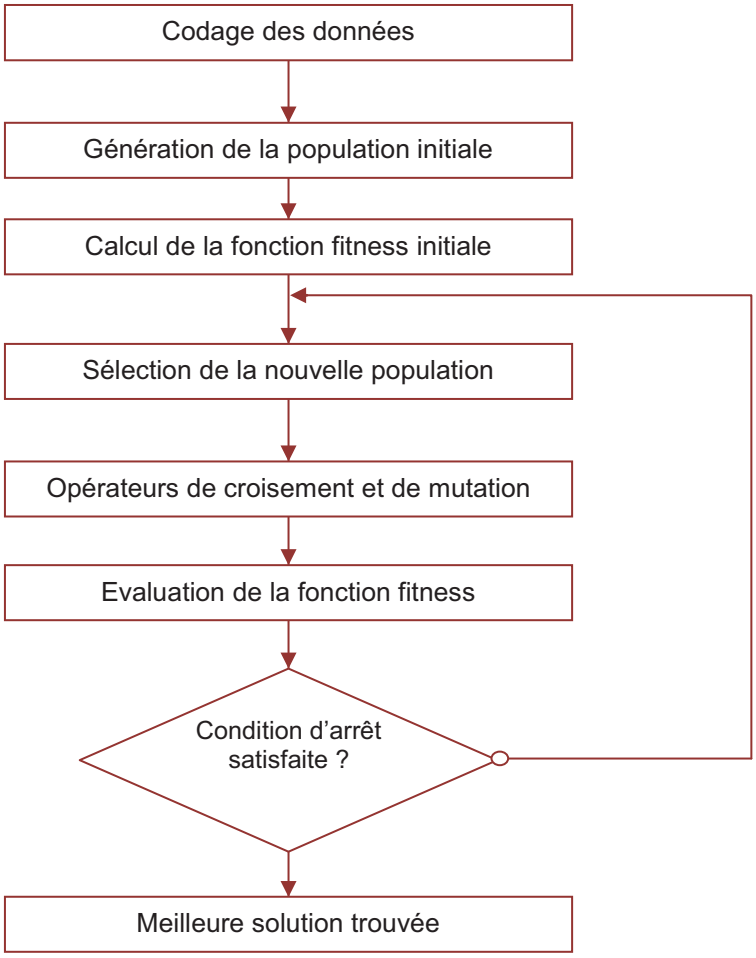



**Figure III.4.** Architecture générale d'un algorithme génétique

## III. 6 - Etat de l'art des CHSP résolus par des algorithmes génétiques :

Les algorithmes génétiques ont été relativement peu utilisés pour résoudre les HSP cycliques, et essentiellement dans des cas mono-produits. En 1997, Lim a développé un AG pour résoudre le problème de base [Lim, 1997]. Chaque chromosome utilisé pour représenter une solution est la liste ordonnée des cuves déchargées par le robot pendant un cycle. Comme le problème est mono-produit, les cuves sont numérotées dans l'ordre de la gamme opératoire, et toute tâche de transport s'effectue entre deux cuves voisines $k$ et $(k+1)$. Par conséquent, le numéro de la cuve de départ suffit à caractériser toute tâche de transport. L'auteur suppose de plus qu'un cycle commence toujours par un chargement (transport entre la cuve fictive 0 et la cuve 1). Ainsi, le chromosome {2 1 3 5 4} décrit un cycle où l'opération de chargement sera

suivie par un transport de la cuve 2 à la cuve 3, puis un autre de la cuve 1 à la cuve 2, etc. La population initiale est constituée de séquences générées aléatoirement. Les ordonnancements cycliques ainsi représentés sont évalués en utilisant la procédure développée par [Armstrong et al, 1994] pour déterminer les dates de début de transport lorsque leur ordre d'exécution est connu. Les chromosomes se voient ainsi attribuer une force dépendant de la période du cycle déterminée par la procédure d'évaluation. Le schéma classique de sélection par roulette (défini dans [Goldberg, 94]) est ensuite utilisé pour sélectionner les chromosomes devant être soumis aux opérateurs génétiques. Ces derniers sont également choisis parmi les techniques traditionnelles employées dans les AGs. Le croisement est celui défini par [Falkenauer et Buffoix, 91] et nommé *Linear Order Crossover* (ou *LOX*). Il consiste à choisir aléatoirement deux points de croisement, de manière à définir trois zones dans chacun des parents sélectionnés. Puis la zone centrale du parent 1 (resp. parent 2) est recopiée dans le fils 2 (resp. fils 1). Puis, les gènes manquants dans le fils 1 (resp. fils 2) sont complétés selon l'ordre de parent 1 (resp. parent 2) en commençant par la première zone et en terminant par la troisième. La mutation, quant à elle, est réalisée par permutation de deux gènes choisis aléatoirement dans la séquence. Dans chaque génération, les plus mauvais éléments de la population initiale sont remplacés par les nouveaux individus créés si ceux-ci sont de meilleure qualité. Les tailles de population utilisées varient de 10 à 30 individus et le critère d'arrêt est un nombre fixe de générations (égal à 1000 pour les tests expérimentaux décrits).

Un problème dans l'article de Lim est que l'utilisation d'opérateurs classiques entraîne forcément la génération de solutions non faisables. Or Lim n'évoque pas l'existence de procédures de réparation dans ces cas.

Manier et Lamrous (2006 et 2008) ont proposé un algorithme évolutionniste pour la résolution des problèmes cycliques mono-produit mais multi-robots avec zones de chevauchement. Les cuves de chargement et déchargement sont associées. Le codage des solutions est fait sur la base des mouvements du robot à vide contrairement aux codages habituels qui est fait sur les mouvements du robot en charge. Un avantage de ce codage est que le même formalisme permet de représenter des solutions à 1, 2 ou h robots. L'inconvénient majeur est de limiter l'exploration de l'espace de recherche et donc empêche parfois de trouver des solutions optimales. Dans cette approche, une procédure récursive est définie pour la construction d'un chromosome. La population initiale est générée aléatoirement, composée d'un ensemble de chromosomes de plusieurs tailles. Les auteurs proposent un programme linéaire mixte pour l'évaluation de la force (fitness) de chaque chromosome. Un sous-ensemble de solutions est sélectionné en adoptant une méthode élitiste et une sélection universelle stochastique. La

descendance est obtenue en adoptant un croisement en 2-points et une mutation qui consiste à permuter deux gènes sélectionnés aléatoirement. Après mutation, le nouvel individu remplace le chromosome parent dans la prochaine population si la solution produite est réalisable, sinon les auteurs utilisent deux procédures de réparation.

## III. 7 - Algorithme génétique pour le CHSP multi-produits

Compte tenu de l'intérêt de cet outil présenté précédemment et étant donné la complexité du problème traité [Lei et Wang, 1989][b], il nous a semblé judicieux de modéliser et résoudre le problème étudié par un AG. Pour cela, nous utilisons un codage direct pour la représentation des chromosomes et le principe du rang pour la sélection. L'originalité de notre approche réside dans la modélisation proposée de l'ordonnancement cyclique multi-produits hétérogène des lignes de traitement de surface et dans les algorithmes de correction appliqués et faisant intervenir des méthodes par voisinage.

Notre algorithme commence par générer une population initiale de *2*S* individus, pour lesquels nous calculons la fonction objectif et nous sélectionnons les individus les mieux appropriés en utilisant le principe de la roulette, une fois que cette population initiale a été testée et corrigée. Les individus, sujets de croisement, sont croisés par un opérateur de croisement en deux points classique avec une probabilité $P_c$, dite probabilité de croisement. Le reste des individus est muté par un opérateur de mutation avec une probabilité de mutation $P_m$ (où $P_m=1-P_c$).

Les individus ainsi générés seront insérés dans la nouvelle population, en utilisant une méthode d'insertion élitiste. A la fin, un test d'arrêt sera effectué, s'il est vérifié, c'est-à-dire si le nombre d'itérations maximales a été atteint, alors l'algorithme s'arrête et affiche la meilleure solution obtenue.

### III. 7. 1 - Représentation d'une solution

Notre algorithme parcourt l'espace des séquences admissibles d'opérations de transport, à la recherche d'une solution minimisant le temps de cycle relatif aux produits présents sur la ligne de traitement. Chacune des séquences 2-cycliques générée est représentée sous la forme suivante : $\{O_1^1 ; O_2^2 ; O_2^1 ; O_3^1 ; O_1^2 ; O_2^2\}$ où $O_i^j$ est la $i^{ème}$ opération de transport du produit $j$.

Considérons l'exemple d'une ligne de traitement à 5 cuves incluant les cuves de chargement (cuve 0) et de déchargement (cuve 4) et à 2 jobs suivants :

Job i $O^H_{(i,j,c)}$ : la $i^{ème}$ Opération de Transport du job $j$ de la Cuve $c$ à la Cuve $(c+1) \Rightarrow$ (*Gène*)

| | | | | |
|---|---|---|---|---|
| Job 1 | $O^H_{(1,1,0)} \Rightarrow (0)$ | $O^H_{(2,1,1)} \Rightarrow (1)$ | $O^H_{(3,1,2)} \Rightarrow (2)$ | $O^H_{(4,1,3)} \Rightarrow (3)$ |
| Job 2 | $O^H_{(1,2,0)} \Rightarrow (4)$ | $O^H_{(2,2,1)} \Rightarrow (5)$ | $O^H_{(3,2,2)} \Rightarrow (6)$ | $O^H_{(4,2,3)} \Rightarrow (7)$ |

Le nombre total des opérations de cet exemple est 8. Nous accordons à chaque opération de transport un entier entre 0 et 7 en respectant l'ordre des opérations pour chaque produit. Ainsi, les opérations de transport du premier produit auront comme code 0, 1, 2 et 3. Celles du produit 2 sont numérotées de 4 à 7.

Ainsi, un chromosome est représenté par un tableau d'entiers de 0 à 7 et le chromosome de la figure III.5 correspond à un ordonnancement cyclique faisable.




**Figure III.5.** Ordonnancement cyclique faisable des opérations de transport

En effet, cet ordonnancement respecte à la fois les contraintes d'antériorité, les contraintes d'occupation des ressources ainsi que les contraintes de temps. Un tel ordonnancement est décrit ainsi : tout d'abord, le produit 1 est chargé (opération de transport n°0) par le robot qui lui fait visiter successivement les trois cuves de traitement et pour le déposer à la fin dans la cuve de déchargement (opération de transport n°3). Ensuite, c'est le tour du second produit d'être chargé (opération n°4), traité et enfin déchargé (opération n°7). Durant le processus de traitement, le robot charge le produit d'une cuve de départ, le transporte et le dépose dans une cuve arrivée et pendant le processus de traitement dans la cuve arrivée, le robot reste immobile au dessus de la cuve de traitement, le temps que la durée minimale de traitement du produit dans cette cuve soit atteinte. Le processus de traitement du premier produit prend fin quand il est mis dans la cuve de déchargement et c'est au tour du produit 2 d'être traité.

## III. 7. 2 - Génération de la population initiale

La population initiale est générée d'une façon aléatoire (voir *procédure III.6*).

**Procédure III.6 :** *Création de la population initiale*

```
Fonction création de la pop. init. (Pop : Tableau de N colonnes et 2*S lignes) : Pop
1   |   individu : tableau d'entiers de dimension N,
2   |   individu[1] ← 0 ;
3   |   pour j de 2 à N faire
4   |   |   individu[j] ← (-1) ;
5   |   fin
6   |   pour j de 2 à N faire
7   |   |   rechercher aléatoirement un indice pour lequel individu[indice]=-1 ;
8   |   |   individu[indice] ← (j-1) ;
9   |   fin
10  |   copier individu dans Pop[1] ;
11  |   pour j de 1 à 2*S-1 faire
12  |   |   Muter individu ;
13  |   |   copier individu dans Pop[j] ;
14  |   |   copier Pop[1] dans individu ;
15  |   fin
    fin
```

Cette méthode, qui se base sur une méthode de recherche par voisinage, consiste à créer un premier individu (lignes : 2→5) d'une façon aléatoire ; puis à lui appliquer l'opérateur de mutation, pour générer le reste des individus de la population (lignes : 6→15).

Par ailleurs, la fixation du premier élément (ligne 2) du chromosome permet d'avoir un codage qui représente une application bijective entre l'ensemble de solutions et l'ensemble de recherche et aussi de minimiser le nombre de redondances dans la même population. Par conséquent l'opérateur de mutation n'agit pas sur le premier gène de l'individu qui reste inchangé et fixé à la première opération de transport du premier produit (0).

Les individus ainsi obtenus ne sont pas tous faisables. Par conséquent, pour remédier à ce problème, nous faisons appel à un premier algorithme de correction, basé sur un opérateur de mutation et sur l'avantage fourni par le codage direct proposé et dont les étapes sont détaillées dans la procédure suivante :

**Procédure III.7 :** *Procédure de correction de la population initiale : COR1*

```
1  k ← 0
2  tant que k < nb_itérations et le chromosome est non faisable faire
3  |   Mutation du chromosome ;
4  |   k++ ;
5  fin
6  si k = nb_itérations_MAX et le chromosome est non faisable alors
7  |  substituer le chromosome avec le vecteur [0,1, 2, …, N-1] ;
8  fin
9  passer au chromosome suivant ;
```

## III. 7. 3 - Opérateur de croisement

Nous utilisons l'opérateur de croisement en deux points illustré par la figure suivante :

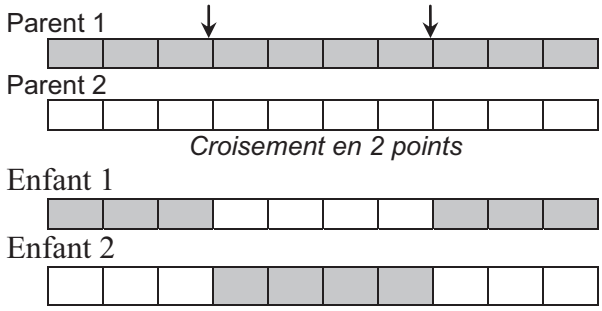



**Figure III.6.** Illustration de l'opérateur de croisement utilisé

Ces deux points de croisement ($i$ et $j$) sont déterminés d'une façon aléatoire tel que :

$$\forall i, j \in \{2, ..., N - 1\} \text{ on a } i, j \succ 1 \text{ et } i \neq j \ ;$$

Evidemment, l'utilisation d'un opérateur de croisement classique présente l'inconvénient majeur de ne pas tenir compte de certaines des contraintes du problème et donc entraîne un fort risque de génération d'individus non faisables. La mise en œuvre d'un tel opérateur doit donc s'accompagner de procédures de réparation permettant de pallier ce problème. Ces procédures sont décrites plus loin dans le mémoire.

## III. 7. 4 - Opérateur de mutation

L'opérateur de mutation est généralement utilisé pour éviter d'établir des populations uniformes incapables d'évoluer. Dans notre cas, cet opérateur est également utilisé pour corriger la population. En effet, quand un chromosome n'est pas faisable, on permute les valeurs de ses gènes en lui appliquant l'opérateur de mutation (voir figure III.7). Les deux

gènes à modifier : *i* et *j* sont déterminés d'une façon aléatoire et vérifient la condition précédemment définie pour les points de croisement. En effet, ces deux points doivent être différents pour que la fonction mutation à appliquer ait un sens et en plus, ils doivent être tous les deux différents de 0, pour garder le premier élément.

Notons que l'application de cet opérateur peut poser le même problème que pour le croisement, c'est-à-dire qu'il ne garantit pas la génération d'une solution faisable.

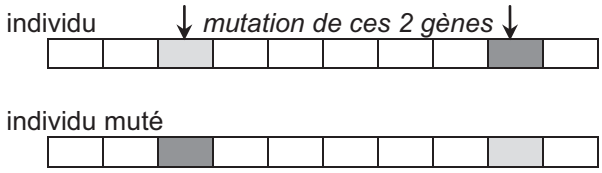



**Figure III.7.** Illustration de l'opérateur de mutation utilisé

## III. 7. 5 - Sélection

La méthode que nous avons utilisée pour la sélection est la sélection par rang, définie dans l'annexe 11. Ainsi, grâce à cette méthode, les meilleurs individus ayant les plus faibles fonctions objectifs sont plus sollicités à être sélectionnés pour l'opération de croisement.

## III. 7. 6 - Approche polynomiale pour le temps de cycle

Pour le calcul d'une borne de la durée du cycle T, nous nous sommes inspirés des travaux de [Armstrong et al, 1994] adaptés au cas mono-produit, et qui définissent une borne minimale pour cette durée. Cette borne minimale peut-être définie comme le chemin le plus long dans un graphe orienté. Les étapes la procédure que nous avons développée sur cette base, seront détaillées dans ce paragraphe ainsi qu'un exemple de calcul.

Pour un individu donné, c'est-à-dire, pour une séquence cyclique donnée $S = \langle [0], [1], [2], ..., [N-1] \rangle$, un graphe orienté G(V,E) peut-être construit. Chaque nœud dans V(G) = {0, 1, 2, …, N-1} représente un mouvement de S. A titre illustratif, on propose la représentation nodale suivante pour un nœud i de V(G), ($i \in V(G)$):

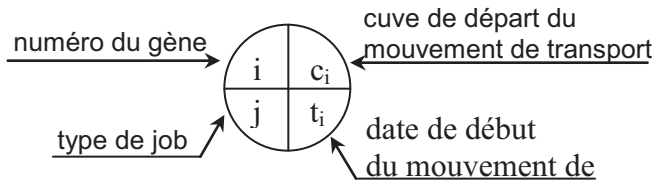



**Figure III.8.** Représentation nodale d'un mouvement de transport (gène)

Dans le graphe établi, on définit deux types d'arcs orientés. Le premier type garantit le respect de la durée de trempe minimale alors que le second assure que le chariot a suffisamment du temps pour se déplacer à vide entre deux mouvements de transport de la séquence cyclique (figures III.9 et III.10) [El Amraoui et al, 2010][c].

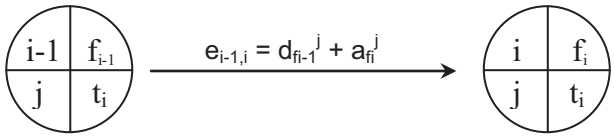



**Figure III.9.** Modélisation de la contrainte du respect de la durée minimale de trempe

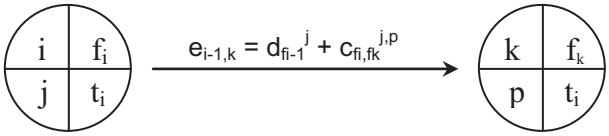



**Figure III.10.** Modélisation de la contrainte du déplacement à vide du robot

Pour tout ordonnancement cyclique S et SM = (S-S), SM étant constitué de deux cycles consécutifs, un graphe orienté G(V,E) peut être élaboré. Chaque nœud de V(G) représente un mouvement de transport dans SM. On définit la fonction $\begin{matrix} e: E \rightarrow \mathbb{R}^+ \\ (i,j) \mapsto e_{i,j} \end{matrix}$ ;

$e_{i,j}$ étant le poids de l'arc joignant le nœud *i* au nœud *j*.

En utilisant le graphe de SM au lieu de S, on cherche à simplifier la démarche de calcul du temps de cycle. Pour cela, on définit une nouvelle variable $Z_{k,k}$, définie comme le plus long chemin entre deux différents nœuds *k*, du graphe SM et appartenant chacun à un cycle S.

Ainsi, pour une séquence $S = \langle [0], [1], [2], [3] \rangle = \langle 0, 1, 3, 2 \rangle$, la séquence SM qui en découle est donnée par : $SM = \langle 0, 1, 3, 2, 0, 1, 3, 2 \rangle$, et les différentes longueurs $Z_{k,k}$ à calculer sont illustrées sur la figure suivante :

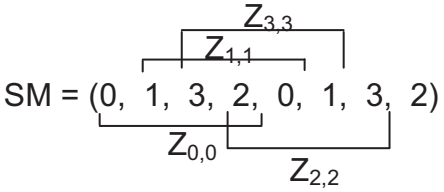



**Figure III.11.** Illustration de la longueur $Z_{k,k}$

Pour une séquence cyclique donnée et pour toute variable $k$, la longueur optimale $Z_{k,k}$ peut-être déterminée par un calcul récursif simple illustré par la procédure III.8.

**Procédure III.8 :** *Calcul des variables $Z_{k,k}$*

// Initialisation des paramètres $Z_{k,k}$
**pour** *k = 0, ..., N-1,* **faire**
  $Z_{[k],[k]}=0$;
**fin**
// Calcul des paramètres $Z_{k,k}$
**tant que (**$i < k < N-1$**) faire**
  $$Z_{[i][k]} = \max\left\{\begin{array}{l}\alpha.(Z_{[i][\beta]} + d_{[i]}^{[j]} + a_{[i]}^{[j]}),\\ Z_{[i][k-1]} + d_{[k-1]}^{[j,p]} + C^{[j,p]}_{[k-1]+1,[k]}\end{array}\right\};$$
  k++ ;
**fin**

$Z_{i,j}$, est définie comme le plus long chemin entre deux différents nœuds $i$ et $j$ du graphe. $\alpha$ est une variable binaire, non nulle dans le cas où il existe un entier $\beta$ de SM, tel que $i < \beta < k$ et vérifiant $[\beta] = [k]-1$. $[k]$ est un mouvement de transport dont la cuve d'origine n'est pas la cuve de chargement.

Cette procédure récursive utilisée pour le calcul du temps de cycle nous permet de déterminer les dates de début des mouvements de transport en charge pour la séquence cyclique considérée, en utilisant la relation suivante :

$$t_{[k]} = Z_{[0][k]} \qquad \text{(III.1)}$$

La borne minimale ($LB$) de la durée du temps de cycle S peut être par ailleurs déterminée par :

$$LB = \max_{0 \le k \le N-1} \left\{ Z_{[k][k]} \right\} \qquad \text{(III.2)}$$

Cette valeur, non forcément atteignable, correspond au temps de cycle du problème relaxé par rapport aux contraintes de capacité de cuves et de temps de trempe maximaux.

En conclusion, pour un individu donné (chromosome) la procédure de calcul de cette borne minimale du temps de cycle peut-être décrite comme suit.

**Procédure III.9 :** *Calcul de la Lower Bound pour un individu donné*

```
// Pour un individu donné (chromosome donné)
définir la séquence cyclique S ;
définir la séquence cyclique SM et construire son graphe orienté ;
calculer toutes les variables Z[k][k] ;
déterminer la valeur de la Lower Bound du temps de cycle ;
```

## III. 7. 7 - Faisabilité d'un chromosome

La faisabilité d'un chromosome ou encore de la séquence cyclique dépend du respect des contraintes du problème. En effet, il existe deux contraintes strictes du problème qui peuvent entrainer la non faisabilité de la séquence et par conséquent du chromosome.

Le premier type de contraintes est lié à la capacité des ressources et au type de produits à traiter. En effet, contrairement au cas mono-produit, nous devons impérativement vérifier le type de produit en plus de la contrainte de capacité. A titre d'exemple, si deux mouvements de transport sont à réaliser successivement dans la séquence cyclique, sans transport à vide entre les deux, il faut s'assurer qu'ils concernent bien le même produit. Dans le cas contraire, la séquence ne serait pas faisable. Pour vérifier la faisabilité du chromosome, la procédure III.10 doit être exécutée.

**Procédure III.10 :** *Vérification de la contrainte d'occupation des cuves*

```
// Initialisation du tableau EtatCuves
pour k = 0, ..., nbcuves-1, faire
    EtatCuves[k] ← 0 ;
fin
pour k = 0, ..., N-1, faire
    si un mouvement k de transport a été sélectionné alors
        si la cuve f_k, cuve de départ du mouvement i est occupé par un produit du
           même type que celui transporté alors
            si la cuve f_{k+1}, cuve de destination du mouvement k est libre alors
                EtatCuves[f_k]=0;
                EtatCuves[f_{k+1}]=1;
                i++;
                sinon
                contrainte d'occupation des cuves est violée et chromosome non
                faisable ;
            fin
        sinon
        contrainte d'occupation des cuves est violée et chromosome non faisable ;
    fin
fin
fin
```

La disponibilité des cuves de la ligne de traitement de surface est traduite par les zéros du tableau EtatCuves alors que leur indisponibilité est traduite par des uns. Au début, toutes les cuves sont libres : initialisation du tableau EtatCuves à zéro (lignes : 1→3). Ensuite, pour chaque mouvement de transport sélectionné dans la séquence cyclique, un premier test est lancé pour vérifier le type de produit qui occupe la cuve de départ, dans le cas où elle est déjà occupée (lignes : 6→7). Si ce test est vérifié, un second test est exécuté pour vérifier la disponibilité de la cuve de destination du mouvement de transport (ligne : 8). Dans le cas où ces deux tests sont vrais, la séquence partielle est faisable, alors la cuve de départ est libérée (ligne : 9) et celle d'arrivée est occupée (ligne : 10).

Quant au second type de contraintes, il est lié aux durées de trempe qui doivent respecter les bornes minimales et maximales des fenêtres de temps. Ce type de contraintes est vérifié au moyen de la procédure de test III.11.

**Procédure III.11 :** *Vérification du respect de la contrainte de fenêtres de temps*

```
// Vérification de la contrainte de fenêtre de temps
1  pour i = 0, ..., N-1, faire
2      si ( min{ t_{i+1} − t_i − d_i − a_{i+1}, b_{i+1} − t_{i+1} + t_i + d_i } ≥ 0 ) alors
3          i++ ;
4          sinon
5          contrainte du respect des fenêtres de temps est violée et chromosome non faisable ;
6      fin
7  fin
```

## III. 7. 8 - Méthode d'insertion

Pour l'insertion, nous utilisons la méthode d'insertion élitiste dont le principe consiste à recopier le meilleur chromosome de l'ancienne population dans la nouvelle. Cette dernière est complétée, petit à petit, par les individus générés grâce aux opérateurs génétiques, jusqu'à ce que la taille de la population atteigne la taille fixée (constante de génération en génération).

## III. 7. 9 - Notre démarche de résolution

Les séquences 2-cycliques générées par notre approche évolutionnaire doivent impérativement respecter les contraintes liées au partage des ressources et les contraintes de bornes maximales caractérisant les durées opératoires. Nous utilisons notre procédure polynomiale pour le calcul de la borne du temps de cycle ainsi que pour la détermination des dates de début des mouvements de transport. Cette borne minimale correspond à la valeur du plus long chemin dans un graphe orienté. Sans toutefois oublier, qu'il existe toujours, au moins une solution réalisable de qualité médiocre mais respectant les différentes contraintes du problème. Cette solution correspond à la borne maximale (*Upper Bound*) du critère à optimiser et consiste à faire traiter intégralement un produit sur la ligne avant d'introduire le produit suivant. Cette solution potentielle sera utilisée au cours de notre approche pour remédier au problème du nombre élevé des solutions non faisables. En effet, citons à titre indicatif, que pour l'exemple de référence [Phillips et Unger, 1976] seulement 233 solutions parmi les 479 millions possibles sont faisables. Rappelons qu'il s'agit ici d'un problème cyclique mono-produit, moins contraignant que le cas multi-produits hétérogène.

Le problème de la multitude des solutions non faisables, inhérent au HSP, nous a incités à chercher une démarche pour remédier à ce problème. L'algorithme (COR1) présenté précédemment, nous a permis dans un premier temps de corriger la population initiale. Dans un second temps, nous avons utilisé l'algorithme (COR2) pour corriger les populations suivantes. Cette démarche évolutive est résumée dans l'algorithme principal III.2.

**Algorithme III.2 :** *Algorithme Génétique pour le CHSP multi-produits*

0 Initialisation de la population et réparation avec COR1
1 Générer une nouvelle population de *2*S* chromosomes en appliquant l'opérateur de *croisement* sur les couples de chromosomes tirés aléatoirement avec une probabilité de $p_c$ et en appliquant l'opérateur de *mutation* avec la probabilité ($1$-$p_c$);
2 Tester si la contrainte d'occupation des cuves est respectée et corriger la population en appliquant la procédure *COR2* ;
3 Evaluer les dates de début des mouvements de transport ainsi que la valeur du temps de cycle pour les différents chromosomes de la population ;
4 Tester si la contrainte de durée de trempe est respectée et corriger la population en appliquant la procédure *COR2* ;
5 Sélectionner les S meilleurs chromosomes de la population ;
6 Répéter depuis 1 jusqu'à ce que le nombre maximal d'itérations soit atteint ;

Les individus créés au cours du processus évolutionnaire sont pour la plupart non faisables. Par conséquent, nous avons utilisé une seconde procédure de correction, basé sur le principe d'élitisme et dont les étapes sont détaillées dans la procédure suivante :

**Procédure III.12 :** *Correction de la population n : COR2*

```
1  k ← 0
2  tant que k < nb_itérations et le chromosome est non faisable faire
3  |   Mutation du chromosome ;
4  |   k++ ;
5  fin
6  si k = nb_itérations_MAX et le chromosome est non faisable alors
7  |  substituer le chromosome avec le meilleur chromosome de la population précédente ;
8  fin
9  passer au chromosome suivant ;
```

## III. 7. 10 - Illustration de la démarche de calcul sur un chromosome

Considérons l'exemple du tableau III.3. On suppose, dans cet exemple, le traitement de deux produits sur une ligne à 5 cuves non inclus les postes de chargement et de déchargement (considérés dissociés) ; la gamme de traitement de chaque produit consiste à le faire passer successivement dans toutes les cuves et dans l'ordre de leur disposition et indexation sur la ligne.

On suppose également que la durée de déplacement du robot en charge (respectivement à vide) entre deux cuves successives est de 10 u.t. (respectivement 5 u.t.).

**Tableau III.3.** Exemple de gammes opératoires partiellement identiques [Mateo, 2001]

| Type de produit | Fenêtre de temps | Cuve i | | | |
|---|---|---|---|---|---|
| | | 0 | 1 | 2 | 3 |
| Job 1 | $[a_{2i}, b_{2i}]$ | [0 - +∞] | [40 - 100] | [20 - 80] | [25 - 75] |
| Job 2 | $[a_{2i+1}, b_{2i+1}]$ | [0 - +∞] | [35 - 105] | [25 - 80] | [35 - 100] |

Dans cet exemple illustratif, il existe 8 mouvements de transport en charge dans une séquence cyclique, numérotés de 0 à 7. Nous avons appliqué notre procédure de calcul sur la séquence cyclique de transport $S = \langle 0, 6, 1, 7, 2, 4, 3, 5 \rangle$. Comme on peut le constater, S respecte les contraintes relatives à la disponibilité des cuves. Pour vérifier la contrainte sur les fenêtres de durées de trempe, nous devons générer la séquence cyclique SM à partir de S :

$SM = \langle 0, 6, 1, 7, 2, 4, 3, 5, 0, 6, 1, 7, 2, 4, 3, 5 \rangle$. Cette séquence et le graphe orienté associé, sont donnés respectivement par les figures III.12 et III.13.

Pour calculer les variables $Z_{k,k}$, nous commencerons par calculer $Z_{0,0}$ (figure III.13). Pour le premier type d'arcs du graphe, les valeurs des poids sont calculées ainsi :
$e_{[0],[1]}=d_0^{\,1}+c_{1,2}^{\,1,2}$ =15+10.(2-1)=25; $e_{[1],[2]}=d_2^{\,2}+c_{3,1}^{\,2,1}$ =15.(3-2)+10.(3-1)=35; $e_{[2],[3]}=d_1^{\,1}+c_{2,3}^{\,1,2}$ =15.(2-1)+10.(3-2)=25; $e_{[3],[4]}=d_3^{\,2}+c_{4,2}^{\,2,1}$ =15.(4-3)+10.(4-2)=35; $e_{[4],[5]}=d_2^{\,2}+c_{12}^{\,12}$ =15.(3-2)+10.(3-0)=45; $e_{[5],[6]}=d_0^{\,1}+c_{12}^{\,12}$ =15.(1-0)+10.|1-3|=35; $e_{[6],[7]}=d_0^{\,1}+c_{12}^{\,12}$ =15.(4-3)+10.(4-1)=45; $e_{[7],[8]}=d_0^{\,1}+c_{12}^{\,12}$ =15.|1-2|+10.(2-0)=35; $e_{[8],[9]} = e_{[0],[1]}$ =25; $e_{[9],[10]} = e_{[1],[2]}$ =35; $e_{[10],[11]} = e_{[2],[3]}$ =25; $e_{[11],[12]} = e_{[3],[4]}$ =35; $e_{[12],[13]} = e_{[4],[5]}$ =45; $e_{[13],[14]} = e_{[5],[6]}$ =35; $e_{[14],[15]} = e_{[6],[7]}$ $=d_0^{\,1}+c_{12}^{\,12}$ =45;

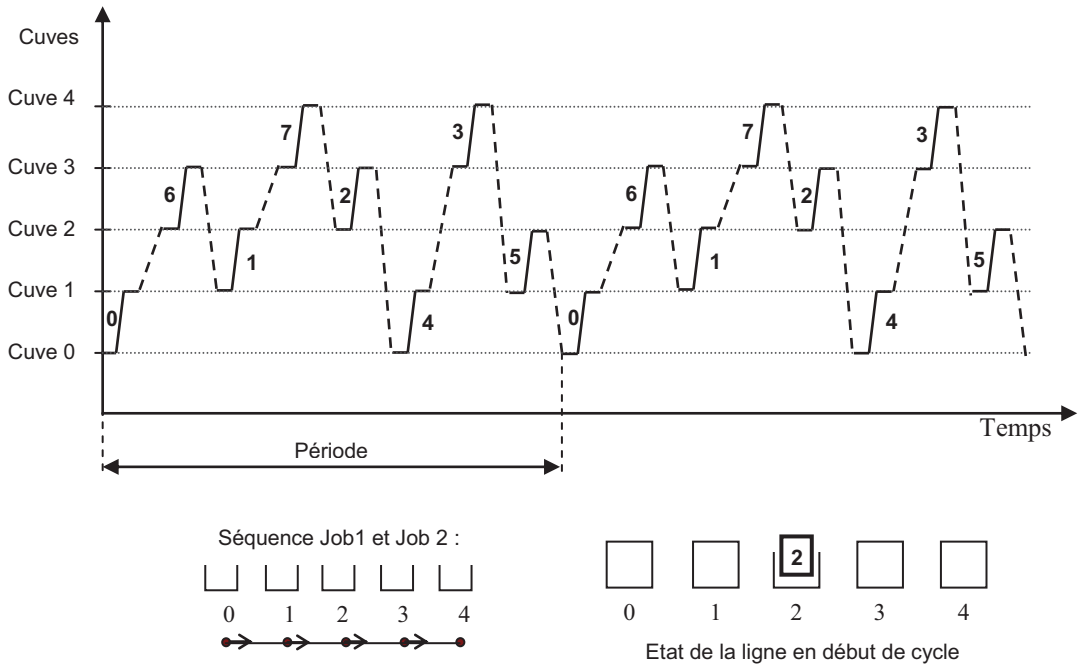



**Figure III.12.** Diagramme de Gantt de la séquence cyclique SM

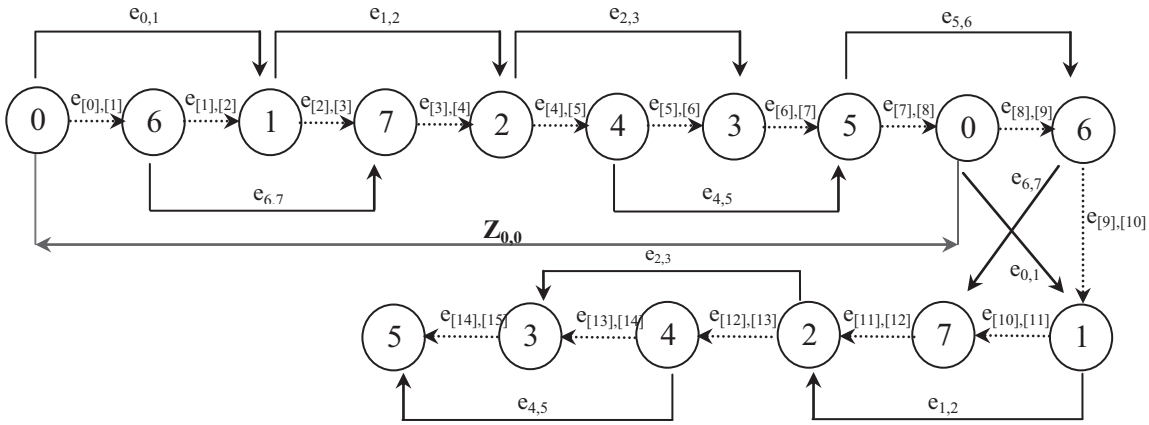



**Figure III.13.** Schéma du graphe orienté de la séquence cyclique SM

Quant au second type d'arc, les valeurs des poids sont :

$e_{0,1}= d_0^1+a_1^1=15.(1-0)+40=55$; $e_{1,2}= d_1^1+a_2^1=15.(1-0)+20=35$;

$e_{2,3}= d_2^1+a_3^1=15.(3-2)+25=40$; $e_{4,5}= d_1^2+a_1^2=15.(1-0)+35=50$;

$e_{5,6}= d_2^2+a_2^2=15.(2-1)+25=40$; $e_{6,7}= d_3^2+a_3^2=15.(3-2)+35=50$;

Le calcul des variables $Z_{[0],[k]}$, se fait de manière récursive (procédure III.8) et nous en déduisons les dates de début des mouvements de transport de la séquence cyclique.

$t_{[0]}= z_{[0],[0]}= 0$;

$t_{[1]}= z_{[0],[1]}= z_{[0],[0]}+e_{[0],[1]}= 25$;

$t_{[2]}= z_{[0],[2]}= \max\{z_{[0],[1]} + e_{[1],[2]}; z_{[0],[0]}+e_{0,1}\}= \max\{60;55\}= 60$;

$t_{[3]}= z_{[0],[3]}= \max\{z_{[0],[2]} + e_{[2],[3]}; z_{[0],[1]}+e_{6,7}\}= \max\{85;75\}= 85$;

$t_{[4]}= z_{[0],[4]}= \max\{z_{[0],[3]} + e_{[3],[4]}; z_{[0],[2]}+e_{1,2}\}= \max\{120;95\} =120$;

$t_{[5]}= z_{[0],[5]}= z_{[0],[4]} + e_{[4],[5]} = 165$;

$t_{[6]}= z_{[0],[6]}= \max\{z_{[0],[5]} + e_{[5],[6]}; z_{[0],[4]}+e_{2,3}\} = \max\{200;160\}= 200$; $t_{[7]}= z_{[0],[7]}= \max\{z_{[0],[6]} + e_{[6],[7]}; z_{[0],[5]}+e_{4,5}\} = \max\{245;215\}= 245$;

et en conclusion: $Z_{0,0}= t_{[8]}=z_{[0],[7]} + e_{[7],[8]} = 280$;

En appliquant cette même démarche de calcul pour le reste des variables $Z_{k,k}$, nous pouvons déterminer la valeur de notre *Lower Bound* $T_{LB}$ où $T_{LB}= \max\{Z_{0,0}; Z_{1,1}; Z_{2,2}; Z_{3,3}; Z_{4,4}; Z_{5,5}; Z_{6,6}; Z_{7,7}\}= 280$ u.t..

Les durées opératoires, pour chaque type de produit, sont déduites des dates de début des mouvements de transport (tableau III.4). [El Amraoui et al, 2010][c]

**Tableau III.4.** Durées opératoires effectives pour chaque type de produit

| Type de produit | Cuve i | | |
|---|---|---|---|
| | 1 | 2 | 3 |
| Job 1 | 80 | 41 | 61 |
| Job 2 | 76 | 25 | 30 |

Ces durées opératoires effectives vérifient bien les contraintes de fenêtres de temps. Par conséquent, la séquence 2-cyclique S est faisable et la durée calculée de 280u.t. correspond à la durée du temps de cycle optimale.

# Partie 3 : Résultats, comparaison et bilan

## III. 8 - Résultats des simulations

Dans la littérature, il existe très peu de benchmarks sur le HSP, et encore moins sur le CHSP multi-produits. Nous avons donc en majorité validé nos heuristique et métaheuristique pour des instances proposées par Mateo dans [Mateo, 2001] (180 exemples). A notre connaissance, ce sont les seules instances disponibles traitant du CHSP multi-produits suivant la même séquence opératoire mais avec des bornes de durées différentes.

Les résultats des tests effectués sont donnés en annexes 12 et 13. Le tableau III.5 présente les résultats obtenus pour quelques unes des instances que nous avons simulées avec notre programme linéaire 2 et/ou 4-cyclique, avec l'heuristique EST de Song et al. (1995), avec notre algorithme de liste OCMPEST, et enfin avec notre algorithme génétique. Pour les algorithmes génétiques, les solutions fournies sont la meilleure période trouvée ($T_{best}$) et la moyenne du temps de cycle 2-périodique obtenue sur 5 réplications ($T_{moyen}$). La taille de la population est de 15 et le nombre d'itérations est de 1000.

L'analyse de l'ensemble des résultats montre que :

- ✓ les solutions 2-cycliques obtenues avec EST sont loin de l'optimum ;
- ✓ notre heuristique OCMPEST fournit de bien meilleurs résultats que EST : par exemple pour les instances 102040605 et 102070705, EST se situe respectivement à 89.05% et 124.56% de l'optimum, contre 18,44% et 2.66% pour OCMPEST. Ceci n'a rien d'étonnant puisque la fixation des durées opératoires et la non remise en cause des décisions à une itération précédente font perdre à la méthode EST une certaine flexibilité, et figent certainement le degré des cycles générés. Ceci dit, cette flexibilité n'a pas que des avantages : avec OCMPEST, on ne maitrise pas le degré du cycle de la solution obtenue et on peut se poser la question de la difficulté de la mise en œuvre pratique d'un cycle *r*-périodique quand r est relativement grand ;
- ✓ les exemples du tableau II.3 montrent des écarts relatifs tout à fait acceptables entre les solutions r-cycliques de OCMPEST et les cycles optimaux de même degré : entre 2.66% et 18,44%, mais en réalité entre 0% et 28.82% pour l'ensemble des tests en annexe 12 pour des lignes entre 5 et 10 cuves. Ainsi, pour certains exemples en particulier notre heuristique atteint la solution optimale.

**Tableau III.5 :** Résultats des simulations des benchmarks de [Mateo, 2001] et autres

| Référence exemple | Solutions Optimales | | | Solution EST | | | Solution OCMPEST | | | Solution AG | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | r | T | T/r | r | T | T/r | r | T | T/r | r | $T_{best}$ | $T_{moyen}$ |
| 105040905 | 2<br>4 | 319<br>622 | 159.5<br>155.5 | 2 | 719 | 259.5 | 4 | 662 | 165.5 | 2 | 319 | 334 |
| 102040605 | 2 | 347 | 173.5 | 2 | 657 | 328 | 2 | 411 | 205.5 | 2 | 347 | 470.7 |
| 102070705 | 2 | 338 | 169 | 2 | 759 | 379.5 | 2 | 347 | 173.5 | 2 | 347 | 388.6 |
| 203050905 | 2<br>4 | 260<br>520 | 130<br>130 | 2 | 487 | 243.5 | 4 | 589 | 147.25 | 2 | 260 | 268 |
| 209050705 | 2 | 377 | 188.5 | 2 | 167 | 383.5 | 6 | 1168 | 194.67 | 2 | 377 | 411.2 |
| 204080905 | 2<br>4 | 325<br>646 | 162.5<br>161.5 | 2 | 631 | 315.5 | 2 | 363 | 181.5 | 2 | 325 | 357 |
| 117071706 | 2 | 362 | 181 | 2 | 746 | 373 | 4 | 700 | 175 | 2 | 362 | 535.4 |
| 125043007 | 2 | 653 | 326,5 | 2 | 1115 | 557,5 | 6 | 1868 | 311 | 2 | 653 | 730.6 |
| 137073308 | 2 | 583 | 291,5 | 2 | 1157 | 578,5 | 10 | 2858 | 285 | 2 | 583 | 606.8 |
| 159045910 | 2 | 737 | 368.5 | 2 | 1324 | 662 | 4 | 1256 | 314 | 2 | 737 | 737 |
| 20 cuves | - | - | - | 2 | 2212 | 1106 | 20 | 5924 | 296,2 | 2 | 1111 | 1331,2 |
| 30 cuves | - | - | - | 2 | 3135 | 1567,5 | 20 | 8863 | 443,15 | 2 | 1280 | 1709,8 |
| 40 cuves | - | - | - | 2 | 3997 | 1998,5 | 20 | 12566 | 628,3 | 2 | 1783 | 2092,8 |
| 50 cuves | - | - | - | 2 | 4849 | 2474 | 20 | 15110 | 755,5 | 2 | 2724 | 3037,1 |

- ✓ pour des degrés de cycle supérieurs à 2, notre approche exacte ne parvient pas toujours à trouver la solution optimale en temps raisonnable (limité à 24h de simulation), ou en tout cas à prouver l'optimalité de la meilleure solution trouvée, et ce même pour certaines instances de faible taille (5 cuves). Ceci implique deux conséquences : la première est que la résolution du modèle linéaire doit être améliorée pour repousser les limites de son utilisation ; la deuxième conséquence est que, dans l'état actuel de nos modèles, cela confirme encore l'intérêt de l'utilisation d'une méthode approchée. Pour les cycles r-périodiques (r>2) pour lesquels nous n'avons pas de solutions optimales, nous avons comparé les solutions de OCMPEST avec les 2-cycles optimaux. Dans ce cas les écarts relatifs observés se situent entre -14.79% et 23.29%. Les écarts relatifs font état d'une amélioration, pour certaines instances, dont l'importance n'est pas

toujours proportionnelle avec le degré du cycle trouvé par OCMPEST : ainsi la meilleure amélioration d'un 2-cycle optimal est fournie par une solution 4-cyclique (instance 159145910 du tableau III.5), alors que certaines solutions de degré de cycle supérieur ne conduisent pas toujours à une amélioration (c'est le cas par exemple pour l'instance 209050705) ;

- ✓ Les 4 dernières lignes du tableau montrent que nous avons pu résoudre des problèmes de taille conséquente (problèmes de 20 à 50 cuves générés aléatoirement : voir données en annexe 15). Toutefois, même notre heuristique OCMPEST a parfois du mal à résoudre ce type d'instances en temps raisonnable. Ceci dépend de la largeur des fenêtres temporelles associées à l'instance considérée. En effet, plus les fenêtres sont étroites, plus le nombre de backtracks est important. De plus, la valeur importante du degré du cycle obtenu ($r$=20 pour ces 4 exemples) rend les résultats obtenus difficilement exploitables en pratique, comme nous l'avons déjà signalé.
- ✓ Les performances de notre algorithme génétique sont encore meilleures a priori que celles de OCMPEST, car nous retrouvons la solution optimale 2-cyclique pour presque toutes les instances de Mateo traitées entre 5 et 10 cuves. Evidemment, si on compare avec les solutions $r$-cycliques de OCMPEST, cela n'est pas toujours le cas (comparaison des temps de cycle divisés par le degré $r$).
- ✓ Certes, nous n'avons pas forcément de solutions optimales de référence pour toutes les instances de taille importante. Cependant, nous pensons pouvoir déduire de nos résultats sur les autres benchmarks que notre algorithme génétique peut nous permettre d'atteindre ou en tout cas d'approcher de l'optimum, aussi pour des problèmes de grande taille. Evidemment, cela peut demander un plus grand nombre d'itérations et donc un temps d'exécution plus important. Pour pallier ce problème, une amélioration de notre algorithme génétique serait nécessaire, par exemple en travaillant sur les opérateurs mis en œuvre.
- ✓ Pour les instances de taille importante présentées dans le tableau III.5 (20 à 50 cuves), une comparaison établie uniquement sur la base du calcul T/r favorise l'heuristique OCMPEST. Toutefois, la maîtrise du degré du cycle (r=2) est ici un avantage certain de l'AG en termes de mise en œuvre de l'ordonnancement trouvé.
- ✓ Les temps de simulation avec OCMPEST ne dépassent pas les 5 secondes pour l'ensemble des benchmarks. Ils sont de l'ordre de 3 minutes au maximum avec notre algorithme génétique.

Les figures III.14 et III.15 présentent les résultats des simulations effectuées sur les mêmes 180 exemples (entre 5 et 10 cuves) associés à diverses fourchettes de fenêtres de temps et de vitesses de chariot. Elles donnent l'écart moyen (%) pour les heuristiques EST et OCMPEST par rapport à la durée moyenne d'un 2-cycle optimal, en fonction du nombre de cuves de traitement. Alors que les résultats de l'heuristique EST présentent un grand écart par rapport à la solution optimale atteignant 97,61% pour le cas des lignes à 7 cuves et en moyenne 86%, notre heuristique OCMPEST n'est qu'à 7,61% en moyenne du temps de cycle moyen optimal. Au vu des résultats de l'heuristique OCMPEST, on remarque une nette amélioration du temps de cycle moyen par rapport à la première heuristique atteignant 39,41% en moyenne (voir figure III.16).

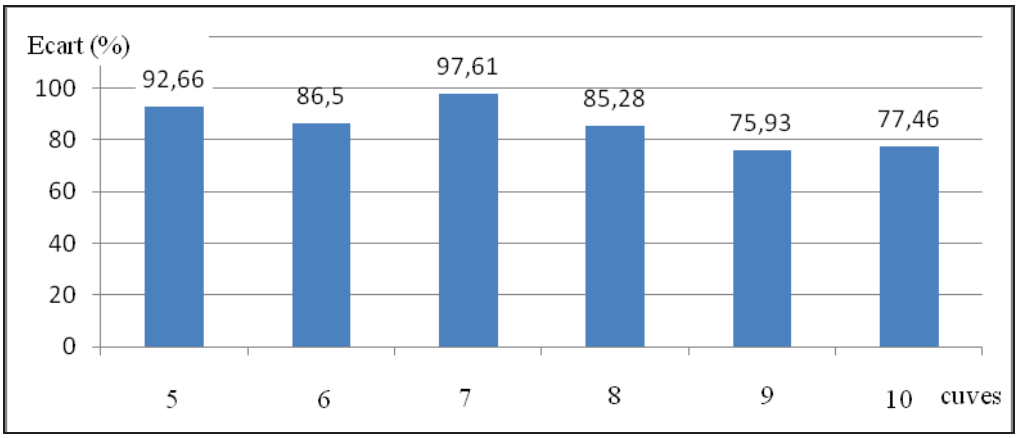



**Figure III.14.** Ecart Moyen (%) de la solution de l'heuristique EST par rapport à la durée moyenne optimale d'un 2-cycle

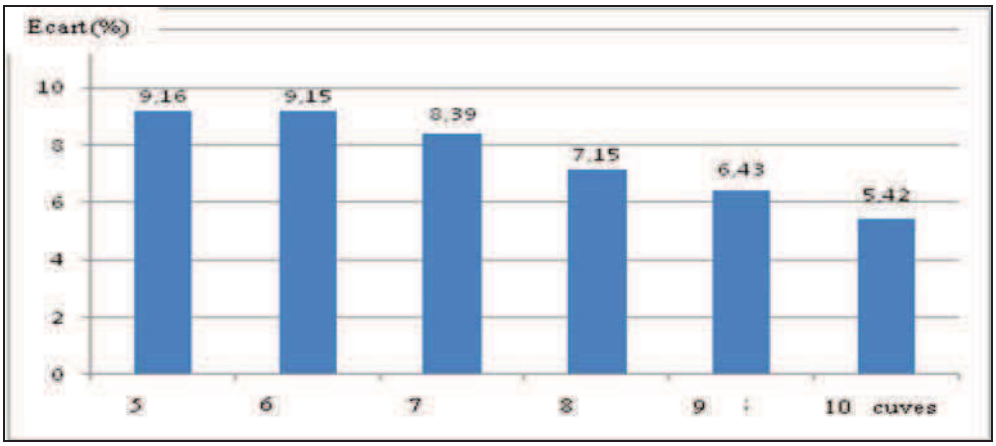



**Figure III.15.** Ecart Moyen (%) de la solution de l'heuristique OCMPEST par rapport à la durée moyenne optimale d'un 2-cycle

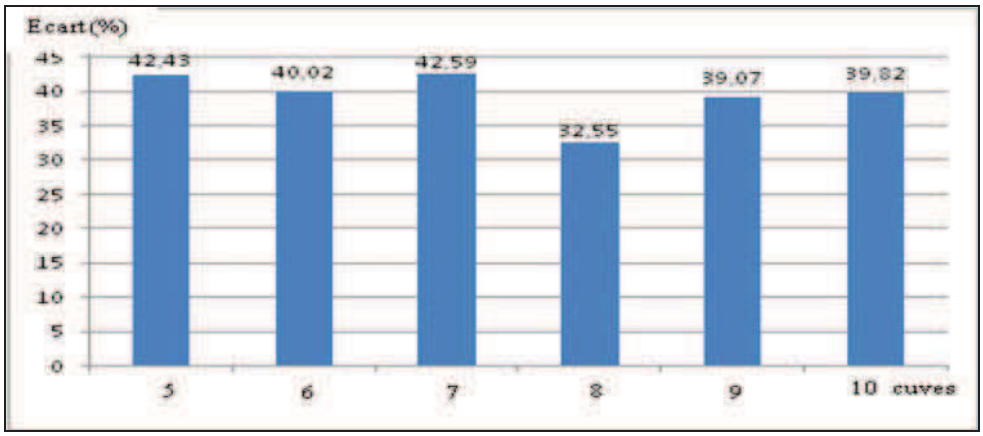



**Figure III.16.** Ecart Moyen (%) entre les solutions des heuristiques EST et OCMPEST

## III. 9 - Influence du panachage sur la durée du cycle

Durant le processus d'ordonnancement, une des questions qui se posent est : quel est le l'ordre d'introduction sur la ligne des types de produit qui minimise le temps de cycle total. Notre problème d'ordonnancement est ainsi décomposé en deux problèmes distincts de séquencement. Le premier concerne l'ordre d'exécution des opérations de transport par le robot, alors que le second s'intéresse à l'ordre d'entrée des produits sur la ligne. Ainsi et comme le planning des types de produits à traiter est connu à l'avance, il est judicieux de choisir un ordre de passage des gammes qui minimise la durée du cycle considéré. Malgré que ce type d'études permette d'obtenir des gains de productivité sensibles [Varnier, 1996], peu de travaux se sont intéressés à l'étude du problème de panachage en entrée de ligne.

### III. 9. 1 - Description de la démarche de résolution

La résolution du premier problème considéré est garantie par la mise en œuvre de l'une des méthodes que nous avons développées (programme linéaire, heuristique OCMPEST ou AG), alors que le second est assuré au moyen d'une nouvelle procédure intégrée en amont et dont le principe est donné par le schéma synoptique de la figure III.17. Notre démarche consiste en la recherche de la meilleure solution cyclique multi-produits hétérogène. Pour cela, on suppose qu'il existe $p$ différents produits et on se propose de définir différentes configurations d'insertion des produits sur la ligne de traitement $C_{[1]-[2]-\ldots-[p]}$ où [i] désigne le type de produit à insérer au $i^{ème}$ rang. Tenant compte du ratio de chaque type de produit à fabriquer, une liste des différentes configurations est générée. A titre d'exemple, si on souhaite produire un motif réduit d'un produit A, deux produits B et un produit C, le nombre de configurations possibles

est à première vue égale à 6 (3 ! et non 4 !) compte tenu de l'aspect cyclique (par exemple, $C_{A\text{-}B\text{-}A\text{-}C}$ est équivalente aux configurations $C_{A\text{-}C\text{-}A\text{-}B}$, $C_{B\text{-}A\text{-}C\text{-}A}$ et $C_{C\text{-}A\text{-}B\text{-}A}$). Ce nombre est réduit à 3 vu l'existence de deux types de produits A dans le motif. Il reste au final 3 configurations distinctes : $C_{A\text{-}A\text{-}B\text{-}C}$, $C_{A\text{-}A\text{-}C\text{-}B}$ et $C_{A\text{-}B\text{-}A\text{-}C}$.

Les différentes configurations générées définissent les listes cycliques d'insertion des produits qui constitueront chacune une donnée d'entrée pour l'algorithme de résolution choisi (PLM, OCMPEST ou AG). Celui-ci permet par la suite de considérer l'ensemble des contraintes propres à chaque configuration et de déterminer la solution cyclique qui correspond à chacune d'entre elles. La meilleure configuration est celle qui fournit l'ordonnancement ayant le temps de cycle moyen minimal.

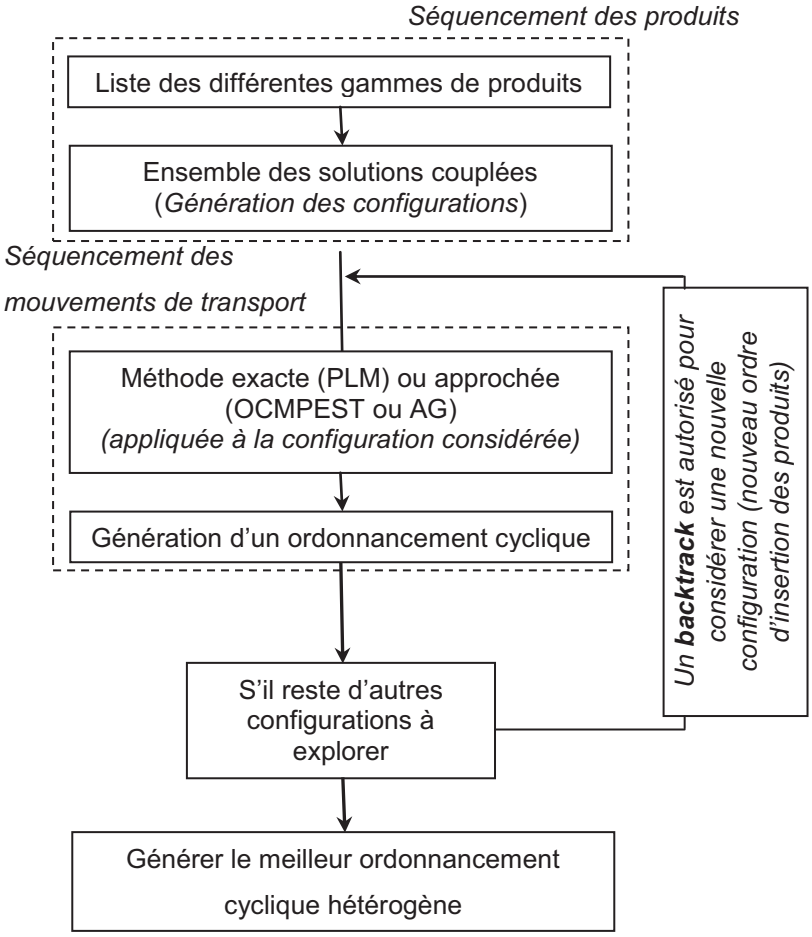



**Figure III.17.** Schéma synoptique du principe de fonctionnement de l'algorithme

### III. 9. 2 - Résultats des simulations

On considère le cas de trois types de produits {A, B, C} à produire en quantités égales. Toutes les configurations admissibles sont équivalentes aux deux configurations suivantes : $C_{A\text{-}B\text{-}C}$ et $C_{A\text{-}C\text{-}B}$. Ces instances correspondent à des lignes à 5, 6, 7, 8, 9 et 10 cuves tirées des benchmarks de Mateo (2001) (voir données en annexe 14) [El Amraoui et al, 2011][a].

Pour illustrer notre propos et résoudre cette instance, nous avons choisi d'utiliser un de nos algorithmes approchés, en l'occurrence OCMPEST. Le choix d'une méthode approchée à ce stade nous semble plus judicieux compte tenu de la complexité du problème associé à chaque configuration. Par ailleurs, il s'agit ici de comparer les performances des différentes configurations, et non d'évaluer l'algorithme utilisé.

Le tableau III.6 montre clairement l'influence du choix de l'ordre de passage des produits sur les performances de l'ordonnancement, en termes de gain de productivité (la ligne réduction (%) indique l'écart relatif par rapport à la moins bonne des 2 solutions).

**Tableau III.6.** Comparaison entre les deux configurations $C_{A\text{-}B\text{-}C}$ et $C_{A\text{-}C\text{-}B}$

| configuration | nombre de cuves | | | | | |
|---|---|---|---|---|---|---|
| $C_{A,B,C}$ - $C_{A,C,B}$ | 5 | 6 | 7 | 8 | 9 | 10 |
| degré cycle r | 3 | 3 | 12 | 12 | 6 | 3 |
| $TC_{A,B,C}$ | 616 | 587 | 2924 | 2735 | 2509 | 1117 |
| $TC_{A,C,B}$ | 603 | 563 | 2958 | 2584 | 2516 | 1107 |
| réduction (%) | 2.1 | 4.1 | 1.1 | 5.5 | 0.3 | 0.9 |

## III. 10 - Bilan

Dans ce chapitre, nous avons développé deux méthodes approchées pour résoudre le CHSP multi-produits, en supposant que l'ordre d'entrée des produits sur la ligne est connu à l'avance et suit une séquence cyclique prédéterminée (P1-P2-P3 - P1-P2-P3 - P1-P2-P3 - …). La première méthode est un algorithme de liste que nous avons élaboré en nous basant sur les travaux de recherches de Song (1995). Cette nouvelle heuristique tire son originalité du fait que le degré du cycle n'est pas fixé au préalable, ce qui permet d'obtenir des temps de cycle très intéressants. Elle a prouvé sa performance, tout d'abord par rapport à l'approche proposée précédemment [Song et al. 1995] et ensuite, par rapport au temps de cycle moyen optimal pour le cas de fonctionnement 2-cyclique optimal.

La deuxième méthode est un Algorithme Génétique dont la modélisation est basée sur un codage direct et la fonction *fitness* ou d'évaluation de la solution cyclique est réduite à la recherche du chemin le plus long d'un graphe orienté. Par rapport à la méthode précédente, le degré de cycle est ici imposé (r=2), mais à degré équivalent, l'AG donne de meilleurs résultats en moyenne pour des instances de taille réduite (tableau III.7, pour les instances de [Mateo, 2001]), avec un temps de simulation néanmoins plus important. Nos deux méthodes approchées développées ont donc révélé leur efficacité pour la résolution des problèmes *r*-cycliques.

**Tableau III.7.** Comparaison entre les différentes méthodes proposées [El Amraoui et al, 2011][c]

| Programmation Linéaire Mixte | | Heuristique | | Algorithmes Génétiques | |
|---|---|---|---|---|---|
| CPU (s) | $Ecart_{moyen}$ (%) | CPU (s) | $Ecart_{moyen}$ (%) | CPU (s) | $Ecart_{moyen}$ (%) |
| de 134 à 778 | 0 (optimum) | de 2 à 75 | de 5,42 à 9,15 | de 129 à 341 | 0 (optimum) |

# Conclusion générale

Le HSP est un problème très spécifique par rapport aux autres problèmes d'ordonnancement classiques d'ateliers (job shop, flow shop, ..). Cette spécificité est traduite par un ensemble de contraintes strictes qui entrainent une croissance exponentielle de sa complexité.

De nombreux problèmes de type HSP demeurent encore irrésolus et peu de chercheurs semblent s'être intéressés au cas des lignes multi-produits et/ou multi-robots. C'est dans ce contexte, que nous avons proposé dans ce mémoire des approches de résolution dédiées aux problèmes cycliques multi-produits.

Dans le chapitre II nous avons présenté une modélisation linéaire du problème 2-cyclique multi-produits pour des lignes simples. Ce modèle est caractérisé par son caractère évolutif. Une généralisation du modèle 2-cyclique multi-produits a donc été enfin élaborée pour la recherche de solutions optimales r-cycliques multi-produits ($r \geq 2$).
Notre modèle multi-produits a prouvé sa fiabilité en terme de temps de cycle. En effet, en comparant les résultats des simulations présentées dans ce chapitre pour un ordonnancement 2-cyclique hétérogène avec ceux des ordonnancements cycliques mono-produit, nous somme parvenus à réduire la durée totale du cycle pour la plupart des exemples traités. De même, notre modèle r-cyclique (pour $r>2$) permet d'améliorer les résultats des modèles 2-cycliques de la littérature de base, ce qui est normal puisque toute solution r-périodique (r étant un multiple de 2) est au pire égale à la duplication du 2-cycle optimal. En plus de notre contribution relative à la généralisation aux r-cycles, nous avons également étendu notre modèle pour traiter le cas des lignes plus complexes (multi-bacs, multi-fonctions) et diverses configurations des postes de chargement et de déchargement.
Toutefois, cette approche reste limitée aux problèmes de petites tailles, ce qui nous a amené à proposer des méthodes approchées alternatives permettant de résoudre des instances plus réalistes, même sans en fournir la solution optimale.

C'est pourquoi, dans le chapitre III, nous avons décrit deux méthodes approchées pour la résolution du problème r-cyclique multi-produits afin de remédier à ces problèmes de

résolution avec des méthodes exactes pour des lignes de grandes tailles ou pour des degrés supérieur à deux. La première méthode proposée est une heuristique, basée sur un algorithme de liste. La seconde méthode est une méthode de type métaheuristique, qui consiste à utiliser un algorithme génétique pour la modélisation et la résolution du problème généralisé 2-cyclique.

La première approche nous a permis de trouver cycles de degré supérieur à deux pour deux différents types de produits, qui parfois conduisent à une amélioration notable du temps de cycle moyen. La deuxième approche, quoi que non exacte, nous conduit à retrouver les solutions optimales 2-cycliques pour des instances de taille réduite.

Bien que ces deux méthodes soient plus intéressantes que la méthode exacte pour des problèmes de grande taille, il n'en reste pas moins qu'elles trouvent leurs limites au niveau de la taille maximale des instances (en termes de nombre de cuves ou de degré de cycle).

Cette analyse nous conduit donc à envisager en perspective directe de nos travaux l'optimisation de nos algorithmes approchés. Par exemple, pour notre algorithme de liste, il serait utile d'optimiser l'exploration de l'arbre de recherche (en coupant plus de branches, ou plus tôt), ce qui pourrait limiter les effets néfastes des nombreux backtracks. En ce qui concerne l'algorithme génétique, la mise en œuvre d'opérateurs dédiés pourrait améliorer le temps de résolution en permettant de générer moins de solutions non faisables, et donc en limitant le temps passé en réparations. Toutefois, la conception de tels opérateurs suppose une analyse pointue de la structure des solutions. D'autre part, les procédures de réparation pourraient elles aussi être optimisées : au niveau de COR2, et moyennant l'identification de la source de l'infaisabilité d'un chromosome, une réparation plus ciblée pourraient être mise en œuvre par mutation directe du gène « défectueux ».

Bien entendu, les performances de notre méthode exacte sont à améliorer, par exemple en optimisant la procédure de parcours de l'espace de recherche, ce qui pourrait conduire à une amélioration notable des capacités de résolution de ce modèle.

Dans la continuité de nos travaux, nous envisageons également d'adopter des méthodes de recherche hybrides combinant par exemple un algorithme génétique avec une méthode exacte, pour traiter des cas plus complexes tels que les lignes multi-robots.

En termes de perspectives à plus long terme, des extensions de ces approches sont envisageables pour la résolution de problèmes connexes au HSP, tels que les problèmes

d'ordonnancement des ateliers flexibles et des cellules robotisées qui comportent également des ressources de transport.

Enfin, pour des raisons environnementales, des réglementations strictes ont été mises en place afin d'inciter les entreprises à mettre en place des dispositifs pour le traitement des rejets en eaux usées dans leurs ateliers de traitement de surface. Il serait ainsi intéressant d'élaborer une modélisation multi-objectifs du problème qui, à la fois maximise la productivité et minimise l'impact des effluents émis sur l'environnement.

# Bibliographie


[Armstrong et al, 1996] Armstrong, R., Gu, S. et Lei, L. (1996). A greedy algorithm to determine the number of transporters in a cyclic electroplating process. *Institute of Industrial Engineers Transactions, 28(5), pp.347-355*.

[Armstrong et al, 1994] Armstrong, R., Lei, L. et Gu, S. (1994). A Bounding Scheme for Deriving the Minimal Cycle Time of a Single-Transporter N-Stage Process with Time-Window Constraints. *European Journal of Operational Research, 75, pp.1–11*.

[Armstrong et al, 1992] Armstrong, R., Lei, L. et Shanhong, G. (1992). A Bounding Scheme for Deriving the Minimal Cycle Time of a Single-Transporter N-Stage Process with Time-Window Constraints. *GSM Working paper #92-07, Rutgers University*.

[Bäck, 1996] Bäck, T. (1996). *Evolutionary Algorithms in Theory and Practice*. Oxford University Press, New-York.

[Backer, 1974] Backer, K. (1974). *Introduction to sequencing and scheduling*. John Wiley & Sons, New York.

[Baptiste et al, 1994] Baptiste, P., Legeard, B., Manier, M.-A. et Varnier, C. (1994). A scheduling problem optimization solved with Constraint Logic Programming. *The 2cd International Conference on the Practical Application of Prolog (ICPAP), London.*

[Baptiste et al, 1995] Baptiste, P., Grunder, O. et Chappe, D. (1995). A branch and bound procedure for the optimal lay-out of a satured production line fed with a single hoist. *International Conference on industrial Ingineering and Production Management (IEPM), Marrakech*.

[Baptiste et al, 1996] Baptiste, P., Legeard, B., Manier, M.-A. et Varnier, C., Résolution d'un problème d'ordonnancement avec la PLC. *European Journal of Automated Systems, 30(2-3), pp.201-230*.

[Barnier et Brisset, 1999] Barnier, N. et Brisset, P. (1999). Optimisation par algorithme génétique sous contraintes. *Technique et science informatique, 18, pp.1-29*.

[Billaut et al, 2005] Billaut, J.C., Moukrim, A. et Sanlaville, E. (2005). *Flexibilité et robustesse en ordonnancement*. Edition Lavoisier, Paris.

[Blazewicz et al, 1996] Blazewicz, J., Ecker, K., Pesch, E., Schmidt, G., et Weglarz, J. (1996). *Scheduling* Computer and Manufacturing Processes. Springer, Berlin.

[Bloch et al, 2008] Bloch, C., Manier, M-A, Baptiste, P. et Varnier, C. (2008). *Production Scheduling*, chapter 8 Hoist scheduling Problem, pp. 193-231, Edited by Pierre Lopez and François Roubellat, ISTE, Wiley, London , april 2008, ISBN: 9781848210172.

[Bloch et al, 1999] Bloch, C., Varnier, C. et Baptiste, P. (1999). A taboo search combined with a modified shifting bottleneck heuristic for solving a blocking scheduling problem with bounded processing times. *15th Conference of the International Federation of Operational Research Societies, Beijing, pp.125-126*.

[Bloch, 1999] Bloch-Rossé, C. (1999). *Contribution à l'ordonnancement dynamique de lignes de traitement de surface*. Thèse de doctorat de l'université de Franche-Comté en Automatique et Informatique.

[Boch et al, 1997] Bloch, C., Varnier, C. et Baptiste, P. (1997). Algorithmes stochastiques et codage pour le Hoist Scheduling Problem multi-produits réactif. *Colloque de Recherche Doctorale AGIS'97 "Automatique, Génie Informatique , Image, Signal", pp.101-105*.

[Bloch et al, 1996][a] Bloch, C., Bachelu, A. et Varnier, C. (1996). Présentation d'une typologie des problèmes de type HSP. *Réunions du Groupe de travail "Bermudes", Tours*.

[Bloch et al, 1996][b] Bloch, C., Varnier, C. et Baptiste, P. (1996). Applying Stochastic Methods to Real Time Hoist Scheduling Problem. *The Computational Engineering in Systems Application Multiconference (CESA'96)-Symposium on Discrete Events and Manufacturing Systems ,Lille, pp.479-484*.

[Boukef, 2009] Boukef, H. (2009). *Sur l'ordonnancement d'ateliers job-shop flexibles et flow-shop en industrie pharmaceutiques : optimisation par algorithmes génétiques et essaims particulaires*. Thèse de Doctorat, Ecole Centrale de Lille, Ecole Nationale d'Ingénieurs de Tunis.

[Bourazza, 2006] Bourazza, S. (2006). *Variantes d'algorithmes génétiques appliquées aux problèmes d'ordonnancement*. Thèse de Doctorat, Université du Havre.

[Bracker et Chapman, 1985] Bracker, W.E. et Chapman, W. (1985). Optimization of a programmable Hoist Using Linear Programming. *Printed Circuit Fabrication, 8, pp.39-42*.

[Brucker et al, 2002] Brucker, P., Dhaenens, C., Knust, S., Kravchenko, S.A., Werner, F. (2002). Complexity results for parallel machine problems with a single server. *Journal of scheduling, pp.429-457*.

[Brucker, 1998] Brucker, P. (1998). *Scheduling Algorithms*. Springer-Verlag, Berlin Heidelberg.

[Calvez et al., 1997] Calvez, S., Aygalinc, P. et Khansa, W. (1997). P-time Petri nets for manufacturing systems with setting time constraints. *The IFAC International Conference on Control of Industrial Systems (IFAC-IFIP-IMACS CIS'97), Belfort, 3, pp.495-500*.

[Carlier et Chretienne, 1988] Carlier, J. et Chretienne, P. (1988). *Problème d'ordonnancement : Modélisation, Complexité, Algorithmes*. Edition Masson, Paris.

[Caux et Pierreval, 1997] Caux, C. et Pierreval, H. (1997). Solving a hoist scheduling problem as a sequencing problem. *Conference on Management and Control of Production and Logistics, Campinas, Brazil, pp.372-376.*

[Caux et al, 1995][a] Caux, C., Fleury, G., Gourgand, M. et Kellert, P. (1995). Couplage méthodes d'ordonnancement – simulation pour l'ordonnancement de systèmes industriels de traitement de surface. *RAIRO-Recherche Opérationnelle, 29(4), pp.391, 1995*.

[Caux et al, 1995][b] Caux, C., Pierreval, H. et Portmann, M.C. (1995). Les algorithmes génétiques et leur application aux problèmes d'ordonnancement. *Revue APII, 29(4-5), pp.409-443*.

[Caux, 1993] Caux, C. (1993). *Analyse et spécification de systèmes de production pour l'évaluation des performances et la recherche d'ordonnancement*. Thèse de doctorat de l'Université Blaise Pascal, Ecole doctorale sciences pour l'ingénieur de Clermont-Ferrand, spécialité Informatique.

[Caux et al, 1992] Caux, C., Fleury, G., Gourgand, M. et Kellert, P. (1992). Couplage simulation à évènements discrets – méthodes stochastiques pour l'ordonnancement d'un atelier de traitement de surface. *Congrès European Simulation Multiconference (ESM'92), York*.

[Che et al, 2009] Che, A., Yan, P., Yang, P.N. et Chu, C. (2009). Optimal cyclic scheduling of a hoist and multi-parts with fixed processing times. *International Journal of Production Research, 48(2), pp.1366-588X*.

[Che et Chu, 2005] Che, A. et Chu, C. A polynomial algorithm for no-wait cyclic hoist scheduling in an extended electroplating line. *Operations research letters, 33(3), pp.274-284*.

[Chen et al, 1998] Chen, H., Chu, C. et Proth, J.M. (1998). Cyclic Scheduling of a hoist with time window constraints. *IEEE Transactions on Robotics and Automation, 14(1), pp.144-152*.

[Chen et Su, 1994] Chen, F.F. et SU, Q. (1994). Sequencing Gantry Robots in Tightly-Coupled Automated Serial Production Systems. *The 5th World Conference on Robotics Research, Cambridge (Massachussetts)*.

[Chen et al, 1994] Chen, H., Chu, C. et Proth, J.M. (1994). Cyclic Scheduling of a Hoist with Time Window Constraints. *Rapport interne n°2307, INRIA*.

[Cheng et Smith, 1995] Cheng, C.-C. et Smith, S. F. (1995). A constraint-posting framework for scheduling under complex constraints. *Symposium on Emerging Technologies and Factory Automation, Paris*, *pp. 269-280*.

[Chrétienne, 2006] Chrétienne, P. (2006). On Single Machine Scheduling with no Intermediate Delays. *In Graphs and Optimization ,Leukerbaad.*

[Chrétienne et Sourad, 2007] Chrétienne, P. et Sourd, F. (2007). Selecting and scheduling tasks with agreeable time windows and setup costs. *The 3rd Multidisciplinary International Conference on Scheduling: Theory and Applications, pp. 561-563*.

[Collette et Siarry, 2002] Collette, Y. et Siarry, P. (2002). *Optimisation multiobjectif. Edition Eyrolles*. ISBN : 2212111681.

[Conway et al, 1967] Conway, R., Maxwell, W., et Miller, L. (1967). *Theory of scheduling*. Addison Wesley, Reading, Massachussets.

[Crama et al, 2000] Crama, Y., Kats, V., Van de Klundert, J..et Levner, E. (2000). Cyclic scheduling in robotic flowshops, Annals of Operations Research: Mathematics of Industrial Systems, 96, pp.97-124.

[Daganzo, 1989] C. F. Daganzo (1989). The crane scheduling. *Transportation Research Part B: Methodological, 23(3), pp.159-175*.

[Davis et al, 1993] Davis, L., Orvosh, D., Cox, A. et Qiu, Y. (1993). *A Genetic Algorithm for Survivable Network Design. ICGA, pp.408-415*.

[Davis, 1991] Davis, L. (1991). *Handbook of genetic algorithms*. Van Nostrand Reinhold Computer Library.

[Deb et al, 2006] Deb, K., Sinha, A. et Kukkonen, S. (2006). Multi-objective test problems, linkages, and evolutionary methodologies. *Genetic and Evolutionary Computation Conference, GECCO 2006, Proceedings Seattle, Washington, pp.8-12*.

[Dridi, 2004] Dridi, M. (2004). *Contribution à la résolution des problèmes de régulation dans les systèmes de transport dans un contexte multicritère par approche évolutionniste*. Thèse de doctorat de l'Université des Sciences et Technologies de Lille et l'Ecole Centrale de Lille en Automatique et Informatique.

[Dupas, 2004] Dupas, R. (2004). *Amélioration de performance des systèmes de production : apport des algorithmes évolutionnistes aux problèmes d'ordonnancement cycliques et flexibles*. Habilitation à Diriger des Recherches, Université d'Artois.

[El Amraoui et al, 2012] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2012). *Resolution of the Multi-Part Cyclic Hoist Scheduling Problem with bounded Processing Times in Complex Lines' Configuration. European Journal of Industrial Engineering, 6(4). Accepté et sera publié en 2012*.

[El Amraoui et al, 2011][a] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2011). Heuristics for the Resolution of the Cyclic Hoist Scheduling Problem with Multi-Items. *18th IFAC World Congress, Milano*.

[El Amraoui et al, 2011][b] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2011). Hoist Scheduling for Multi-Parts CHSP in Complex Lines' Configuration. *43th Southeastem Symposium on System Theory, IEEE ICIT-SSST, Auburn, Alabama*.

[El Amraoui et al, 2011][c] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2011). Ordonnancement Cyclique Multi-Produits des Lignes de Traitement de Surface. *12ème congrès annuel de la société française de recherché opérationnelle et d'aide à la décision Roadef 2011, Saint-Etienne*.

[El Amraoui et al, 2010][a] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2010). Exact method for solving the r-cyclic hoist scheduling problem with heterogeneous parts. *Third International Conference on Information Systems, Logistics and Supply Chain ILS'2010, Casablanca*.

[El Amraoui et al, 2010][b] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2010). Heuristique pour l'ordonnancement cyclique multi-produits des lignes de traitement de surface. *Sixième Conférence Internationale Francophone d'Automatique CIFA 2010, Nancy*.

[El Amraoui et al, 2010][c] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2010). Genetic Algorithm for a Cyclic Hoist Scheduling Problem with Time-Window Constraints and Heterogeneous Part Jobs. *18th IEEE conference: Mediterranean Conference on Control and Automation MED'10, Marrakech*.

[El Amraoui et al, 2009][a] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2009). On the Maximization of the Throughput for a Single Hoist Cyclic Scheduling. *IEEE Conference on Systems, Signals and Devices SSD'09, Djerba*.

[El Amraoui et al, 2009][b] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2009). Robustness Integration in a Transport Scheduling Problem. *International Revue of Automatic Control (IREACO), 4(2), pp.476-480*.

[El Amraoui et al, 2008][a] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2008). A Comprehensive Linear Program to Solve Cyclic Hoist Scheduling Problem. *The 9th International Conference on Sciences and Techniques of Automatic control and computer engineering STA'2008, STA-SSI-487, Sousse*.

[El Amraoui et al, 2008][b] El Amraoui, A., Manier, M.-A., El Moudni, A. et Benrejeb, M. (2008). A Mixed Linear Program for a Multi-Part Cyclic Hoist Scheduling Problem. *International Journal of Sciences and Techniques of Automatic control and computer engineering IJ-STA, (Special Issue, CEM), pp.612-623*.

[Falkenauer et Bouffouix, 1991] Falkenauer, E. et Bouffouix, S. (1991). A genetic algorithm for job shop. *IEEE International Conference on Robotics and Automation, Sacramento, California*.

[Fargier et Lamothe, 2001] Fargier, H. et Lamothe, J. (2001). Handling soft constraints in hoist scheduling problems: the fuzzy approach. *Engineering Applications of Artificial Intelligence, 14, pp. 387-399*.

[Fleury et al, 2001] Fleury, G., Gourgand, M. et Lacomme, P. (2001). Metaheuristics for the Stochastic Hoist Scheduling Problem (SHSP). *International Journal of Production Research, 39(15), pp.3419-3457*.

[Fleury, 1995] Fleury, G. (1995). Applications de méthodes stochastiques inspirées du recuit simulé à des problèmes d'ordonnancement. *Automatique productique informatique industrielle, 29(4-5), pp.445-470*.

[Fleury, 1993] Fleury G. (1993). *Méthodes stochastiques et déterministes pour les problèmes NP-difficiles*. Thèse de doctorat de l'université Blaise Pascal, Ecole doctorale sciences pour l'ingénieur de Clermond-Ferrand, en Informatique.

[Gargouri, 2003] Gargouri, E. (2003). *Ordonnancement coopératif en industries agroalimentaires*. Thèse de Doctorat, Université des Sciences et Technologies de Lille 1.

[Gargouri et al, 1999] Gargouri, E., Hammadi, S., et Borne, P. (1999). Study of scheduling problems in agro-food chain. *Int. Journal of Math. and Computers in Simulation, Intelligent Forecasting, Fault Diagnosis, Scheduling and Control, (60), pp.277-291*.

[Garey et Johnson, 1979] Garey, R. et Johnson, D. S. (1979). *Computers and Intractability: A guide to the theory of NP-Competeness*. Edition Freeman and Co.

[Gentina et al, 2001] Gentina, J.C., Korbaa, O. et Camus, H., Problèmes d'ordonnancement cyclique , Chapitre 7 dans Ordonnancement de la Production, eds Lopez P. et Roubellat F., Hermès Sciences, IC2 Productique, pp.197-223, 2001.

[Ghedira, 2006] Ghedira, K. (2006). *Logistique de la production*. Edition Technip, Paris.

[Ge et Yih, 1995] Ge, Y. et Yih, Y. (1995). Crane scheduling with time windows in circuit board production Lines. *International Journal of Production Research, 33(5), pp.1187-1189*.

[Gen et Cheng, 1997] Gen M., Cheng R., Genetic algorithms & Engineering Design and Automation, Wiley-Interscience Publication, John Wiley &Sons, INC., 1997.

[Ghedira, 2007] Ghedira, K. (2007). *Optimisation combinatoire par métaheuristiques*. Edition Technip, Paris.

[Goldberg, 1994] Goldberg, D.E. (1994). *Algorithmes génétiques : exploitation, optimisation et apprentissage automatique*. Editions Addison-Wesley, France.

[Golberg, 1989][a] Golberg, D. (1989). *Genetic Algorithm*. Addison Wesley.

[Golberg, 1989][b] Golberg, D. (1989). *Genetic Algorithm in Search, Optimization, and Machine Learning*. Reading Mass, Addison-Wesley.

[Chrétien-Goni, 2002] Chrétien-Goni, J.P. (2002). *Encyclopédie Universélle, 11, pp.291,* Paris, Corpus.

[Gotha, 1993] Gotha, G. (1993). Les problems d'ordonnancement. *RAIRO Recherche Opérationnelle, 27(1), pp.77-150*.

[Graham et al, 1979] Graham, R.L., Lawler, E.L., Lenstra, J.K. et Rinnooy Kan A.H.G. (1979). Optimization and approximation in deterministic sequencing and scheduling theory : a survey. *Annals of discrete Mathematics, 5, pp.287-326*.

[Grunder et al, 1997] Grunder, O., Baptiste, P. et Chappe, D. (1997). The relationship between the physical layout of the work stations and the productivity of a saturated single-hoist production line. *Int. J. Prod. Res. 35(8), pp.2189–2211*.

[Gupta et Sivakumar, 2006] Gupta, A. K. et Sivakumar, A. I. (2006). Job shop scheduling techniques in semi conductor manufacturing. *International Journal of Advanced Manufacturing Technologies, 27, pp.1163-1169.*

[Hanen, 1994] Hanen, C. (1994). Periodic scheduling of several hoists. *4th international workshop on project management and scheduling, Leuven, pp.108-110*.

[Hanen et Munier, 1993] Hanen, C. et Munier, A. (1993). Ordonnancement cyclique d'un robot sur une ligne de galvanoplastie : modèles et algorithmes. *Rapport de recherche LITP 93.30, université Pierre et Marie Curie, Paris*.

[Haouari et Daouars, 1999] Haouari, M. et Daouars, T. (1999). Optimal Scheduling of the 3 machine assembly-type Flow Shop. *RAIRO Operations Research, 33, pp.439-445*.

[Henrik et al, 2005] Henrik, J.P., Christian, B. et Herbert, K. (2005). A heuristic scheduling procedure for multi-item hoist production lines. *International journal of production economics, 105(1), pp.54-69*.

[Hindi et Fleszar, 2004] Hindi, K. S. et Fleszar, K. (2004). A constraint propagation heuristic for the single hoist, multiple products scheduling problem. *Computers and Industrial Engineering, 47, pp.91-101*.

[Holland, 1975] Holland, J.H. (1975). *Adaptation in Natural and Artificial Systems*. The University of Michigan.

[Hosein et Ross, 1975] Hosein, R. W., Ross, S. M. (1975). *Job shop scheduling in an electroplating environment. The Production Engineer, pp.303-307*.

[Jeunehomme, 1999] Jeunehomme, N. (1999). *Contribution à l'ordonnancement cyclique multi-produits dans le cadre des lignes de traitement de surface*. Rapport de diplôme d'études approfondies en informatique, automatique et productique, université de Besançon.

[Jégou et al, 2006] Jégou, D., Kim, D.-W., Baptiste, P. et Lee Kwang, H. (2006). A contract net based intelligent agent system for solving the reactive hoist scheduling problems. *Expert Systems with Applications, 30, pp.156-167*.

[Ji et Wan, 2001] Ji, P. et Wan, Y. F. (2001). Planning for printed circuit board assembly : the state-of-the-art review. *International Journal of Computer Applications in Technology, 14(4/5/6), pp.136-144*.

[Jiang et Liu, 2007] Jiang, Y. et Liu, J. (2007). Multihoist cyclic scheduling with fixed processing and transfer times. *IEEE Transactions on Automation science and engineering, 4(3), pp.435-450*.

[Kacem, 2003] Kacem, I. (2003). *Ordonnancement multicritère des job-shops flexibles : formulation, bornes inférieures et approche évolutionniste coopérative*. Thèse de Doctorat, Université de Lille 1.

[Kacem et al, 2001] Kacem, I., Hammadi, S. et Borne, P. (2001). Approach by localization and genetic manipulation algorithm for flexible job-shop scheduling problem. *IEEE International Conference on Systems, Man and Cybenetics, Tucson, AZ., pp.2599-2604*.

[Kats et al, 2006] Kats, V., Lei, L. et Levner, E. (2006). Minimizing the cycle time of multiple-product processing networks with a fixed operation sequence, setups, and time-window constraints'. *European Journal of Operational Rrsearch, 187(3), pp.1196-1211*.

[Kats et Levner, 1997] Kats, V. et Levner, E. (1997). A stongly polynominal for no-wait cyclic robotic floxshop scheduling. *Operations Research Letters, 21, pp.171-179*.

[Khamphang, 1998] Khamphang, C. (1998). *Algorithmes heuristiques et évolutionnistes : Application à la résolution du problème de placement de formes irrégulières*. Thèse de doctorat à l'université des sciences et technologies de Lille.

[Kuntay et al, 2006] Kuntay, I., Xu, Q., Uygun, K. et Huang, Y. L. (2006). Environmentally Conscious Hoist Scheduling for Electroplating Facilities. *Chemical Engineering Communications, 193, pp.273-292*.

[Lacomme, 1998] Lacomme, P. (1998). *Optimisation des systèmes de production: méthodes stochastiques et approche multi-agents*. Thèse de doctorat de l'université de Clermont-Ferrand.

[Lamothe et Dlemas, 1997] Lamothe, J. et Dlemas, J. (1997). Ordonnancement dynamique d'un atelier de traitement de surface avec cuves de capacité multiple. *2ème Congrès International Franco-québécois de Génie Industriel, Albi*.

[Lamothe et al, 1996] Lamothe, J., Thierry, C. et Delmas, J. (1996). A mutltihoist model for the Real Time Hoist Scheduling Problem. *The Computational Engineering in Systems Application Multiconference (CESA'96): Symposium on Discrete Events and Manufacturing Systems, Lille, pp.461-466*.

[Lamothe, 1996] Lamothe, J. (1996). *Une approche pour l'ordonnancement dynamique d'un atelier de traitement de surface*. Thèse de doctorat de l'Ecole Nationale Supérieure de l'Aéronautique et de l'Espace de Toulouse.

[Lamothe et al, 1995] Lamothe, J., Correge, M. et Delmas, J. (1995). A dynamic Heuristic for the real Time Hoist Scheduling Problem. *Symposium on Emerging Technologies and Factory Automation (INRIA/IEEE), ETFA'95, Paris, pp.161-168*.

[Lamothe et al, 1994] Lamothe, J., Correge, M. et Delmas, J. (1994). Hoist Scheduling Problem in a real Time Context. *11ème Conférence internationale sur l'analyse et l'optimisation des systèmes, Sophia Antipolis*.

[Lei et Liu, 2001] Lei, L. et Liu, Q. (2001). Optimal cyclic scheduling of a robotic processing line with two-product and time-window constraints. *INFOR, 39 (2), pp.185-199*.

[Lei et Wang, 1994] Lei, L. et Wang, T.-J. (1994). Determining the optimal cyclic hoist schedules in a single-hoist electroplating line. *IIE transactions, 2, pp.25-33*.

[Lei et al, 1993] Lei, L., Armstrong, R. et Gu, S. (1993). Minimizing the fleet size with dependent time-window and single-track constraint. *Operations Research Letters, 14, pp.91-98*.

[Lei, 1993] Lei L. (1993). Determining the optimal starting time in a cyclic schedule with a given route. *Computers Operations Research, 20(8), pp.807-816*.

[Lei et Wang, 1991] Lei, L. et Wang, T.-J. (1991). The minimum common cycle algorithm for cycle scheduling of two material handling hoists with time window constraints. *Management Science, 37(12), pp. 1629-1639*.

[Lei et Wang, 1989][a] Lei, L. et Wang, T. J. (1989). On The Optimal Cyclic Schedules of Single Hoist Electroplating Processes. *Working paper # 89-0006, Rutgers University*.

[Lei et Wang, 1989][b] Lei, L. et Wang, T. J. (1989). A Proof: The Cyclic Hoist Scheduling Problem is NP-Complete. *Working paper # 89-0016, Rutgers University, New Jersey*.

[Leung et Levner, 2006] Leung, J., M. Y. et Levner, E. (2006). An efficient algorithm for multi-hoist cyclic scheduling with fixed processing times. *Operations research letters, 34(4), pp.465-472*.

[Levner et al, 1995] Levner, E., Kats, V. et Levit, V.E. (1995). An improved Algorithm for a Cyclic Robotic Scheduling Problem. *Internatonal workshop on intelligent scheduling of robots and flexible manufacturing systems, CTEH, Holon, pp.129-141*.

[Lieberman et Turksen, 1981] Lieberman, R. W., Turksen, I.B. (1981). Crane Scheduling Problems. *AIIE Transactions, 13(4), pp.304-311*.

[Lieberman et Turksen, 1982] Lieberman, R. W., Turksen, I.B. (1982). Two-operation Crane Scheduling Problems. *IIE Transactions, 14(3), pp.147-155*.

[Lim, 1997] Lim, J. M. (1997). A genetic algorithm for a single hoist scheduling in the printed-circuit-board electroplating line. *Computer and Industrial Engineering, 33, pp.789-792*.

[Liu et al, 2002] Liu, J., Jiang, Y. et Zhou, Z. (2002). Cyclic scheduling of a single hoist in extended electroplating lines: a comprehensive integer programming solution. *IIE Transaction, 34, pp.905-914*.

[Lopez et Esquirol, 1999] Lopez, P. et Esquirol, P. (1999). *L'ordonnancement*. Edition Economica, Paris.

[Lopez et Roubellat, 2008] Lopez, P. et Roubellat, F. (2008). *Production Scheduling*. Edition John Wiley & Sons Inc., Hoboken.

[Mahfoud et Golberg, 1995] Mahfoud, S.W. et Golberg, D.E. (1995). Parallele recombinative simulated annealing: a genetic algorithm. *Parallel Computing, 21, pp.1-28*.

[Mangione, 2003] Mangione, F. (2003). *Ordonnancement des ateliers de traitement de surface pour une production cyclique et mono-produit*. Thèse de doctorat de l'Institut National Polytechnique de Grenoble en Génie Industriel.

[Manier et Lamrous, 2008] Manier, M.-A. et Lamrous, S. (2008). An evolutionary approach for the design and scheduling of electroplating facilities. Journal of Mathematical Modelling and Algorithms.

[Manier, 2006] Manier, M.-A. (2006). *Planification et ordonnancement des tâches de transport*. Habilitation à Diriger des Recherches, Université de Franche-Comté.

[Manier et Lamrous, 2006] Manier, M.-A. et Lamrous, S. (2006). Design and scheduling of electroplating Facilities. *IEEE International Conference on Service Systems and Service Management, Troyes, pp.1114-1119*.

[Manier et al, 2000] Manier, M.-A., Varnier, C. et Baptiste, P. (2000). Constraint-based model for the cyclic multi-hoists scheduling problem. *Production Planning and Control*, *11(3), pp.244-257*.

[Manier et Bloch, 2003] Manier, M.-A. et Bloch, C. (2003). A classification for Hoist Scheduling Problem. *The International Journal of Flexible Manufacturing Systems, 15(1), pp.37-55*.

[Manier, 1994] Manier-Lacoste, M.-A. (1994). *Contribution à l'ordonnancement cyclique du système de manutention d'une ligne de galvanoplastie*. Thèse de doctorat de l'université de Franche-Comté en Automatique et Informatique.

[Manier et Baptiste, 1994] Manier, M.-A. et Baptiste, P. (1994). Etat de l'art : ordonnancement de robots de manutention en galvanoplastie. *Automatique productique informatique industrielle, 28(1), pp.7-35*.

[Mateo et Companys, 2006] Mateo, M., et Companys, R. (2006). Hoist Scheduling in a chemical line to produce batches with identical sizes of different products. *6ème Conférence Francophone de Modélisation et SIMulation, MOSIM'06, Rabat, pp.677-684*.

[Mateo, 2001] Mateo, M. (2001). *New computational experiences on the Hoist Scheduling Problem for cyclic manufacturing of different products*. Thèse de doctorat de l'université polytechnique de Catalunya, en Génie Industriel.

[Mateo et al, 2000] Mateo, M., Companys, R. et Bautista, J. (2000). Bounded cycle time for the cyclic hoist scheduling problem. *1st world conference on production and operations management, Seville, pp.1-10*.

[Matsuo et al, 1991] Matsuo, H., Shang, J. S., Sullivan, R.S. (1991). A Crane Scheduling Problem in a Computer-integrated Manufacturing Environment. *Management Science, 37(5), pp.587-606*.

[Mesghouni, 1999] Mesghouni, K. (1999). *Application des algorithmes évolutionnistes dans les problèmes d'optimisation en ordonnancement de production*. Thèse de doctorat de l'Université de Lille 1.

[Michalewicz et al., 1992] Michalewicz, Z., Janikow, C., Krawczyk, J. (1992). A modofoed genetic algorithm for optimal control problems. *Computers and Mathematics with Applications, 23(12), pp.83-94*.

[Ng et Leung, 1997] Ng, W.C. et Leung, J. (1997). Determining the optimal move times for a given cyclic schedule of a material hoist. *Computers industrial Engineering, 32(3), pp.595-606*.

[Ng, 1995] Ng, W.C. (1995). Determining the optimal number of duplicated process tanks in a single hoist circuit board production line. *Computer and industrial enginering, 28(4), pp.681-688*.

[Papadimitriou et Steiglitz, 1998] Papadimitriou, C. et Steiglitz, K. (1998). *Combinatorial optimization: algorithms and complexity*. Mineola, N.Y., Dover publications.

[Patnaik, 1994] Patnaik, S. (1994). Genetic algorithms : a survey. *IEEE computer society, 27(6), pp.17-26*.

[Paul et al, 2007] Paul, H. J., Bierwirth, C. et Kopfer, H. (2007). A heuristic procedure for multi-item hoist production line. *International Journal of Production Economics, 105, pp.54-69*.

[Pinedo, 1955] Pinedo, M. (1955). *Scheduling : Theory, Algorithms and Systems*. Prentice-Hall, Englewood Clis, New Jersey.

[Pirlot, 2002] Pirlot, M. (2002). *Optimisation approchée en recherché opérationnelle*, chapitre 1 Métaheuristiques pour l'optimisation combinatoire: un aperçu général, pp. 25-56, Edité par Jacques Teghem et Marc Pirlot, Lavoisier, Paris, 2008, ISBN: 2-7462-0462-2.

[Phillips et Unger, 1976] Phillips, L. W. et Unger, P. S. (1976). Mathematical Programming Solution of a Hoist Scheduling Program', *AIIE Transactions, 8(2), pp.219-225*.

[Portmann, 1996] Portmann, M.C. (1996). Genetic Algorithms and Scheduling : A State of the Art and some Propositions, *Proceedings of the Workshop on Production Planning and Control, Mons*.

[Ptuskin, 1995] Ptuskin, A. S. (1995). No-wait Periodic Scheduling of Non-identical Parts in Flexible Manufacturing Lines with Fuzzy Processing Times', *International Workshop on Intelligent Scheduling of Robots and Flexible Manufacturing Systems, Holon, pp.210-222*.

[Riera et Yorke-Smith, 2002] Riera, D. et Yorke-Smith, N. (2002). An Improved Hybrid Model for the Generic Hoist Scheduling Problem. *Annals of Operations Research, 115, pp.173-191*.

[Saad, 2007] Saad, I. (2007). *Conception d'un système d'aide à l'ordonnancement tenant compte des impératifs économiques*. Thèse de Doctorat, Ecole Centrale de Lille, Ecole Nationale d'Ingénieurs de Tunis.

[Sakarovitch, 1984] Sakarovitch, M. (1984). *Optimisation combinatoire : Méthodes mathématiques et algorithmiques- Programmation discrète*. Edition Hermann, Paris.

[Shapiro et Nuttle, 1988] Shapiro, G.W. et Nuttle, H.L.W. (1988). Hoist Scheduling for a PCB Electroplating Facility. *IIE Transactions, 20(2), pp.157-167*.

[Shapiro, 1985] Shapiro, G.W. (1985). *Hoist Scheduling for a PCB Electoplating Facility*. Thesis for the Degree of Master Science (Program in Operations Research), Graduate Faculty of North Carolina state University.

[Shoup, 2009] Shoup, V. (2009). *A Computational Introduction to Number Theory and Algebra*. Cambridge UP, second edition.

[Song et al, 1995] Song, W., Storch, R. L. et Zabinsky, Z. B. (1995). An example for scheduling a chemical processing tank line. *Proceedings of ETFA'95, Symposium on Emerging Tachnologies and Factory Automation, INRIA, IEEE, 1, pp.475-482*.

[Song et al, 1993] Song, W., Zabinsky, Z. B. et Storch, R. L. (1993). An algorithm for scheduling a chemical processing tank line. *Production Planning and Control, 4(4), pp.323-332*.

[Spacek et al, 1999] Spacek, P., Manier, M.-A., et El Moudni, A. (1999). Control of an electroplating line in the Max and Min algebras. *International Journal of Systems Science, 30(7), pp.759-778*.

[Storch et al, 1993] Storch, R L., Song, W. et Zabinsky, Z.B. (1993). Scheduling a chemical processing tank line. *The 5th international conference on advances in production management systems (B-13), IFIP W.G. 5.7, Athènes, pp.457-468*.

[Subaï et al, 2005] Subaï, C., Baptiste, P. et Niel, E. (2006). Scheduling issues for environmentally responsible manufacturing: The case of hoist scheduling in an electroplating line. *International Journal of Production Economics, 99 (1-2), pp.74-87*.

[Subaï et al, 2003] Subaï, C., Niel, E. et Baptiste, P. (2003). Vers un pilotage propre des lignes de traitement de surface. *Quatrième conférence francophone de MOdélisation et SIMulation : organisation et conduite d'activités dans l'industrie et les services, MOSIM'03, Toulouse*.

[Syswarda, 1989] Syswarda, G. (1989). Uniform cross-over in genetic algorithms. *Proceedings of 3rd international conference on genetic algorithms, pp.2-9*.

[Tangour, 2007] Tangour, F. (2007). *Ordonnancement dynamique dans les industries agroalimentaires*. Thèse de Doctorat, Ecole Centrale de Lille, Ecole Nationale d'Ingénieurs de Tunis.

[Thesen et Lei, 1990] Thesen, A., et Lei, L. (1990). An expert scheduling system for material handling Hoists. *Journal of Manufacturing Systems, 9(3), pp.247-252*.

[Tsang, 1993] Tsang, E.P.K. (1993). *Foundations of constraint satisfaction*. ISBN : 0127016104.

[Varnier et Jeunehomme, 2000] Varnier, C. et Jeunehomme, N. (2000). A cyclic approach for the multi-product hoist scheduling problem. *In 7th International Workshop on Project Management and Scheduling – PMS, Osnabrueck*.

[Varnier et al, 1997] Varnier, C., Bachelu, A. et Baptiste, P. (1997). Resolution of the cyclic multi-hoist scheduling problem. *Information Systems and Operational Research 35(4), pp.309-324*.

[Varnier, 1996] Varnier, C. (1996). *Extensions du "Hoist Scheduling Problem" cyclique. Résolution basée sur un traitement des contraintes disjonctives en programmation en logique avec contraintes*. Thèse de doctorat de l'université de Franche-Comté en Automatique et Informatique.

[Xu et Huang, 2004] Xu, Q. et Huang, Y. (2004). Graph-Assisted Cyclic Hoist Scheduling for Environmentally Benign Electroplating. *Industrial and Engineering Chemistry Research, 43(26), pp.8307-8316*.

[Yang et al, 2001] Yang, G.W., Ju, D.P., Zheng, W.M. et Lam, K. (2001). Solving multiple hoist scheduling problems by use of simulated annealing. *Ruan Jian Xue Bao/Journal of Software, 12(1), pp.11-17*.

[Yih, 1994] Yih, Y. (1994). An algorithm for hoist scheduling problems. *International Journal of Production Research, pp.501-16*.

[Yih et al, 1993] Yih, Y., Liang, T.P. et Moskowitz, H. (1993). Robot scheduling in a circuit board production line: an OR/ANN approach. *Institute of Industrial Engineers Transactions, 25(2), pp.26-33*.

[Yih et Chiu, 1993] Yih, Y. et Chiu, C. (1993). Incremental learning for hoist scheduling problems in circuit board production lines. *Manufacturing Aspects in Electronics Packaging ASME, EPP, 5, pp.119-124*.

[Yih, 1990] Yih Y. (1990). Trace-driven knowledge acquisition (TDKA) for rule based real time scheduling systems. *Journal of Intelligent Manufacturing, 1, pp.217-230*.

[Yih et Thesen, 1989] Yih, Y. et Thesen, A. (1989). An evaluation of user interfaces for interactive knowledge acquisition for expert scheduling systems. *Designing and using Human-Computer Interfaces and Knowledge Based Systems, pp.720-726*.

[Zhou et Li, 2009] Zhou, Z. et Li, L. (2009). A solution for cyclic scheduling of multi-hoists without overlapping. *Annals of Operations Research, 168, pp.5-21*.

[Zhou et Liu, 2008] Zhou, Z. et Liu, J. (2008). A heuristic algorithm for the two-hoist cyclic scheduling problem with overlapping hoist coverage ranges. *IIE Transactions, 40, pp.782-794*.

[Zhou et Ling, 2003] Zhou, Z. et Ling, L. (2003). Single hoist cyclic scheduling with multiple tanks: a material handling solution. *Computers & Operations Research, 30, pp.811-819*.

# ANNEXE 1 :
# notion d'optimisation combinatoire

**[Sakarovitch, 1984], [Papadimitriou et Steiglitz, 1998]**

Le problème d'ordonnancement que nous traitons dans le cadre de cette thèse relève de l'optimisation combinatoire. Ce type de d'optimisation occupe de plus en plus d'importance et plus particulièrement dans le domaine de la Recherche Opérationnelle. Ce qui pourrait-être expliqué, d'une part, par la grande difficulté de résolution des problèmes d'optimisation, et d'autre part, par la facilité de formalisation de ce type de problèmes en des problèmes d'optimisation combinatoire. Toutefois, ces derniers se caractérisent par leur difficulté de résolution, puisqu'ils sont pour la plupart des problèmes NP-difficiles et ne possèdent donc pas à ce jour de solution algorithmique efficace valable pour toutes les données.
Un problème d'optimisation combinatoire peut-être défini par :

**Définition 1.** [Sakarovitch, 1984]

Soit $S$ l'ensemble des solutions possibles d'un problème. Soit $X \subseteq S$, l'ensemble des solutions admissibles vérifiant un ensemble de contraintes $C$. Soit $f$: X $\mapsto \mathbb{R}$, une fonction que l'on nomme fonction objectif. Un problème d'optimisation se définit par :

$$\begin{cases} \min f(x) \\ x \in X \end{cases}$$

Lorsque l'ensemble $S$ des solutions est discret, on parle de problème d'optimisation combinatoire.

Nous pouvons définir les problèmes de maximisation, de façon similaire, en remplaçant **max** ***f(x)*** par **- min (-*f(x)*)**.
Ainsi, à chaque instance du problème d'optimisation mono-objectif est associé :

- ✓ un ensemble discret de solution $S$ ;
- ✓ une fonction objectif $f$ (ou fonction de coût) ;
- ✓ un ensemble de contraintes $C$ que doit satisfaire une solution afin d'être admissible.

Toutefois, il pourrait exister d'autres variantes dans la définition des problèmes d'optimisation combinatoire à l'exemple des problèmes d'optimisation multi-objectifs ou les problèmes de satisfaction de contraintes. Une définition adéquate de ces deux derniers est donnée respectivement dans [Collette et Siarry, 2002] et [Tsang, 1993].

La résolution d'un problème d'optimisation combinatoire, ou plus précisément d'une telle instance du problème, consiste à chercher une solution $s^* \in \mathrm{X}$ optimisant la fonction objectif. Cette solution $s^*$, s'appelle *solution optimale* ou *optimum global* et d'où la définition suivante.

**Définition 2.** inspirée de [Papadimitriou et Steiglitz, 1998]

Une instance $I$ d'un problème de minimisation est un couple de (X, $f$) où $\mathrm{X} \subseteq S$ est un ensemble fini de solutions admissibles, et $f$ une fonction objectif à minimiser $f$ : X $\mapsto \mathbb{R}$. Le problème est de trouver $s^* \in \mathrm{X}$ tel que $f(s^*) \leq f(s)$ pour tout élément $s \in \mathrm{X}$.

D'une manière similaire, on peut définir les problèmes de maximisation en remplaçant simplement $\leq$ par $\geq$.

Pour plus de précision, il est important d'expliquer la différence qui existe entre les notions de *problème* et d'*instance de problème*. La première se réfère à une question générale. Cette question, exprimée d'une façon générique, possède généralement des paramètres dont les valeurs doivent être instanciées. On parle ainsi d'instance de problème ou de problème instancié. A titre d'exemple, un problème d'optimisation peut être décrit par la question : *Quel est l'ordonnancement le plus adéquat pour augmenter la production d'un atelier ?* Toutefois, une instance du problème est décrite quand on fixe le type d'atelier, les gammes opératoires ainsi que les durées d'exécution de chaque type de produit sur chaque machine.

# ANNEXE 2 :
# notation dédiée au HSP

**[Manier et Bloch, 2003]**

Afin, de définir un cadre de description systématique permettant de spécifier précisément le problème traité et de le situer parmi les différentes classes du HSP, une notation ainsi qu'une typologie ont été proposé par Manier et Bloch [Manier et Bloch, 2003].

La notation proposée repose sur quatre champs, faisant apparaitre quatre facteurs principaux :

*Le type de HSP |les paramètres physiques |les paramètres logiques |le critère à optimiser*

$$\alpha \mid \beta \mid \delta \mid \gamma$$

- ✓ *Le premier champs* $\alpha$ = XHSP correspond au type de problème étudié.
  - $\alpha$ = {CHSP ou PHSP} selon que le problème statique (prédictif) est cyclique ou pas.
  - $\alpha$ = {DHSP ou RHSP} selon que le problème dynamique est avec prévision dans ce cas on parle d'un ordonnancement DHSP ou non et dans ce cas, on parle d'un problème d'ordonnancement réactif.

- ✓ *Le second champs* $\beta = \beta_1, \beta_2, \beta_3, \beta_4 / \beta_5, \beta_6, \beta_7, \beta_8 / \beta_9$ spécifie certaines caractéristiques du système physique à travers onze paramètres.
  - $\beta_1 \in \{\emptyset, \text{nl}\}$, $\beta_1$ = nl, si le nombre de lignes est égal à nl (avec nl > 1) et $\beta_1 = \emptyset$ dans le cas contraire, c'est-à-dire que le nombre de ligne est limité à une seule.
  - $\beta_2 \in \{\emptyset, \text{ntransfer}\}$, $\beta_2$ = ntransfer, si le nombre total de transporteurs interlignes est supérieur à 1 et $\beta_2 = \emptyset$, dans le cas contraire.
  - $\beta_3 \in \{\emptyset, \text{synchro}\}$ décrit le protocole d'échange entre robots et chariots de transfert : $\beta_3$ = synchro, si l'échange d'un porteur entre deux lignes nécessite, à un moment donné, la présence simultanée d'un chariot de transfert et d'un robot. $\beta_3 = \emptyset$ dans le cas où les mouvements d'un transporteur interligne sont indépendants de ceux des robots.

- $\beta_4 \in \{\beta_{41}, \beta_{42}, \beta_{43}\}_{i=1 \text{ à } nl}$, symbolise à la fois la configuration de la ligne, le nombre de cuves ainsi que le nombre de robots pour chaque tronçon de ligne considéré :

$\beta_{41} \in$ {I, U, O, L, H, …},

$\beta_{42}$ = mt, définit le nombre de cuves, non incluant les postes de chargement et de déchargement,

$\beta_{43} = \{\emptyset, mh\}$ où mh est le nombre de robots présents sur la ligne quand ce nombre est supérieur à 1.

- $\beta_5 \in \{\emptyset, ct\}$ spécifie la capacité maximale des cuves.

$\beta_5 = \emptyset$, si toutes les cuves sont mono-bac et s'il n'est pas nécessaire de le distinguer de β6 qui prend lui aussi la valeur $\emptyset$,

$\beta_5 = 1$, si toutes les cuves sont mono-bac et $\beta_6$ est différent de $\emptyset$,

$\beta_5$ = ct, si la cuve de plus grande capacité comporte ct emplacements.

- $\beta_6 \in \{\emptyset, circ\}$ traduit l'hypothèse de circulation des porteurs définit par [Shapiro et Nuttle, 1988] ; ainsi $\beta_6 = \emptyset$, s'il n y'a pas de circulation des porteurs et dans le cas contraire $\beta_6$ = circ.

- $\beta_7 \in \{\emptyset, ret\}$. Ce paramètre précise l'existence d'un système dédié au retour des porteurs vides entre les postes de déchargement et de chargement : $\beta_7 = \emptyset$, si le robot assure le retour des porteurs vides ou si les postes de chargement et de déchargement sont associés, sinon $\beta_7$ = ret.

- $\beta_8 \in \{\emptyset, empty\}$ est associé à la nécessité de gérer les porteurs vides qui ne peuvent être évacués de la ligne. $\beta_8$ = empty, si les porteurs vides ne peuvent pas être stockés hors ligne ou au poste de chargement / déchargement, sinon $\beta_8 = \emptyset$.

- $\beta_9 \in \{\emptyset, ass, diss\}$ décrit les postes de chargement et de déchargement, trois cas sont à distinguer : $\beta_9 = \emptyset$ si la configuration de ces postes est quelconque, $\beta_9$=ass, si les postes sont associés et $\beta_9$=diss, s'ils sont dissociés.

*Remarque* : dans certains articles la notation load/unload est employée pour traduire l'absence d'information sur la configuration étudiée.

- ✓ *Le troisième champs* $\delta = \delta_1 / \delta_2, \delta_2, \delta_3, \delta_4, \delta_5$ traduit les caractéristiques du système logique.

  - $\delta_1$ = nparts est le nombre total de produits à traiter.

  - $\delta_2 \in \{\emptyset, \text{nps}\}$ correspond au nombre de gammes considérées. $\delta_2 = \emptyset$, si le problème est mono-gamme, sinon $\delta_2$ = nps, où nps est le nombre de gammes différentes.

  - $\delta_3$ = nop correspond au nombre maximal d'opérations dans l'ensemble de toutes les gammes considérées.

  - $\delta_4 \in \{\emptyset, \text{clean}\}$ précise si la gamme comporte des opérations de nettoyage. Ainsi, si le déchargement est la dernière opération de la gamme, $\delta_4 = \emptyset$, sinon $\delta_4$=clean.

  - $\delta_5 \in \{\emptyset, \text{recrc}\}$ spécifie si une contrainte de recirculation est associée aux ressources de traitement. $\delta_5 = \emptyset$, si toutes les cuves sont mono-fonction, sinon et si au moins une cuve est multi-fonctions alors $\delta_5$ = recrc.

- ✓ *Le quatrième et le dernier champ* $\gamma \in \{C_{max}, T_{min}, \ldots\}$ correspond au critère à optimiser ; en d'autres termes, l'objectif à atteindre. Pour le HSP, cet objectif peut être :

  - Minimisation la période d'un cycle (T ou $T_{min}$) pour le cas d'un ordonnancement périodique.
  - Minimisation la date de fin de production ($C_{max}$)
  - Minimisation des encours
  - Maximisation de l'utilisation des ressources
  - Maximisation de la production
  - Minimisation du nombre de chariots utilisés sur une ligne et un horizon donné

  Dans le cas d'une approche multi-critères, le champ $\gamma$ peut contenir autant de paramètres que de critères à optimiser.

Cette typologie est présentée sous la forme arborescente et chaque type de problèmes **X**HSP est arboré sur l'une des figures de l'annexe 3.

Cette notation a été appliquée à un grand nombre d'articles trouvés dans la littérature (voir annexe 4). A première vue, elle peut paraître complexe. Toutefois, il très important de rappeler que plusieurs simplifications sont réalisées puisque la plupart du temps, on fait appel aux valeurs par défaut des paramètres.

**ANNEXE 3 :**
**typologie arborescente des travaux de recherche mentionnés dans notre état de l'art, en utilisant la notation élaborée par**

**Manier et Bloch [Manier et Bloch, 2003]**

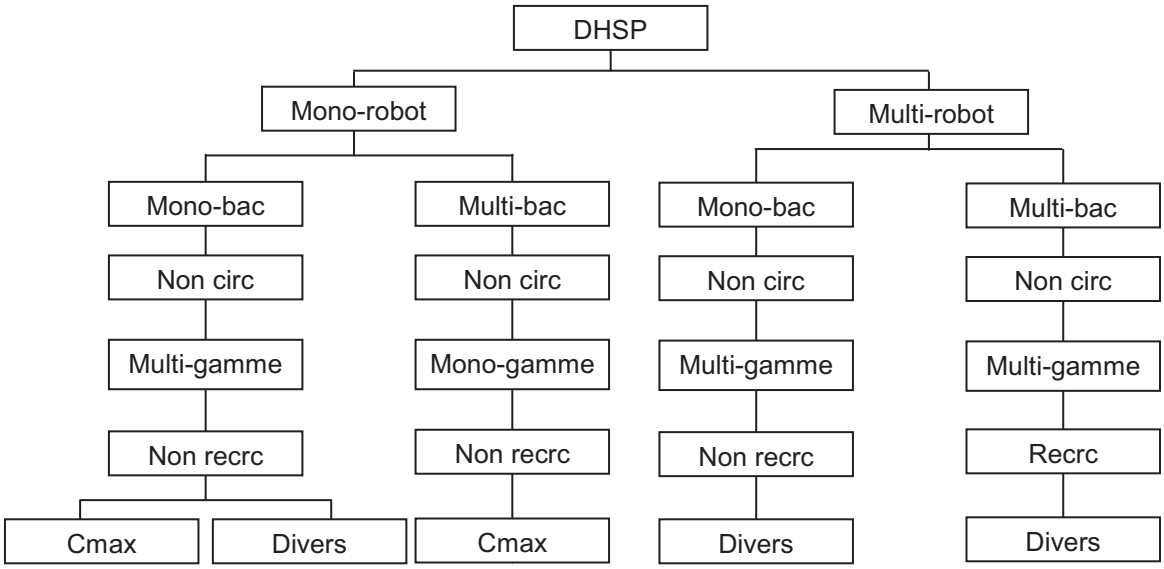



**Figure A_3.1. Typologie du cas dynamique**

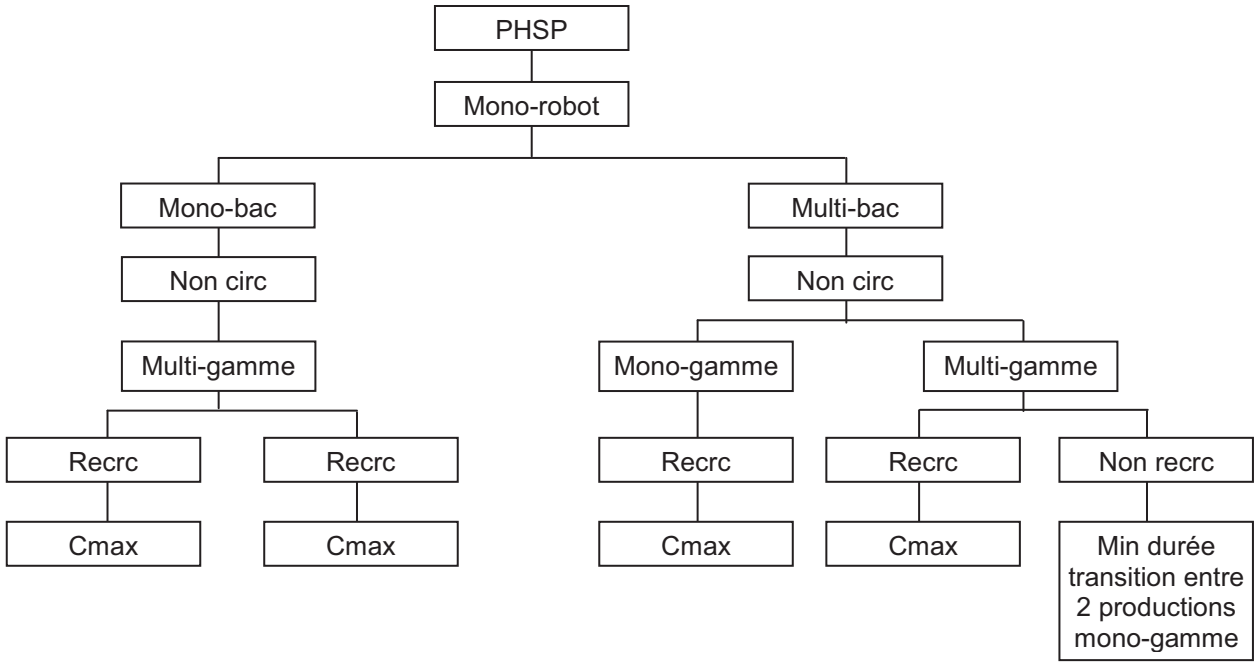



**Figure A_3.2.** Typologie du cas prédictif non cyclique

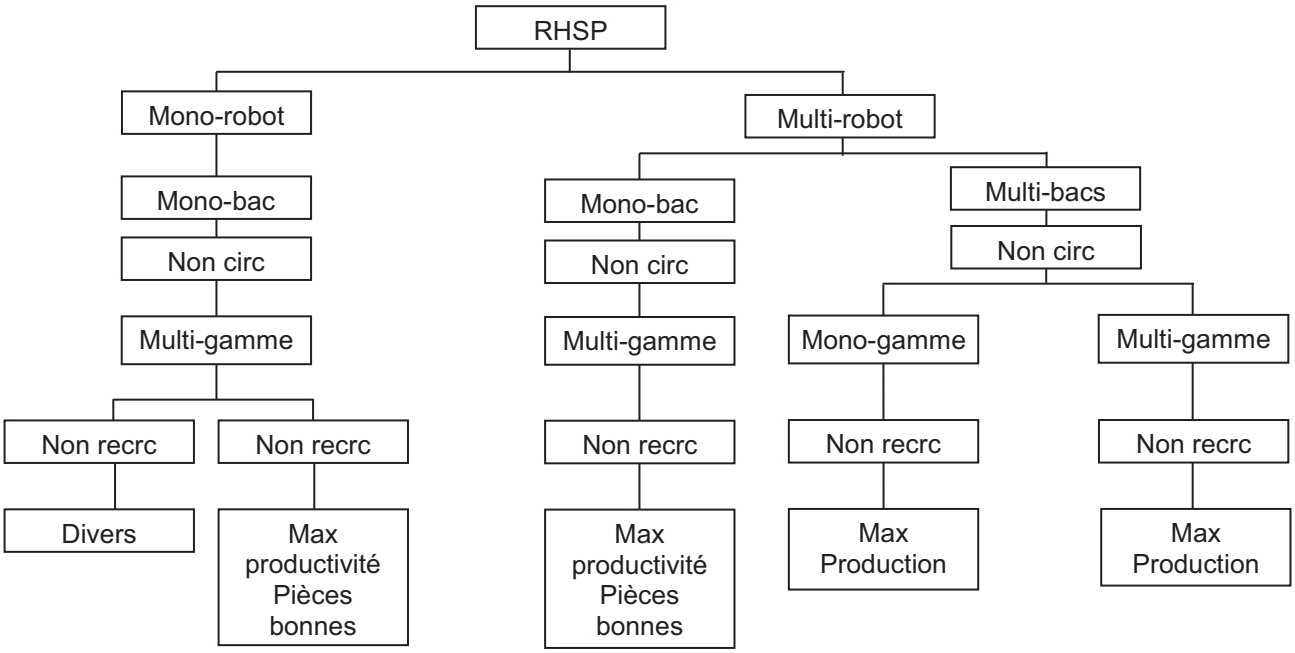



**Figure A_3.3.** Typologie du cas réactif

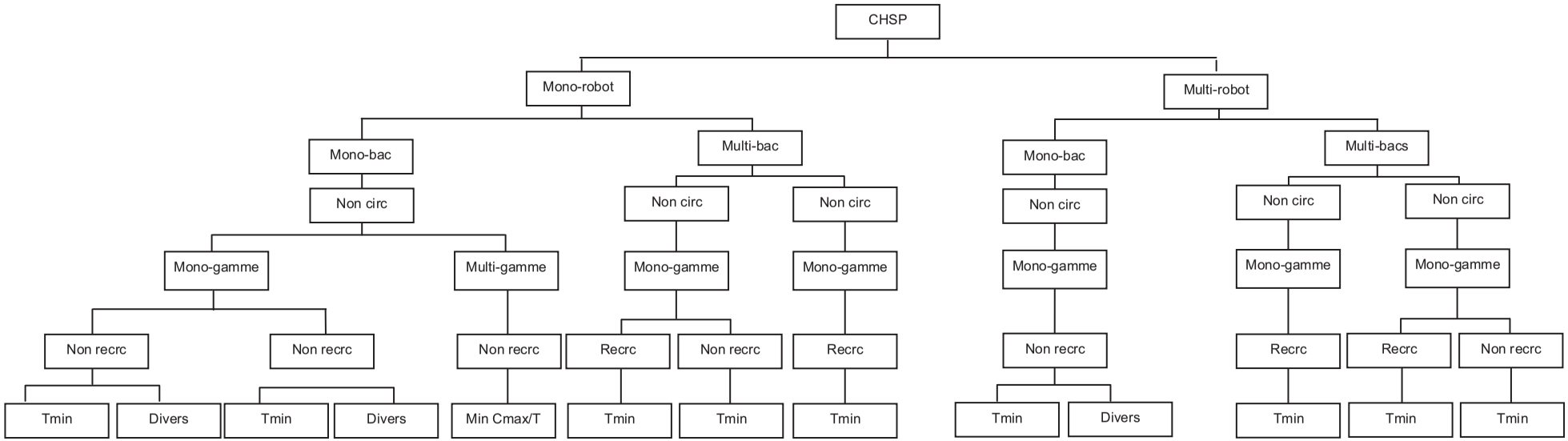



**Figure A_3.4.** Typologie du cas cyclique

# ANNEXE 4 :
# classification d'une partie des travaux de recherches mentionnés dans notre état de l'art, en utilisant la notation élaborée par Manier et Bloch

**[Manier et Bloch, 2003]**

(Rappel sur la notation complète en annexe 2)

| N° | Référence | Notation |
|---|---|---|
| 1 | [Armstrong et al, 1994] | CHSP \|19//diss\|/21\|$T_{min}$ |
| 2 | [Armstrong et al, 1996] | CHSP \|mh,50,ct//diss\|/52\|mhmin |
| 3 | [Baptiste et al, 1996] | CHSP \|3,18,10//\|/19,recrc\|Tmin |
| 4 | [Bloch, 1999] | PHSP\|mh,mt,1/ret/\|nparts/nps,nop\|Cmax |
| 5 | [Caux et al, 1995]a | PHSP \|3,4,synchro,(15),(15),(15)// /diss\|16/24,17,recrc\|Cmax |
| 6 | [Caux et Pierreval, 1997] | PHSP \|16//diss\|36/10,12,recrc\|Cmax |
| 7 | [Cheng et Smith, 1995] | DHSP \|5//diss\|100/100,7\|Cmax |
| 8 | [Chen et al, 1994] | CHSP\|13//diss\|/14\|Tmin |
| 9 | [El Amraoui et al, 2011][a] | CHSP \|mt//diss\|/2/nop,recrc\|Tmin |
| 10 | [El Amraoui et al, 2009][a] | CHSP \|mt,ct//diss\|/2/mt+2\|Tmin |
| 11 | [El Amraoui et al, 2008][b] | CHSP \|mt//diss\|/2/mt+2\|Tmin |
| 12 | [Fargier et Lamothe, 2001] | DHSP\|mt/nc/\|nparts/nop\|(Tmin,Qmax) |
| 13 | [Fleury, 1995] | PHSP \|1,12,ct/nc/\|20/nps,4,recrc\|Cmax |
| 14 | [Fleury et al, 2001]* | SHSP*\| mh,16 / 6, empty/ diss \| nparts/ / 12, nop\| (Cmax, Autre) |
| 15 | [Ge et Yih, 1995] | DHSP \|mt//diss\|100/nps,nop\|Cmax |
| 16 | [Hanen et Munier, 1993] | CHSP \|1,14,ct//diss\|/16,recrc\|Tmin |
| 17 | [Hanen, 1994] | CHSP \|mh,mt,ct//diss\|/nop\|Tmin |
| 18 | [Hindi et Fleszar, 2004] | DHSP \| 5 / / diss \| 100 / 100,6\| Cmax |
| 19 | [Jégou et al, 2006] | RHSP\|2,16,1//load-unload\| nparts/ /14\|(Min Produit Défectueux, Max Production) |
| 20 | [Lacomme, 1998] | PHSP\|16/6,empty/diss\|nparts/12,16,recrc\|Cmax |
| 21 | [Kuntay et al, 2006] | CHSP \| 1, 10, 8 / / ass \| / 12\| (Tmin, Min Perte) |
| 22 | [Lamothe, 1996] | DHSP\|2,1,synchro,(mh,17,1),(mh,17,2)//diss\|50/ 4,25,recrc\|Autre |
| 23 | [Lamothe et al, 1996] | DHSP\|2,12,1//diss\|/nps,14\|Autre |
| 24 | [Lei et Wang, 1989]a | CHSP\|1,12,ct//\|/14\|Tmin |

| | | |
|---|---|---|
| 25 | [Lei et Wang, 1991] | CHSP \| 2,25,ct / / diss \|/ 27\| Tmin |
| 26 | [Lei, 1993] | CHSP\|5//diss\|/7\|Tmin |
| 27 | [Lei et al, 1993] | CHSP\|mh,12,1//diss\|/14\|mhmin |
| 28 | [Lei et Liu, 2001] | CHSP\|1,20,ct//\|/42\|Tmin |
| 29 | [Lei et Wang, 1994] | CHSP\|12//diss\|/14\|Tmin |
| 30 | [Lim, 1997] | CHSP\|12//load-unload\|/14\|Tmin |
| 31 | [Liu et al, 2002] | CHSP \| 1, 8, 5 / / load-unload \| / 11, recrc\| Tmin |
| 32 | [Manier-Lacoste, 1994] | CHSP\|mh,mt,ct/circ/\|/nop,clean,recrc\|Tmin |
| 33 | [Manier et al, 2000] | CHSP\|3,17,10//load-unload\|/20,clean,recrc\|Tmin |
| 34 | [Manier et Lamrous, 2006] | CHSP \| mh, 12, 1 / / ass \| / 14\| (Tmin, mhmin) |
| 35 | [Manier et Lamrous, 2008] | CHSP \| mh, 14, 1 / circ/ ass \| / 16\| (Tmin, mhmin) |
| 36 | [Mateo et al, 2000] | CHSP\|8//\|/nop\|Tmin |
| 37 | [Mateo et Companys, 2006] | CHSP \| mt / / diss \| / 2, mt+2\| Tmin |
| 38 | [Paul et al, 2007] | DHSP \| 18 / / diss \| 40 / 4, 12\| Cmax |
| 39 | [Phillips et Unger, 1976] | CHSP \| 12/ / ass \| / 14, recrc\| Tmin |
| 40 | [Ptuskin, 1995] | CHSP\|50/nc/\|nparts/nps,nop\|Cmax |
| 41 | [Riera et Yorke-Smith, 2002] | CHSP\|4,13,ct//\|/nop,recrc\|Tmin |
| 42 | [Shapiro et Nuttle, 1988] | CHSP \|1,15,6/nc,circ/\|/17,clean,recrc\|Tmin |
| 43 | [Song et al, 1995] | CHSP\|10//diss\|/12\|Autre |
| 44 | [Spacek et al, 1999] | DHSP\|1,12,3//\|/14\|Cmax |
| 45 | [Subaï et al, 2006] | CHSP \| 12/ / ass \| / 14\| (Tmin, Min Perte, Autres) |
| 46 | [Thesen et Lei, 1990] | RHSP\|3,20,1/nc/\|nparts/3,nop\| Max Production |
| 47 | [Varnier, 1996] | PHSP\|mh,mt,1//ass\|/2,nop, recrc\|Min Trans |
| 48 | [Varnier et al, 1997] | CHSP\|4,30,ct//ass\|/30,recrc\| Tmin |
| 49 | [Varnier et Jeunehomme, 2000] | CHSP\|mt//load-unload\| /nps, nop\| Tmin |
| 50 | [Xu et Huang, 2004] | CHSP \| 1, 15, 8 / / ass \| / 17\| (Tmin, Min Perte) |
| 51 | [Yang et al, 2001] | CHSP \| mh, mt, 1 / / diss \| / mt+2\| Tmin |
| 52 | [Yih et al, 1993] | RHSP\|5//diss\|nparts/nps, 7\| Max Production |
| 53 | [Yih, 1994] | DHSP\|5//diss\|100/100, 7\|Cmax |
| 54 | [Zhou et Ling, 2003] | CHSP \| 1, 10, 5 / / ass \| / 12\| Tmin |

* SHSP = PHSP avec des temps de transport, générés aléatoirement.

# ANNEXE 5 :
# caractéristiques et données des benchmarks 2-cyclique multi-produits hétérogènes de

**[Matéo, 2001]**

A notre connaissance, ce sont les seuls instances disponibles traitant du CHSP multi-produits. Dans ces exemples, on traite deux types de produits possédant la même séquence opératoire mais ayant des bornes de durées différentes.

Ces données sont générées d'une façon aléatoire. Les bornes minimales des fenêtres de temps ($a_i$) sont générées entre 20 et 80 par une loi de distribution uniforme. Et les bornes maximales sont générées à la base des valeurs des $a_i$ et de la largeur de la fenêtre de temps.

| |
|---|
| Numéro du benchmark; Réference [Matéo, 2001] |
| Nombre de cuves de traitements |
| Durée de trempe minimale  Durée de trempe maximmle pour chaque traitement du premier produit [$a_{2i}$, $b_{2i}$] |
| Durée de trempe minimale  Durée de trempe maximmle pour chaque traitement du second produit [$a_{2i+1}$, $b_{2i+1}$] |
| Durée du déplacement en charge du robot entre deux cuves successives |
| Durée du déplacement à vide du robot entre deux cuves successives |

Exemples d'une ligne à 5 cuves de traitements

1 104040705
5
50 114  80 184  60 145  80 191  28 67
49 126  71 178  54 152  53 159  56 166
12
8

2 105040905
5
55 123  57 138  43 104  67 151  62 143
45 123  20 52  68 204  73 207  25 73
10
7

3 109040505
5
20 47  49 121  22 54  68 167  67 148
49 131  78 206  27 78  30 88  30 81
15
10

4 102040605
5
46 107  51 115  79 194  24 52  35 87
43 125  22 55  40 103  49 131  64 169
10
7

5 103040805
5
50 117  57 134  51 117  44 96  54 131
72 207  27 76  79 213  40 112  46 119
13
9

6 107070405
5
77 184  57 142  67 160  45 105  27 65
33 361  33 249  43 286  28 112  23 208
12
8

7 109070605
5
20 47  49 121  22 54  68 167  67 148
71 417  43 412  44 372  62 320  27 215
15
10

8 102070205
5
46 107  51 115  79 194  24 52  35 87
43 279  73 762  24 168  56 274  76 693
10
7

9 102070705
5
46 107 51 115 79 194 24 52 35 87
76 299 26 268 72 673 72 484 74 726
10
7
10 101070405
5
25 55 34 78 79 182 48 111 25 55
33 361 33 249 43 286 28 112 23 208
12
8
11 205050605
5
71 167 63 146 49 109 45 109 58 141
40 109 45 130 40 109 59 162 34 99
16
8
12 203050905
5
43 98 61 143 75 167 35 79 68 160
57 168 72 196 44 130 75 193 69 177
10
5
13 208050705
5
68 149 23 54 29 65 74 183 66 160
48 139 26 75 77 224 72 190 20 58
14
7
14 209050705
5
68 167 39 95 69 165 51 122 45 110
48 139 26 75 77 224 72 190 20 58
14
7
15 204050805
5
32 72 70 168 31 75 20 50 28 67
53 135 62 177 47 121 50 142 58 159
12
6
16 201080405
5
28 67 40 95 26 59 29 66 43 106
28 164 61 378 26 225 32 203 57 486
20
10
17 207080605
5
63 144 63 146 37 88 30 70 53 124
27 153 30 183 22 207 38 308 55 371
16
8
18 209080305
5
68 167 39 95 69 165 51 122 45 110
44 257 32 339 54 179 71 290 69 259
14
7
19 205080205
5
71 167 63 146 49 109 45 109 58 141
31 231 78 423 21 128 59 178 78 663
16
8
20 204080905
5
32 72 70 168 31 75 20 50 28 67
24 134 37 135 69 572 58 246 46 332
12
6
21 308060205
5
40 89 63 155 34 84 73 171 33 78
44 129 27 75 80 238 55 148 27 70
18
6
22 301060405
5
63 157 70 161 80 199 73 174 74 185
55 158 41 121 30 86 50 138 45 134
27
9
23 304060405
5
54 128 42 94 47 113 32 77 70 166
55 158 41 121 30 86 50 138 45 134
27
9
24 303060905
5
35 81 80 187 38 94 58 130 45 111
32 92 48 123 77 215 67 194 33 83
15
5
25 309060805
5
58 128 76 185 64 156 48 118 72 172
43 110 21 60 63 179 26 72 23 60
24
8
26 309090705
5
58 128 76 185 64 156 48 118 72 172
32 352 43 210 40 135 23 90 43 158
24
8
27 308090205
5
40 89 63 155 34 84 73 171 33 78
46 403 63 370 78 386 24 227 45 411
18
6
28 303090905
5
35 81 80 187 38 94 58 130 45 111
59 260 74 662 43 248 50 464 71 457
15
5

29 302090205
5
40 92 62 148 47 116 66 159 67 157
46 403 63 370 78 386 24 227 45 411
18
6

30 304090405
5
54 128 42 94 47 113 32 77 70 166
75 727 72 249 56 492 51 316 74 289
27
9

## Exemples d'une ligne à 6 cuves de traitements

1 111041606
6
60 134 63 147 80 199 71 161 62 145 72 161
38 110 63 188 45 117 34 86 37 95 78 213
12
8
2 111042006
6
60 134 63 147 80 199 71 161 62 145 72 161
75 210 72 189 22 62 66 188 52 150 70 191
12
8
3 116041806
6
30 70 48 119 70 155 45 105 24 57 72 170
73 212 55 146 69 189 44 122 34 86 48 128
9
6
4 116041706
6
30 70 48 119 70 155 45 105 24 57 72 170
73 212 38 113 69 183 23 59 64 179 49 131
9
6
5 117041706
6
30 70 59 144 44 108 45 109 24 57 32 76
73 212 38 113 69 183 23 59 64 179 49 131
9
6
6 120072006
6
63 153 49 120 62 155 59 133 31 76 26 62
74 310 67 673 22 219 44 444 59 360 21 190
7
5
7 116071306
6
30 70 48 119 70 155 45 105 24 57 72 170
40 381 40 403 27 116 50 159 24 160 44 197
9
6
8 119071306
6
31 74 29 65 56 137 71 160 49 117 72 167
40 381 40 403 27 116 50 159 24 160 44 197
9
6
9 112071606
6
71 158 21 49 57 130 31 68 29 68 29 64

30 184 28 270 47 162 65 391 67 403 25 219
15
10
10 117071706
6
30 70 59 144 44 108 45 109 24 57 32 76
51 477 53 366 35 304 68 741 35 225 60 462
9
6
11 213051106
6
33 77 32 74 44 98 23 57 53 122 56 134
77 228 31 82 23 63 36 93 40 105 44 125
14
7
12 217051806
6
73 172 65 152 50 115 79 186 37 83 74 173
59 175 76 192 30 84 51 148 72 182 57 147
18
9
13 215051806
6
47 114 42 95 47 106 72 168 74 181 57 140
59 175 76 192 30 84 51 148 72 182 57 147
18
9
14 211051406
6
39 93 65 155 78 171 69 152 24 59 35 79
26 70 32 85 54 143 22 63 70 182 50 126
18
9
15 211051806
6
39 93 65 155 78 171 69 152 24 59 35 79
59 175 76 192 30 84 51 148 72 182 57 147
18
9
16 218081906
6
70 155 56 140 62 139 43 102 34 81 79 173
33 144 79 788 73 686 43 140 54 398 24 177
16
8
17 217081606
6
73 172 65 152 50 115 79 186 37 83 74 173
42 229 70 399 20 88 26 123 56 601 36 200
18
9

18 219081406
6
73 161 63 151 44 101 35 84 61 144 69 167
55 397 28 245 62 532 39 217 60 537 56 378
14
7
19 215081106
6
47 114 42 95 47 106 72 168 74 181 57 140
49 437 56 435 56 392 37 383 66 648 75 322
18
9
20 216081606
6
26 57 65 146 66 147 64 143 37 85 60 136
42 229 70 399 20 88 26 123 56 601 36 200
18
9
21 314061406
6
66 156 52 119 54 120 75 182 45 99 30 71
24 70 30 76 30 83 33 92 79 217 31 80
30
10
22 318061506
6
24 52 56 128 68 165 59 135 37 88 36 90
30 84 25 71 44 124 27 75 32 85 63 178
18
6
23 316061906
6
21 50 20 46 38 84 21 49 59 130 28 63
27 68 54 147 26 68 63 186 45 126 29 82
21
7
24 311062006
6
68 159 47 116 68 149 35 81 26 61 23 55
31 93 23 65 71 210 68 171 28 73 76 196
24
8
25 319061906
6
59 138 26 61 44 101 30 67 28 65 38 92
27 68 54 147 26 68 63 186 45 126 29 82
21
7
26 315091306
6
21 46 47 105 45 107 77 183 37 83 21 48
25 232 71 218 61 502 47 447 58 615 41 124
24
8
27 318091606
6
24 52 56 128 68 165 59 135 37 88 36 90
34 274 26 252 57 172 56 515 67 646 45 380
18
6
28 311091306
6
68 159 47 116 68 149 35 81 26 61 23 55
25 232 71 218 61 502 47 447 58 615 41 124
24
8
29 316091906
6
21 50 20 46 38 84 21 49 59 130 28 63
23 95 57 335 40 333 26 85 34 247 29 118
21
7
30 312091706
6
39 86 59 136 52 126 39 88 77 170 68 160
78 642 45 352 70 269 23 236 46 172 27 281
15
5

## Exemples d'une ligne à 7 cuves de traitements

1 130042607
7
75 173 67 156 80 183 65 152 45 99 71 169
75 184
54 150 49 127 76 220 24 71 57 152 67 189
29 75
10
7
2 125043007
7
74 171 57 139 56 132 70 165 55 128 65 145
33 80
23 67 62 177 35 99 32 87 67 177 70 181
64 164
13
9
3 127042807
7
21 49 28 66 61 136 46 102 28 69 24 56
72 169
58 169 71 199 73 208 59 154 32 84 58 163
41 110
15
10
4 128042407
7
68 161 79 181 37 86 46 111 28 63 55 127
35 78
31 91 49 131 39 115 36 99 78 205 56 158
30 88
9
6
5 126043007
7
53 131 33 75 24 56 54 125 31 72 70 159
30 72

23 67 62 177 35 99 32 87 67 177 70 181
64 164
13
9
6 130072707
7
75 173 67 156 80 183 65 152 45 99 71 169
75 184
41 409 67 731 34 331 30 150 50 205 34 213
30 329
10
7
7 130072707
7
75 173 67 156 80 183 65 152 45 99 71 169
75 184
41 409 67 731 34 331 30 150 50 205 34 213
30 329
10
7
8 126073007
7
53 131 33 75 24 56 54 125 31 72 70 159
30 72
79 247 31 124 45 458 39 330 29 242 36 176
47 206
13
9
9 126073007
7
53 131 33 75 24 56 54 125 31 72 70 159
30 72
79 247 31 124 45 458 39 330 29 242 36 176
47 206
13
9
10 130072507
7
75 173 67 156 80 183 65 152 45 99 71 169
75 184
61 374 47 403 73 238 29 310 32 330 29 153
52 265
10
7
11 228052607
7
50 119 58 131 28 66 50 124 56 138 32 74
60 140
41 102 62 178 70 197 76 199 76 216 78 226
58 168
14
7
12 228052607
7
50 119 58 131 28 66 50 124 56 138 32 74
60 140
41 102 62 178 70 197 76 199 76 216 78 226
58 168
14
7
13 223052407
7
55 134 59 147 43 106 37 89 36 88 73 177
20 48
47 129 37 107 61 182 55 138 59 150 36 107
38 98
16
8
14 228052907
7
50 119 58 131 28 66 50 124 56 138 32 74
60 140
52 143 48 141 21 63 48 138 21 56 41 107
41 104
14
7
15 226052507
7
74 168 35 79 71 177 54 118 61 140 37 81
52 122
64 162 80 215 75 215 54 136 23 59 73 197
36 106
10
5
16 226082807
7
74 168 35 79 71 177 54 118 61 140 37 81
52 122
24 126 36 143 24 248 66 522 69 299 60 312
25 259
10
5
17 223082207
7
55 134 59 147 43 106 37 89 36 88 73 177
20 48
78 672 66 709 58 419 27 235 38 119 41 172
69 351
16
8
18 223082507
7
55 134 59 147 43 106 37 89 36 88 73 177
20 48
30 282 65 340 66 426 64 679 28 188 52 206
30 314
16
8
19 227082707
7
57 136 62 147 46 106 38 91 62 139 42 97
39 93
76 568 42 233 21 99 76 387 50 205 43 326
77 772
12
6
20 228082607
7
50 119 58 131 28 66 50 124 56 138 32 74
60 140

79 279 62 340 43 203 51 279 36 150 31 142
32 289
14
7
21 325062907
7
31 74 52 115 74 172 50 114 30 68 43 103
21 51
31 87 62 160 29 79 52 138 28 79 62 176
57 160
27
9
22 330062207
7
79 187 73 181 26 58 61 138 51 122 29 63
73 174
63 187 61 166 51 149 36 107 43 113 34 97
51 130
15
5
23 328062307
7
60 133 45 105 67 163 24 60 72 170 72 168
36 83
78 209 76 206 22 65 50 142 66 182 36 98
51 142
24
8
24 328062707
7
60 133 45 105 67 163 24 60 72 170 72 168
36 83
35 100 61 159 64 182 65 167 68 180 56 160
77 211
24
8
25 329062707
7
70 159 31 75 27 60 76 170 47 107 30 69
53 123
35 100 61 159 64 182 65 167 68 180 56 160
77 211
24
8
26 323092507
7
63 142 37 91 68 162 22 52 32 73 66 156
50 121
41 318 37 378 32 165 73 498 56 581 54 472
31 223
30
10
27 326092807
7
68 168 39 85 58 131 32 73 27 66 27 66
44 110
71 339 53 551 58 281 62 316 57 443 79 544
76 775
21
7
28 325092707
7
31 74 52 115 74 172 50 114 30 68 43 103
21 51
60 470 52 546 70 632 42 409 30 292 24 96
47 462
27
9
29 328092907
7
60 133 45 105 67 163 24 60 72 170 72 168
36 83
37 294 80 655 52 494 72 734 43 345 48 243
30 324
24
8
30 327092307
7
48 106 37 92 72 166 54 123 64 151 58 131
69 170
45 460 59 319 42 137 57 438 34 248 25 178
60 538
15
5

## Exemples d'une ligne à 8 cuves de traitements

1 140043108
8
73 168 66 148 75 178 41 100 61 150 22 53
66 146 45 110
33 97 31 92 28 71 27 79 61 154 42 117
37 110 63 175
10
7
2 139044008
8
36 87 33 78 35 85 37 91 38 86 41 100
46 107 40 93
63 173 49 140 72 203 53 138 46 132 76 199
40 116 24 65
7
5
3 140043108
8
73 168 66 148 75 178 41 100 61 150 22 53
66 146 45 110
33 97 31 92 28 71 27 79 61 154 42 117
37 110 63 175
10
7
4 133043108
8
75 170 30 72 24 56 36 87 51 112 50 112
23 51 77 171
33 97 31 92 28 71 27 79 61 154 42 117
37 110 63 175
10
7

5 138043708
8
60 142 67 147 53 125 32 80 35 86 29 68
45 108 26 64
26 77 27 74 48 135 71 212 44 113 58 147
80 201 27 70
7
5
6 137073308
8
45 112 64 142 47 110 76 187 30 70 29 67
24 53 77 178
60 343 61 463 41 141 52 276 61 635 57 239
75 793 52 465
10
7
7 134073908
8
67 150 40 94 75 166 28 63 46 107 52 126
36 85 63 156
55 215 73 231 77 240 46 432 50 152 71 608
62 663 52 260
12
8
8 136073908
8
53 120 75 183 65 156 52 117 33 81 24 53
36 86 75 171
55 215 73 231 77 240 46 432 50 152 71 608
62 663 52 260
12
8
9 131073308
8
76 177 49 108 56 125 65 157 60 137 31 74
50 114 22 54
60 343 61 463 41 141 52 276 61 635 57 239
75 793 52 465
10
7
10 131073608
8
76 177 49 108 56 125 65 157 60 137 31 74
50 114 22 54
56 598 26 98 36 394 21 169 31 319 51 391
29 279 55 233
10
7
11 237053208
8
43 99 21 47 34 83 66 162 67 162 47 111
73 168 21 50
67 196 38 98 44 130 64 175 78 234 37 100
46 134 37 96
12
6
12 236053208
8
61 146 26 64 73 174 61 149 20 47 20 49
33 73 42 92
67 196 38 98 44 130 64 175 78 234 37 100
46 134 37 96
12
6
13 234053408
8
70 166 39 96 63 152 59 139 22 49 42 100
58 136 50 117
41 110 69 189 62 174 53 144 52 130 71 184
53 148 61 167
16
8
14 232053108
8
24 59 50 121 37 88 37 86 79 193 22 50
35 85 32 71
28 77 35 105 39 109 33 91 63 183 52 151
45 117 30 86
16
8
15 239053208
8
41 101 32 79 64 145 74 180 50 111 40 89
21 50 69 166
67 196 38 98 44 130 64 175 78 234 37 100
46 134 37 96
12
6
16 232083608
8
24 59 50 121 37 88 37 86 79 193 22 50
35 85 32 71
40 284 29 284 41 380 55 179 55 174 43 255
58 594 69 275
16
8
17 233083908
8
64 150 29 65 27 64 42 97 60 137 52 122
53 131 26 62
62 680 57 460 58 371 71 582 63 224 52 550
78 312 31 309
20
10
18 240083108
8
61 139 21 49 60 135 24 54 60 137 61 148
26 64 41 98
73 582 59 311 74 519 52 288 80 802 60 372
55 533 71 733
10
5
19 240083808
8
61 139 21 49 60 135 24 54 60 137 61 148
26 64 41 98
45 238 53 517 57 437 36 165 45 453 53 578
36 354 34 205
10
5

20 231083608
8
47 108 39 95 69 160 74 162 58 142 28 61
59 143 58 142
40 284 29 284 41 380 55 179 55 174 43 255
58 594 69 275
16
8
21 335063308
8
41 95 52 125 52 114 48 118 22 51 77 185
80 185 66 156
44 122 66 168 44 111 79 216 50 135 49 144
74 211 37 102
27
9
22 335063708
8
41 95 52 125 52 114 48 118 22 51 77 185
80 185 66 156
53 150 53 134 38 109 68 186 71 210 80 234
27 71 60 178
27
9
23 337063608
8
28 65 49 112 37 84 41 100 50 117 54 121
78 178 74 168
53 140 44 117 34 87 63 169 45 134 29 77
27 73 39 104
24
8
24 336063608
8
62 140 67 166 74 172 70 172 54 132 53 130
76 173 71 168
53 140 44 117 34 87 63 169 45 134 29 77
27 73 39 104
24
8
25 339063908
8
20 48 46 103 47 105 36 88 21 51 41 95
72 171 38 94
40 113 71 187 59 162 43 127 46 127 44 126
56 149 35 96
24
8
26 336094008
8
62 140 67 166 74 172 70 172 54 132 53 130
76 173 71 168
52 455 68 647 45 275 24 168 35 240 35 339
71 294 33 172
24
8
27 340093208
8
41 98 78 185 48 110 70 157 41 91 75 177
49 120 35 84
31 228 56 230 71 456 50 159 61 533 50 408
29 222 45 381
15
5
28 335093508
8
41 95 52 125 52 114 48 118 22 51 77 185
80 185 66 156
64 363 52 295 45 395 42 180 39 158 23 181
61 244 78 486
27
9
29 335093708
8
41 95 52 125 52 114 48 118 22 51 77 185
80 185 66 156
36 121 74 623 52 478 39 363 36 301 59 184
31 170 62 289
27
9
30 338093908
8
52 122 39 89 63 144 74 175 46 106 55 132
68 155 67 147
50 174 79 814 63 395 32 281 30 259 61 373
24 184 56 256
18
6

## Exemples d'une ligne à 9 cuves de traitements

1 144044509
9
42 100 66 152 36 87 21 50 29 68 32 74
68 168 60 146 36 85
65 184 65 189 76 192 21 53 38 113 56 160
49 128 51 152 22 56
13
9
2 142044509
9
39 97 67 152 41 92 39 88 30 66 47 108
32 76 71 174 78 194
65 184 65 189 76 192 21 53 38 113 56 160
49 128 51 152 22 56
13
9
3 148044309
9
52 128 68 170 43 101 57 133 32 72 40 90
66 159 53 129 30 73
48 134 29 74 72 201 69 187 39 103 59 150
20 55 49 136 29 85
10
7
4 146044509

9
49 114 41 95 58 133 42 93 66 162 30 73
63 142 58 142 32 79
65 184 65 189 76 192 21 53 38 113 56 160
49 128 51 152 22 56
13
9
5 148044709
9
52 128 68 170 43 101 57 133 32 72 40 90
66 159 53 129 30 73
53 151 80 215 24 65 27 67 39 105 40 107
50 133 44 131 56 156
10
7
6 148074809
9
52 128 68 170 43 101 57 133 32 72 40 90
66 159 53 129 30 73
33 288 57 362 47 207 80 387 43 309 65 622
38 416 79 516 64 687
10
7
7 142074509
9
39 97 67 152 41 92 39 88 30 66 47 108
32 76 71 174 78 194
21 109 62 576 59 211 47 260 74 253 53 212
42 207 50 178 52 568
13
9
8 141074409
9
50 124 33 78 67 162 21 49 30 68 23 54
26 64 33 75 64 159
34 328 45 221 76 577 74 757 71 410 36 395
34 226 45 290 60 285
9
6
9 146074509
9
49 114 41 95 58 133 42 93 66 162 30 73
63 142 58 142 32 79
21 109 62 576 59 211 47 260 74 253 53 212
42 207 50 178 52 568
13
9
10 141074609
9
50 124 33 78 67 162 21 49 30 68 23 54
26 64 33 75 64 159
79 784 55 424 52 347 39 153 24 235 52 371
72 601 60 499 29 284
9
6
11 245054909
9
64 159 28 61 25 56 40 97 32 80 32 78
53 116 26 60 50 124
36 91 78 204 68 188 60 156 76 201 28 81
58 153 36 99 30 82
14
7
12 247054709
9
60 136 59 136 76 189 32 70 48 110 42 102
65 148 64 154 24 53
55 143 57 142 61 181 57 164 45 117 54 153
58 169 42 121 20 60
12
6
13 242054709
9
71 171 48 110 76 174 47 109 69 159 35 78
69 163 44 107 40 90
55 143 57 142 61 181 57 164 45 117 54 153
58 169 42 121 20 60
12
6
14 246054609
9
65 154 55 126 31 68 34 80 21 52 35 78
76 189 44 105 51 123
54 154 40 108 77 223 58 151 55 161 69 198
20 58 52 132 61 178
15 245054909
9
64 159 28 61 25 56 40 97 32 80 32 78
53 116 26 60 50 124
36 91 78 204 68 188 60 156 76 201 28 81
58 153 36 99 30 82
14
7
16
8
16 243084309
9
44 102 62 141 30 71 33 74 76 180 45 100
32 73 72 178 49 115
46 297 42 203 39 425 41 310 61 651 48 313
64 242 49 261 60 202
16
8
17 249084509
9
76 180 71 163 25 60 24 54 66 151 53 121
56 129 69 154 47 105
43 298 24 192 37 339 58 633 37 338 76 501
22 122 21 92 27 209
20
10
18 244084409
9
24 57 51 126 74 172 32 75 61 135 47 109
24 58 52 122 79 185
63 218 69 658 75 557 38 343 36 181 28 240
41 297 21 84 22 200
16
8

19 244084309
9
24 57 51 126 74 172 32 75 61 135 47 109
24 58 52 122 79 185
46 297 42 203 39 425 41 310 61 651 48 313
64 242 49 261 60 202
16
8
20 242084909
9
71 171 48 110 76 174 47 109 69 159 35 78
69 163 44 107 40 90
62 439 41 439 74 351 53 463 76 575 50 172
53 168 46 376 71 310
12
6
21 343064609
9
22 50 68 167 62 155 75 171 41 98 47 108
36 86 39 87 68 156
35 102 51 137 41 113 60 163 24 61 30 86
74 204 28 71 36 95
27
9
22 350064809
9
56 133 74 169 65 145 62 138 72 164 23 57
77 183 54 119 77 187
78 218 61 157 35 98 26 73 80 212 57 163
29 84 49 125 77 209
24
8
23 346064309
9
31 71 26 61 63 152 27 66 39 92 55 126
65 149 70 171 59 137
65 166 53 132 77 198 54 156 50 128 33 91
45 114 30 76 70 180
21
7
24 349064709
9
68 163 63 148 33 74 77 184 58 142 45 111
76 185 49 117 40 90
28 81 29 79 77 197 23 61 30 86 73 195
70 199 36 99 68 197
15
5
25 343065009
9
22 50 68 167 62 155 75 171 41 98 47 108
36 86 39 87 68 156
78 203 21 53 60 152 68 170 58 164 68 200
56 149 43 123 39 105
27
9
26 350094809
9
56 133 74 169 65 145 62 138 72 164 23 57
77 183 54 119 77 187
34 264 29 141 62 369 56 547 33 350 59 431
70 393 30 164 70 561
24
8
27 348094609
9
69 168 52 129 21 47 27 66 35 81 79 194
62 153 62 152 74 171
67 701 24 83 39 293 61 635 69 227 60 225
39 348 28 124 68 394
21
7
28 344094909
9
28 64 45 110 53 127 73 166 70 156 78 195
28 63 79 185 21 48
47 474 44 238 46 500 30 111 27 106 52 546
77 644 75 799 63 282
18
6
29 348095009
9
69 168 52 129 21 47 27 66 35 81 79 194
62 153 62 152 74 171
57 262 32 136 72 611 40 190 51 347 80 475
72 333 56 366 35 262
21
7
30 343094409
9
22 50 68 167 62 155 75 171 41 98 47 108
36 86 39 87 68 156
32 320 49 344 70 669 41 447 38 407 56 368
55 329 71 521 22 84
27
9

## Exemples d'une ligne à 10 cuves de traitements

1 159045910
10
42 98 58 127 36 82 70 170 22 55 46 112
71 158 44 108 72 173 21 47
78 214 31 90 49 141 33 88 32 93 74 197
44 111 69 202 37 97 65 174
9
6
2 152045310
10
43 99 74 183 67 148 51 118 55 125 55 135
36 79 75 165 62 144 76 180
79 222 59 168 36 107 50 125 21 61 64 189
66 180 43 121 70 201 33 85
10
7
3 156045610
10

```
 22  52   75 171   21  46   36  88   37  84   49 114
54 131   45 109   38  91   70 167
 54 138   68 202   37 103   47 134   74 190   25  67
42 110   41 103   50 130   68 204
 13
  9
 4 156046010
10
 22  52   75 171   21  46   36  88   37  84   49 114
54 131   45 109   38  91   70 167
 24  70   57 149   67 185   35  93   73 205   21  55
59 173   68 185   62 156   69 182
 13
  9
 5 155045910
10
 73 167   45 104   40  88   34  75   62 140   33  73
33  76   50 120   41  97   46 110
 78 214   31  90   49 141   33  88   32  93   74 197
44 111   69 202   37  97   65 174
  9
  6
 6 159075810
10
 42  98   58 127   36  82   70 170   22  55   46 112
71 158   44 108   72 173   21  47
 43 388   72 545   37 355   57 439   54 249   21  84
27 159   44 346   76 407   57 227
  9
  6
 7 158075710
10
 58 129   41  93   73 163   37  89   68 162   43  95
66 156   51 117   37  85   21  46
 68 503   48 239   51 254   27 144   58 354   48 209
46 494   24 177   78 611   60 440
 15
 10
 8 152075210
10
 43  99   74 183   67 148   51 118   55 125   55 135
36  79   75 165   62 144   76 180
 54 472   39 276   43 181   59 581   54 229   29 100
26 161   28 142   41 235   51 449
 10
  7
 9 157075110
10
 70 156   76 177   23  52   26  65   69 161   43  95
76 169   69 171   76 174   59 133
 47 231   70 600   73 746   39 135   54 489   33 159
64 466   44 270   42 294   28 208
  7
  5
10 156075910
10
 22  52   75 171   21  46   36  88   37  84   49 114
54 131   45 109   38  91   70 167
 41 422   37 247   66 292   26 155   29 316   29 317
30 217   77 617   48 237   33 333
 13
  9
11 252055710
10
 57 134   79 179   24  57   80 177   39  96   22  49
63 139   68 159   20  46   55 125
 36  90   60 179   47 123   61 180   73 198   71 188
24  70   79 201   40 107   23  65
 16
  8
12 257055410
10
 24  57   69 167   30  72   77 189   55 128   72 171
46 105   79 194   60 138   76 174
 52 155   42 121   77 213   57 155   56 150   74 206
65 191   67 196   36  98   78 228
 18
  9
13 256055910
10
 61 151   61 136   34  76   35  81   57 136   44 106
44  98   73 182   74 166   38  87
 66 197   33  90   38 112   77 216   79 229   66 197
65 164   43 110   51 137   68 201
 12
  6
14 260055210
10
 48 114   50 112   40  89   79 196   49 111   58 135
65 149   45 111   38  85   57 133
 67 187   79 221   77 193   43 123   64 166   40 111
44 112   45 123   52 139   67 200
 14
  7
15 255055310
10
 67 159   60 138   37  88   77 173   73 174   21  48
31  77   43  97   22  52   67 155
 50 139   42 124   52 139   54 161   73 189   62 183
36 102   50 148   31  80   45 134
 20
 10
16 256085510
10
 61 151   61 136   34  76   35  81   57 136   44 106
44  98   73 182   74 166   38  87
 55 603   71 514   47 429   68 270   55 547   31 125
38 343   63 308   27 110   38 282
 12
  6
17 259085210
10
 23  56   79 178   21  46   63 154   59 139   25  55
51 113   48 111   27  62   48 119
 28 141   75 300   42 367   45 214   53 564   48 155
50 231   60 550   43 294   38 346
 10
  5
18 260085910
10
```

48 114 50 112 40 89 79 196 49 111 58 135
65 149 45 111 38 85 57 133
78 384 61 217 47 359 52 216 56 321 79 764
59 189 56 286 69 421 30 295
14
7
19 255085810
10
67 159 60 138 37 88 77 173 73 174 21 48
31 77 43 97 22 52 67 155
34 132 27 258 49 282 58 544 28 279 70 315
42 204 72 784 50 343 43 415
20
10
20 254086010
10
23 54 80 182 30 71 66 164 45 100 33 72
26 61 24 59 36 85 27 66
70 363 22 200 21 161 71 365 66 570 49 377
43 285 69 272 47 334 25 117
18
9
21 359065910
10
60 146 23 56 43 106 79 178 34 75 68 155
56 129 32 79 68 151 25 62
66 180 36 102 59 150 62 170 21 59 35 94
21 52 42 122 63 166 63 175
18
6
22 355065710
10
21 51 62 143 58 142 55 132 29 66 39 87
77 179 63 140 63 147 71 162
43 107 54 148 31 80 55 142 22 65 21 58
28 74 73 216 49 127 40 101
27
9
23 356065510
10
58 144 32 80 35 79 78 194 70 161 40 99
78 194 56 124 25 58 76 187
47 129 22 64 78 225 21 61 50 131 71 213
31 88 26 71 45 119 52 148
21
7
24 358065310
10
75 167 26 58 61 139 33 80 60 138 43 104
60 147 44 98 59 130 53 126
51 144 67 190 39 115 40 108 45 117 68 173
35 89 24 65 78 232 68 196
24
8
25 357065510
10
75 186 25 62 52 125 43 105 78 186 20 48
59 134 31 68 72 172 41 91
47 129 22 64 78 225 21 61 50 131 71 213
31 88 26 71 45 119 52 148
21
7
26 355095210
10
21 51 62 143 58 142 55 132 29 66 39 87
77 179 63 140 63 147 71 162
32 160 44 202 50 240 27 165 30 285 72 706
74 482 62 430 70 383 38 210
27
9
27 358095410
10
75 167 26 58 61 139 33 80 60 138 43 104
60 147 44 98 59 130 53 126
65 318 78 738 27 269 80 373 64 637 66 639
37 154 42 461 31 237 78 687
24
8
28 353095910
10
65 153 52 121 65 146 73 177 24 58 38 91
43 98 74 185 26 58 57 132
78 694 64 325 68 620 37 117 58 264 49 233
75 803 55 570 21 108 24 165
21
7
29 359095710
10
60 146 23 56 43 106 79 178 34 75 68 155
56 129 32 79 68 151 25 62
54 366 28 291 46 397 31 285 27 105 64 227
60 394 25 119 73 260 62 238
18
6
30 359095810
10
60 146 23 56 43 106 79 178 34 75 68 155
56 129 32 79 68 151 25 62
61 558 20 153 76 700 77 246 77 238 66 348
39 335 54 396 69 531 76 609
18
6

# ANNEXE 6 :

## résultats de la PLM sur les benchmarks 2-cyclique multi-produits hétérogènes de l'annexe 5

PARAMÈTRES DES TABLEAUX DE RÉSULTATS

| Symbole | Signification |
|---|---|
| N° | le numéro de l'exemple dans la série d'exemples. |
| Référence | la référence de l'exemple d'après [Matéo, 2001]. |
| CPU(s) | la durée du temps de simulation, en secondes. |
| Nœuds | correspond au nombre de nœuds visités ou encore, au nombre de disjonctions établis. |
| T | la durée minimale du cycle 2-cyclique hétérogène. |
| $T_1$ | la durée minimale du cycle 1-cyclique homogène pour le produit 1. |
| $T_2$ | la durée minimale du cycle 1-cyclique homogène pour le produit 2. |
| Ecart(%) | l'écart de réduction définit par la formule suivante : $\frac{(T_1+T_2)-T}{(T_1+T_2)}\times 100.$ |

| Nombre de cuves : 5 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **N°** | **Référence** | **CPU (secondes)** | **Nœuds** | **T** | **$T_1$** | **$T_2$** | **Ecart** | **Dates de début des opérations de transport du robot** |
| 1 | 104040705 | 115 | 140 | 368 | 198 | 184 | 3,66% | [0 256 120 340 276 92 28 176 148 296 196 48] |
| 2 | 105040905 | 110 | 97 | 319 | 158 | 155 | -1,92% | [0 164 78 295 181 17 264 95 34 198 112 233] |
| 3 | 109040505 | 116 | 22 | 489 | 252 | 254 | 3,36% | [0 70 35 134 99 227 159 287 252 369 334 414] |
| 4 | 102040605 | 97 | 26 | 347 | 185 | 144 | -5,47% | [0 155 86 244 172 285 261 24 302 110 41 199] |
| 5 | 103040805 | 119 | 211 | 405 | 240 | 214 | 10,79% | [0 182 111 374 204 22 268 133 325 226 53 294] |
| 6 | 107070405 | 115 | 497 | 370 | 195 | 168 | -1,93% | [0 195 131 342 215 20 306 151 40 235 79 270] |
| 7 | 109070605 | 118 | 133 | 466 | 252 | 210 | -0,87% | [0 70 35 249 99 329 159 411 284 124 366 184] |
| 8 | 102070205 | 118 | 75 | 333 | 185 | 158 | 2,92% | [0 141 86 230 158 17 247 103 288 175 41 264] |
| 9 | 102070705 | 110 | 54 | 338 | 185 | 154 | 0,29% | [0 141 86 227 158 17 247 103 293 185 41 269] |
| 10 | 101070405 | 115 | 54 | 373 | 214 | 168 | 2,36% | [0 65 37 165 85 213 185 273 245 321 293 113] |
| 11 | 205050605 | 115 | 227 | 432 | 224 | 208 | 0,00% | [0 248 120 360 272 24 384 80 48 192 160 304] |
| 12 | 203050905 | 115 | 65 | 260 | 140 | 130 | 3,70% | [0 155 85 240 170 65 20 125 105 210 190 35] |
| 13 | 208050705 | 110 | 169 | 376 | 196 | 190 | 2,59% | [0 126 98 266 147 306 196 28 327 168 49 217] |
| 14 | 209050705 | 116 | 119 | 377 | 201 | 190 | 3,58% | [0 199 171 349 224 21 314 136 42 245 101 279] |
| 15 | 204050805 | 124 | 59 | 325 | 182 | 156 | 3,85% | [0 78 54 162 138 253 181 24 223 108 277 199] |
| 16 | 201080405 | 163 | 170 | 494 | 346 | 204 | 10,18% | [0 88 48 158 118 308 188 354 258 424 384 218] |
| 17 | 207080605 | 143 | 36 | 384 | 224 | 192 | 7,69% | [0 296 160 352 320 128 32 184 88 240 208 56] |
| 18 | 209080305 | 110 | 242 | 368 | 201 | 168 | 0,27% | [0 126 98 214 151 263 235 28 319 172 49 291] |
| 19 | 205080205 | 143 | 302 | 421 | 224 | 208 | 2,55% | [0 184 112 280 208 24 304 136 365 232 56 333] |
| 20 | 204080905 | 143 | 59 | 325 | 182 | 156 | 3,85% | [0 78 54 162 138 253 181 24 223 108 277 199] |
| 21 | 308060205 | 160 | 117 | 392 | 192 | 207 | 1,75% | [0 224 98 362 188 24 290 122 56 260 146 314] |
| 22 | 301060405 | 143 | 158 | 594 | 314 | 354 | 11,08% | [0 450 171 549 324 36 495 117 72 270 225 369] |
| 23 | 304060405 | 163 | 84 | 576 | 409 | 354 | 24,51% | [0 396 81 531 162 36 243 117 324 198 459 279] |
| 24 | 303060905 | 174 | 139 | 323 | 160 | 170 | 2,12% | [0 100 75 205 180 20 255 125 40 230 145 278] |
| 25 | 309060805 | 143 | 151 | 546 | 284 | 368 | 16,26% | [0 359 104 426 311 498 458 136 48 208 168 255] |
| 26 | 309090705 | 174 | 390 | 512 | 284 | 263 | 6,40% | [0 304 136 472 336 96 424 168 48 240 200 376] |
| 27 | 308090205 | 174 | 241 | 388 | 192 | 186 | -2,65% | [0 222 96 286 256 24 310 120 54 174 144 334] |
| 28 | 303090905 | 143 | 174 | 331 | 160 | 160 | -3,44% | [0 100 75 246 180 20 266 125 40 200 145 286] |
| 29 | 302090205 | 143 | 260 | 383 | 192 | 186 | -1,32% | [0 120 90 270 240 24 305 150 48 204 174 329] |
| 30 | 304090405 | 143 | 151 | 558 | 409 | 270 | 17,82% | [0 198 153 333 234 504 369 45 450 270 81 405] |

| Nombre de cuves : 6 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **N°** | **Référence** | **CPU (secondes)** | **Nœuds** | **T** | **$T_1$** | **$T_2$** | **Ecart** | **Dates de début des opérations de transport du robot** |
| 1 | 111041606 | 174 | 424 | 450 | 220 | 221 | -2,04% | [0 358 145 422 229 108 386 165 36 249 185 298 269 56] |
| 2 | 111042006 | 225 | 424 | 440 | 220 | 236 | 3,51% | [0 236 136 412 256 108 376 156 36 276 176 340 296 56] |
| 3 | 116041806 | 174 | 240 | 337 | 171 | 160 | -1,81% | [0 87 39 177 102 241 193 54 256 117 292 208 132 271] |
| 4 | 116041706 | 174 | 419 | 336 | 171 | 165 | 0,00% | [0 87 39 177 102 240 192 54 255 117 291 207 132 270] |
| 5 | 116041706 | 175 | 337 | 366 | 189 | 165 | -3,39% | [0 182 60 264 204 339 279 75 27 111 90 225 131 300] |
| 6 | 120072006 | 205 | 1020 | 269 | 144 | 124 | -0,37% | [0 90 73 251 190 56 17 151 112 202 163 29 214 124] |
| 7 | 116071306 | 174 | 505 | 316 | 171 | 153 | 2,47% | [0 87 39 208 102 18 181 54 235 117 271 153 132 250] |
| 8 | 119071306 | 186 | 376 | 332 | 194 | 153 | 4,32% | [0 98 75 218 113 305 191 21 278 128 36 164 143 251] |
| 9 | 112071606 | 205 | 383 | 525 | 413 | 257 | 21,64% | [0 121 86 226 146 480 251 35 311 171 371 276 415 336] |
| 10 | 117071706 | 174 | 592 | 362 | 189 | 161 | -3,43% | [0 200 63 275 215 15 290 78 30 161 93 236 134 311] |
| 11 | 213051106 | 163 | 200 | 461 | 287 | 217 | 8,53% | [0 328 49 433 98 21 156 70 205 120 272 177 377 235] |
| 12 | 217051806 | 184 | 443 | 596 | 331 | 297 | 5,10% | [0 272 144 398 227 551 308 36 434 180 497 335 72 461] |
| 13 | 215051806 | 184 | 308 | 621 | 351 | 297 | 4,17% | [0 135 99 288 162 423 243 504 333 45 459 198 540 360] |
| 14 | 211051406 | 163 | 107 | 606 | 499 | 319 | 25,92% | [0 526 63 570 146 27 242 99 329 180 392 269 445 356] |
| 15 | 211051806 | 175 | 262 | 662 | 499 | 297 | 16,83% | [0 279 81 423 164 617 315 36 459 117 501 207 554 351] |
| 16 | 218081906 | 184 | 785 | 499 | 286 | 248 | 6,55% | [0 200 128 296 224 24 320 152 379 248 435 347 64 403] |
| 17 | 217081606 | 190 | 701 | 581 | 331 | 261 | 1,86% | [0 207 171 290 254 536 322 36 419 117 482 349 72 446] |
| 18 | 219081406 | 184 | 1211 | 420 | 231 | 203 | 3,23% | [0 196 133 392 217 21 301 154 350 238 84 322 259 56] |
| 19 | 215081106 | 174 | 530 | 567 | 351 | 261 | 7,35% | [0 101 65 168 128 276 195 384 303 459 411 222 486 330] |
| 20 | 216081606 | 174 | 111 | 583 | 342 | 261 | 3,32% | [0 144 63 252 171 27 279 90 361 198 424 306 502 388] |
| 21 | 314061406 | 163 | 30 | 739 | 602 | 414 | 27,26% | [0 635 96 689 186 40 276 136 385 226 475 335 535 425] |
| 22 | 318061506 | 184 | 1184 | 510 | 284 | 279 | 9,41% | [0 168 42 216 132 286 240 348 317 402 372 456 426 84] |
| 23 | 316061906 | 163 | 26 | 548 | 383 | 307 | 20,58% | [0 77 42 140 105 215 180 278 243 362 323 428 390 478] |
| 24 | 311062006 | 173 | 186 | 653 | 491 | 395 | 26,30% | [0 558 92 613 167 32 259 127 359 219 431 291 478 391] |
| 25 | 319061906 | 143 | 95 | 568 | 421 | 307 | 21,98% | [0 485 80 533 143 40 227 108 293 192 356 258 415 321] |
| 26 | 315091306 | 234 | 140 | 586 | 472 | 297 | 23,80% | [0 85 45 157 117 277 237 530 357 197 429 317 474 389] |
| 27 | 318091606 | 178 | 312 | 477 | 284 | 216 | 4,60% | [0 72 42 182 116 272 206 387 296 140 351 230 417 321] |
| 28 | 311091306 | 174 | 243 | 632 | 491 | 297 | 19,80% | [0 527 92 576 164 39 256 124 328 196 400 288 447 360] |
| 29 | 316091906 | 174 | 87 | 525 | 383 | 252 | 17,32% | [0 77 42 140 105 315 168 392 231 469 350 196 427 259] |
| 30 | 312091706 | 205 | 1047 | 386 | 207 | 190 | 2,77% | [0 100 75 235 210 20 351 125 40 260 175 321 280 150] |

| Nombre de cuves : 7 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **N°** | **Référence** | **CPU (secondes)** | **Nœuds** | **T** | **$T_1$** | **$T_2$** | **Ecart** | **Dates de début des opérations de transport du robot** |
| 1 | 130042607 | 230 | 2207 | 446 | 220 | 249 | 4,90% | [0 190 166 329 243 17 346 219 34 260 121 363 277 51 380 90] |
| 2 | 125043007 | 151 | 1043 | 653 | 371 | 361 | 10,79% | [0 586 87 622 157 44 226 126 309 179 377 259 455 342 501 424] |
| 3 | 127042807 | 115 | 271 | 690 | 480 | 431 | 24,26% | [0 71 36 144 109 230 195 318 262 392 346 452 417 539 504 595] |
| 4 | 128042407 | 208 | 1610 | 408 | 216 | 204 | 2,86% | [0 287 150 327 302 15 348 123 30 171 67 363 186 45 230 84] |
| 5 | 126043007 | 112 | 403 | 622 | 471 | 361 | 25,24% | [0 546 66 582 119 35 188 88 255 141 335 221 418 304 461 387] |
| 6 | 130072707 | 183 | 2022 | 418 | 220 | 205 | 1,65% | [0 291 171 394 308 147 24 191 116 325 208 41 342 85 58 225] |
| 7 | 130072707 | 183 | 2022 | 418 | 220 | 205 | 1,65% | [0 291 171 394 308 147 24 191 116 325 208 41 342 85 58 225] |
| 8 | 126073007 | 173 | 1006 | 598 | 471 | 248 | 16,83% | [0 102 71 257 124 390 177 31 288 146 341 199 470 310 513 439] |
| 9 | 126073007 | 173 | 1006 | 598 | 471 | 248 | 16,83% | [0 102 71 257 124 390 177 31 288 146 341 199 470 310 513 439] |
| 10 | 130072507 | 236 | 4187 | 426 | 220 | 203 | -0,71% | [0 206 182 402 285 48 24 230 144 302 247 72 350 113 89 326] |
| 11 | 228052607 | 175 | 1398 | 558 | 322 | 266 | 5,10% | [0 105 77 177 149 261 198 453 282 35 352 219 411 311 488 383] |
| 12 | 228052607 | 175 | 1398 | 558 | 322 | 266 | 5,10% | [0 105 77 177 149 261 198 453 282 35 352 219 411 311 488 383] |
| 13 | 223052407 | 115 | 528 | 635 | 403 | 374 | 18,28% | [0 540 71 603 146 24 205 101 279 172 335 247 424 303 460 359] |
| 14 | 228052907 | 138 | 557 | 578 | 322 | 349 | 13,86% | [0 431 64 550 136 34 185 85 249 157 319 206 368 270 487 340] |
| 15 | 226052507 | 231 | 2818 | 370 | 217 | 180 | 6,80% | [0 165 95 265 180 15 280 110 30 195 125 230 210 325 305 50] |
| 16 | 226082807 | 352 | 6092 | 360 | 217 | 170 | 6,98% | [0 204 175 340 220 90 20 150 105 285 240 35 300 120 50 255] |
| 17 | 223082207 | 175 | 1132 | 579 | 403 | 273 | 14,35% | [0 114 82 547 178 50 237 146 293 202 350 261 439 318 475 407] |
| 18 | 223082507 | 175 | 1486 | 600 | 403 | 268 | 10,58% | [0 123 72 219 147 315 243 32 339 171 395 267 484 363 520 419] |
| 19 | 227082707 | 124 | 3805 | 452 | 256 | 204 | 1,74% | [0 144 120 416 276 18 386 90 36 294 180 356 234 54 318 198] |
| 20 | 228082607 | 118 | 1103 | 539 | 322 | 224 | 1,28% | [0 189 126 294 210 21 259 147 357 231 427 329 476 399 63 448] |
| 21 | 325062907 | 113 | 27 | 775 | 589 | 502 | 28,96% | [0 672 89 730 170 44 271 125 352 226 441 307 525 396 573 480] |
| 22 | 330062207 | 155 | 4788 | 460 | 250 | 233 | 4,76% | [0 280 135 365 250 20 305 155 390 220 45 325 90 410 185 65] |
| 23 | 328062307 | 86 | 236 | 788 | 473 | 400 | 9,74% | [0 192 152 294 224 422 326 468 374 542 500 636 596 708 668 72] |
| 24 | 328062707 | 136 | 292 | 784 | 473 | 435 | 13,66% | [0 192 152 264 224 355 315 443 387 532 483 624 579 704 664 72] |
| 25 | 329062707 | 146 | 472 | 778 | 590 | 435 | 24,10% | [0 192 152 264 224 349 296 437 396 526 469 618 558 698 650 72] |
| 26 | 323092507 | 76 | 855 | 928 | 658 | 420 | 13,91% | [0 180 130 350 220 708 390 858 442 60 532 270 628 482 778 578] |
| 27 | 326092807 | 94 | 508 | 658 | 519 | 291 | 18,77% | [0 126 91 245 154 392 273 539 350 42 427 189 490 308 581 455] |
| 28 | 325092707 | 91 | 366 | 741 | 589 | 360 | 21,92% | [0 126 81 315 216 36 351 171 432 252 513 387 594 468 642 549] |
| 29 | 328092907 | 91 | 267 | 724 | 473 | 320 | 8,70% | [0 128 88 200 160 676 280 40 352 240 448 312 544 384 604 480] |
| 30 | 327092307 | 237 | 5281 | 438 | 260 | 200 | 4,78% | [0 148 63 283 168 20 303 83 398 188 103 323 238 368 343 213] |

| Nombre de cuves : 8 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **N°** | **Référence** | **CPU (secondes)** | **Nœuds** | **T** | **$T_1$** | **$T_2$** | **Ecart** | **Dates de début des opérations de transport du robot** |
| 1 | 140043108 | 172 | 1466 | 585 | 374 | 370 | 21,37% | [0 518 167 561 267 17 390 72 473 127 41 198 96 298 215 345 315 428] |
| 2 | 139044008 | 202 | 5034 | 412 | 238 | 188 | 3,29% | [0 88 71 182 111 265 194 374 277 22 352 128 34 211 140 294 223 325] |
| 3 | 140043108 | 172 | 1466 | 585 | 374 | 370 | 21,37% | [0 518 167 561 267 17 390 72 473 127 41 198 96 298 215 345 315 428] |
| 4 | 133043108 | 172 | 4070 | 605 | 407 | 370 | 22,14% | [0 149 125 192 168 233 209 279 255 359 316 440 376 492 409 539 509 59] |
| 5 | 138043708 | 172 | 2757 | 390 | 225 | 208 | 9,93% | [0 334 80 373 234 17 351 92 29 246 104 297 145 41 197 128 268 162] |
| 6 | 137073308 | 172 | 983 | 583 | 368 | 223 | 1,35% | [0 217 55 305 186 17 243 72 374 155 414 260 455 343 517 431 110 493] |
| 7 | 134073908 | 176 | 924 | 645 | 569 | 277 | 23,76% | [0 176 148 243 200 328 287 417 348 593 437 44 501 373 549 457 72 521] |
| 8 | 136073908 | 165 | 1388 | 680 | 416 | 277 | 1,88% | [0 93 65 180 152 265 229 357 293 533 377 620 441 313 489 397 576 461] |
| 9 | 131073308 | 374 | 20288 | 579 | 330 | 223 | -4,70% | [0 110 86 180 145 266 211 341 286 48 371 228 414 303 474 388 506 450] |
| 10 | 131073608 | 277 | 14105 | 573 | 330 | 209 | -6,31% | [0 128 86 200 145 272 217 24 292 162 362 234 408 309 468 379 500 444] |
| 11 | 237053208 | 116 | 2145 | 582 | 344 | 297 | 9,20% | [0 118 55 202 88 286 142 405 226 540 310 160 369 244 465 333 498 441] |
| 12 | 236053208 | 112 | 1296 | 563 | 498 | 297 | 29,18% | [0 97 73 181 115 266 200 322 284 467 340 139 382 224 431 358 497 407] |
| 13 | 234053408 | 104 | 1620 | 776 | 619 | 472 | 28,87% | [0 666 86 723 142 32 221 110 310 179 366 247 435 334 512 403 578 480] |
| 14 | 232053108 | 76 | 224 | 676 | 532 | 541 | 37,00% | [0 600 56 644 122 24 201 80 257 146 352 225 408 315 464 376 512 432] |
| 15 | 239053208 | 184 | 857 | 570 | 342 | 297 | 10,80% | [0 77 53 163 97 249 181 311 267 474 329 121 381 207 423 347 504 399] |
| 16 | 232083608 | 128 | 277 | 627 | 532 | 296 | 24,28% | [0 72 40 138 106 194 162 251 219 363 331 483 387 555 443 291 515 411] |
| 17 | 233083908 | 169 | 1278 | 805 | 623 | 372 | 19,10% | [0 150 110 232 192 434 262 745 332 50 474 292 546 364 619 504 665 576] |
| 18 | 240083108 | 726 | 30780 | 490 | 307 | 200 | 3,35% | [0 120 100 230 135 300 205 384 250 25 320 155 404 265 440 335 50 419] |
| 19 | 240083808 | 278 | 11570 | 410 | 307 | 195 | 18,33% | [0 130 110 295 145 15 270 85 315 160 35 220 175 365 235 50 340 190] |
| 20 | 231083608 | 168 | 467 | 712 | 648 | 296 | 24,58% | [0 95 63 191 119 287 215 664 311 143 385 239 451 335 526 409 600 494] |
| 21 | 335063308 | 88 | 255 | 1065 | 603 | 603 | 11,69% | [0 201 68 300 147 457 246 528 345 634 394 711 573 792 747 893 840 957] |
| 22 | 335063708 | 141 | 378 | 1066 | 603 | 774 | 22,59% | [0 113 68 282 149 439 228 504 327 603 376 710 549 817 656 871 755 958] |
| 23 | 337063608 | 148 | 146 | 831 | 601 | 622 | 32,05% | [0 92 52 169 125 241 201 313 273 400 347 474 432 575 534 672 632 735] |
| 24 | 336063608 | 182 | 2369 | 922 | 544 | 622 | 20,93% | [0 140 86 217 177 315 275 409 369 496 447 568 528 669 628 763 723 826] |
| 25 | 339063908 | 146 | 212 | 809 | 609 | 544 | 29,84% | [0 84 44 156 116 251 188 334 291 406 366 478 438 574 534 654 606 713] |
| 26 | 336094008 | 368 | 7758 | 878 | 544 | 400 | 6,99% | [0 126 86 246 206 32 326 166 420 286 498 380 587 452 687 547 782 619] |
| 27 | 340093208 | 381 | 25326 | 515 | 289 | 230 | 0,77% | [0 100 75 300 275 20 390 125 40 210 145 330 235 415 350 460 435 170] |
| 28 | 335093508 | 176 | 887 | 980 | 603 | 462 | 7,98% | [0 844 81 935 162 36 243 117 383 198 432 279 536 329 643 468 736 573] |
| 29 | 335093708 | 134 | 2503 | 972 | 603 | 468 | 9,24% | [0 113 68 194 149 295 250 439 331 538 380 601 484 694 637 775 730 864] |
| 30 | 338093908 | 345 | 13251 | 627 | 410 | 288 | 10,17% | [0 144 114 285 171 447 309 585 471 36 549 201 60 339 225 393 363 507] |

| Nombre de cuves : 9 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **N°** | **Référence** | **CPU (secondes)** | **Nœuds** | **T** | **$T_1$** | **$T_2$** | **Ecart** | **Dates de début des opérations de transport du robot** |
| 1 | 144044509 | 115 | 139 | 794 | 610 | 546 | 31,31% | [0 86 55 165 134 243 210 332 273 385 354 438 407 519 488 592 561 656 625 691] |
| 2 | 142044509 | 115 | 211 | 799 | 664 | 546 | 33,97% | [0 83 52 163 132 241 186 346 263 380 306 433 402 502 455 570 539 661 630 696] |
| 3 | 148044309 | 310 | 3606 | 711 | 482 | 401 | 19,48% | [0 629 62 687 140 17 202 99 269 178 320 227 370 296 446 346 509 422 549 463] |
| 4 | 146044509 | 116 | 269 | 793 | 659 | 546 | 34,19% | [0 93 62 171 122 249 193 338 271 391 360 444 413 520 489 591 560 655 624 690] |
| 5 | 148044709 | 296 | 2191 | 692 | 482 | 524 | 31,21% | [0 340 131 454 309 661 364 24 478 93 547 162 616 264 55 395 186 509 226 578] |
| 6 | 148074809 | 248 | 8456 | 660 | 482 | 254 | 10,33% | [0 128 69 171 147 255 200 404 274 31 316 217 366 292 477 340 540 442 580 516] |
| 7 | 142074509 | 139 | 333 | 773 | 664 | 323 | 21,68% | [0 143 72 236 165 22 258 94 311 187 379 280 442 346 495 401 579 464 670 548] |
| 8 | 141074409 | 189 | 940 | 569 | 479 | 222 | 18,83% | [0 81 60 124 102 199 178 461 214 27 257 151 311 229 374 272 416 326 500 395] |
| 9 | 146074509 | 155 | 533 | 723 | 659 | 323 | 26,37% | [0 103 72 197 126 683 219 32 274 148 353 241 406 307 482 375 553 438 598 522] |
| 10 | 141074609 | 96 | 830 | 568 | 479 | 222 | 18,97% | [0 144 123 237 165 301 280 391 316 529 364 33 412 192 448 337 490 427 60 469] |
| 11 | 245054909 | 93 | 1307 | 793 | 560 | 457 | 22,03% | [0 106 78 156 127 248 177 330 282 404 376 494 425 543 515 615 564 665 637 709] |
| 12 | 247054709 | 414 | 12669 | 697 | 446 | 350 | 12,44% | [0 173 72 281 143 401 251 553 305 30 365 102 431 215 511 329 589 449 625 481] |
| 13 | 242054709 | 849 | 18424 | 712 | 435 | 350 | 9,30% | [0 357 111 424 219 520 381 676 448 33 544 141 598 303 57 472 165 568 267 622] |
| 14 | 246054609 | 183 | 313 | 852 | 625 | 542 | 26,99% | [0 750 81 820 152 24 236 117 323 191 379 262 435 347 527 403 587 478 654 555] |
| 15 | 245054909 | 93 | 1307 | 793 | 560 | 457 | 22,03% | [0 106 78 156 127 248 177 330 282 404 376 494 425 543 515 615 564 665 637 709] |
| 16 | 243084309 | 353 | 1094 | 786 | 683 | 338 | 23,02% | [0 104 72 216 184 384 240 32 296 144 416 264 481 328 537 440 625 505 690 581] |
| 17 | 249084509 | 158 | 665 | 929 | 787 | 475 | 26,39% | [0 136 96 227 187 297 257 367 327 453 413 526 486 622 562 692 652 762 722 809] |
| 18 | 244084409 | 243 | 848 | 759 | 684 | 363 | 27,51% | [0 88 56 184 152 24 248 120 304 216 381 272 444 349 500 412 568 468 663 536] |
| 19 | 244084309 | 207 | 816 | 768 | 684 | 338 | 24,85% | [0 77 40 139 107 251 219 347 275 624 371 179 436 299 500 395 568 460 672 536] |
| 20 | 242084909 | 1628 | 32910 | 600 | 435 | 251 | 12,54% | [0 209 155 576 227 29 373 173 494 245 53 391 107 457 269 522 325 77 421 287] |
| 21 | 343064609 | 142 | 5 | 991 | 818 | 739 | 36,35% | [0 94 49 189 144 299 233 380 335 467 422 548 503 629 584 730 665 811 766 874] |
| 22 | 343064609 | 359 | 912 | 1026 | 880 | 785 | 38,38% | [0 94 49 189 144 299 233 380 335 467 422 548 503 629 584 730 665 811 766 874] |
| 23 | 346064309 | 180 | 527 | 919 | 715 | 548 | 27,24% | [0 87 52 173 115 247 201 377 275 454 335 540 412 594 498 674 622 737 702 828] |
| 24 | 349064709 | 672 | 15490 | 628 | 424 | 370 | 20,91% | [0 560 115 603 240 20 325 215 520 260 45 345 150 490 285 65 400 170 455 375] |
| 25 | 343065009 | 410 | 110 | 1056 | 818 | 851 | 36,73% | [0 560 115 603 240 20 325 215 520 260 45 345 150 490 285 65 400 170 455 375] |
| 26 | 350094809 | 711 | 1517 | 987 | 880 | 445 | 25,51% | [0 170 104 242 202 379 291 499 411 48 531 323 603 443 704 563 782 635 883 742] |
| 27 | 348094609 | 387 | 20524 | 849 | 700 | 350 | 19,14% | [0 224 189 312 272 375 340 438 403 543 466 786 571 126 654 494 737 599 77 688] |
| 28 | 344094909 | 419 | 12208 | 732 | 461 | 320 | 6,27% | [0 76 46 141 109 210 180 337 271 433 361 549 457 295 513 390 615 483 654 585] |
| 29 | 348095009 | 841 | 11151 | 847 | 700 | 392 | 22,44% | [0 210 175 288 248 351 316 484 379 603 442 49 561 407 680 526 763 645 91 728] |
| 30 | 343094409 | 552 | 566 | 971 | 818 | 483 | 25,37% | [0 94 49 189 144 278 233 380 335 461 416 542 497 625 580 707 661 805 760 854] |

| Nombre de cuves : 10 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| **N°** | **Référence** | **CPU (secondes)** | **Nœuds** | **T** | **T1** | **T2** | **Ecart** | **Dates de début des opérations de transport du robot** |
| 1 | 159045910 | 614 | 8121 | 737 | 448 | 376 | 10,56% | [0 533 51 716 118 19 163 77 242 133 282 178 337 261 417 314 470 392 617 438 647 596] |
| 2 | 152045310 | 541 | 6527 | 848 | 532 | 459 | 14,43% | [0 77 53 189 137 258 214 323 275 388 340 429 405 503 451 579 536 632 608 718 694 761] |
| 3 | 156045610 | 148 | 517 | 875 | 689 | 748 | 39,11% | [0 66 35 154 123 235 176 288 257 348 310 435 372 488 457 546 515 600 568 682 651 763] |
| 4 | 156046010 | 124 | 186 | 918 | 689 | 643 | 31,08% | [0 71 35 154 123 224 176 304 246 357 326 443 388 496 465 568 537 649 590 724 693 806] |
| 5 | 155045910 | 307 | 3573 | 761 | 486 | 376 | 11,72% | [0 153 132 240 186 280 255 338 298 390 369 432 411 515 453 568 547 646 597 692 661 63] |
| 6 | 159075810 | 118 | 17081 | 652 | 448 | 264 | 8,43% | [0 99 51 190 118 15 163 66 242 142 278 221 333 257 413 293 466 348 547 433 577 499] |
| 7 | 158075710 | 184 | 1654 | 910 | 770 | 425 | 23,85% | [0 108 73 191 133 256 221 322 281 399 364 524 424 590 549 766 615 805 675 459 711 640] |
| 8 | 152075210 | 78 | 77485 | 781 | 532 | 321 | 8,44% | [0 412 53 543 246 17 374 70 474 263 574 450 660 491 715 591 108 629 180 684 308 156] |
| 9 | 157075110 | 288 | 41941 | 456 | 268 | 202 | 2,98% | [0 134 117 305 288 12 318 151 358 266 29 330 80 370 217 41 382 178 53 229 190 394] |
| 10 | 156075910 | 238 | 2041 | 797 | 689 | 340 | 22,55% | [0 66 35 154 123 207 176 286 229 348 308 441 370 623 463 712 521 392 574 485 663 543] |
| 11 | 252055710 | 149 | 695 | 952 | 771 | 586 | 29,85% | [0 865 73 917 168 41 224 115 320 192 400 281 465 368 551 424 654 519 690 575 761 614] |
| 12 | 257055410 | 274 | 595 | 1032 | 885 | 750 | 36,88% | [0 865 73 917 168 41 224 115 320 192 400 281 465 368 551 424 654 519 690 575 761 614] |
| 13 | 256055910 | 298 | 50290 | 781 | 494 | 437 | 16,11% | [0 102 78 180 151 282 198 378 252 587 348 36 404 222 460 318 545 436 653 499 703 629] |
| 14 | 260055210 | 306 | 3356 | 911 | 571 | 553 | 18,95% | [0 90 62 171 126 264 192 355 313 412 376 490 448 556 527 614 586 673 640 739 711 820] |
| 15 | 255055310 | 811 | 124 | 1056 | 828 | 825 | 36,12% | [0 90 62 171 126 264 192 355 313 412 376 490 448 556 527 614 586 673 640 739 711 820] |
| 16 | 256085510 | 173 | 28154 | 756 | 494 | 270 | 1,05% | [0 97 73 170 146 299 192 586 239 30 323 214 384 257 440 341 538 416 628 458 678 508] |
| 17 | 259085210 | 660 | 10131 | 632 | 405 | 238 | 1,71% | [0 104 33 142 122 227 172 279 245 334 314 397 351 457 412 547 472 367 509 427 567 489] |
| 18 | 260085910 | 200 | 37555 | 805 | 571 | 329 | 10,56% | [0 90 62 182 126 280 203 341 301 427 364 511 448 231 532 385 591 469 643 553 714 615] |
| 19 | 255085810 | 314 | 741 | 1038 | 828 | 531 | 23,62% | [0 90 62 182 126 280 203 341 301 427 364 511 448 231 532 385 591 469 643 553 714 615] |
| 20 | 254086010 | 240 | 2706 | 916 | 687 | 415 | 16,88% | [0 77 41 221 185 347 248 500 374 853 455 131 536 284 601 410 664 563 727 628 772 691] |
| 21 | 359065910 | 230 | 2154 | 911 | 752 | 582 | 31,71% | [0 797 78 881 132 24 212 102 309 182 363 279 449 333 530 403 584 473 670 554 713 640] |
| 22 | 355065710 | 914 | 676 | 1106 | 934 | 812 | 36,66% | [0 90 62 182 126 280 203 341 301 427 364 511 448 231 532 385 591 469 643 553 714 615] |
| 23 | 356065510 | 784 | 318 | 1005 | 856 | 751 | 37,46% | [0 77 41 221 185 347 248 500 374 853 455 131 536 284 601 410 664 563 727 628 772 691] |
| 24 | 358065310 | 937 | 4318 | 1062 | 866 | 867 | 38,72% | [0 797 78 881 132 24 212 102 309 182 363 279 449 333 530 403 584 473 670 554 713 640] |
| 25 | 357065510 | 124 | 225 | 960 | 804 | 751 | 38,26% | [0 857 99 925 162 28 235 127 299 190 398 263 461 363 541 426 604 489 697 569 759 642] |
| 26 | 355095210 | 619 | 1477 | 1093 | 934 | 608 | 29,12% | [0 136 91 255 164 374 283 899 402 42 447 199 510 318 586 475 681 551 744 614 822 709] |
| 27 | 358095410 | 514 | 5147 | 1004 | 866 | 464 | 24,51% | [0 108 78 265 132 519 193 30 295 163 349 217 435 319 561 397 615 459 701 585 744 665] |
| 28 | 353095910 | 171 | 1092 | 948 | 825 | 423 | 24,04% | [0 136 91 255 164 374 283 899 402 42 447 199 510 318 586 475 681 551 744 614 822 709] |
| 29 | 359095710 | 19758 | 6475 | 828 | 752 | 372 | 26,33% | [0 108 78 265 132 519 193 30 295 163 349 217 435 319 561 397 615 459 701 585 744 665] |
| 30 | 359095810 | 778 | 10399 | 821 | 752 | 336 | 24,54% | [0 108 78 198 132 295 222 42 319 156 373 253 459 343 533 400 608 483 694 570 737 664] |

# ANNEXE 7 :
# notre modèle linéaire mono-produit :
**modèle linéaire pour une production mono-produit et une configuration de base de la ligne de traitement**

**Inspiré des modèles de [Zhou et Ling, 2003], [Manier, 1994] et [Phillips et Unger, 1976]**

- *Fonction objectif :*

| | |
|---|---|
| *Minimiser* $T$ | (A_7.1) |

- *Contraintes du problème*

| | | |
|---|---|---|
| $d_i + a_{i+1} - M(1 - y_{i,i+1}) \leq t_{i+1} - t_i$ | $\forall i = 0,...,n-2$ | (A_7.2) |
| $t_{i+1} - t_i \leq d_i + b_{i+1} + M(1 - y_{i,i+1})$ | $\forall i = 0,...,n-2$ | (A_7.3) |
| $d_i + a_{i+1} - My_{i,i+1} \leq t_{i+1} - t_i + T$ | $\forall i = 0,...,n-2$ | (A_7.4) |
| $t_{i+1} - t_i + T \leq d_i + b_{i+1} + My_{i,i+1}$ | $\forall i = 0,...,n-2$ | (A_7.5) |
| $t_i \geq t_0 + d_0 + c_{s_i,s_i}$ | $\forall i = 1,...,n-1$ | (A_7.6) |
| $t_j - t_i \geq d_i + c_{s_{i+1},s_j} - M(1-y_{i,j})$ | $i \prec j$ $\forall i, j = 1,...,n-1$ | (A_7.7) |
| $t_i - t_j \geq d_j + c_{s_{j+1},s_i} - My_{i,j}$ | $i \prec j$ $\forall i, j = 1,...,n-1$ | (A_7.8) |
| $T - t_i \geq d_i + C_{s_{i+1},s_0}$ | $\forall i = 1,...,n-2$ | (A_7.9) |
| $y_{i,j} \in \{0, 1\}$ | $\forall i, j = 0,...,n-1$ | (A_7.10) |

| | | |
|---|---|---|
| $y_{0,i} = 1$ | $\forall i = 1,...,n-1$ | (A_7.11) |

| | | | |
|---|---|---|---|
| $(1-y_{i,j}) \geq y_{j,i}$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (A_7.12) |

| | | | |
|---|---|---|---|
| $y_{j,i} \geq (1-y_{i,j})$ | $i \prec j$ | $\forall i, j = 1,...,n-1$ | (A_7.13) |

| | | |
|---|---|---|
| $t_i \geq 0$ | $\forall i = 1,...,n-1$ | (A_7.14) |

| | |
|---|---|
| $t_0 = 0$ | (A_7.15) |

| | |
|---|---|
| $T \geq 0$ | (A_7.16) |

PARAMÈTRES DU MODÈLE

| Symbole | Signification |
|---|---|
| i | le numéro de l'opération de transport, i=0, …, n-1. |
| $s_i$ | la cuve de départ de l'opération de transport i. |
| $a_i$ et $b_i$ | les durées minimale et maximale de la durée de trempe dans la cuve $s_i$, spécifiée par la gamme. |
| $d_i$ | le temps nécessaire pour exécuter l'opération de transport i (de $s_i$ à $s_{i+1}$). |
| $c_{si,sj}$ | la durée constante de déplacement à vide du robot entre $s_i$ et $s_j$. |
| $t_i$ | la date de début du mouvement de transport i. |
| M | un nombre très grand (représentant la valeur $+\infty$ ) |
| $y_{i,j}$ | variable binaire traduisant l'ordre d'exécution des mouvements de transport I et j (si le mouvement i est exécute avant le mouvement j dans la séquence cyclique alors $y_{i,j}$=1. Sinon, $y_{i,j}$=0) |
| T | la période de l'ordonnacement cyclique recherché. |

# ANNEXE 8 :
# modèle référence de Phillips [Phillips et Unger, 1976] :
## modèle linéaire pour une production mono-produit, mono-robot

**Inspiré de [Phillips et Unger, 1976]**

- *Fonction objectif :*

| | |
|---|---|
| *Minimiser* $t_{max} + \sum_{i=1}^{n} (C'_{i,i+1} + C_{i+1,0}) z_i$ | (A_8.1) |

- *Contraintes du problème*

| | | | |
|---|---|---|---|
| $t_{max} \geq t_i$ | | $\forall i = 1,...,n$ | (A_8.2) |
| $t_{max} \leq t_i - (z_i - 1)M$ | | $\forall i = 1,...,n$ | (A_8.3) |
| $\sum_{i=1}^{n} z_i = 1$ | | | (A_8.4) |
| $t_j - t_i \geq C'_{i,i+1} + C_{i+1,j} - y_{i,j}M$ | $j \prec i$ | $\forall i, j = 0,...,n$ | (A_8.5) |
| $t_i - t_j \geq C'_{j,j+1} + a_{j+1} + (y_{i,j} - 1)M$ | $i = j + 1$ | $\forall i, j = 1,...,n-1$ | (A_8.6) |
| $t_i - t_j \geq C'_{j,j+1} + C_{j+1,i} + (y_{i,j} - 1)M$ | $i \succ j + 1$ | $\forall i, j = 1,...,n-1$ | (A_8.7) |
| $d = \sum_{i=1}^{n} (C'_{i,i+1} + C_{i+1,0}) z_i$ | | $\forall i = 0,...,n-1$ | (A_X.8) |
| $t_i + t_{max} + d - \left[ t_{i-1} + C'_{i-1,i} \right] \geq a_i - y_{i,i-1}M$ | | $\forall i = 1,...,n$ | (A_8.9) |

| | | |
|---|---|---|
| $t_i + t_{max} + d - \left[ t_{i-1} + C'_{i-1,i} \right] \leq b_i - y_{i,i-1}M$ | $\forall i = 1,...,n$ | (A_8.10) |

| | | |
|---|---|---|
| $t_i - \left[ t_{i-1} + C'_{i-1,i} \right] \leq b_i + (1 - y_{i,i-1})M$ | $\forall i = 1,...,n$ | (A_8.11) |

| | | |
|---|---|---|
| $t_i \geq 0$ | $\forall i = 0,...,n$ | (A_8.12) |

| | |
|---|---|
| $t_{max} \geq 0$ | (A_8.13) |

| | |
|---|---|
| $d \geq 0$ | (A_8.14) |

PARAMÈTRES DU MODÈLE

| Symbole | Signification |
|---|---|
| N | le nombre de cuves de traitements incluant la cuve de chargement et de déchargement (cuve fictive : cuve 0). |
| n | le nombre de cuves de traitements. |
| $a_i$ et $b_i$ | les durées minimales et maximales de la durée de trempe dans la cuve i. |
| $C_{i,j}=C_{j,i}$ | la durée du déplacement à vide du robot entre les cuves i et j. |
| $C_{i,i}= 0$ | |
| $C'_{i,i+1}$ | la durée de déplacement en charge du robot de la cuve i à la cuve i+1. |
| $C'_{i,j}= C'_{j,i}$ | la durée de déplacement en charge du robot de la cuve i à la cuve j est la même que la durée de déplacement en charge de la cuve j à la cuve i. |
| M | un nombre très grand (représentant la valeur $+\infty$ ) |
| $t_i$ | date de début de l'opération de transport i. |
| $t_{max}$ | la date de début du dernier movement de transport dans un cycle. |
| $y_{i,j}$ | variable binaire et où $y_{i,j}=1$, si $t_i > t_j$. Sinon, $y_{i,j}=0$. |
| $z_i$ | variable binaire et où $z_i =1$, si i est le dernier mouvement. Sinon, $z_i =0$. |

# ANNEXE 9 :
# Données et résultats des benchmarks 2-cyclique hétérogènes du tableau II.9

## ✓ Données :

| |
|---|
| Numéro du benchmark; |
| Durée de trempe minimale  Durée de trempe maximmle pour chaque traitement du premier produit [$a_{2i}$, $b_{2i}$] |
| Durée de trempe minimale  Durée de trempe maximmle pour chaque traitement du second produit [$a_{2i+1}$, $b_{2i+1}$] |
| Durée du déplacement en charge du robot entre deux cuves successives |
| Durée du déplacement à vide du robot entre deux cuves successives |

Exemples d'une ligne à 5 cuves de traitements

1
100 110  30 40  125 145  120 130  100  110
100 110  20 35  40 50  50 60  56 66
10
5

2
95  110  41 60  25  42  222  243  120 130
115 123  80  95  140 150  57  69  56  67
15
5

3
35 80  75 85  125 140  122 135  90 110
85 120  100 125  80 120  78  100  75 95
10
5

Exemples d'une ligne à 6 cuves de traitements

4
60 130  60 147  80 199  70  161  62 70  72 120
100 110  30  70  120 135  100 111  110 120  125 200
10
5

5
85 90  75 100  90 110  120 155  70 90  62 70
90 115  45  67  80  98  110 130  112 124  132 155
15
5

6
87 95  74 90  50  80  100 154  70  90  62 70
81 99  50 65  76  88  120 135  112 124  120 145
10
5

Exemples d'une ligne à 7 cuves de traitements

```
7
100 110   50 75   120 140   100 120   140  155   110 130   101 121
100 120   30 40    20  25     110 123   110 121    54  65      50  62
15
5

8
90 110    45 75   110 140   125 129   132 155   110 131   91 100
100 120   31 40    20  27     111 133   115 121   52 76      70 88
15
10

9
80 110   45 75   122 140   100 120   140  175   100  132   101 132
95 120   31 40   55  67     110 124   100  121    54  68     43 87
15
5
```

Exemples d'une ligne à 8 cuves de traitements

```
10
125 119   37 31   105 25   100 40   120 105   125 55   95 77     50 30
130 125   45 40   120 35    110 70  135 145   140 70  120 100   75  55
10
5

11
100 100   20 35   100 15   85  110   100  85    90  45   101 40   15 25
130 125   45 70   145 55  100 140   145 112   130 90   135 85   45  80
15
5

12
105  90   44 25   100 70    85 100   110 125  117 64   121 65   15 25
110 110  73 40   110 85  120 123   170 156  142 71   135 79   71 175
15
5
```

Exemples d'une ligne à 9 cuves de traitements

```
13
90 120   120 180    100 140   110  130   100  160   50  80   100 130   100 150   170  190
90 160    80 100     100 120   120  180   120  190   20  40   140 170   150 170   150  200
10
5

14
80 100   120 130  100 140   115  130    100  160   50 80   100  130   100 150   160 190
70 100    80 100   90 120   120  180    120  190   20 40   140  170   150 170    150 190
10
5
```

15
75 100 120 140 120 140 110 130 100 160 47 65 100 130 110 145 160 180
80 110 75 101 90 120 130 142 115 150 21 65 140 162 175 190 135 170
10
5

Exemples d'une ligne à 10 cuves de traitements

16
40 50 75 90 75 100 90 95 130 135 85 40 25 95 125 110 160 140 75 45
80 90 110 140 100 145 125 130 145 160 130 65 65 130 162 145 190 180 170 75
10
5

17
55 65 75 95 65 95 95 95 105 45 25 75 85 85 130 155 160 145 120 75
85 90 110 120 90 125, 125 130 185 75 50 100 95 110 190 210 195 190 185 95
10
5

18
50 45 95 85 70 90 95 90 115 55 30 70 85 75 130 150 150 145 125 190
85 70 110 100 90 125 120 120 185 75 50 100 95 110 185 215 195 190 180 225
10

✓ **Résultats :**

| N° | Nombre de traitements | Modèle Linéaire | CPU(s) | Noeuds | T | Ecart(%) | Dates de début des opérations de transport | Attentes en charge |
|---|---|---|---|---|---|---|---|---|
| 1 | 5 | (S1) | 11 | 8 | 485 | **22,68** | [0 355 120 465 170 25 305 85 435 145 60 211] | 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 5 | (S2) | 7 | 21 | 375 | | [0 245 110 355 160 15 315 125 80 210 190 286] | 0 0 0 0 0 50 0 15 0 0 0 0 |
| 2 | 5 | (S1) | 10 | 10 | 573 | **00,87** | [0 318 110 448 170 543 227 130 473 202 35 273] | 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 5 | (S2) | 8 | 162 | 568 | | [0 363 125 493 215 20 272 175 518 247 85 318] | 0 0 15 0 0 0 0 0 0 0 0 0 |
| 3 | 5 | (S1) | 9 | 9 | 340 | **05,88** | [0 200 90 295 175 65 310 155 110 265 230 30] | 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 5 | (S2) | 12 | 40 | 320 | | [0 90 70 200 155 15 290 135 115 245 225 35] | 0 0 0 0 0 0 0 0 0 5 0 0 |
| 4 | 6 | (S1) | 9 | 11 | 405 | **06,17** | [0 160 140 275 210 350 330 75 25 185 105 305 235 40] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 6 | (S2) | 8 | 358 | 380 | | [0 245 70 355 220 15 330 150 110 271 190 35 300 170] | 0 0 0 0 0 0 0 0 0 14 0 0 0 0 |
| 5 | 6 | (S1) | 11 | 70 | 650 | **28,46** | [0 135 100 265 190 342 295 455 430 580 523 57 600 220] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 6 | (S2) | 14 | 461 | 465 | | [0 320 105 425 220 20 355 125 40 250 145 377 270 60] | 0 0 0 0 10 0 0 0 0 0 40 0 0 0 |
| 6 | 6 | (S1) | 12 | 69 | 454 | **01,10** | 0 308 97 399 181 15 271 112 419 242 62 364 142 40 | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 6 | (S2) | 9 | 918 | 449 | | [0 206 97 314 181 374 259 20 389 152 40 274 127 404] | 0 0 0 0 0 0 0 0 0 0 7 0 0 0 |
| 7 | 7 | (S1) | 79 | 66 | 814 | **22,11** | [0 524 125 639 190 684 329 719 444 30 599 155 744 224 55 289] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 7 | (S2) | 73 | 669 | 634 | | [0 494 125 609 245 20 409 55 544 180 75 305 205 374 341 439] | 0 0 30 0 9 0 0 0 0 0 0 0 0 0 0 0 |
| 8 | 7 | (S1) | 80 | 41 | 583 | **05, 32** | [0 316 120 446 186 496 336 533 471 75 30 206 150 271 251 369] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 7 | (S2) | 143 | 2222 | 552 | | [0 416 111 526 231 15 351 45 486 166 76 291 196 371 306 456] | 0 0 35 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 9 | 7 | (S1) | 111 | 242 | 688 | **04,80** | [0 550 125 662 211 20 348 93 467 232 622 368 50 437 166 495] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 7 | (S2) | 135 | 1657 | 655 | | [0 454 95 564 210 610 369 25 504 150 45 265 170 334 286 399] | 0 0 25 0 4 0 0 0 6 0 0 0 0 0 0 0 |

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 10 | 8 | (S1) | 11 | 29 | 892 | **23,43** | [0 332 135 467 182 517 297 562 407 612 537 742 672 822 777 40 837 85] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 8 | (S2) | 17 | 4525 | 683 | | [0 197 132 322 172 422 282 467 392 567 522 777 647 839 752 42 813 82] | 2 0 0 50 0 0 0 0 0 55 0 0 0 0 0 0 0 0 |
| 11 | 8 | (S1) | 11 | 561 | 735 | **10,89** | [0 515 140 630 195 705 310 25 425 165 560 265 665 335 55 390 85 455] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 8 | (S2) | 15 | 4835 | 655 | | [0 512 122 627 157 22 277 52 387 177 547 297 77 357 202 412 232 452] | 0 0 0 0 0 0 0 0 0 0 40 0 0 0 0 0 0 0 |
| 12 | 8 | (S1) | 10 | 67 | 928 | **02,69** | [0 450 120 575 195 623 310 723 410 838 535 69 667 155 803 235 868 345] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 8 | (S2) | 17 | 7794 | 903 | | [0 375 120 480 195 520 320 620 420 758 545 35 677 155 813 235 843 275] | 0 0 0 0 0 0 0 0 0 9 0 34 0 0 0 0 0 0 |
| 13 | 9 | (S1) | 134 | 2421 | 705 | **12,06** | [0 480 110 630 280 15 390 125 510 255 665 405 40 435 180 585 340 55 540 215] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 9 | (S2) | 234 | 10174 | 620 | | [0 425 130 545 300 15 445 145 585 275 95 460 185 505 330 45 480 205 60 365] | 0 0 0 0 0 0 0 0 0 0 0 0 0 5 0 0 0 0 0 0 0 |
| 14 | 9 | (S1) | 170 | 1027 | 770 | **07,79** | [0 445 110 540 240 630 350 740 475 130 600 265 690 315 40 495 185 665 385 55] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 9 | (S2) | 403 | 22734 | 710 | | [0 550 105 635 265 15 375 140 515 280 670 410 40 440 165 590 305 55 475 215] | 0 0 20 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 15 | 9 | (S1) | 75 | 284 | 700 | **00,64** | [0 530 110 620 250 15 370 125 495 265 660 390 40 420 170 570 290 55 455 200] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 9 | (S2) | 580 | 36748 | 700 | | [0 539 119 629 259 24 379 134 504 274 660 399 49 429 179 579 299 64 464 209] | 9 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 16 | 10 | (S1) | 12 | 1728 | 870 | **02,87** | [0 470 50 535 150 680 260 835 360 70 500 235 600 285 635 420 770 575 100 725 195 785] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 10 | (S2) | 123 | 65621 | 845 | | [0 345 50 405 165 545 265 700 370 810 510 135 605 190 640 295 775 450 100 655 215 740] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 15 0 0 0 0 |
| 17 | 10 | (S1) | 11 | 555 | 775 | **01,29** | [0 350 65 425 155 555 240 665 375 25 525 90 585 180 690 275 110 465 290 620 485 710] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 10 | (S2) | 45 | 18218 | 765 | | [0 200 75 275 175 405 250 555 375 660 530 715 575 35 680 135 115 340 320 495 465 600] | 0 0 0 0 0 15 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 18 | 10 | (S1) | 13 | 1915 | 910 | **01,65** | [0 405 75 460 185 570 265 670 370 770 540 835 600 35 695 145 855 330 120 505 305 725] | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| | 10 | (S2) | 124 | 58738 | 895 | | [0 495 65 550 185 660 310 760 460 860 630 30 690 130 830 240 105 400 265 595 415 800] | 0 0 0 0 25 0 20 0 0 0 0 0 35 0 0 0 0 0 0 0 0 0 |

# ANNEXE 10 :

# résultats 4-périodique vs 2-périodique de la PLM des benchmarks multi-produits hétérogènes de l'annexe 5

PARAMÈTRES DES TABLEAUX DE RÉSULTATS

| Symbole | Signification |
|---|---|
| N° | le numéro de l'exemple dans la série d'exemples. |
| Référence | la référence de l'exemple d'après [Matéo, 2001]. |
| CPU(s) | la durée du temps de simulation, en secondes. |
| Nœuds | correspond au nombre de nœuds visités ou encore, au nombre de disjonctions établis. |
| $Tc_1$, $Tc_2$ | la durée minimale du cycle 1-périodique homogène. |
| T | la durée minimale du cycle 2-périodique hétérogène. |
| $T_1$ | la durée minimale du cycle 4-périodique hétérogène. |
| r, $r_1$ | degré du cycle. |
| Ecart(%) | l'écart de réduction définit par la formule suivante : $\frac{r_1 T - r T_1}{r T_1} \times 100.$ |

| N° | Référence | CPU (secondes) | Nœuds | $T_1$ | $Tc_1$ | $Tc_2$ | Ecart | Dates de début des opérations de transport du robot de l'ordonnancement 4-périodique |
|---|---|---|---|---|---|---|---|---|
| 1 | 104040705 | 90 | 43292 | 736 | 198 | 184 | 3,80 | [0 256 368 624 120 340 488 708 276 460 644 92 396 544 28 176 516 664 148 296 564 48 196 416] |
| 2 | 105040905 | 55 | 20299 | 622 | 158 | 155 | 0,64 | [0 347 189 472 103 448 261 598 206 489 364 17 278 567 417 120 393 65 537 223 513 137 41 295] |
| 3 | 109040505 | 35 | 15726 | 978 | 252 | 254 | 3,48 | [0 559 489 70 35 623 524 134 99 716 588 227 159 798 648 287 252 858 741 369 334 903 823 414] |
| 4 | 102040605 | 59 | 23593 | 637 | 185 | 144 | 3,30 | [0 208 510 400 103 294 582 486 225 338 17 527 314 424 120 599 355 544 163 65 441 41 249 139] |
| 5 | 103040805 | 178 | 50449 | 810 | 240 | 214 | 12,10 | [0 182 405 587 111 374 516 779 204 427 609 22 268 538 673 133 325 631 730 226 458 699 53 290] |
| 6 | 107070405 | 440 | 252816 | 708 | 195 | 168 | 2,54 | [0 168 560 484 140 456 680 532 224 504 112 616 308 588 196 28 365 636 280 76 404 48 328 252] |
| 7 | 109070605 | 71 | 43623 | 904 | 252 | 210 | 2,21 | [0 135 579 322 35 287 639 544 160 347 744 604 197 407 804 664 372 489 70 769 454 689 222 829] |
| 8 | 102070205 | 38 | 129890 | 642 | 185 | 158 | 6,85 | [0 127 515 316 103 292 587 429 223 405 17 532 340 446 151 604 374 549 192 65 463 41 247 168] |
| 9 | 102070705 | 40 | 13461 | 645 | 185 | 154 | 5,12 | [0 398 508 127 103 484 580 247 223 525 17 336 312 607 151 422 353 65 192 542 439 168 278 41] |
| 10 | 101070405 | 154 | 89666 | 718 | 214 | 168 | 6,41 | [0 523 254 346 49 690 318 495 139 20 366 543 282 75 459 598 386 159 563 646 423 194 618 95] |
| 11 | 205050605 | 171 | 95784 | 842 | 224 | 208 | 2,61 | [0 693 192 432 160 805 360 528 272 24 456 653 384 80 552 725 480 232 613 48 576 304 749 104] |
| 12 | 203050905 | 29 | 4787 | 520 | 140 | 130 | 3,85 | [0 415 105 210 85 500 190 320 170 65 300 430 280 150 385 20 365 260 470 130 450 345 35 240] |
| 13 | 208050705 | 55 | 15928 | 700 | 196 | 190 | 10,29 | [0 301 539 168 140 420 672 273 189 504 21 322 238 602 70 441 378 42 210 574 462 91 350 623] |
| 14 | 209050705 | 100 | 53020 | 721 | 201 | 190 | 8,46 | [0 182 546 420 150 266 693 518 203 315 25 567 287 448 115 658 371 588 224 46 469 622 343 80] |
| 15 | 204050805 | 91 | 36944 | 638 | 182 | 156 | 5,96 | [0 294 482 174 72 398 554 270 246 500 18 344 320 572 90 416 362 36 132 518 434 108 210 590] |
| 16 | 201080405 | 29 | 12695 | 988 | 346 | 204 | 11,34 | [0 582 494 88 48 652 542 158 118 802 612 308 188 848 682 354 258 918 752 424 384 218 878 712] |
| 17 | 207080605 | 279 | 104518 | 768 | 224 | 192 | 8,33 | [0 296 384 680 160 352 544 736 320 512 704 128 416 568 32 184 472 624 88 240 592 56 208 440] |
| 18 | 209080305 | 371 | 122637 | 678 | 201 | 168 | 8,85 | [0 601 339 262 140 650 479 311 283 112 622 451 367 161 28 500 416 213 77 552 524 388 182 49] |
| 19 | 205080205 | 341 | 181574 | 826 | 224 | 208 | 4,6 | [0 565 288 144 112 661 453 256 224 786 589 421 320 32 685 477 381 184 746 613 501 344 56 709] |
| 20 | 204080905 | 62 | 32242 | 646 | 182 | 156 | 4,64 | [0 286 481 220 72 397 553 262 238 499 18 334 310 580 90 415 361 36 132 517 433 108 172 598] |
| 21 | 308060205 | 82 | 44826 | 784 | 192 | 207 | 1,79 | [0 196 580 412 58 286 670 550 220 376 24 604 310 478 118 742 448 628 244 88 502 706 334 142] |
| 22 | 301060405 | > 24h | 298587 | 1188 | 314 | 354 | 12,46 | [0 450 171 549 324 36 495 117 72 270 225 369 594 1044 765 1143 918 630 1089 711 666 864 819 963 ] |
| 23 | 304060405 | > 24h | 326547 | 1152 | 409 | 354 | 32,47 | [0 396 81 531 162 36 243 117 324 198 459 279 576 972 657 1107 738 612 819 693 900 774 1035 855] |
| 24 | 303060905 | 100 | 50323 | 646 | 160 | 170 | 2,17 | [0 100 323 423 75 205 398 528 180 343 503 20 258 448 581 125 363 553 40 230 468 601 145 278] |
| 25 | 309060805 | > 24h | 501491 | 1092 | 284 | 368 | 19,41 | [0 359 104 426 311 498 458 136 48 208 168 255 546 905 650 972 857 1044 1004 682 594 754 714 801] |
| 26 | 309090705 | > 24h | 347871 | 1024 | 284 | 263 | 6,84 | [0 304 136 472 336 96 424 168 48 240 200 376 512 816 648 984 848 608 936 680 560 752 712 888] |
| 27 | 308090205 | 277 | 149949 | 772 | 192 | 186 | -2,07 | [0 160 550 352 58 250 640 442 184 376 24 574 274 472 82 670 400 598 208 724 496 694 298 106] |
| 28 | 303090905 | 102 | 51380 | 650 | 160 | 160 | -1,54 | [0 165 515 355 50 270 565 430 185 375 21 535 290 450 130 615 395 590 235 105 470 80 310 210] |
| 29 | 302090205 | 309 | 141390 | 756 | 192 | 186 | 0,00 | [0 120 456 576 90 354 546 726 252 480 696 60 378 606 30 222 504 660 192 312 630 162 282 402] |
| 30 | 304090405 | > 24h | 398521 | 1116 | 409 | 270 | 21,68 | [0 198 153 333 234 504 369 45 450 270 81 405 558 756 711 891 792 1062 927 603 1008 828 639 963] |

| Données | | PLM 2-cyclique | | | | | PLM 4-cyclique | | | | | AMELIORATION |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | r | T | T/r | Nœuds | CPU | $r_1$ | T | $T/r_1$ | Nœuds | CPU | Ecart (%) |
| 1 | 104040705 | 2 | 368 | 184 | 140 | 614 | 4 | 736 | 184 | 43292 | 690 | 0,00 |
| 2 | 105040905 | 2 | 319 | 159,5 | 97 | 541 | 4 | 622 | 155,5 | 20299 | 785 | 2,57 |
| 3 | 109040505 | 2 | 489 | 244,5 | 22 | 148 | 4 | 978 | 244,5 | 15726 | 235 | 0,00 |
| 4 | 102040605 | 2 | 347 | 173,5 | 26 | 124 | 4 | 637 | 159,25 | 23593 | 59 | **8,21** |
| 5 | 103040805 | 2 | 405 | 202,5 | 211 | 307 | 4 | 810 | 202,5 | 50449 | 178 | 0,00 |
| 6 | 107070405 | 2 | 370 | 185 | 497 | 118 | 4 | 708 | 177 | 252816 | 440 | 4,32 |
| 7 | 109070605 | 2 | 466 | 233 | 133 | 184 | 4 | 904 | 226 | 43623 | 271 | 3,00 |
| 8 | 102070205 | 2 | 333 | 166,5 | 75 | 78 | 4 | 642 | 160,5 | 129890 | 98 | 3,60 |
| 9 | 102070705 | 2 | 338 | 169 | 54 | 288 | 4 | 645 | 161,25 | 13461 | 140 | 4,59 |
| 10 | 101070405 | 2 | 373 | 186,5 | 54 | 238 | 4 | 718 | 179,5 | 89666 | 154 | 3,75 |
| 11 | 205050605 | 2 | 432 | 216 | 227 | 149 | 4 | 842 | 210,5 | 95784 | 171 | 2,55 |
| 12 | 203050905 | 2 | 260 | 130 | 65 | 274 | 4 | 520 | 130 | 411529 | 29 | 0,00 |
| 13 | 208050705 | 2 | 376 | 188 | 169 | 298 | 4 | 700 | 175 | 15928 | 55 | 6,91 |
| 14 | 209050705 | 2 | 377 | 188,5 | 119 | 306 | 4 | 721 | 180,25 | 53020 | 100 | 4,38 |
| 15 | 204050805 | 2 | 325 | 162,5 | 59 | 811 | 4 | 638 | 159,5 | 36944 | 91 | 1,85 |
| 16 | 201080405 | 2 | 494 | 247 | 170 | 173 | 4 | 988 | 247 | 12695 | 29 | 0,00 |
| 17 | 207080605 | 2 | 384 | 192 | 36 | 660 | 4 | 768 | 192 | 104518 | 279 | 0,00 |
| 18 | 209080305 | 2 | 368 | 184 | 242 | 200 | 4 | 678 | 169,5 | 122637 | 371 | 7,88 |
| 19 | 205080205 | 2 | 421 | 210,5 | 302 | 314 | 4 | 826 | 206,5 | 389475 | 341 | 1,93 |
| 20 | 204080905 | 2 | 325 | 162,5 | 59 | 240 | 4 | 646 | 161,5 | 32242 | 624 | 0,62 |
| 21 | 308060205 | 2 | 392 | 196 | 117 | 230 | 4 | 784 | 196 | 44826 | 821 | 0,00 |
| 22 | 301060405 | 2 | 594 | 297 | 158 | 914 | 4 | 1188 | 297 | 298587 | > 24h | 0,00 |
| 23 | 304060405 | 2 | 576 | 288 | 84 | 784 | 4 | 1152 | 288 | 326547 | > 24h | 0,00 |
| 24 | 303060905 | 2 | 323 | 161,5 | 139 | 937 | 4 | 646 | 161,5 | 50323 | 990 | 0,00 |
| 25 | 309060805 | 2 | 546 | 273 | 151 | 124 | 4 | 1092 | 273 | 501491 | > 24h | 0,00 |
| 26 | 309090705 | 2 | 512 | 256 | 390 | 619 | 4 | 1024 | 256 | 347871 | > 24h | 0,00 |
| 27 | 308090205 | 2 | 388 | 194 | 241 | 514 | 4 | 772 | 193 | 312497 | 977 | 0,52 |
| 28 | 303090905 | 2 | 331 | 165,5 | 174 | 171 | 4 | 650 | 162,5 | 478784 | 192 | 1,85 |
| 29 | 302090205 | 2 | 383 | 191,5 | 260 | 19758 | 4 | 756 | 189 | 457124 | 309 | 1,32 |
| 30 | 304090405 | 2 | 558 | 279 | 151 | 778 | 4 | 1116 | 279 | 398521 | > 24h | 0,00 |

# ANNEXE 11 :
# composante des algorithmes génétiques

**[Goldberg, 1989][a], [Pirlot, 2002], [Ghedira, 2007]**

Pour utiliser un algorithme génétique nous devons disposer des éléments suivants :

**1 - Codage du chromosome**

Le codage est une modélisation d'une solution d'un problème quelconque en un chromosome. Le choix du codage dépend de la spécificité du problème traité. Une solution particulière a différentes manières d'être codée selon l'alphabet utilisé. En effet, on distingue trois différents types de codages. Si l'alphabet utilisé est constituée de chiffres (resp. de lettres alphabétiques ou de symboles) alors on parle de codage numérique (resp. symbolique). Et si on utilise un alphabet combinant à la fois les lettres et les chiffres, on parle alors d'un codage alpha-numérique. Le codage est dit direct quand le passage du génotype au phénotype est immédiat ou indirect dans le cas contraire et dans ce cas le passage est assuré par l'intermédiaire d'une procédure de transition. Le codage classique correspond au codage binaire où le chromosome est représenté par un vecteur dont les éléments appartiennent à l'ensemble {0,1} comme le montre la figure suivante :




**Figure A_11.1.** Exemple d'un codage numérique classique d'un chromosome

**2 - Génération de la population initiale**

La rapidité de la convergence vers la solution optimale est conditionnée par le point d'initialisation. En effet, la connaissance à priori de solutions candidates de bonne qualité comme solution de départ accélère la convergence de l'AG. Par ailleurs, si la position de l'optimum est totalement inconnue dans l'ensemble des solutions réalisables alors dans ce cas les individus doivent être générés d'une façon aléatoire. Les tirages sont réalisés en respectant les contraintes et d'une manière uniforme dans chacun des domaines associés aux composantes de l'ensemble des solutions. Dans le but de parcourir l'ensemble des solutions le plus largement possible, des opérateurs de croisement et de mutation sont introduits afin d'entretenir la diversité d'une population non homogène au cours des générations.

**3 - Méthodes de sélection**

La sélection consiste à choisir, dans une population, les individus susceptibles de se reproduire. Plusieurs principes de sélection sont présentés dans la littérature et ils sont détaillés dans la suite de ce paragraphe.

**a. Sélection aléatoire**

Ce type de sélection se fait aléatoirement et d'une façon uniforme. Tous les individus de la population possèdent la même probabilité pour être sélectionné ($1/N$). L'inconvénient de ce type de sélection est que la convergence de l'algorithme est lente.

**b. Sélection par rang**

Il s'agit de faire correspondre à chaque individu un classement traduisant son ordre d'adaptation. Dans le cas d'un problème de minimisation, les individus sont classés par ordre décroissant des valeurs de la fonction objectif. Ainsi, l'individu ayant la plus grande valeur de la fonction objectif est l'individu le plus mauvais et on lui affectera le rang 1 (tableau A_11.1). Toutefois, dans le cas d'un problème de maximisation, les individus sont classés par ordre croissant des valeurs de la fonction objectif. A partir de cet ensemble d'individus ordonnés, on extrait une nouvelle population en utilisant les probabilités indexées sur les rangs des individus. Ces probabilités sont définies par l'équation suivante :

$$\text{Probabilité de sélection } (Parent_i) = \frac{\text{Rang } (Parent_i)}{\sum_{j \in \text{population}} \text{Rang } (Parent_j)} \qquad \text{(A_11.1)}$$

Cette procédure se caractérise par sa simplicité toutefois, le poids du meilleur élément pourrait être nettement supérieur aux autres éléments exploitables. Ce qui pourrait augmenter son rôle au détriment du reste des éléments. L'élément de rang 3, par exemple, bien qu'il soit situé dans une région d'intérêt, aura une probabilité plus faible que l'élément de rang 4, pour être sélectionné.

**Tableau A_11.1.** Sélection par rang pour un problème de minimisation

| | Chromosome | Fitness | Rang | Prob. de sélection = $\frac{Rang}{Total(Rang)}$ |
|---|---|---|---|---|
| | Parent 1 | 20 | 2 | 20% |
| | Parent 2 | 55 | 1 | 10% |
| | Parent 3 | 15 | 3 | 30% |
| | Parent 4 | 10 | 4 | 40% |
| Total | | 100 | 10 | 100% |

### c. Sélection par roulette

Il s'agit de faire correspondre à chaque individu *i* une probabilité de sélection, noté $Prob_i$, proportionnelle à la valeur de sa fonction objectif. Dans un problème de minimisation cette probabilité correspond à l'expression donnée par l'équation (A_11.2) et chaque individu a $Prob_i$ chances d'être sélectionné.

$$Prob_i = 1 - \frac{f_i}{\sum_{j \in population} f_j} \qquad \text{(A_11.2)}$$

L'individu ayant la probabilité de sélection la plus importante est considéré comme le meilleur individu et par conséquent l'individu à reproduire le plus (tableau A_11.2)

**Tableau A_11.2.** Sélection par roulette pour un problème de minimisation

| | Chromosome | Fitness | Prob. de sélection = $\frac{(1 - Prob_i)}{(N-1)}$ |
|---|---|---|---|
| | Parent 1 | 20 | 27% |
| | Parent 2 | 55 | 15% |
| | Parent 3 | 15 | 28% |
| | Parent 4 | 10 | 30% |
| Total | | 100 | 100% |

Dans le cas d'un problème de maximisation, il suffit d'utiliser $Prob_i$, comme probabilité de sélection.

### d. Sélection par tournoi

Le principe de cette méthode de sélection consiste à augmenter les chances des individus de mauvaise qualité de participer à l'amélioration de la population. En effet, cette méthode consiste à créer une sous-population de taille $M$ ($M \leq N$) prise au hasard dans la population. Le paramètre $M$ est fixé à priori par l'opérateur. L'individu, dans la sous-population, ayant la meilleure probabilité de sélection sera considéré comme vainqueur et sera sélectionné pour l'application de l'opérateur de croisement.

Deux cas particuliers sont à distinguer quant à la valeur du paramètre $M$ :

*Cas 1* : si ($M = N$) où $N$ est la taille de la population. Le résultat par la sélection de la méthode du tournoi donne à chaque fois un seul individu qui est le meilleur individu par rapport à la valeur de la fonction objectif ce qui réduit l'algorithme génétique à un algorithme de

recherche local travaillant sur une seule solution à la fois. L'inconvénient d'un tel algorithme est de converger parfois rapidement vers un optimal local.

*Cas 2* : si ($M = 1$), la méthode de sélection du tournoi correspond à la sélection aléatoire.

**4 - Opérateurs de croisement**

Le principe du croisement consiste à recombiner des gènes parentaux pour former des descendants possédant des caractéristiques issues des deux parents. Il a pour but d'enrichir la diversité de la population en manipulant les composantes des chromosomes. Classiquement, les croisements sont envisagés avec deux parents et génèrent deux enfants. Ils consistent à faire l'échange des empreintes génétiques entre les deux parents qui sont choisis d'une manière aléatoire parmi les géniteurs sélectionnés dans la population précédente pour former les nouveaux chromosomes. Ce processus est appliqué à chaque paire de chromosomes sélectionnée avec une certaine probabilité $P_{cross}$, communément appelée probabilité de croisement. Cette probabilité est fixée par l'utilisateur selon le problème considéré [Caux et al, 1995][b], [Khamphang, 1998]. Après l'utilisation de la méthode de sélection pour le choix de deux individus, nous générons un nombre aléatoire $\alpha \in [0,1]$. Et si ($\alpha \leq P_{cross}$), nous appliquons l'opérateur de croisement sur le couple. Il existe plusieurs types d'opérateurs de croisement :

**a. Croisement à un point**

C'est l'opérateur de croisement le plus simple, il consiste à diviser chacun des deux parents en deux parties à la même position, choisie au hasard. L'enfant1 est composé de la première partie du premier parent et de la deuxième partie du deuxième parent.

Pour obtenir le chromosome de l'enfant 2, il suffit d'échanger parent 1 et parent 2 (figure A_11.2) [Haouari et Daouars, 1999].

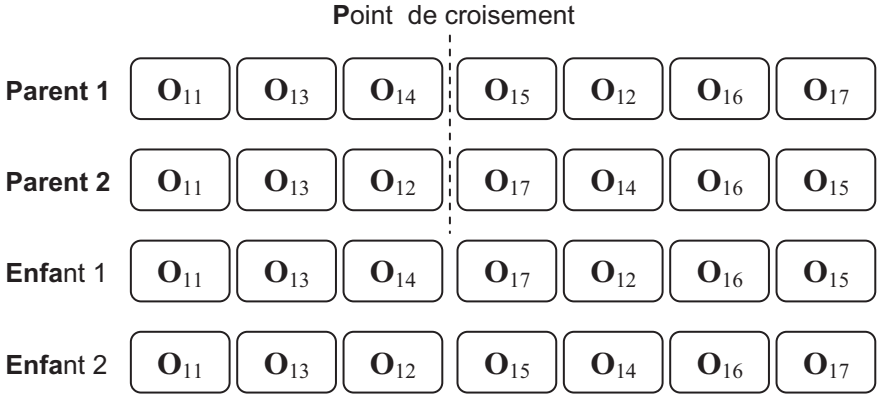



**Figure A_11.2.** Opérateur de croisement à un point

### b. Croisement multi-points

Dans ce cas, plusieurs points de coupe sont effectués dans le chromosome. L'opérateur à deux points de croisement est l'opérateur multi-points le plus répondu. Il s'agit de fixer deux points de croisement au hasard, et de prendre une section entre les points d'un parent et les autres sections en dehors des points de l'autre parent puis de les recombiner. Dans l'exemple suivant (figure A_11.3), les deux points de croisement sont placés respectivement après le deuxième et le cinquième gène [Tangour 2007].

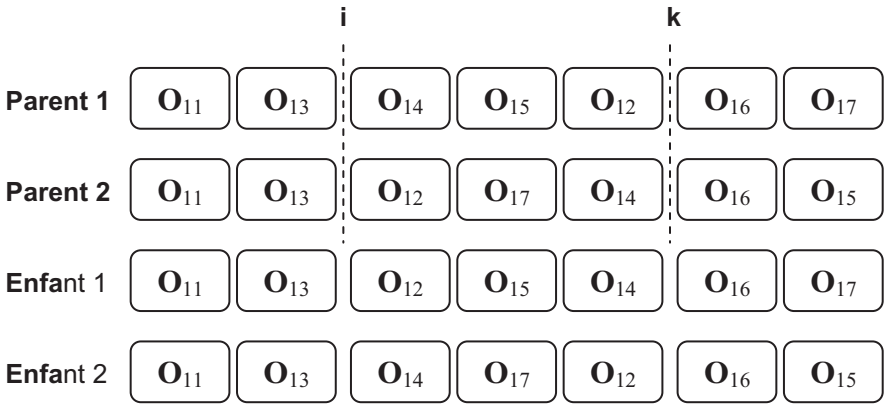



**Figure A_11.3.** Opérateur de croisement à deux points

### c. Croisement uniforme

Ce type de croisement a été proposé par Syswarda dans [Syswarda, 1989]. Il consiste à choisir avec la même probabilité un allèle de l'un ou de l'autre parent, pour transmettre sa valeur à la même position, aux enfants. L'exemple suivant (figure A_11.4) présente le masque de transmission dont le principe est le suivant : si la valeur dans le masque est une opération impaire, alors la valeur de l'allèle du parent 1 passe à l'enfant 1 (resp. du parent 2 passe à l'enfant 2) ; sinon la valeur de l'allèle du parent 1 passe à l'enfant 2 (resp. du parent 2 passe à l'enfant 1), tout en respectant la même position de l'allèle.

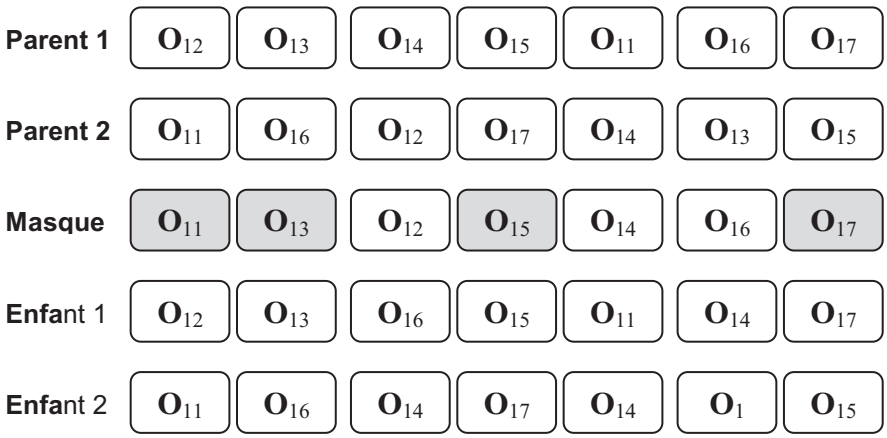



**Figure A_11.4.** Opérateur de croisement uniforme

**5 - Opérateurs de mutation**

La mutation opère sur un ou plusieurs génotypes d'un seul individu. Elle consiste à faire remplacer la valeur d'un gène par une autre appartenant au même domaine. Le rôle de cet opérateur est d'assurer la diversité de la population en empêchant ainsi l'évolution de se figer. La mutation est appliquée avec une probabilité généralement très faible $P_m$ variant de 1 à 5% [Dridi 2004]. Il existe plusieurs autres façons d'effectuer la mutation :

- Transposition de deux gènes consécutifs, c'est à dire échange de deux gènes consécutifs dans un chromosome.
- Transposition de deux gènes quelconques, ce qui correspond à l'échange de deux gènes quelconques dans un chromosome (figure A_11.5).

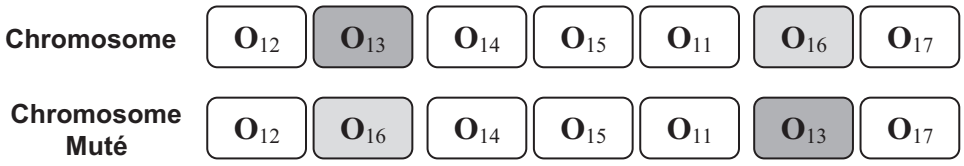



**Figure A_11.5.** Opérateur de mutation par transposition de deux gènes quelconques

**6 - Méthode d'insertion**

Une fois que l'étape de mutation est achevée, une nouvelle population doit être générée, pour cela, nous devons mettre en place une procédure d'insertion adéquate à notre problème.

Le problème auquel on se trouve confronté peut-être résumé par la question suivante : faut-il garder les enfants ou les parents ou bien un certain pourcentage des deux en conservant une taille de population ($N$) constante ? [Bourazza, 2006]

Pour assurer l'évolution de la population, nous pouvons adopter une des deux stratégies présentes dans la littérature. La première stratégie, consiste à choisir les $N$ individus à partir de $N_f$ enfants déjà créés par les opérateurs de croisement et de mutation. Généralement, on a $N_f \geq N$ et dans le cas particulier où $N_f = N$, on parle de la méthode générationnelle où les parents sont remplacés par les enfants.

Quant à la seconde stratégie, elle consiste à choisir les $N$ individus parmi la population précédente et les $N_f$ nouveaux enfants. Un cas particulier de cette stratégie, appelé méthode d'état d'équilibre, a pour principe de sauvegarder une grande partie de la population dans la génération suivante. A chaque itération quelques chromosomes (parents) ayant les meilleurs coûts seront sélectionnés afin de créer des chromosomes fils qui remplaceront les plus mauvais parents. Le reste de la population survit et sera copié dans la nouvelle génération.

L'élitisme est une stratégie complémentaire de la première stratégie. Le principe de cette stratégie consiste à copier quelques meilleurs chromosomes dans la nouvelle population, ce qui permet d'augmenter l'efficacité de l'algorithme génétique basé sur la méthode d'insertion générationnelle. Le but principal d'une telle stratégie est d'éviter que les meilleurs chromosomes soient perdus après les opérations de croisement et de mutation. Cette méthode améliore considérablement les algorithmes génétiques, car elle permet de conserver, à une itération $K$, le meilleur individu trouvé dans toutes les populations générées antérieurement.

**7 - Test d'arrêt**

La qualité des individus est jugée à partir du test d'arrêt adopté. Le but d'un tel test consiste à assurer l'optimalité de la solution finale obtenue par l'algorithme génétique. Les critères d'arrêt, répandus dans la littérature sont de deux natures [Bourazza, 2006] :

- Arrêt après un nombre de générations fixé a priori par l'opérateur. C'est la solution retenue lorsqu'une durée maximale de temps de calcul est imposée.
- Arrêt lorsque la population cesse d'évoluer ou n'évolue plus suffisamment. Et dans ce cas, nous sommes en présence d'une population homogène dont on peut penser qu'elle se situe à proximité de l'optimum.

Il est à noter qu'aucune certitude concernant la bonne convergence de l'algorithme n'est assurée. Comme dans toute procédure d'optimisation l'arrêt est arbitraire, et la solution "en temps fini" ne constitue qu'une approximation de l'optimum.

**8 - Schéma de principe**

Les étapes décrites ci-dessus, expliquent en détail le fonctionnement des différents opérateurs de l'AG appliqués aux différents types de problèmes sans aucune connaissance préalable sur le problème traité. D'une façon itérative, les processus de sélection, de croisement et de mutation, sont appliqués aux populations d'individus [Patnaik, 1994].

Partant d'une génération *i* et afin d'aboutir à la génération (*i+1*), la population est d'abord reproduite par sélection dans le but de mieux reproduire les bons individus, au sens du critère considéré, plutôt que les mauvais. Ensuite, un croisement d'une portion de la population (avec une probabilité $P_{cross}$, généralement supérieure à 0.6), est appliqué, pour produire de nouveaux enfants. Un opérateur de mutation est également appliqué (avec une probabilité $P_m$, généralement très inférieure à $P_{cross}$) à une certaine proportion de la population. Enfin, les nouveaux individus sont évalués et rajoutés à la population de la génération suivante. Cette démarche est modélisée par le schéma synoptique de la figure A_11.6 [Barnier et Brisset, 1999].

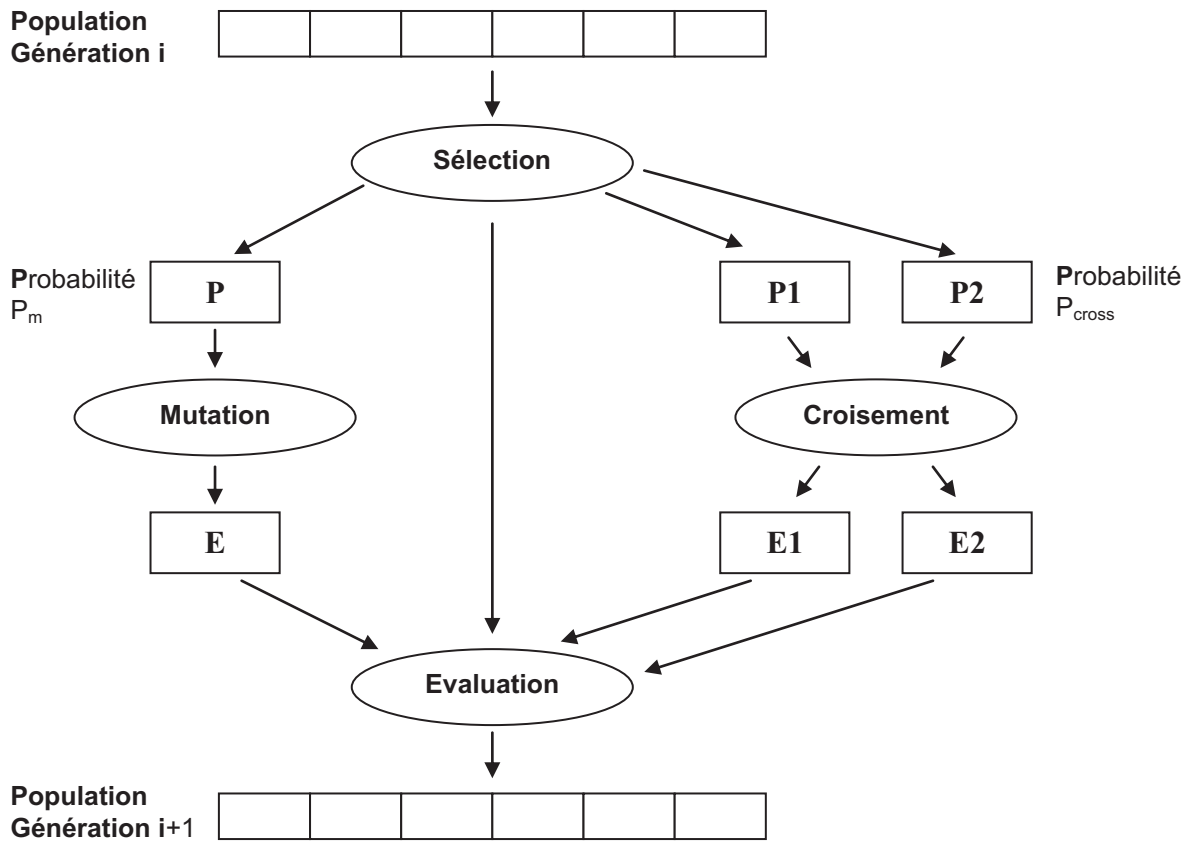



**Figure A_11.6:** Schéma synoptique de l'évolution d'un AG [Alliot et Durand, 2005]

# ANNEXE 12

## Comparaison entre la PLM, l'heuristique OCMPEST et EST sur les benchmarks 2-cyclique multi-produits hétérogènes de l'annexe 4

PARAMÈTRES DES TABLEAUX DE RÉSULTATS

| Symbole | Signification |
|---|---|
| r, $r_1$ et $r_2$ | Degré du cycle. |
| T | Période du cycle. |
| T/r, $T/r_1$ et $T/r_2$ | Temps de cycle moyen. |
| E(%) | Ecart (%) entre le temps de cycle moyen retourné par l'heuristique EST ou CMPEST et le temps de cycle optimal 2-cyclique. |
| EM(%) | Ecart Moyen (%) : la moyenne des écarts pour une largeur donnée des fenêtres de temps (FL, FM, FE) et une vitesse donnée du chariot (CR, CM, CL). |
| Ecart (%) (OCMPEST-EST) | Ecart (%) entre le temps de cycle moyen obtenu par l'heuristique OCMPEST et celui retourné par l'heuristique EST. |

| Données | | Exacte: PLM | | | Heuristique EST | | | | | Heuristique OCMPEST | | | | | AMELIORATION | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | n | T | T/n | $r_1$ | T | $T/r_1$ | E(%) | EM(%) | $r_2$ | T | $T/r_2$ | E(%) | EM(%) | Ecart (%) (OCMPEST-EST) | EM (%) |
| 1 | 104040705 | 2 | 368 | 184 | 2 | 821 | 410,5 | 123,10 | | 4 | 782 | 195,50 | 6,25 | | 52,38 | |
| 2 | 105040905 | 2 | 319 | 159,5 | 2 | 719 | 359,5 | 125,39 | | 4 | 662 | 165,50 | 3,76 | | 53,96 | |
| 3 | 109040505 | 2 | 489 | 244,5 | 2 | 740 | 370 | 51,33 | 96,55 | 2 | 489 | 244,50 | **0,00** | 8,51 | 33,92 | **43,75** |
| 4 | 102040605 | 2 | 347 | 173,5 | 2 | 657 | 328,5 | 89,34 | | 2 | 411 | 205,50 | 18,44 | | 37,44 | |
| 5 | 103040805 | 2 | 405 | 202,5 | 2 | 784 | 392 | 93,58 | | 2 | 462 | 231,00 | 14,07 | | 41,07 | |
| 6 | 107070405 | 2 | 370 | 185 | 2 | 673 | 336,5 | 81,89 | | 2 | 379 | 189,50 | 2,43 | | 43,68 | |
| 7 | 109070605 | 2 | 466 | 233 | 2 | 773 | 386,5 | 65,88 | | 2 | 500 | 250,00 | 7,30 | | 35,32 | |
| 8 | 102070205 | 2 | 333 | 166,5 | 2 | 711 | 355,5 | 113,51 | 89,93 | 2 | 412 | 206,00 | 23,72 | 9,05 | 42,05 | **41,75** |
| 9 | 102070705 | 2 | 338 | 169 | 2 | 759 | 379,5 | 124,56 | | 2 | 347 | 173,50 | 2,66 | | 54,28 | |
| 10 | 101070405 | 2 | 373 | 186,5 | 2 | 611 | 305,5 | 63,81 | | 2 | 407 | 203,50 | 9,12 | | 33,39 | |
| 11 | 205050605 | 2 | 432 | 216 | 2 | 792 | 396 | 83,33 | | 6 | 1394 | 232,33 | 7,56 | | 41,33 | |
| 12 | 203050905 | 2 | 260 | 130 | 2 | 487 | 243,5 | 87,31 | | 4 | 589 | 147,25 | 13,27 | | 39,53 | |
| 13 | 208050705 | 2 | 376 | 188 | 2 | 755 | 377,5 | 100,80 | 96,02 | 2 | 425 | 212,50 | 13,03 | 9,70 | 43,71 | **43,91** |
| 14 | 209050705 | 2 | 377 | 188,5 | 2 | 767 | 383,5 | 103,45 | | 6 | 1168 | 194,67 | 3,27 | | 49,24 | |
| 15 | 204050805 | 2 | 325 | 162,5 | 2 | 667 | 333,5 | 105,23 | | 2 | 362 | 181,00 | 11,38 | | 45,73 | |
| 16 | 201080405 | 2 | 494 | 247 | 2 | 730 | 365 | 47,77 | | 2 | 520 | 260,00 | 5.26 | | 38,08 | |
| 17 | 207080605 | 2 | 384 | 192 | 2 | 706 | 353 | 83,85 | | 2 | 452 | 226,00 | 17,71 | | 35,98 | |
| 18 | 209080305 | 2 | 368 | 184 | 2 | 794 | 397 | 115,76 | 88,26 | 8 | 1548 | 193,50 | 5,16 | 9,91 | 51,26 | **42,57** |
| 19 | 205080205 | 2 | 421 | 210,5 | 2 | 841 | 420,5 | 99,76 | | 4 | 924 | 231,00 | 9,74 | | 45,07 | |
| 20 | 204080905 | 2 | 325 | 162,5 | 2 | 631 | 315,5 | 94,15 | | 2 | 363 | 181,50 | 11,69 | | 42,47 | |
| 21 | 308060205 | 2 | 392 | 196 | 2 | 764 | 382 | 94,90 | | 2 | 463 | 231,50 | 18,11 | | 39,40 | |
| 22 | 301060405 | 2 | 594 | 297 | 2 | 1013 | 506,5 | 70,54 | | 2 | 657 | 328,50 | 10,61 | | 35,14 | |
| 23 | 304060405 | 2 | 576 | 288 | 2 | 898 | 449 | 55,90 | 83,05 | 2 | 593 | 296,50 | 2,95 | 10,09 | 33,96 | **38,71** |
| 24 | 303060905 | 2 | 323 | 161,5 | 2 | 753 | 376,5 | 133,13 | | 4 | 707 | 176,75 | 9,44 | | 53,05 | |
| 25 | 309060805 | 2 | 546 | 273 | 2 | 878 | 439 | 60,81 | | 2 | 597 | 298,50 | 9,34 | | 32,00 | |
| 26 | 309090705 | 2 | 512 | 256 | 2 | 883 | 441,5 | 72,46 | | 2 | 580 | 290,00 | 13,28 | | 34,31 | |
| 27 | 308090205 | 2 | 388 | 194 | 2 | 787 | 393,5 | 102,84 | | 4 | 841 | 210,25 | 8,38 | | 46,57 | |
| 28 | 303090905 | 2 | 331 | 165,5 | 2 | 793 | 396,5 | 139,58 | 102,13 | 6 | 1029 | 171,50 | 3,63 | 7,67 | 56,75 | **45,76** |
| 29 | 302090205 | 2 | 383 | 191,5 | 2 | 826 | 413 | 115,67 | | 6 | 1188 | 198,00 | 3,39 | | 52,06 | |
| 30 | 304090405 | 2 | 558 | 279 | 2 | 1005 | 502,5 | 80,11 | | 2 | 612 | 306,00 | 9,68 | | 39,10 | |

| Données | | Exacte: PLM | | | Heuristique EST | | | | | Heuristique OCMPEST | | | | | AMELIORATION | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | r | T | T/r | $r_1$ | T | $T/r_1$ | E (%) | EM(%) | $r_2$ | T | $T/r_2$ | E (%) | EM(%) | ecart (%) (OCMPEST-EST) | EM (%) |
| 1 | 111041606 | 2 | 450 | 225 | 2 | 983 | 491,5 | 118,44 | | 8 | 2118 | 264,75 | 17,67 | | 46,13 | |
| 2 | 111042006 | 2 | 440 | 220 | 2 | 1045 | 522,5 | 137,50 | | 2 | 539 | 269,50 | 22,50 | | 48,42 | |
| 3 | 116041806 | 2 | 337 | 168,5 | 2 | 822 | 411 | 143,92 | 124,06 | 2 | 399 | 199,50 | 18,40 | 19,52 | 51,46 | **46,48** |
| 4 | 116041706 | 2 | 336 | 168 | 2 | 715 | 357,5 | 112,80 | | 2 | 435 | 217,50 | 29,46 | | 39,16 | |
| 5 | 116041706 | 2 | 366 | 183 | 2 | 760 | 380 | 107,65 | | 2 | 401 | 200,50 | 9,56 | | 47,24 | |
| 6 | 120072006 | 2 | 269 | 134,5 | 2 | 745 | 372,5 | 176,95 | | 6 | 918 | 153,00 | 13,75 | | 58,93 | |
| 7 | 116071306 | 2 | 316 | 158 | 2 | 474 | 237 | 50,00 | | 4 | 682 | 170,50 | 7,91 | | 28,06 | |
| 8 | 119071306 | 2 | 332 | 166 | 2 | 743 | 371,5 | 123,80 | 103,75 | 4 | 796 | 199,00 | 19,88 | 9,67 | 46,43 | **43,70** |
| 9 | 112071606 | 2 | 525 | 262,5 | 2 | 850 | 425 | 61,90 | | 2 | 578 | 289,00 | 10,10 | | 32,00 | |
| 10 | 117071706 | 2 | 362 | 181 | 2 | 746 | 373 | 106,08 | | 4 | 700 | 175,00 | **-3,31** | | 53,08 | |
| 11 | 213051106 | 2 | 461 | 230,5 | 2 | 786 | 393 | 70,50 | | 2 | 501 | 250,50 | 8,68 | | 36,26 | |
| 12 | 217051806 | 2 | 596 | 298 | 2 | 1101 | 550,5 | 84,73 | | 8 | 2513 | 314,13 | 5,41 | | 42,94 | |
| 13 | 215051806 | 2 | 621 | 310,5 | 2 | 1062 | 531 | 71,01 | 67,55 | 4 | 1316 | 329,00 | 5,96 | 6,12 | 38,04 | **36,41** |
| 14 | 211051406 | 2 | 606 | 303 | 2 | 942 | 471 | 55,45 | | 2 | 659 | 329,50 | 8,75 | | 30,04 | |
| 15 | 211051806 | 2 | 662 | 331 | 2 | 1033 | 516,5 | 56,04 | | 2 | 674 | 337,00 | 1,81 | | 34,75 | |
| 16 | 218081906 | 2 | 499 | 249,5 | 2 | 986 | 493 | 97,60 | | 2 | 535 | 267,50 | 7,21 | | 45,74 | |
| 17 | 217081606 | 2 | 581 | 290,5 | 2 | 1006 | 503 | 73,15 | | 4 | 1262 | 315,50 | 8,61 | | 37,28 | |
| 18 | 219081406 | 2 | 420 | 210 | 2 | 939 | 469,5 | 123,57 | 88,56 | 10 | 2332 | 233,20 | 11,05 | 7,93 | 50,33 | **42,17** |
| 19 | 215081106 | 2 | 567 | 283,5 | 2 | 1056 | 528 | 86,24 | | 2 | 618 | 309,00 | 8,99 | | 41,48 | |
| 20 | 216081606 | 2 | 583 | 291,5 | 2 | 946 | 473 | 62,26 | | 2 | 605 | 302,50 | 3,77 | | 36,05 | |
| 21 | 314061406 | 2 | 739 | 369,5 | 2 | 1109 | 554,5 | 50,07 | | 2 | 791 | 395,50 | 7,04 | | 28,67 | |
| 22 | 318061506 | 2 | 510 | 255 | 2 | 837 | 418,5 | 64,12 | | 6 | 1551 | 258,50 | 1,37 | | 38,23 | |
| 23 | 316061906 | 2 | 548 | 274 | 2 | 823 | 411,5 | 50,18 | 54,19 | 2 | 602 | 301,00 | 9,85 | 5,73 | 26,85 | **31,31** |
| 24 | 311062006 | 2 | 653 | 326,5 | 2 | 1012 | 506 | 54,98 | | 2 | 700 | 350,00 | 7,20 | | 30,83 | |
| 25 | 319061906 | 2 | 568 | 284 | 2 | 861 | 430,5 | 51,58 | | 2 | 586 | 293,00 | 3,17 | | 31,94 | |
| 26 | 315091306 | 2 | 586 | 293 | 2 | 999 | 499,5 | 70,48 | | 2 | 636 | 318,00 | 8,53 | | 36,34 | |
| 27 | 318091606 | 2 | 477 | 238,5 | 2 | 901 | 450,5 | 88,89 | | 6 | 1441 | 240,17 | 0,70 | | 46,69 | |
| 28 | 311091306 | 2 | 632 | 316 | 2 | 1018 | 509 | 61,08 | 80,90 | 2 | 632 | 316,00 | **0,00** | 5,93 | 37,92 | **40,03** |
| 29 | 316091906 | 2 | 525 | 262,5 | 2 | 788 | 394 | 50,10 | | 2 | 584 | 292,00 | 11,24 | | 25,89 | |
| 30 | 312091706 | 2 | 386 | 193 | 2 | 903 | 451,5 | 133,94 | | 8 | 1686 | 210,75 | 9,20 | | 53,32 | |

| Données | | Exacte : PLM | | | Heuristique EST | | | | | Heuristique OCMPEST | | | | | AMELIORATION | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | r | T | T/r | $r_1$ | T | $T/r_1$ | E(%) | EM(%) | $r_2$ | T | $T/r_2$ | E (%) | EM(%) | ecart (%) (OCMPEST-EST) | EM (%) |
| 1 | 130042607 | 2 | 446 | 223 | 2 | 1106 | 553 | 147,98 | | 4 | 1042 | 260,50 | 16,82 | | 52,89 | |
| 2 | 125043007 | 2 | 653 | 326,5 | 2 | 1115 | 557,5 | 70,75 | | 6 | 1868 | 311,33 | **-4,65** | | 44,16 | |
| 3 | 127042807 | 2 | 690 | 345 | 2 | 1072 | 536 | 55,36 | 104,96 | 2 | 763 | 381,50 | 10,58 | 7,69 | 28,82 | **42,17** |
| 4 | 128042407 | 2 | 408 | 204 | 2 | 907 | 453,5 | 122,30 | | 6 | 1361 | 226,83 | 11,19 | | 49,98 | |
| 5 | 126043007 | 2 | 622 | 311 | 2 | 1000 | 500 | 60,77 | | 2 | 650 | 325,00 | 4,50 | | 35,00 | |
| 6 | 130072707 | 2 | 418 | 209 | 2 | 1036 | 518 | 147,85 | | 4 | 1008 | 252,00 | 20,57 | | 51,35 | |
| 7 | 130072707 | 2 | 418 | 209 | 2 | 1036 | 518 | 147,85 | | 4 | 1008 | 252,00 | 20,57 | | 51,35 | |
| 8 | 126073007 | 2 | 598 | 299 | 2 | 953 | 476,5 | 59,36 | 113,26 | 2 | 686 | 343,00 | 14,72 | 16,72 | 28,02 | **42,77** |
| 9 | 126073007 | 2 | 598 | 299 | 2 | 953 | 476,5 | 59,36 | | 2 | 686 | 343,00 | 14,72 | | 28,02 | |
| 10 | 130072507 | 2 | 426 | 213 | 2 | 1073 | 536,5 | 151,88 | | 4 | 963 | 240,75 | 13,03 | | 55,13 | |
| 11 | 228052607 | 2 | 558 | 279 | 2 | 1131 | 565,5 | 102,69 | | 2 | 576 | 288,00 | 3,23 | | 49,07 | |
| 12 | 228052607 | 2 | 558 | 279 | 2 | 1131 | 565,5 | 102,69 | | 2 | 576 | 288,00 | 3,23 | | 49,07 | |
| 13 | 223052407 | 2 | 635 | 317,5 | 2 | 1040 | 520 | 63,78 | 102,05 | 2 | 635 | 317,50 | **0,00** | 3,55 | 38,94 | **46,99** |
| 14 | 228052907 | 2 | 578 | 289 | 2 | 942 | 471 | 62,98 | | 4 | 1190 | 297,50 | 2,94 | | 36,84 | |
| 15 | 226052507 | 2 | 370 | 185 | 2 | 1029 | 514,5 | 178,11 | | 2 | 401 | 200,50 | 8,38 | | 61,03 | |
| 16 | 226082807 | 2 | 360 | 180 | 2 | 928 | 464 | 157,78 | | 4 | 761 | 190,25 | 5,69 | | 59,00 | |
| 17 | 223082207 | 2 | 579 | 289,5 | 2 | 1048 | 524 | 81,00 | | 2 | 681 | 340,50 | 17,62 | | 35,02 | |
| 18 | 223082507 | 2 | 600 | 300 | 2 | 1042 | 521 | 73,67 | 104,83 | 2 | 695 | 347,50 | 15,83 | 10,59 | 33,30 | **44,45** |
| 19 | 227082707 | 2 | 452 | 226 | 2 | 1019 | 509,5 | 125,44 | | 4 | 939 | 234,75 | 3,87 | | 53,93 | |
| 20 | 228082607 | 2 | 539 | 269,5 | 2 | 1004 | 502 | 86,27 | | 4 | 1185 | 296,25 | 9,93 | | 40,99 | |
| 21 | 325062907 | 2 | 775 | 387,5 | 2 | 1198 | 599 | 54,58 | | 2 | 802 | 401,00 | 3,48 | | 33,06 | |
| 22 | 330062207 | 2 | 460 | 230 | 2 | 1051 | 525,5 | 128,48 | | 6 | 1451 | 241,83 | 5,14 | | 53,98 | |
| 23 | 328062307 | 2 | 788 | 394 | 2 | 1267 | 633,5 | 60,79 | 74,99 | 2 | 803 | 401,50 | 1,90 | 4,24 | 36,62 | **39,26** |
| 24 | 328062707 | 2 | 784 | 392 | 2 | 1314 | 657 | 67,60 | | 2 | 784 | 392,00 | **0,00** | | 40,33 | |
| 25 | 329062707 | 2 | 778 | 389 | 2 | 1272 | 636 | 63,50 | | 2 | 861 | 430,50 | 10,67 | | 32,31 | |
| 26 | 323092507 | 2 | 928 | 464 | 2 | 1302 | 651 | 40,30 | | 2 | 928 | 464,00 | **0,00** | | 28,73 | |
| 27 | 326092807 | 2 | 658 | 329 | 2 | 1199 | 599,5 | 82,22 | | 2 | 709 | 354,50 | 7,75 | | 40,87 | |
| 28 | 325092707 | 2 | 741 | 370,5 | 2 | 1202 | 601 | 62,21 | 79,15 | 2 | 869 | 434,50 | 17,27 | 8,13 | 27,70 | **37,86** |
| 29 | 328092907 | 2 | 724 | 362 | 2 | 1250 | 625 | 72,65 | | 4 | 1587 | 396,75 | 9,60 | | 36,52 | |
| 30 | 327092307 | 2 | 438 | 219 | 2 | 1044 | 522 | 138,36 | | 4 | 929 | 232,25 | 6,05 | | 55,51 | |

| Données | | Exacte : PLM | | | Heuristique EST | | | | | Heuristique OCMPEST | | | | | AMELIORATION | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | r | T | T/r | $r_1$ | T | $T/r_1$ | E(%) | EM(%) | $r_2$ | T | $T/r_2$ | E (%) | EM(%) | ecart (%) (OCMPEST-EST) | EM (%) |
| 1 | 140043108 | 2 | 585 | 292,5 | 2 | 1077 | 538,5 | 84,10 | | 6 | 1694 | 282,33 | **-3,48** | | 47,57 | |
| 2 | 139044008 | 2 | 412 | 206 | 2 | 945 | 472,5 | 129,37 | | 6 | 1442 | 240,33 | 16,67 | | 49,14 | |
| 3 | 140043108 | 2 | 585 | 292,5 | 2 | 1077 | 538,5 | 84,10 | 96,82 | 6 | 1694 | 282,33 | **-3,48** | 5,77 | 47,57 | **39,13** |
| 4 | 133043108 | 2 | 605 | 302,5 | 2 | 994 | 497 | 64,30 | | 4 | 1305 | 326,25 | 7,85 | | 34,36 | |
| 5 | 138043708 | 2 | 390 | 195 | 2 | 944 | 472 | 142,05 | | 4 | 868 | 217,00 | 11,28 | | 54,03 | |
| 6 | 137073308 | 2 | 583 | 291,5 | 2 | 1157 | 578,5 | 98,46 | | 10 | 2858 | 285,80 | **-1,96** | | 50,60 | |
| 7 | 134073908 | 2 | 645 | 322,5 | 2 | 1253 | 626,5 | 94,26 | | 2 | 723 | 361,50 | 12,09 | | 42,30 | |
| 8 | 136073908 | 2 | 680 | 340 | 2 | 1259 | 629,5 | 85,15 | 91,73 | 2 | 755 | 377,50 | 11,03 | 6,81 | 40,03 | **33,88** |
| 9 | 131073308 | 2 | 579 | 289,5 | 2 | 1174 | 587 | 102,76 | | 10 | 2808 | 280,80 | **-3,01** | | 52,16 | |
| 10 | 131073608 | 2 | 573 | 286,5 | 2 | 1020 | 510 | 78,01 | | 2 | 664 | 332,00 | 15,88 | | 34,90 | |
| 11 | 237053208 | 2 | 582 | 291 | 2 | 1107 | 553,5 | 90,21 | | 10 | 2757 | 275,70 | **-5,26** | | 50,19 | |
| 12 | 236053208 | 2 | 563 | 281,5 | 2 | 1071 | 535,5 | 90,23 | | 2 | 666 | 333,00 | 18,29 | | 37,82 | |
| 13 | 234053408 | 2 | 776 | 388 | 2 | 1297 | 648,5 | 67,14 | 80,77 | 2 | 841 | 420,50 | 8,38 | 5,67 | 35,16 | **30,98** |
| 14 | 232053108 | 2 | 676 | 338 | 2 | 1073 | 536,5 | 58,73 | | 2 | 727 | 363,50 | 7,54 | | 32,25 | |
| 15 | 239053208 | 2 | 570 | 285 | 2 | 1126 | 563 | 97,54 | | 4 | 1133 | 283,25 | **-0,61** | | 49,69 | |
| 16 | 232083608 | 2 | 627 | 313,5 | 2 | 1138 | 569 | 81,50 | | 2 | 641 | 320,50 | 2,23 | | 43,67 | |
| 17 | 233083908 | 2 | 805 | 402,5 | 2 | 1365 | 682,5 | 69,57 | | 2 | 850 | 425,00 | 5,59 | | 37,73 | |
| 18 | 240083108 | 2 | 490 | 245 | 2 | 1148 | 574 | 134,29 | 100,25 | 4 | 955 | 238,75 | **-2,55** | 7,37 | 58,41 | **36,65** |
| 19 | 240083808 | 2 | 410 | 205 | 2 | 983 | 491,5 | 139,76 | | 4 | 1011 | 252,75 | 23,29 | | 48,58 | |
| 20 | 231083608 | 2 | 712 | 356 | 2 | 1254 | 627 | 76,12 | | 2 | 771 | 385,50 | 8,29 | | 38,52 | |
| 21 | 335063308 | 2 | 1065 | 532,5 | 2 | 1529 | 764,5 | 43,57 | | 2 | 1157 | 578,50 | 8,64 | | 24,33 | |
| 22 | 335063708 | 2 | 1066 | 533 | 2 | 1536 | 768 | 44,09 | | 2 | 1146 | 573,00 | 7,50 | | 25,39 | |
| 23 | 337063608 | 2 | 831 | 415,5 | 2 | 1321 | 660,5 | 58,97 | 52,41 | 2 | 883 | 441,50 | 6,26 | 8,31 | 33,16 | **23,94** |
| 24 | 336063608 | 2 | 922 | 461 | 2 | 1437 | 718,5 | 55,86 | | 2 | 922 | 461,00 | **0,00** | | 35,84 | |
| 25 | 339063908 | 2 | 809 | 404,5 | 2 | 1291 | 645,5 | 59,58 | | 2 | 964 | 482,00 | 19,16 | | 25,33 | |
| 26 | 336094008 | 2 | 878 | 439 | 2 | 1466 | 733 | 66,97 | | 2 | 980 | 490,00 | 11,62 | | 33,15 | |
| 27 | 340093208 | 2 | 515 | 257,5 | 2 | 1190 | 595 | 131,07 | | 4 | 1069 | 267,25 | 3,79 | | 55,08 | |
| 28 | 335093508 | 2 | 980 | 490 | 2 | 1490 | 745 | 52,04 | 81,55 | 2 | 1075 | 537,50 | 9,69 | 6,65 | 27,85 | **32,81** |
| 29 | 335093708 | 2 | 972 | 486 | 2 | 1475 | 737,5 | 51,75 | | 2 | 999 | 499,50 | 2,78 | | 32,27 | |
| 30 | 338093908 | 2 | 627 | 313,5 | 2 | 1291 | 645,5 | 105,90 | | 6 | 1982 | 330,33 | 5,37 | | 48,83 | |

| Données | | Exacte : PLM | | | Heuristique EST | | | | | Heuristique OCMPEST | | | | | AMELIORATION | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | r | T | T/r | $r_1$ | T | $T/r_1$ | E(%) | EM(%) | $r_2$ | T | $T/r_2$ | E (%) | EM(%) | ecart (%) (OCMPEST-EST) | EM (%) |
| 1 | 144044509 | 2 | 794 | 397 | 2 | 1273 | 636,5 | 60,33 | | 2 | 837 | 418,50 | 5,42 | | 34,25 | |
| 2 | 142044509 | 2 | 799 | 399,5 | 2 | 1327 | 663,5 | 66,08 | | 2 | 820 | 410,00 | 2,63 | | 38,21 | |
| 3 | --- | 2 | --- | --- | --- | --- | --- | --- | 66,42 | --- | --- | --- | --- | 4,38 | --- | **37,21** |
| 4 | 146044509 | 2 | 793 | 396,5 | 2 | 1322 | 661 | 66,71 | | 2 | 868 | 434,00 | 9,46 | | 34,34 | |
| 5 | 148044709 | 2 | 692 | 346 | 2 | 1194 | 597 | 72,54 | | 6 | 2076 | 346,00 | **0,00** | | 42,04 | |
| 6 | --- | 2 | --- | --- | --- | --- | --- | --- | | --- | --- | --- | --- | | --- | |
| 7 | 142074509 | 2 | 773 | 386,5 | 2 | 1344 | 672 | 73,87 | | 2 | 894 | 447,00 | 15,65 | | 33,48 | |
| 8 | 141074409 | 2 | 569 | 284,5 | 2 | 1122 | 561 | 97,19 | 87,88 | 2 | 733 | 366,50 | 28,82 | 16,15 | 34,67 | **38,10** |
| 9 | 146074509 | 2 | 723 | 361,5 | 2 | 1339 | 669,5 | 85,20 | | 6 | 2340 | 390,00 | 7,88 | | 41,75 | |
| 10 | 141074609 | 2 | 568 | 284 | 2 | 1109 | 554,5 | 95,25 | | 4 | 1275 | 318,75 | 12,24 | | 42,52 | |
| 11 | 245054909 | 2 | 793 | 396,5 | 2 | 1240 | 620 | 56,37 | | 2 | 889 | 444,50 | 12,11 | | 28,31 | |
| 12 | 247054709 | 2 | 697 | 348,5 | 2 | 1279 | 639,5 | 83,50 | | 6 | 2023 | 337,17 | **-3,25** | | 47,28 | |
| 13 | 242054709 | 2 | 712 | 356 | 2 | 1308 | 654 | 83,71 | 86,33 | 4 | 1234 | 308,50 | **-13,34** | 2,72 | 52,83 | **38,24** |
| 14 | 246054609 | 2 | 852 | 426 | 2 | 1378 | 689 | 61,74 | | 2 | 903 | 451,50 | 5,99 | | 34,47 | |
| 15 | 245054909 | 2 | 793 | 396,5 | 2 | 1240 | 620 | 56,37 | | 2 | 889 | 444,50 | 12,11 | | 28,31 | |
| 16 | 243084309 | 2 | 786 | 393 | 2 | 1373 | 686,5 | 74,68 | | 2 | 834 | 417,00 | 6,11 | | 39,26 | |
| 17 | 249084509 | 2 | 929 | 464,5 | 2 | 1432 | 716 | 54,14 | | 2 | 929 | 464,50 | **0,00** | | 35,13 | |
| 18 | 244084409 | 2 | 759 | 379,5 | 2 | 1317 | 658,5 | 73,52 | 82,42 | 2 | 799 | 399,50 | 5,27 | 6,11 | 39,33 | **40,88** |
| 19 | 244084309 | 2 | 768 | 384 | 2 | 1374 | 687 | 78,91 | | 2 | 877 | 438,50 | 14,19 | | 36,17 | |
| 20 | 242084909 | 2 | 600 | 300 | 2 | 1385 | 692,5 | 130,83 | | 4 | 1260 | 315,00 | 5,00 | | 54,51 | |
| 21 | 343064609 | 2 | 991 | 495,5 | 2 | 1557 | 778,5 | 57,11 | | 2 | 991 | 495,50 | **0,00** | | 36,35 | |
| 22 | 343064609 | 2 | 1026 | 513 | 2 | 1692 | 846 | 64,91 | | 2 | 1052 | 526,00 | 2,53 | | 37,83 | |
| 23 | 346064309 | 2 | 919 | 459,5 | 2 | 1472 | 736 | 60,17 | 70,82 | 2 | 932 | 466,00 | 1,41 | 3,62 | 36,68 | **38,84** |
| 24 | 349064709 | 2 | 628 | 314 | 2 | 1343 | 671,5 | 113,85 | | 8 | 2868 | 358,50 | 14,17 | | 46,61 | |
| 25 | 343065009 | 2 | 1056 | 528 | 2 | 1669 | 834,5 | 58,05 | | 2 | 1056 | 528,00 | **0,00** | | 36,73 | |
| 26 | 350094809 | 2 | 987 | 493,5 | 2 | 1643 | 821,5 | 66,46 | | 4 | 2219 | 554,75 | 12,41 | | 32,47 | |
| 27 | 348094609 | 2 | 849 | 424,5 | 2 | 1496 | 748 | 76,21 | | 2 | 967 | 483,50 | 13,90 | | 35,36 | |
| 28 | 344094909 | 2 | 732 | 366 | 2 | 1416 | 708 | 93,44 | 76,69 | 10 | 3793 | 379,30 | 3,63 | 7,95 | 46,43 | **38,70** |
| 29 | 348095009 | 2 | 847 | 423,5 | 2 | 1536 | 768 | 81,35 | | 2 | 930 | 465,00 | 9,80 | | 39,45 | |
| 30 | 343094409 | 2 | 971 | 485,5 | 2 | 1612 | 806 | 66,01 | | 2 | 971 | 485,50 | **0,00** | | 39,76 | |

| Données | | Exacte : PLM | | | Heuristique EST | | | | | Heuristique OCMPEST | | | | | AMELIORATION | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | r | T | T/r | $r_1$ | T | $T/r_1$ | E(%) | EM(%) | $r_2$ | T | $T/r_2$ | E (%) | EM(%) | ecart (%) (OCMPEST-EST) | EM (%) |
| 1 | 159045910 | 2 | 737 | 368,5 | 2 | 1324 | 662 | 79,65 | | 4 | 1256 | 314,00 | **-14,79** | | 52,57 | |
| 2 | 152045310 | 2 | 848 | 424 | 2 | 1489 | 744,5 | 75,59 | | 6 | 2694 | 449,00 | 5,90 | | 39,69 | |
| 3 | 156045610 | 2 | 875 | 437,5 | 2 | 1437 | 718,5 | 64,23 | 69,97 | 2 | 975 | 487,50 | 11,43 | 4,65 | 32,15 | **38,19** |
| 4 | 156046010 | 2 | 918 | 459 | 2 | 1466 | 733 | 59,69 | | 2 | 949 | 474,50 | 3,38 | | 35,27 | |
| 5 | 155045910 | 2 | 761 | 380,5 | 2 | 1299 | 649,5 | 70,70 | | 2 | 893 | 446,50 | 17,35 | | 31,25 | |
| 6 | 159075810 | 2 | 652 | 326 | 2 | 1300 | 650 | 99,39 | | 4 | 1173 | 293,25 | **-10,05** | | 54,88 | |
| 7 | 158075710 | 2 | 910 | 455 | 2 | 1553 | 776,5 | 70,66 | | 2 | 959 | 479,50 | 5,38 | | 38,25 | |
| 8 | 152075210 | 2 | 781 | 390,5 | 2 | 1392 | 696 | 78,23 | 102,45 | 4 | 1425 | 356,25 | **-8,77** | 4,43 | 48,81 | **46,63** |
| 9 | 157075110 | 2 | 456 | 228 | 2 | 1345 | 672,5 | 194,96 | | 4 | 1105 | 276,25 | 21,16 | | 58,92 | |
| 10 | 156075910 | 2 | 797 | 398,5 | 2 | 1347 | 673,5 | 69,01 | | 2 | 912 | 456,00 | 14,43 | | 32,29 | |
| 11 | 252055710 | 2 | 952 | 476 | 2 | 1549 | 774,5 | 62,71 | | 2 | 995 | 497,50 | 4,52 | | 35,77 | |
| 12 | 257055410 | 2 | 1032 | 516 | 2 | 1786 | 893 | 73,06 | | 2 | 1059 | 529,50 | 2,62 | | 40,71 | |
| 13 | 256055910 | 2 | 781 | 390,5 | 2 | 1503 | 751,5 | 92,45 | 71,40 | 2 | 997 | 498,50 | 27,66 | 8,20 | 33,67 | **36,78** |
| 14 | 260055210 | 2 | 911 | 455,5 | 2 | 1569 | 784,5 | 72,23 | | 6 | 2585 | 430,83 | **-5,42** | | 45,08 | |
| 15 | 255055310 | 2 | 1056 | 528 | 2 | 1653 | 826,5 | 56,53 | | 2 | 1179 | 589,50 | 11,65 | | 28,68 | |
| 16 | 256085510 | 2 | 756 | 378 | 2 | 1410 | 705 | 86,51 | | 6 | 2128 | 354,67 | **-6,17** | | 49,69 | |
| 17 | 259085210 | 2 | 632 | 316 | 2 | 1256 | 628 | 98,73 | | 4 | 1199 | 299,75 | **-5,14** | | 52,27 | |
| 18 | 260085910 | 2 | 805 | 402,5 | 2 | 1578 | 789 | 96,02 | 79,71 | 2 | 983 | 491,50 | 22,11 | 5,57 | 37,71 | **40,55** |
| 19 | 255085810 | 2 | 1038 | 519 | 2 | 1631 | 815,5 | 57,13 | | 2 | 1215 | 607,50 | 17,05 | | 25,51 | |
| 20 | 254086010 | 2 | 916 | 458 | 2 | 1467 | 733,5 | 60,15 | | 2 | 916 | 458,00 | **0,00** | | 37,56 | |
| 21 | 359065910 | 2 | 911 | 455,5 | 2 | 1484 | 742 | 62,90 | | 2 | 928 | 464,00 | 1,87 | | 37,47 | |
| 22 | 355065710 | 2 | 1106 | 553 | 2 | 1766 | 883 | 59,67 | | 2 | 1106 | 553,00 | **0,00** | | 37,37 | |
| 23 | 356065510 | 2 | 1005 | 502,5 | 2 | 1607 | 803,5 | 59,90 | 61,53 | 2 | 1005 | 502,50 | **0,00** | 2,47 | 37,46 | **36,56** |
| 24 | 358065310 | 2 | 1062 | 531 | 2 | 1733 | 866,5 | 63,18 | | 2 | 1086 | 543,00 | 2,26 | | 37,33 | |
| 25 | 357065510 | 2 | 960 | 480 | 2 | 1555 | 777,5 | 61,98 | | 2 | 1039 | 519,50 | 8,23 | | 33,18 | |
| 26 | 355095210 | 2 | 1093 | 546,5 | 2 | 1829 | 914,5 | 67,34 | | 2 | 1093 | 546,50 | **0,00** | | 40,24 | |
| 27 | 358095410 | 2 | 1004 | 502 | 2 | 1786 | 893 | 77,89 | | 2 | 1180 | 590,00 | 17,53 | | 33,93 | |
| 28 | 353095910 | 2 | 948 | 474 | 2 | 1662 | 831 | 75,32 | 79,73 | 2 | 1036 | 518,00 | 9,28 | 7,17 | 37,67 | **40,19** |
| 29 | 359095710 | 2 | 828 | 414 | 2 | 1486 | 743 | 79,47 | | 2 | 873 | 436,50 | 5,43 | | 41,25 | |
| 30 | 359095810 | 2 | 821 | 410,5 | 2 | 1631 | 815,5 | 98,66 | | 6 | 2552 | 425,33 | 3,61 | | 47,84 | |

# ANNEXE 13 :
# comparaison entre PLM, OCMPEST et AG sur des exemples 2-périodique de [Mateo, 2001]

PARAMÈTRES DES TABLEAUX DE RÉSULTATS

| Symbole | Signification |
|---|---|
| N° | le numéro de l'exemple dans la série d'exemples. |
| Référence | la référence de l'exemple d'après [Matéo, 2001]. |
| CPU(s) | la durée du temps de simulation, en secondes. |
| T | la durée du cycle 2-périodique hétérogène. |
| T_moyen | la durée du cycle moyenne sur les cinq essais réalisés. |
| E(%) | l'écart moyen entre $T_{moyen}$ et T, définit par la formule suivante : $\frac{T_{moyen} - T}{T_{moyen}} \times 100.$ |

| **Données** | | **PLM** | | **OCMPEST** | | **AG** | | | |
|---|---|---|---|---|---|---|---|---|---|
| N° | Référence | CPU(s) | T | CPU(s) | T | CPU(s) | T | $T_{moyen}$ | E(%) |
| 1 | 109040505 | 116 | 489 | < 1 | 489 | 107 | 489 | 501 | 2,39 |
| 2 | 102040605 | 97 | 347 | < 1 | 411 | 111 | 347 | 470,6 | 26,26 |
| 3 | 103040805 | 119 | 405 | < 1 | 462 | 109 | 405 | 477,8 | 15,24 |
| 4 | 107070405 | 115 | 370 | < 1 | 379 | 109 | 370 | 374,4 | 1,17 |
| 5 | 109070605 | 118 | 466 | < 1 | 500 | 107 | 466 | 480,6 | 3,04 |
| 6 | 102070205 | 118 | 333 | < 1 | 412 | 106 | 333 | 351,2 | 5,18 |
| 7 | 102070705 | 110 | 338 | < 1 | 347 | 108 | 347 | 388,6 | 10,70 |
| 8 | 101070405 | 115 | 373 | < 1 | 407 | 109 | 373 | 443,8 | 15,95 |
| 9 | 205050605 | 115 | 432 | < 1 | 1394 | 109 | 432 | 459,6 | 6,01 |
| 10 | 208050705 | 110 | 376 | < 1 | 425 | 110 | 376 | 417,2 | 9,87 |
| 11 | 204050805 | 124 | 325 | < 1 | 362 | 111 | 325 | 346 | 6,07 |
| 12 | 201080405 | 163 | 494 | < 1 | 520 | 111 | 494 | 510 | 3,14 |
| 13 | 207080605 | 143 | 384 | < 1 | 452 | 111 | 384 | 428,8 | 10,45 |
| 14 | 204080905 | 143 | 325 | < 1 | 363 | 109 | 325 | 349,4 | 6,98 |
| 15 | 308060205 | 160 | 392 | < 1 | 463 | 108 | 392 | 413,8 | 5,27 |
| 16 | 301060405 | 143 | 594 | < 1 | 657 | 108 | 594 | 644,4 | 7,82 |
| 17 | 304060405 | 163 | 576 | < 1 | 593 | 106 | 576 | 611,2 | 5,76 |
| 18 | 309060805 | 143 | 546 | < 1 | 597 | 107 | 546 | 570 | 4,21 |
| 19 | 309090705 | 174 | 512 | < 1 | 580 | 108 | 512 | 518,4 | 1,23 |
| 20 | 304090405 | 143 | 558 | < 1 | 612 | 107 | 558 | 639 | 12,68 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 21 | 111042006 | 225 | 440 | < 1 | 539 | 115 | 440 | 547,4 | 19,62 |
| 22 | 116041806 | 174 | 337 | < 1 | 399 | 111 | 337 | 437,6 | 22,99 |
| 23 | 116041706 | 174 | 336 | < 1 | 435 | 111 | 336 | 408 | 17,65 |
| 24 | 116041706 | 175 | 366 | < 1 | 401 | 116 | 366 | 449,6 | 18,59 |
| 25 | 112071606 | 205 | 525 | < 1 | 578 | 109 | 525 | 565,8 | 7,21 |
| 26 | 213051106 | 163 | 461 | < 1 | 501 | 101 | 461 | 500,6 | 7,91 |
| 27 | 211051406 | 163 | 606 | < 1 | 659 | 111 | 606 | 638,4 | 5,08 |
| 28 | 211051806 | 175 | 662 | < 1 | 674 | 110 | 662 | 707,6 | 6,44 |
| 29 | 218081906 | 184 | 499 | < 1 | 535 | 118 | 499 | 601,4 | 17,03 |
| 30 | 215081106 | 174 | 567 | < 1 | 618 | 118 | 567 | 624,2 | 9,16 |
| 31 | 216081606 | 174 | 583 | < 1 | 605 | 120 | 583 | 627,8 | 7,14 |
| 32 | 314061406 | 163 | 739 | < 1 | 791 | 119 | 739 | 897,4 | 17,65 |
| 33 | 316061906 | 163 | 548 | < 1 | 602 | 118 | 548 | 564,8 | 2,97 |
| 34 | 311062006 | 173 | 653 | < 1 | 700 | 119 | 653 | 682,2 | 4,28 |
| 35 | 319061906 | 143 | 568 | < 1 | 586 | 121 | 568 | 683 | 16,84 |
| 36 | 315091306 | 234 | 586 | < 1 | 636 | 118 | 586 | 615,6 | 4,81 |
| 37 | 311091306 | 174 | 632 | < 1 | 632 | 122 | 632 | 863,6 | 26,82 |
| 38 | 316091906 | 174 | 525 | < 1 | 584 | 119 | 525 | 676,2 | 22,36 |
| 39 | 127042807 | 115 | 690 | < 1 | 763 | 189 | 690 | 790,2 | 12,68 |
| 40 | 126043007 | 112 | 622 | < 1 | 650 | 191 | 622 | 650,4 | 4,37 |
| 41 | 126073007 | 173 | 598 | < 1 | 686 | 189 | 598 | 643,4 | 7,06 |
| 42 | 228052607 | 175 | 558 | < 1 | 576 | 191 | 558 | 607 | 8,07 |
| 43 | 228052607 | 175 | 558 | < 1 | 576 | 192 | 558 | 609,8 | 8,49 |
| 44 | 223052407 | 115 | 635 | < 1 | 635 | 187 | 635 | 797 | 20,33 |
| 45 | 226052507 | 231 | 370 | < 1 | 401 | 190 | 370 | 433,6 | 14,67 |
| 46 | 223082207 | 175 | 579 | < 1 | 681 | 192 | 579 | 623 | 7,06 |
| 47 | 223082507 | 175 | 600 | < 1 | 695 | 188 | 600 | 709,8 | 15,47 |
| 48 | 325062907 | 113 | 775 | < 1 | 802 | 187 | 775 | 775 | 0 |
| 49 | 328062307 | 86 | 788 | < 1 | 803 | 187 | 788 | 834,2 | 5,54 |
| 50 | 328062707 | 136 | 784 | < 1 | 784 | 188 | 784 | 818,8 | 4,25 |
| 51 | 329062707 | 146 | 778 | < 1 | 861 | 192 | 778 | 816,4 | 4,70 |
| 52 | 323092507 | 76 | 928 | < 1 | 928 | 191 | 928 | 1152,4 | 19,47 |
| 53 | 326092807 | 94 | 658 | < 1 | 709 | 191 | 658 | 681,8 | 3,49 |
| 54 | 325092707 | 91 | 741 | < 1 | 869 | 201 | 741 | 748,8 | 1,04 |
| 55 | 134073908 | 176 | 645 | < 1 | 723 | 185 | 645 | 720,6 | 10,49 |
| 56 | 136073908 | 165 | 680 | < 1 | 755 | 211 | 680 | 693,2 | 1,90 |
| 57 | 131073608 | 277 | 573 | < 1 | 664 | 209 | 573 | 622,4 | 7,94 |
| 58 | 236053208 | 112 | 563 | < 1 | 666 | 209 | 563 | 599 | 6,01 |
| 59 | 234053408 | 104 | 776 | < 1 | 841 | 211 | 776 | 800,8 | 3,10 |
| 60 | 232053108 | 76 | 676 | < 1 | 727 | 210 | 676 | 680,4 | 0,65 |
| 61 | 232083608 | 128 | 627 | < 1 | 641 | 210 | 627 | 691,6 | 9,34 |
| 62 | 233083908 | 169 | 805 | < 1 | 850 | 245 | 805 | 867,4 | 7,19 |
| 63 | 231083608 | 168 | 712 | < 1 | 771 | 232 | 712 | 750,6 | 5,14 |
| 64 | 337063608 | 148 | 831 | < 1 | 883 | 236 | 831 | 871,4 | 4,64 |
| 65 | 336063608 | 182 | 922 | < 1 | 922 | 211 | 922 | 949 | 2,85 |
| 66 | 336094008 | 368 | 878 | < 1 | 980 | 220 | 878 | 957,4 | 8,29 |
| 67 | 335093508 | 176 | 980 | < 1 | 1075 | 210 | 980 | 980 | 0,00 |
| 68 | 335093708 | 134 | 972 | < 1 | 999 | 223 | 972 | 975,6 | 0,37 |
| 69 | 144044509 | 115 | 794 | < 1 | 837 | 295 | 794 | 794 | 0,00 |
| 70 | 142044509 | 115 | 799 | < 1 | 820 | 300 | 799 | 842,2 | 5,13 |
| 71 | 146044509 | 116 | 793 | < 1 | 868 | 296 | 793 | 807,4 | 1,78 |
| 72 | 142074509 | 139 | 773 | < 1 | 894 | 295 | 773 | 810,8 | 4,66 |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 73 | 141074409 | 189 | 569 | < 1 | 733 | 298 | 569 | 578,8 | 1,69 |
| 74 | 245054909 | 93 | 793 | < 1 | 889 | 299 | 793 | 796,2 | 0,40 |
| 75 | 246054609 | 183 | 852 | < 1 | 903 | 301 | 852 | 874 | 2,52 |
| 76 | 243084309 | 353 | 786 | < 1 | 834 | 300 | 786 | 815,2 | 3,58 |
| 77 | 249084509 | 158 | 929 | < 1 | 929 | 298 | 929 | 696,2 | 4,15 |
| 78 | 244084409 | 243 | 759 | < 1 | 799 | 297 | 759 | 811 | 6,41 |
| 79 | 244084309 | 207 | 768 | < 1 | 877 | 290 | 768 | 812 | 5,42 |
| 80 | 346064309 | 180 | 919 | < 1 | 932 | 301 | 919 | 929,4 | 1,12 |
| 81 | 348095009 | 841 | 847 | < 1 | 930 | 299 | 847 | 902,2 | 6,12 |
| 82 | 343094409 | 552 | 971 | < 1 | 971 | 298 | 971 | 971 | 0,00 |
| 83 | 156046010 | 124 | 918 | < 1 | 949 | 340 | 918 | 933,2 | 1,63 |
| 84 | 155045910 | 307 | 761 | < 1 | 893 | 326 | 761 | 841,8 | 9,60 |
| 85 | 158075710 | 184 | 910 | < 1 | 959 | 341 | 910 | 960,8 | 5,29 |
| 86 | 260085910 | 200 | 805 | < 1 | 983 | 341 | 805 | 911 | 11,64 |
| 87 | 359065910 | 230 | 911 | < 1 | 928 | 334 | 911 | 942,8 | 3,37 |
| 88 | 359095710 | 197 | 828 | < 1 | 873 | 337 | 828 | 924,8 | 10,47 |

# ANNEXE 14 :
# données des benchmarks multi-produits hétérogènes utilisées pour tester la procédure de panachage

**construites à partir des données de [Matéo, 2001]**



| Produit | Cuves de traitements |
|---|---|
| Numéro | Fenêtre de temps |
| Durée du déplacement (en charge; à vide) entre 2 cuves successives | |

*Benchmark à 5 cuves*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 |
|---|---|---|---|---|---|
| 1 | 50-114 | 80-184 | 60-145 | 80-191 | 28-67 |
| 2 | 49-126 | 71-178 | 54-152 | 53-159 | 56-166 |
| 3 | 77-184 | 57-142 | 67-160 | 45-105 | 27-65 |
| 12; 8 | | | | | *( benchmark à 5 cuves )* |

*Benchmark à 6 cuves*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 |
|---|---|---|---|---|---|---|
| 1 | 30-70 | 48-119 | 70-155 | 45-105 | 24-57 | 72-170 |
| 2 | 73-212 | 38-113 | 69-183 | 23-59 | 64-179 | 49-131 |
| 3 | 30-70 | 59-144 | 44-108 | 45-109 | 24-57 | 32-76 |
| 9; 6 | | | | | | *( benchmark à 6 cuves )* |

*Benchmark à 7 cuves*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 |
|---|---|---|---|---|---|---|---|
| 1 | 75-173 | 67-156 | 80-183 | 65-152 | 45-99 | 71-169 | 75-184 |
| 2 | 54-150 | 49-127 | 76-220 | 24-71 | 57-152 | 67-189 | 29-75 |
| 3 | 41-409 | 67-731 | 34-331 | 30-150 | 50-205 | 34-213 | 30-329 |
| 10; 7 | | | | | | | *( benchmark à 7 cuves )* |

*Benchmark à 8 cuves*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 | cuve 8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 36-87 | 33-78 | 35-85 | 37-91 | 38-86 | 41-100 | 46-107 | 40-93 |
| 2 | 63-173 | 49-140 | 72-203 | 53-138 | 46-132 | 76-199 | 40-116 | 24-65 |
| 3 | 60-142 | 67-147 | 53-125 | 32-80 | 35-86 | 29-68 | 45-108 | 26-64 |
| 7; 5 | | | | | | | | *(benchmark à 8 cuves)* |

*Benchmark à 9 cuves*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 | cuve 8 | cuve 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 42-100 | 66-152 | 36-87 | 21-50 | 29-68 | 32-74 | 68-168 | 60-146 | 36-85 |
| 2 | 65-184 | 65-189 | 76-192 | 21-53 | 38-113 | 56-160 | 49-128 | 51-152 | 22-56 |
| 3 | 40-122 | 40-122 | 40-122 | 40-122 | 40-122 | 40-122 | 40-122 | 40-122 | 40-122 |
| 13; 9 | | | | | | | *(benchmark à 9 cuves)* | | |

*Benchmark à 10 cuves*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 | cuve 8 | cuve 9 | cuve 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 42-98 | 58-127 | 36-82 | 70-170 | 22-55 | 46-112 | 71-158 | 44-108 | 72-173 | 21-47 |
| 2 | 78-214 | 31-90 | 49-141 | 33-88 | 32-93 | 74-197 | 44-111 | 69-202 | 37-97 | 65-174 |
| 3 | 73-167 | 45-104 | 40-88 | 34-75 | 62-140 | 33-73 | 33-76 | 50-120 | 41-97 | 46-110 |
| 9; 6 | | | | | | | | | *(benchmark à 10 cuves)* | |

# ANNEXE 15 :
# données des benchmarks multi-produits hétérogènes à 20, 30, 40 et 50 cuves de traitements



| Produit | Cuves de traitements |
|---|---|
| Numéro | Fenêtre de temps |
| Durée du déplacement (en charge; à vide) entre 2 cuves successives | |

*Benchmark à 20 cuves de traitement*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 | cuve 8 | cuve 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 42-3200 | 58-3540 | 36-3800 | 70-3000 | 22-3660 | 46-3100 | 71-3200 | 44-3011 | 72-3011 |
| 2 | 78-3109 | 31-3341 | 49-2780 | 33-2980 | 32-3365 | 74-3000 | 44-3200 | 69-3011 | 37-2811 |
| 6; 3 | | | | | | | | | *(benchmark à 20 cuves de traitement)* |

| cuve 10 | cuve 11 | cuve 12 | cuve 13 | cuve 14 | cuve 15 | cuve 16 | cuve 17 | cuve 18 | cuve 19 | cuve 20 |
|---|---|---|---|---|---|---|---|---|---|---|
| 21-3012 | 73-3100 | 45-3220 | 40-3005 | 34-3019 | 62-2950 | 33-2890 | 33-2930 | 50-3001 | 41-3120 | 46-3055 |
| 65-3112 | 78-3105 | 31-3120 | 49-3105 | 33-2913 | 32-2850 | 74-2690 | 44-2930 | 69-2871 | 37-2920 | 65-3145 |
| 6; 3 | | | | | | | | | | *(benchmark à 20 cuves de traitement)* |

*Benchmark à 30 cuves de traitement*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 | cuve 8 | cuve 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 42-3200 | 58-3540 | 36-3800 | 70-3000 | 22-3660 | 46-3100 | 71-3200 | 44-3011 | 72-3011 |
| 2 | 78-3109 | 31-3341 | 49-2780 | 33-2980 | 32-3365 | 74-3000 | 44-3200 | 69-3011 | 37-2811 |
| 6; 3 | | | | | | | | | *(benchmark à 30 cuves de traitement)* |

| cuve 10 | cuve 11 | cuve 12 | cuve 13 | cuve 14 | cuve 15 | cuve 16 | cuve 17 | cuve 18 | cuve 19 | cuve 20 |
|---|---|---|---|---|---|---|---|---|---|---|
| 21-3012 | 73-3100 | 45-3220 | 40-3005 | 34-3019 | 62-2950 | 33-2890 | 33-2930 | 50-3001 | 41-3120 | 46-3055 |
| 65-3112 | 78-3105 | 31-3120 | 49-3105 | 33-2913 | 32-2850 | 74-2690 | 44-2930 | 69-2871 | 37-2920 | 65-3145 |
| 6; 3 | | | | | | | | | | *(benchmark à 30 cuves de traitement)* |

| cuve 21 | cuve 22 | cuve 23 | cuve 24 | cuve 25 | cuve 26 | cuve 27 | cuve 28 | cuve 29 | cuve 30 |
|---|---|---|---|---|---|---|---|---|---|
| 56-3097 | 43-3210 | 20-2980 | 34-2790 | 21-2919 | 23-2560 | 54-3000 | 12-3120 | 39-3013 | 17-3000 |
| 35-3097 | 45-3110 | 65-2880 | 52-2690 | 51-2519 | 32-2860 | 25-3012 | 12-3020 | 13-3101 | 51-3000 |
| 6; 3 | | | | | | | | | *(benchmark à 30 cuves de traitement)* |

*Benchmark à 40 cuves de traitement*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 | cuve 8 | cuve 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 42-3200 | 58-3540 | 36-3800 | 70-3000 | 22-3660 | 46-3100 | 71-3200 | 44-3011 | 72-3011 |
| 2 | 78-3109 | 31-3341 | 49-2780 | 33-2980 | 32-3365 | 74-3000 | 44-3200 | 69-3011 | 37-2811 |
| 6; 3 | | | | | | | | | *(benchmark à 40 cuves de traitement)* |

| cuve 10 | cuve 11 | cuve 12 | cuve 13 | cuve 14 | cuve 15 | cuve 16 | cuve 17 | cuve 18 | cuve 19 | cuve 20 |
|---|---|---|---|---|---|---|---|---|---|---|
| 21-3012 | 73-3100 | 45-3220 | 40-3005 | 34-3019 | 62-2950 | 33-2890 | 33-2930 | 50-3001 | 41-3120 | 46-3055 |
| 65-3112 | 78-3105 | 31-3120 | 49-3105 | 33-2913 | 32-2850 | 74-2690 | 44-2930 | 69-2871 | 37-2920 | 65-3145 |
| 6; 3 | | | | | | | | | | *(benchmark à 40 cuves de traitement)* |

| cuve 21 | cuve 22 | cuve 23 | cuve 24 | cuve 25 | cuve 26 | cuve 27 | cuve 28 | cuve 29 | cuve 30 | cuve 31 |
|---|---|---|---|---|---|---|---|---|---|---|
| 56-3097 | 43-3210 | 20-2980 | 34-2790 | 21-2919 | 23-2560 | 54-3000 | 12-3120 | 39-3013 | 17-3000 | 23-3020 |
| 35-3097 | 45-3110 | 65-2880 | 52-2690 | 51-2519 | 32-2860 | 25-3012 | 12-3020 | 13-3101 | 51-3000 | 25-2989 |
| 6; 3 | | | | | | | | | | *(benchmark à 40 cuves de traitement)* |

| cuve 32 | cuve 33 | cuve 34 | cuve 35 | cuve 36 | cuve 37 | cuve 38 | cuve 39 | cuve 40 | |
|---|---|---|---|---|---|---|---|---|---|
| 34-3103 | 44-2910 | 13-2875 | 57-3100 | 32-2800 | 13-2910 | 19-2880 | 23-2970 | 10-2790 | |
| 34-3102 | 23-2910 | 19-2970 | 56-3205 | 34-2876 | 77-2960 | 23-2880 | 65-2970 | 39-2768 | |
| 6; 3 | | | | | | | | | *(benchmark à 40 cuves de traitement)* |

*Benchmark à 50 cuves de traitement*

| Produit | cuve 1 | cuve 2 | cuve 3 | cuve 4 | cuve 5 | cuve 6 | cuve 7 | cuve 8 | cuve 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 42-3200 | 58-3540 | 36-3800 | 70-3000 | 22-3660 | 46-3100 | 71-3200 | 44-3011 | 72-3011 |
| 2 | 78-3109 | 31-3341 | 49-2780 | 33-2980 | 32-3365 | 74-3000 | 44-3200 | 69-3011 | 37-2811 |
| 6; 3 | | | | | | | | | *(benchmark à 50 cuves de traitement)* |

| cuve 10 | cuve 11 | cuve 12 | cuve 13 | cuve 14 | cuve 15 | cuve 16 | cuve 17 | cuve 18 | cuve 19 | cuve 20 |
|---|---|---|---|---|---|---|---|---|---|---|
| 21-3012 | 73-3100 | 45-3220 | 40-3005 | 34-3019 | 62-2950 | 33-2890 | 33-2930 | 50-3001 | 41-3120 | 46-3055 |
| 65-3112 | 78-3105 | 31-3120 | 49-3105 | 33-2913 | 32-2850 | 74-2690 | 44-2930 | 69-2871 | 37-2920 | 65-3145 |
| 6; 3 | | | | | | | | *(benchmark à 50 cuves de traitement)* | | |

| cuve 21 | cuve 22 | cuve 23 | cuve 24 | cuve 25 | cuve 26 | cuve 27 | cuve 28 | cuve 29 | cuve 30 | cuve 31 |
|---|---|---|---|---|---|---|---|---|---|---|
| 56-3097 | 43-3210 | 20-2980 | 34-2790 | 21-2919 | 23-2560 | 54-3000 | 12-3120 | 39-3013 | 17-3000 | 23-3020 |
| 35-3097 | 45-3110 | 65-2880 | 52-2690 | 51-2519 | 32-2860 | 25-3012 | 12-3020 | 13-3101 | 51-3000 | 25-2989 |
| 6; 3 | | | | | | | | *(benchmark à 50 cuves de traitement)* | | |

| cuve 32 | cuve 33 | cuve 34 | cuve 35 | cuve 36 | cuve 37 | cuve 38 | cuve 39 | cuve 40 | cuve 41 | cuve 42 |
|---|---|---|---|---|---|---|---|---|---|---|
| 34-3103 | 44-2910 | 13-2875 | 57-3100 | 32-2800 | 13-2910 | 19-2880 | 23-2970 | 10-2790 | 65-3000 | 34-5000 |
| 34-3102 | 23-2910 | 19-2970 | 56-3205 | 34-2876 | 77-2960 | 23-2880 | 65-2970 | 39-2768 | 45-2900 | 55-4900 |
| 6; 3 | | | | | | | | *(benchmark à 50 cuves de traitement)* | | |

| cuve 43 | cuve 44 | cuve 45 | cuve 46 | cuve 47 | cuve 48 | cuve 49 | cuve 50 |
|---|---|---|---|---|---|---|---|
| 86-4320 | 33-3120 | 23-3050 | 56-3077 | 9-3065 | 23-2890 | 15-2789 | 37-2991 |
| 15-4210 | 45-2912 | 69-3065 | 33-3177 | 13-3035 | 69-2790 | 34-2779 | 56-2891 |
| 6; 3 | | | | | *(benchmark à 50 cuves de traitement)* | | |

**Résumé :**

Cette thèse s'intéresse au fonctionnement cyclique multi-produits des ateliers de traitement de surface, et au problème d'ordonnancement associé (HSP), caractérisé par des contraintes fortes et atypiques, dont certaines sont liées aux ressources de transport. Dans le cas de productions en grandes séries, une commande cyclique de ces systèmes est particulièrement adaptée, permettant notamment de réduire la combinatoire de résolution, et sous réserve que les ratios de produits soient connus à l'avance. Notre objectif est de trouver le meilleur ordonnancement des tâches de traitement et de transport en un temps raisonnable. Pour cela, nous proposons une première approche, basée sur un modèle linéaire et une méthode de résolution arborescente de type séparation et évaluation. Nous présentons des modélisations pour différentes extensions du problème dit de base et nous fournissons des exemples illustratifs et des résultats sur des benchmarks. Par la suite et compte tenu de l'analyse de la littérature relative aux ordonnancements cycliques mono-produit et multi-produits, nous proposons tout d'abord une heuristique dédiée au cas multi-produits étudié, et basée sur un algorithme de liste. Avec ce dernier, nous obtenons un ordonnancement cyclique dont le degré du cycle n'est pas fixé au préalable. Enfin, nous présentons une deuxième modélisation approchée sous la forme d'un algorithme génétique pour résoudre un HSP 2-cyclique. Ces différents modèles sont validés par des tests sur des benchmarks de la littérature pour lesquels nous avons obtenus des résultats prometteurs. Nous terminons par une analyse critique des avantages et inconvénients des modèles élaborés et par quelques propositions de perspectives pour ce travail.

**Mots clés :** Ordonnancement Cyclique Multi-Produits des Lignes de Traitement de Surface, Fenêtres de Temps, Programmation Linéaire Mixte, Heuristique, Algorithme Génétique.

**Abstract:**

In this thesis, we study the Hoist Scheduling Problem (HSP) in automated electroplating lines, when a mass production must be achieved. The HSP is characterized by strong and atypical constraints, some of which are related to transportation resources. In the case of mass production, a cyclic control of these systems is particularly suitable, especially for reducing the complexity resolution when the production rates are known in advance. Our goal is to find the best scheduling of processing transport tasks in a reasonable time. We first elaborate a 2-degree cyclic model and an associated branch and bound algorithm. Then we extend it to more complex configurations. Then, we develop a dedicated heuristic to find a feasible repetitive sequence of hoist moves that minimizes the cycle time, without a priori fixing the cycle degree. Comparisons with existing algorithms are presented to show the efficiency of the proposed heuristic. To improve the performance of our approach and to reduce the cycle time, we integrate in the dedicated heuristic an algorithm to find the best entry order of products into the line. Finally, we describe a genetic algorithm approach to find a schedule which can reach the optimal 2-cycle. We finally discuss the interest of those various models, based on the promising results obtained and we provide some perspectives which could be explored.

**Keywords:** Multi-Parts Cyclic Hoist Scheduling Problem, Time Windows, Mixed Integer Linear Model, Heuristic, Genetic Algorithm.