



**HAL**  
open science

# Planification et affectation de ressources dans les réseaux de soin : analogie avec le problème du bin packing, proposition de méthodes approchées

Nathalie Klement

## ► To cite this version:

Nathalie Klement. Planification et affectation de ressources dans les réseaux de soin : analogie avec le problème du bin packing, proposition de méthodes approchées. Autre [cs.OH]. Université Blaise Pascal - Clermont-Ferrand II, 2014. Français. NNT : 2014CLF22517 . tel-01153498

**HAL Id: tel-01153498**

**<https://theses.hal.science/tel-01153498v1>**

Submitted on 19 May 2015

**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 : D.U. 2517  
EDSPIC : 675

**UNIVERSITÉ BLAISE PASCAL -  
CLERMONT-FERRAND II**

*ÉCOLE DOCTORALE*

*SCIENCES POUR L'INGÉNIEUR DE  
CLERMONT-FERRAND*

# **THÈSE**

présentée par

**Nathalie KLEMENT**

pour obtenir le grade de

**Docteur d'université**

Spécialité : INFORMATIQUE - PRODUCTIQUE

**Planification et affectation de  
ressources dans les réseaux de soin :**

**Analogie avec le problème du bin packing  
Proposition de méthodes approchées**

Soutenue publiquement le 4 décembre 2014 devant le jury :

Madame	Virginie Fortineau	Examinatrice
Monsieur	Michel Gourgand	Directeur de thèse
Madame	Nathalie Grangeon	Co-encadrante
Monsieur	Samir Lamouri	Rapporteur
Monsieur	Aziz Moukrim	Rapporteur
Monsieur	Alain Quilliot	Examineur



# Abstract

**Title :** Planning and resources assignment in healthcare networks : analogy with the bin packing problem, proposition of approximate methods.

The presented work is about optimization of the hospital system. An existing solution is the pooling of resources within the same territory. This may involve different forms of cooperation between several hospitals. Various problems are defined at the decision level : strategic, tactical or operational ; and at the modeling level : macroscopic, mesoscopic and microscopic. Problems of sizing, planning and scheduling may be considered. We define the problem of activities planning with resource allocation. Several cases are dissociated : either human resources are under infinite capacity, or they are under limited capacity and their assignment on a place is given, or they are under limited capacity and their assignment is a variable. These problems are specified and mathematically formalized. All these problems are compared to a bin packing problem : the classical problem of bin packing is used for the problem where human resources are under infinite capacity, the bin packing problem with interdependencies is used in the two other cases. The bin packing problem with incompatibilities is defined.

Many resolution methods have been proposed for the bin packing problem. We make several propositions including a hierarchical coupling between heuristic and metaheuristic. Single based metaheuristics and a population based metaheuristic, the particle swarm optimization, are used. This proposition requires a new encoding inspired by permutation problems. This method gives very good results to solve instances of the bin packing problem. It is easy to apply : it combines already known methods.

With the proposed coupling, the new constraints to be considered need to be integrated only on the heuristic level. The running of the metaheuristic is the same. Thus, our method is easily adaptable to the problem of activities planning with resource allocation. For big instances, the solver used as a reference returns only an interval of solutions. The results of our method are once again very promising : the obtained solutions are better than the upper limit returned by the solver. It is possible to adapt our method on more complex issues through integration into the heuristic of the new constraints to consider. It would be particularly interesting to test these methods on real hospital authorities to assess their significance.

**Keywords :** activities planning, resource allocation, bin packing problems, single based metaheuristics, particle swarm optimization, heuristics, hybridization of methods, cooperation, pooling.



# Résumé

Les travaux de thèse présentés s'intéressent à l'optimisation des systèmes hospitaliers. Une solution existante est la mutualisation de ressources au sein d'un même territoire. Cela peut passer par différentes formes de coopération dont la Communauté Hospitalière de Territoire. Différents problèmes sont définis en fonction du niveau de décision : stratégique, tactique ou opérationnel ; et du niveau de modélisation : macroscopique, mesoscopique et microscopique. Des problèmes de dimensionnement, de planification et d'ordonnancement peuvent être considérés. Nous définissons notamment le problème de planification d'activités avec affectation de ressources. Plusieurs cas sont dissociés : soit les ressources humaines sont à capacité infinie, soit elles sont à capacité limitée et leur affectation sur site est une donnée, soit elles sont à capacité limitée et leur affectation sur site est une variable. Ces problèmes sont spécifiés et formalisés mathématiquement. Tous ces problèmes sont comparés à un problème de bin packing : le problème du bin packing de base pour le problème où les ressources humaines sont à capacité infinie, le problème du bin packing avec interdépendances dans les deux autres cas. Le problème du bin packing avec incompatibilités est ainsi défini.

De nombreuses méthodes de résolution ont déjà été proposées pour le problème du bin packing. Nous faisons plusieurs propositions dont un couplage hiérarchique entre une heuristique et une métaheuristique. Des métaheuristicues basées individu et une métaheuristique basée population, l'optimisation par essaim particulière, sont utilisées. Cette proposition nécessite un nouveau codage inspiré des problèmes de permutation d'ordonnancement. Cette méthode donne de très bons résultats sur les instances du problème du bin packing. Elle est simple à appliquer : elle couple des méthodes déjà connues.

Grâce au couplage proposé, les nouvelles contraintes à considérer nécessitent d'être intégrées uniquement au niveau de l'heuristique. Le fonctionnement de la métaheuristique reste le même. Ainsi, notre méthode est facilement adaptable au problème de planification d'activités avec affectation de ressources. Pour les instances de grande taille, le solveur utilisé comme référence ne donne qu'un intervalle de solutions. Les résultats de notre méthode sont une fois encore très prometteurs : les solutions obtenues sont meilleures que la borne supérieure retournée par le solveur. Il est envisageable d'adapter notre méthode sur d'autres problèmes plus complexes par intégration dans l'heuristique des nouvelles contraintes à considérer. Il serait notamment intéressant de tester ces méthodes sur de réelles instances hospitalières afin d'évaluer leur portée.

**Mots-clés** : planification d'activités, affectation de ressources, problèmes du bin packing, métaheuristicues basées individu, optimisation par essaim particulière, heuristiques, hybridation de méthodes, coopération, mutualisation.



# Remerciements

Ces travaux ont été réalisés au Laboratoire d'Informatique, de Modélisation et d'Optimisation des Systèmes (LIMOS CNRS UMR 6158) dirigé par Alain Quilliot puis par Farouk Toumani. Je tiens à les remercier tous deux pour leur accueil dans le laboratoire.

Cette thèse a été encadrée par Michel Gourgand et Nathalie Grangeon. Je remercie Michel pour avoir fait tout son possible pour m'accueillir au sein de l'équipe en me proposant ce sujet. Je le remercie pour sa disponibilité, son écoute et tous ses conseils qui m'ont permis d'en arriver là où je suis. Je remercie également Nathalie, mon homonyme, pour m'avoir accordée de son temps et avoir su trouver les mots justes pour m'indiquer la voie à suivre. Je remercie également tous les membres de l'équipe : Laurent, Fabrice, Fazia, Benjamin, Sophie, Janvier, Jean-Christophe, Bernard, Johan et bien sûr Sylvie sur qui j'ai pu compter quand le besoin s'en faisait ressentir.

Je remercie les membres du jury d'avoir bien voulu juger mon travail : Alain Quilliot en tant que président du jury, Samir Lamouri et Aziz Moukrim en tant que rapporteurs et Virginie Fortineau en tant qu'examinatrice. Je remercie particulièrement Samir Lamouri de m'avoir suivie tout au long de ma thèse au travers des comités de thèse réguliers. Je le remercie également, ainsi que Virginie Fortineau, d'avoir contribué au financement d'une partie de ma thèse, en instaurant un partenariat avec les hôpitaux de Paris. J'espère qu'un travail collaboratif pourra voir le jour prochainement.

Je remercie d'une façon générale tous les collègues que j'ai côtoyés tout au long de cette thèse, que ce soit sur un aspect recherche, enseignement ou administratif. Toutes ces discussions enrichies auront contribué à consolider mes compétences techniques et relationnelles. Je pense en particulier à Fatiha et à Michelle, mais aussi à Michel, Christophe, Loïc, Violaine, Philippe, Susan, Alain, Jonas, Pascale et Séverine. Je remercie en particulier Béa, administrative de talent, sans qui je n'aurais pas pu en arriver là. Une pensée également aux enseignants de l'IFMA, qui ont été mes enseignants avant de devenir mes collègues : Jean-Luc, Christophe, Séverine, Khalid, Anne-Lise, Michel, Caroline, Olivier, Laurent et tous les autres.

Merci à mes parents de m'avoir permise de briller dans les études, ils ont toujours su me pousser vers le haut. Merci d'avoir cru en moi. Merci à ma petite sœur Laura, tu as été mon premier cobaye.



Merci à tous ceux qui ont partagé mon quotidien, de près ou de loin :

Merci à ma chère Maud, ma première collocataire du temps de la prépa. Merci de m'avoir initiée à l'anglais pendant que je te distraçais avec mes problèmes de math.

Merci à Carmen et Paco de m'avoir enseigné l'espagnol; merci à Wojtek, Michele, Mauro, Stefanuzzo, Tomek et Arnaud de m'avoir fait découvrir les États-Unis; merci à Ade de les avoir aidés.

Merci à Laure la meilleure binôme qui soit; merci à Katy ma maman de cœur; merci à Luc mon daddy de cœur qui a toujours su m'écouter et me conseiller dans tous les domaines.

Merci à Kseniya, la première à m'avoir fait découvrir le monde de la recherche; merci à Maïté ma première copine de sport à supporter mes jérémiades du lundi soir; merci à Bob, parce qu'on connaît tous un Bob; merci à Aurélie et Bastien (et l'asticot) pour toujours trouver les mots ou actes pour me faire rire; merci à Delphine ma deuxième copine de sport, ma copine de soirée, mais qui m'a abandonnée trop vite pour la vie parisienne; merci à Georges et ses robots, ses baignades régulières dans tous les volcans qui nous entourent; merci à Pti Ju de m'avoir initiée aux vrais jeux de société; merci à Sami mon compagnon de toujours, depuis le BdE jusqu'à aujourd'hui au bout de la nuit; merci à Nadège la courageuse doctorante en mécanique qui ne fait pas de la mécanique, tu tiens le bon bout; merci à Noemi de nous faire rêver avec l'Italie et en particulier la Sicile; merci à Benoit pour ses cours de rock; merci à Charles le jeunot de la bande. Merci également à Guillaume, Romain, Benjamin, Lorena, Nicolas, Agnès, Simon, Khaled, Imad, Cédric, Kévin et à tous ceux que j'ai côtoyés au feu cinquième étage, à l'ancien LAMI.

Merci aux geeks 1.0 : merci à Jonathan de m'avoir intégrée à la bande, de m'avoir initiée à tous ces langages de programmation et d'avoir été mon assistant technique; merci à Luc et Jonathan pour leurs discussions à la machine à café qui m'ont fait comprendre qu'il existe plusieurs niveaux d'informaticien; merci à Nico pour sa ponctualité et ses cours de réseau; merci à Clément pour sa ponctualité et ses blagues intarissables; merci à Guillaume, potu du midi mais également en dehors; merci à Thomas pour son expérience du monde de la recherche; merci à Benny, tu es plus qu'un geek mais tu mérites aussi mes remerciements. Merci également à Pierre, Sébastien, Romain, Faouzi et les autres.

Merci aux geeks 2.0 : merci à Arnaud d'avoir pu éclairer mes connaissances approchées des méthodes de résolution et d'avoir su rester patient; merci à Ben pour son introduction aux tsutsumbis, sa description de la femme idéale mais surtout son courage sans faille; merci à Corinne pour son écoute et son soutien au quotidien, pour m'avoir ouvert les yeux sur la réalité du monde du travail; merci à PA pour ses compétences en Perl et en L<sup>A</sup>T<sub>E</sub>X, ses dégustations de thé, son jeu de raquette, ses prises de tête, merci d'avoir fait en sorte qu'aucune journée ne ressemble à une autre, d'avoir été aussi insupportable qu'aimable; merci à Seb le gentil de la bande, qui nous étonne toujours autant avec ses propos politiques. Merci également à Charifa, Damien, Benjamin, Maxime, Maxime,

Raksmey, Djeloul, Firmin et les autres.

Et enfin les trois plus importants, après mes parents et ma sœur bien sûr, ceux qui ont partagé ma vie pendant ces trois dernières années : Julien, Jérémy et Lina. Et non, je ne vous oublie pas. Merci à Juju de m'avoir initiée à la rénovation d'appartement ; merci à Jérémy de m'avoir redonné goût à la lecture ; merci à vous deux de m'avoir supportée au quotidien. Merci pour tous les bons moments partagés (parce qu'il n'y en a eu que des bons). Merci à Lina, ma copine de maternelle de m'avoir rejointe dans mon quotidien. Il fallait bien que ma série de collocations se termine par celle-ci !

Merci à tous d'y avoir cru tout autant que moi =)



# Table des matières

<b>Abstract</b>	<b>iii</b>
<b>Résumé</b>	<b>v</b>
<b>Remerciements</b>	<b>vii</b>
<b>Table des matières</b>	<b>xi</b>
<b>Liste des figures</b>	<b>xv</b>
<b>Liste des tableaux</b>	<b>xvii</b>
<b>Liste des algorithmes</b>	<b>xix</b>
<b>Introduction</b>	<b>1</b>
<b>1 Présentation de la problématique</b>	<b>5</b>
1.1 Introduction . . . . .	7
1.2 Contexte hospitalier . . . . .	7
1.2.1 Définitions . . . . .	7
1.2.2 Plans hôpitaux 2007 et 2012 . . . . .	9
1.2.3 Rapport Larcher . . . . .	10
1.2.4 Loi HPST . . . . .	10
1.3 Réseaux de soin et coopérations multi-sites . . . . .	11
1.3.1 Communauté Hospitalière de Territoire . . . . .	11
1.3.2 Fédération d’Imagerie Médicale . . . . .	19
1.3.3 Autres exemples de coopérations . . . . .	21
1.4 Recherche opérationnelle pour les réseaux de soin . . . . .	23
1.4.1 Dimensionnement . . . . .	23
1.4.2 Planification . . . . .	24
1.4.3 Ordonnancement . . . . .	26
1.4.4 Réseaux hospitaliers multi-sites . . . . .	27
1.4.5 Conclusion . . . . .	29
1.5 Objectifs de la thèse . . . . .	29
1.5.1 Niveaux de décision . . . . .	29

1.5.2	Niveaux de modélisation . . . . .	29
1.5.3	Définition des problèmes . . . . .	30
1.5.4	Outil d'aide à la décision . . . . .	32
1.6	Conclusion . . . . .	34
<b>2</b>	<b>Spécification et formalisation de la Communauté Hospitalière de Terri-</b>	<b>35</b>
	<b>toire</b>	
2.1	Introduction . . . . .	36
2.2	Spécification . . . . .	36
2.2.1	Processus de modélisation . . . . .	36
2.2.2	Sous-système physique . . . . .	37
2.2.3	Sous-système logique . . . . .	38
2.2.4	Sous-système décisionnel . . . . .	38
2.3	Problèmes considérés . . . . .	41
2.3.1	Hypothèses . . . . .	41
2.3.2	Définition des problèmes considérés . . . . .	41
2.4	Formalisation . . . . .	44
2.4.1	Modèle 1 : CHT . . . . .	44
2.4.2	Modèle 2 : CHT_RH non mobiles . . . . .	46
2.4.3	Modèle 3 : CHT_RH mobiles . . . . .	49
2.4.4	Récapitulatif des modèles . . . . .	51
2.5	Le problème du bin packing . . . . .	56
2.5.1	Présentation du problème . . . . .	56
2.5.2	Analogies avec notre problème . . . . .	57
2.6	Conclusion . . . . .	60
<b>3</b>	<b>État de l'art</b>	<b>63</b>
3.1	Introduction . . . . .	64
3.2	Méthodes de résolution pour l'optimisation combinatoire . . . . .	64
3.2.1	Méthodes exactes pour la résolution des problèmes linéaires . . . . .	64
3.2.2	Méthodes approchées . . . . .	66
3.2.3	Approches d'hybridation . . . . .	78
3.3	Le problème du bin packing . . . . .	80
3.3.1	Méthodes exactes . . . . .	81
3.3.2	Heuristiques . . . . .	82
3.3.3	Métaheuristiques . . . . .	88
3.3.4	Conclusion de l'état de l'art sur le problème du bin packing . . . . .	94
3.4	Différentes utilisations de la PSO pour la résolution de problèmes d'opti-	
	misation combinatoire . . . . .	95
3.4.1	Le problème des $n$ dames . . . . .	96
3.4.2	Le problème du voyageur de commerce . . . . .	96
3.4.3	Le RCPSP . . . . .	98
3.4.4	Les problèmes d'ordonnancement . . . . .	99
3.4.5	Le problème du bin packing . . . . .	101
3.4.6	Conclusion sur la PSO . . . . .	102
3.5	Conclusion . . . . .	102

<b>4</b>	<b>Proposition de méthodes pour la résolution de problèmes de bin packing</b>	<b>103</b>
4.1	Introduction . . . . .	105
4.2	Métaheuristiques pour le bin packing de base . . . . .	105
4.2.1	Codage d'une solution . . . . .	105
4.2.2	Métaheuristiques basées individu . . . . .	106
4.2.3	Métaheuristique basée population . . . . .	108
4.2.4	Conclusion du codage classique . . . . .	111
4.3	Couplage heuristiques métaheuristiques pour le bin packing de base . . . . .	112
4.3.1	Codage d'une solution . . . . .	112
4.3.2	Métaheuristiques basées individu . . . . .	114
4.3.3	Métaheuristique basée population . . . . .	116
4.3.4	Conclusion du couplage hiérarchique . . . . .	125
4.4	Extension des méthodes pour le bin packing avec incompatibilités . . . . .	125
4.4.1	Codage . . . . .	127
4.4.2	Proposition d'algorithmes de liste . . . . .	127
4.4.3	Système de voisinage . . . . .	129
4.4.4	Pénalisation de contraintes . . . . .	131
4.4.5	Méthode des critères pondérés . . . . .	132
4.5	Conclusion . . . . .	133
<b>5</b>	<b>Mise en œuvre et résultats</b>	<b>135</b>
5.1	Introduction . . . . .	136
5.2	Le problème du bin packing . . . . .	136
5.2.1	Instances . . . . .	136
5.2.2	Métaheuristiques . . . . .	140
5.2.3	Couplage heuristiques métaheuristiques . . . . .	140
5.2.4	Conclusion de la résolution du problème du bin packing . . . . .	142
5.3	Le modèle 1 de la CHT . . . . .	142
5.3.1	Création des instances . . . . .	142
5.3.2	Résultats référence : CPLEX . . . . .	143
5.3.3	Algorithmes de liste . . . . .	144
5.3.4	Métaheuristiques . . . . .	146
5.3.5	Couplage heuristiques métaheuristiques . . . . .	146
5.4	Les modèles avec ressources humaines . . . . .	147
5.4.1	Résolution par méthode exacte . . . . .	147
5.4.2	Limitation des méthodes exactes . . . . .	150
5.5	Conclusion . . . . .	150
	<b>Conclusion</b>	<b>153</b>
	<b>Bibliographie</b>	<b>157</b>
<b>A</b>	<b>Détail des résultats</b>	<b>165</b>
A.1	Problème du bin packing . . . . .	166
A.2	Problème de planification d'activités avec affectation de ressources . . . . .	175



# Liste des figures

1.1	Chronologie des réformes gouvernementales . . . . .	9
1.2	Exemple d'une CHT . . . . .	13
1.3	Chronologie depuis la création des CHT . . . . .	14
2.1	Communication entre les trois sous-systèmes . . . . .	37
2.2	Diagramme de classes du problème complet de la CHT . . . . .	42
2.3	Diagramme de classes du problème 1 . . . . .	43
2.4	Diagramme de classes du problème 3 . . . . .	43
2.5	Une solution du problème du bin packing 1D . . . . .	56
2.6	Une représentation du problème de planification d'activités avec affectation de ressources . . . . .	58
2.7	Illustration du problème du bin packing interdépendant . . . . .	60
2.8	Une représentation du problème de la CHT avec les ressources humaines . . . . .	61
3.1	Méthodes de résolution exactes . . . . .	67
3.2	Comportement d'une particule . . . . .	75
3.3	Ensemble $\Omega$ des particules $p$ . . . . .	78
3.4	Chainage entre deux méthodes . . . . .	79
3.5	Couplage séquentiel entre deux méthodes . . . . .	80
3.6	Couplage hiérarchique entre deux méthodes . . . . .	80
3.7	Critère de dominance : $B_2$ domine $B_1$ . . . . .	81
4.1	Ensembles de solution $C \subseteq \Omega$ . . . . .	106
4.2	Ensembles de solutions utilisés par l'hybridation . . . . .	113
4.3	Illustration du système de voisinage $V$ . . . . .	131
1	Composition de la CHT . . . . .	155
2	Taux d'occupation des équipements et des ressources humaines . . . . .	156





# Liste des tableaux

1.1	Matrice 3*3 des problèmes de la CHT . . . . .	31
2.1	Les critères rencontrés à chaque niveau de décision . . . . .	40
2.2	Données considérées par chaque modèle . . . . .	52
2.3	Variabes considérées par chaque modèle . . . . .	53
2.4	Contraintes utilisées par chaque modèle . . . . .	54
2.5	Critères pris en compte par chacun des modèles . . . . .	55
2.6	Tableau d’analogies entre les deux problèmes . . . . .	59
3.1	Définition des problèmes . . . . .	65
3.2	Comparatif éléments PSO . . . . .	77
3.3	Fonctions objectifs du problème du bin packing optimisées par la littérature	89
5.1	Instances du problème du bin packing testées par la littérature . . . . .	138
5.2	Résultats de la littérature sur les instances du problème du bin packing . .	139
5.3	Application du couplage heuristique métaheuristique basée individu pour la résolution du problème du bin packing . . . . .	141
5.4	Données générées pour la CHT . . . . .	143
5.5	Résultats pour le modèle 1 avec la méthode exacte . . . . .	144
5.6	Résultats de l’application des algorithmes de liste sur le modèle 1 . . . . .	145
5.7	Résultat du couplage pour le problème de planification d’activités avec affectation de ressources . . . . .	146
5.8	Résolution par méthode exacte du modèle 2 . . . . .	148
5.9	Résolution par méthode exacte du modèle 3 . . . . .	149
5.10	Nombre d’itérations et de nœuds pour la résolution par méthode exacte des cinq modèles . . . . .	151
A.1	Résultats de l’application des métaheuristiques basées individu . . . . .	167
A.2	Nombre de solutions optimales trouvées par les métaheuristiques basées individu . . . . .	168
A.3	Résultats de l’application du couplage du recuit simulé avec les heuristiques	169
A.4	Résultats de l’application du couplage de l’algorithme du kangourou et des heuristiques . . . . .	170
A.5	Nombre de solutions optimales trouvées par le couplage entre les métaheuristiques basées individu et les heuristiques . . . . .	171

A.6	Résultats de l'application du couplage de la PSO et de l'heuristique First Fit	172
A.7	Résultats de l'application du couplage de la PSO et de l'heuristique Best Fit	173
A.8	Nombre de solutions optimales trouvées par l'application du couplage de la PSO et des heuristiques . . . . .	174
A.9	Résultats de l'application des métaheuristiques basées individu . . . . .	176
A.10	Meilleurs résultats de l'application des métaheuristiques basées individu . .	177
A.11	Résultats de l'application du couplage du recuit simulé et des heuristiques	178
A.12	Résultats de l'application du couplage de l'algorithme du kangourou et des heuristiques . . . . .	179
A.13	Meilleurs résultats de l'application du couplage des métaheuristiques basées individu et des heuristiques . . . . .	180
A.14	Résultats de l'application du couplage de la PSO et de l'heuristique First Fit	181
A.15	Résultats de l'application du couplage de la PSO et de l'heuristique Best Fit	182
A.16	Meilleurs résultats de l'application du couplage de la PSO avec une heuristique . . . . .	183

# Liste des algorithmes

1	Algorithme de principe de la descente déterministe . . . . .	68
2	Algorithme de principe de la descente stochastique . . . . .	69
3	Algorithme de principe du recuit simulé inhomogène . . . . .	69
4	Algorithme de principe de l'algorithme du kangourou . . . . .	70
5	Algorithme de principe de la recherche à voisinage variable . . . . .	71
6	Algorithme de principe de la recherche tabou . . . . .	72
7	Algorithme de principe des algorithmes génétiques . . . . .	73
8	Algorithme de principe de l'optimisation par colonie de fourmis . . . . .	74
9	Algorithme de principe de la PSO en optimisation continue . . . . .	77
10	Algorithme de principe de l'heuristique Next Fit . . . . .	83
11	Algorithme de principe de l'heuristique First Fit . . . . .	84
12	Algorithme de principe de l'heuristique Best Fit . . . . .	84
13	Première approche : exploration de $C$ . . . . .	107
14	Seconde approche : exploration de $\Omega$ . . . . .	107
15	Système de voisinage . . . . .	107
16	Détermination de la température initiale $T_0$ . . . . .	108
17	Évaluation de contraintes pour la PSO . . . . .	109
18	Pseudo-code de la PSO pour le BPP . . . . .	110
19	Algorithme de principe de l'hybridation métaheuristique basée individu - algorithme de liste . . . . .	114
20	Système de voisinage $P_i$ . . . . .	115
21	Système de voisinage $P_{i,j}$ . . . . .	115
22	Système de voisinage $I_{i,j}$ . . . . .	115
23	Algorithme de principe l'algorithme hybride du recuit simulé . . . . .	116
24	Algorithme de principe de l'algorithme hybride du kangourou . . . . .	117
25	Initialisation de l'essaim de particules . . . . .	118
26	Principe de la PSO combinatoire hybride . . . . .	119
27	PSOM Mutation . . . . .	119
28	PSOMD Mutation et Dispersion . . . . .	120
29	PSOR dispersion avec influence du Record . . . . .	121
30	Déplacement $PSO1 (Y_{p,t} \oplus V_{p,t+1})$ . . . . .	122
31	Soustraction $(Y_{p,t}^{(2)} \ominus Y_{p,t}^{(1)})$ . . . . .	123
32	Addition $(V_{p,t}^{(1)} \oplus V_{p,t}^{(2)})$ . . . . .	124

33	Multiplication ( $c \otimes V_{p,t}$ ) . . . . .	124
34	Déplacement d'une particule <i>PSO2</i> . . . . .	126
35	Algorithme de principe de l'heuristique Next Fit CHT . . . . .	128
36	Algorithme de principe de l'heuristique First Fit CHT . . . . .	128
37	Algorithme de principe de l'heuristique Best Fit CHT . . . . .	129
38	Algorithme de principe de l'heuristique Worst Fit CHT . . . . .	130
39	Système de voisinage $V$ . . . . .	131

# Introduction

Cette thèse a été réalisée dans le cadre de Péricles : "Programme d'Études et de Recherche en Informatique : Collaborations entre Laboratoires et Établissements de Santé". Cette convention-cadre a permis de créer un partenariat entre les Centres Hospitaliers Régionaux Universitaires de Clermont-Ferrand, Marseille et Montpellier et les laboratoires et universités de ces mêmes secteurs sur les domaines de l'informatique, des systèmes d'information et de l'organisation. Les laboratoires concernés sont le LIMOS CNRS UMR 6158, le LSIS CNRS UMR 6168 et le LIRMM CNRS UMR 5506. Il consiste à identifier les problématiques communes et spécifiques aux établissements hospitaliers en termes d'informatique, de systèmes d'information et d'organisation en vue de mutualiser les compétences des partenaires afin d'apporter des éléments de réponse à ces problématiques par le biais, notamment, de stages et de thèses.

Le 31 décembre 1970, la notion de service public hospitalier a été introduite en France. Elle permet une égalité d'accès, une égalité de traitement et une continuité de service pour tout un chacun ayant besoin de soins hospitaliers. Des réformes hospitalières ont vu le jour depuis afin d'aller vers une meilleure efficacité des systèmes hospitaliers.

De nos jours, le contexte politico-économique oblige le gouvernement à faire des économies là où cela est possible, notamment du côté de la santé. Cela passe par des réglementations sur les modes de financement de la sécurité sociale, la promotion des médicaments génériques, mais également des restrictions budgétaires quant aux dotations pour les Centres Hospitaliers avec l'établissement de la Tarification À l'Activité (T2A). L'accès aux soins étant un droit français, il faut malgré tout pouvoir garantir un service minimum. Des restrictions budgétaires incitent les acteurs des systèmes de santé à trouver les moyens et les solutions d'aller vers une diminution des coûts tout en garantissant le même service.

Depuis le baby-boom d'après guerre, la France, comme d'autres pays, est victime d'un vieillissement de la population. Les meilleures conditions de vie actuelles sont également une des raisons de ce vieillissement de la population. Nous vivons plus longtemps et en meilleure santé. Cependant, il arrive forcément un moment où notre état de santé se dégrade, moment à partir duquel nous sommes amenés à consulter nos médecins de plus en plus souvent. Nous passons plus d'examen médicaux et sommes hospitalisés plus souvent. Cela contribue à une augmentation de l'occupation des Centres Hospitaliers.

Si nous sommes en bonne santé aussi longtemps, c'est parce que nous prenons soin de nous. Nous sommes peut-être même trop attentifs à notre état de santé. En effet, de plus

en plus d'examens médicaux sont prescrits. Ceci pour deux raisons principales :

- Nous voulons nous assurer de notre bonne santé donc demandons plus de prescriptions auprès de nos médecins.
- Les médecins veulent se couvrir en cas d'accident. Pour ne passer à côté d'aucune pathologie, ils prescrivent parfois plus d'examens que nécessaire.

Ceci contribue également à une augmentation de l'occupation des Centres Hospitaliers.

Les ressources sont inégalement occupées sur un même territoire. Certains Centres Hospitaliers sont en sur-occupation alors que d'autres sont en sous-occupation. Ces centres peuvent être géographiquement proches les uns des autres. Dans les premiers, les patients attendent des mois pour obtenir un rendez-vous, alors qu'ils peuvent se rendre dans les seconds presque sans avoir de rendez-vous au préalable.

Les Centres Hospitaliers Universitaires sont localisés dans des grandes villes. Ils sont souvent privilégiés par les patients aux Hôpitaux Locaux situés en périphérie. Ceci peut être dû à une trop forte confiance des patients dans les CHU. Les CHU sont des centres de recherche et sont donc à la pointe de la technologie.

Le personnel hospitalier aussi peut être amené à préférer travailler au sein d'un CHU. Les grosses structures offrent souvent de meilleures conditions de travail, notamment sur la charge horaire. Ceci conduit à une désertification médicale des centres périphériques. Le principe de la libre installation des médecins interdit quiconque à être forcé de combler ces déserts médicaux. La raréfaction des moyens humains n'aide pas à combler ces déserts.

Tous ces contextes mis bout à bout montrent la nécessité de mener des recherches pour améliorer l'efficacité du système hospitalier. Les Centres Hospitaliers ont de moins en moins de moyens financiers dû à des restrictions budgétaires, ils sont de plus en plus sollicités à cause d'un vieillissement de la population et à cause d'une population plus soucieuse de sa santé. Ces Centres Hospitaliers sont occupés de façon inégale sur un même territoire.

Pour remédier à cela, le gouvernement français a exploré plusieurs pistes au cours des dernières années. Plusieurs réformes de santé ont vu le jour, notamment le Plan Hôpital 2007 instauré en 2003, puis le Plan Hôpital 2012 instauré en 2007 et le dernier en date la loi Hôpital, Patients, Santé et Territoires instaurée en 2009.

Une des propositions de ces différentes réformes est la coopération entre les Centres Hospitaliers. Les principales formes de coopération sont la Communauté Hospitalière de Territoire, le Groupement de Coopération Sanitaire, le Groupement d'Intérêt Économique, le Groupement d'Intérêt Public, etc. Chacune de ces formes de coopération diffère par ses objectifs ainsi que par sa forme juridique. Certaines n'autorisent que la coopération entre des établissements publics alors que d'autres autorisent la coopération avec des établissements privés. En particulier, la Communauté Hospitalière de Territoire permet une collaboration entre des établissements publics, qui par une mise en commun de leurs ressources visent à atteindre un objectif commun. C'est par la mutualisation des ressources hospitalières que l'amélioration de l'efficacité du système hospitalier pourra être atteinte. Le manque d'un instrument de planification au service de la recomposition de l'offre hospitalière, de l'efficacité et de l'amélioration du parcours du patient pour favoriser l'ancrage territorial de la coopération se fait sentir.

L'objectif des travaux de thèse présentés est la mise au point d'un outil d'aide à la

décision pour aider à la gestion des Communautés Hospitalières de Territoire ou de toute autre forme de coopération hospitalière. Cet outil doit pouvoir être utilisé à plusieurs niveaux : stratégique, tactique ou opérationnel, pour traiter des problèmes de dimensionnement, de planification, d'affectation de ressources ou d'ordonnancement. Il doit pouvoir être utilisé pour l'anticipation de la création d'une nouvelle coopération, la gestion de l'organisation de cette coopération au quotidien, ou la réaction à l'apparition d'aléas ou de situation de crise. Ce manuscrit est divisé en cinq chapitres.

La problématique étudiée tout au long de nos travaux est présentée par le Chapitre 1. Le contexte actuel a incité le gouvernement à proposer des réformes hospitalières. La notion de coopération hospitalière a été introduite par ces différentes réformes. Ces coopérations existent sous plusieurs formes : nous pouvons citer les Communautés Hospitalières de Territoire purement publiques et les Groupements de Coopération Sanitaire qui autorisent des coopérations public-privées. Les différents problèmes qui peuvent être considérés par nos travaux sont définis : des problèmes de dimensionnement de système, de planification d'activités, d'affectation de ressources ou d'ordonnancement.

Le Chapitre 2 présente l'analyse du domaine hospitalier. Le problème de planification d'activités avec affectation de ressources est défini. Ce problème est décomposé en trois sous-problèmes : les ressources humaines sont à capacité infinie, les ressources humaines sont à capacité limitée et leur affectation sur site est une donnée, les ressources humaines sont à capacité limitée et leur affectation sur site devient une variable. L'étude de ces problèmes passe par la formalisation de modèles mathématiques avec le détail des critères à optimiser et des contraintes à respecter. Ces formalisations nous permettent de faire le lien entre nos problèmes et le problème du bin packing qui est un problème très étudié en recherche opérationnelle. Une analogie entre ces différents problèmes est réalisée.

L'état de l'art est réalisé par le Chapitre 3. De nombreuses méthodes de résolution exactes ou approchées existent pour la résolution de Programmation Linéaire en Nombres Entiers. Une présentation de ces méthodes de résolution est faite. En particulier, les différentes approches déjà utilisées pour la résolution du problème du bin packing sont présentées. Une nouvelle métaheuristique a récemment vu le jour : l'optimisation par essaim particulaire. Elle a été utilisée pour la résolution de problèmes d'optimisation continue ou combinatoire et nous a permis de proposer un cadre de résolution de notre problème.

Nous proposons des nouvelles méthodes de résolution du problème du bin packing dans le Chapitre 4 en concentrant nos efforts à la définition de méthodes simples et efficaces. Les méthodes proposées utilisent un couplage hiérarchique entre une heuristique et une métaheuristique. Des métaheuristicités basées individu ainsi que l'optimisation par essaim particulaire sont utilisées. Ces méthodes sont aisément adaptées à la résolution du problème de planification d'activités avec affectation de ressources grâce aux analogies présentées dans le Chapitre 2.

Le Chapitre 5 présente les différentes expérimentations réalisées, que ce soit sur le problème du bin packing ou le problème de planification d'activités avec affectation de ressources. Nos méthodes permettent d'obtenir de très bons résultats pour ces deux problèmes qui sont détaillés en annexe de ce manuscrit.

Ce manuscrit est achevé par une conclusion et des perspectives de recherche.





# Chapitre 1

## Présentation de la problématique

### Sommaire

---

<b>1.1</b>	<b>Introduction</b>	<b>7</b>
<b>1.2</b>	<b>Contexte hospitalier</b>	<b>7</b>
1.2.1	Définitions	7
1.2.2	Plans hôpitaux 2007 et 2012	9
1.2.3	Rapport Larcher	10
1.2.4	Loi HPST	10
<b>1.3</b>	<b>Réseaux de soin et coopérations multi-sites</b>	<b>11</b>
1.3.1	Communauté Hospitalière de Territoire	11
1.3.1.1	Définition	11
1.3.1.2	Chronologie de la mise en place des CHT	13
1.3.1.3	Exemples de CHT	17
1.3.1.4	Conclusion sur les CHT	18
1.3.2	Fédération d’Imagerie Médicale	19
1.3.2.1	L’examen	19
1.3.2.2	Le patient	20
1.3.2.3	Les sites et services	20
1.3.2.4	Les équipements	20
1.3.2.5	Les ressources humaines	20
1.3.3	Autres exemples de coopérations	21
1.3.3.1	Groupement de Coopération Sanitaire	21
1.3.3.2	Groupement d’Intérêt Économique	22
1.3.3.3	Groupement d’Intérêt Public	22
1.3.3.4	Groupement de Coopération Sociale ou Médico-Sociale	22
1.3.3.5	Syndicat InterHospitalier	23
<b>1.4</b>	<b>Recherche opérationnelle pour les réseaux de soin</b>	<b>23</b>
1.4.1	Dimensionnement	23
1.4.2	Planification	24
1.4.3	Ordonnancement	26
1.4.4	Réseaux hospitaliers multi-sites	27

1.4.5	Conclusion . . . . .	29
<b>1.5</b>	<b>Objectifs de la thèse . . . . .</b>	<b>29</b>
1.5.1	Niveaux de décision . . . . .	29
1.5.2	Niveaux de modélisation . . . . .	29
1.5.3	Définition des problèmes . . . . .	30
1.5.4	Outil d'aide à la décision . . . . .	32
<b>1.6</b>	<b>Conclusion . . . . .</b>	<b>34</b>

---

## 1.1 Introduction

Étant donnée la situation économique actuelle, tout est fait pour aller vers une meilleure utilisation des systèmes de production de biens et de services. Le système hospitalier suit également cette tendance : autant ou de moins en moins de moyens lui sont alloués mais il doit fonctionner de façon de plus en plus efficiente face à une demande qui ne cesse d'augmenter.

En effet, le contexte démographique engendre une augmentation de l'utilisation du système hospitalier. L'Homme vit plus longtemps et dans de meilleures conditions, ce qui tend à augmenter la population terrestre. Même si l'Homme vit mieux, le nombre de passages dans les centres hospitaliers ne fait que croître. Nous sommes plus attentifs à notre corps donc avons tendance à consulter de plus en plus.

C'est par la coopération et la mise en commun des ressources et des moyens qu'une meilleure efficacité du système hospitalier pourra être atteinte. Plusieurs formes de réseaux de soin ont été définies par le gouvernement. Par exemple, la Communauté Hospitalière de Territoire a été créée en 2008. Une CHT est une coopération entre établissements publics qui mettent en œuvre une stratégie commune et gèrent en commun certaines fonctions et activités grâce à des délégations ou des transferts de compétences entre eux. De nombreuses CHT ont été créées ces dernières années et d'autres sont en cours de création en France. Mais elles manquent d'outils pour gérer leur fonctionnement. Des applications de ces CHT existent, c'est le cas notamment de la Fédération d'Imagerie Médicale sur le bassin clermontois. Cette fédération consiste en la mise en commun de ressources entre plusieurs établissements au niveau du secteur de l'imagerie médicale. D'autres formes de réseaux existent, par exemple le Groupement de Coopération Sanitaire qui lui permet une convention entre des établissements publics et privés.

La Section 1.2 présente certaines définitions utilisées tout au long de ce manuscrit. Le contexte hospitalier actuel a conduit à quelques réformes gouvernementales présentées dans cette même section. Ces réformes ont mené à la définition de différentes formes de coopération définies dans la Section 1.3. Un état de l'art sur la recherche opérationnelle au service du système hospitalier est réalisé dans la Section 1.4. La Section 1.5 présente les différents objectifs de cette thèse à travers les différents problèmes considérés dans ce manuscrit.

## 1.2 Contexte hospitalier

### 1.2.1 Définitions

Certaines définitions méritent d'être précisées en préambule.

Un **établissement de soin** est le lieu où sont prodigués des actes de soin.

Le **système hospitalier** désigne d'une façon générale l'ensemble des établissements de soin. Le secteur hospitalier français fait cohabiter des établissements de différents types de statut juridique, combinant différents modes d'organisation et de gestion, de financement et de régulation, ainsi que différents modes de participation aux missions de service public. Les statuts des personnels travaillant dans ces établissements sont également variés. Cette diversité, essentiellement héritée de l'histoire, reste aujourd'hui encore très structurante de l'organisation de l'offre de soin hospitalière et en particulier de la répartition des lits

et de l'activité (Rodier, 2010).

Une première distinction est faite entre secteur hospitalier public et privé. La loi hospitalière de 1970 amène une nouvelle distinction en introduisant la notion de service public hospitalier dont les missions ne sont pas assurées uniquement par le secteur public. Les établissements qui participent au service public hospitalier sont distingués de ceux qui n'y participent pas :

- Les établissements publics représentent en France environ 65% des capacités d'accueil.
- Les établissements Privés à but Non Lucratif représentent environ 28% des 3000 hôpitaux français et 15% des capacités d'accueil. Ce sont en majorité des établissements Participant au Service Public Hospitalier (PSPH).
- Les établissements Privés Lucratifs (PL) dont la place relative tend à augmenter représentent les 20% restant.

Le système hospitalier est donc représenté par de nombreux établissements. Chacun de ces établissements est localisé sur un **site**.

Le **Centre Hospitalier (CH)** est le terme général pour désigner un établissement de soin. Un établissement de soin public peut être représenté sous différentes formes : cet établissement peut être un Centre Hospitalier Universitaire (CHU) ou un Hôpital Local (HL). Un établissement de soin privé peut être une clinique privée. Un Centre Hospitalier peut être réparti sur plusieurs sites.

Un Centre Hospitalier peut être composé de plusieurs **services hospitaliers** tels que l'imagerie médicale, les blocs opératoires, la chirurgie ambulatoire, la maternité, la cardiologie, etc. Ces services peuvent être localisés sur des sites différents.

Un **réseau hospitalier** est composé de plusieurs Centres Hospitaliers. Il est par définition **multi-site**, c'est-à-dire que l'ensemble de ses entités le composant est réparti sur plusieurs sites. Ce réseau permet une coopération entre les établissements en faisant partie. Ce réseau peut aussi être désigné par le terme **réseau de soin** : un réseau où sont prodigués des soins.

La **mutualisation** est le partage par des individus ou groupe d'individus, de biens, de logements, d'équipements (ménagers, outils, moyens informatiques, etc) ou de moyens de transport de manière à optimiser l'accès à ces ressources et optimiser leur rentabilité (par le partage des coûts, frais d'entretien, assurance, réparation, etc.). Afin d'aller vers une meilleure efficacité des réseaux hospitaliers, une mutualisation de leurs ressources peut être envisagée.

La **coopération** est l'action de coopérer, de collaborer à une action commune. Les Centres Hospitaliers peuvent coopérer via une mutualisation de leurs ressources.

Une coopération entre différents Centres Hospitaliers permettrait une meilleure occupation des ressources et des actes réalisés plus rapidement. Les ressources humaines sont concentrées sur un Centre Hospitalier où les temps d'attente sont longs ? Pourquoi ne pas déplacer ces ressources humaines vers les Centres Hospitaliers périphériques ? Les patients se dirigent tous vers le même Centre Hospitalier, souvent malgré eux ? Pourquoi ne pas concevoir une plateforme informatique qui permettrait de diriger le patient au meilleur endroit, en lui expliquant notamment que s'il persiste à aller dans un centre précis il devra patienter quelques mois alors qu'il pourrait obtenir un rendez-vous dès le lendemain dans

un autre Centre Hospitalier ?

Certaines réformes gouvernementales ont été mises en place ces dernières années afin d'améliorer l'efficacité du système hospitalier, notamment par la proposition de nouvelles formes de coopération. Ces réformes sont représentées sur la Figure 1.1.

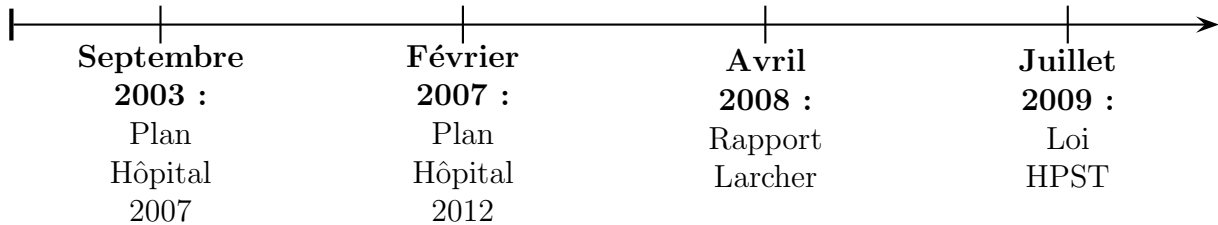


FIGURE 1.1 – Chronologie des réformes gouvernementales

### 1.2.2 Plans hôpitaux 2007 et 2012

En 2003 et 2007 se succèdent deux séries de mesures gouvernementales visant à l'amélioration du système hospitalier français : le plan hôpital 2007 annoncé le 4 septembre 2003 par le ministre de la santé Jean-François Mattéi et le plan hôpital 2012 annoncé le 13 février 2007 par le ministre de la santé Xavier Bertrand. Ces plans visent entre autres à une diminution des dépenses de santé qui ne cessent d'augmenter depuis les dernières décennies.

Les deux principales mesures du plan hôpital 2007 sont :

- La nouvelle gouvernance hospitalière avec l'amélioration de la concertation entre les médecins et les directeurs, ainsi que la création de pôles d'activité.
- La Tarification À l'Activité (T2A) visant à lier les moyens alloués à un établissement en fonction de son activité, ce qui permet de réglementer les dotations allouées à chaque Centre Hospitalier.

Le plan hôpital 2012 va dans la continuité du plan hôpital 2007. Trois des mesures de ce dernier plan sont :

- Un maintien du niveau d'investissement pour les établissements de santé publics et privés.
- L'accompagnement de recompositions hospitalières, de regroupements de plateaux techniques, de partenariats entre établissements publics et privés par la mise en œuvre des Schémas Régionaux d'Organisation Sanitaire (SROS).
- Le développement du Système d'Information Hospitalier.

Les Schémas Régionaux d'Organisation Sanitaire (SROS) ont été créés en 1991. Le SROS détermine l'organisation territoriale des moyens de toute nature, compris ou non dans la carte sanitaire, qui permettra la réalisation des objectifs qu'il fixe. Il peut comporter des recommandations pour la réalisation de ces objectifs. Le SROS procède d'une démarche d'amélioration de la prise en compte par le système de soin des besoins de santé et de promotion de la coordination entre le secteur hospitalier, la médecine de ville et le

secteur médico-social. Le SROS est un document de cadrage des grands axes de la recombinaison hospitalière dans une région. Les objectifs qu'il retient visent à corriger les principaux dysfonctionnements que le bilan de l'existant et les différents diagnostics effectués par les agences régionales de l'hospitalisation ont révélés. Le SROS indique les cibles à atteindre mais laisse des marges de manœuvre aux établissements de santé sur les modalités. Sa mise en œuvre s'appuie sur les contrats d'objectifs et de moyens qui sont négociés entre l'agence régionale de l'hospitalisation et chaque établissement de santé comme sur la délivrance des autorisations ou encore l'approbation des projets d'établissements ou l'agrément des réseaux.

Il doit permettre la rationalisation de l'offre hospitalière, par :

- L'instauration de plateaux techniques sophistiqués,
- La réflexion sur des formes alternatives de soin, comme l'Hospitalisation À Domicile (HAD),
- L'optimisation des moyens, notamment la complémentarité des établissements de santé, particulièrement en ce qui concerne les équipements lourds, comme les scanners,
- L'amélioration de la qualité des soins.

Les Schémas Régionaux d'Organisation Sanitaire existent jusqu'en 2009 : la loi HPST les font devenir Schémas Régionaux d'Organisation des Soins (et non plus sanitaire).

### 1.2.3 Rapport Larcher

Le 11 avril 2008, l'ancien président de la Fédération Hospitalière de France (FHF), Gérard Larcher, remet à Nicolas Sarkozy, alors président de la République, son rapport sur la nouvelle réforme hospitalière qui est à l'origine de la loi HPST. Il préconise le regroupement d'établissements dans le but de ramener les hôpitaux publics à l'équilibre budgétaire en 2012, alors qu'ils affichent un déficit cumulé de huit cents millions d'euros chaque année. Le but de ces regroupements d'hôpitaux est la mutualisation de moyens. Ces regroupements pourraient se faire dans le cadre de Communautés Hospitalières de Territoire encouragées par l'allocation prioritaire des crédits d'investissement du plan hôpital 2012. Dans ces CHT, les établissements de grande taille assureraient les soins de pointe, et les hôpitaux locaux de petite taille concentreraient leurs activités dans les soins de suite et la gériatrie. Trois cents CHT seraient ainsi créées, regroupant le millier de centres hospitaliers actuels.

La CHT publique serait le levier d'une hospitalisation publique cohérente proche de la population, garante de la qualité des soins sur un territoire et partenaire du privé.

### 1.2.4 Loi HPST

À la suite de la remise du rapport Larcher, une loi a été promulguée par Roselyne Bachelot, ministre de la santé : la loi du 21 juillet 2009 dite "Hôpital, Patients, Santé et Territoires" (HPST). C'est un projet d'organisation sanitaire et non de financement. À terme, elle doit permettre la mise en place d'une offre de soin graduée, de qualité, accessible à tous, satisfaisant à l'ensemble des besoins de santé.

Les principales mesures de la loi HPST sont :

- Une organisation régionale de la santé, par la mise en place des Agences Régionales de la Santé qui coordonnent les hôpitaux, la médecine de ville et le secteur médico-social.
- Une nouvelle gouvernance des hôpitaux, via le conseil de surveillance, le directoire et les contrats pluriannuels d'objectifs et de moyens. L'objectif est de faciliter les coopérations entre les établissements publics et privés.
- Un accès aux soins facilité pour tous, en repeuplant les déserts médicaux tout en respectant le principe de la liberté d'installation des personnels libéraux.

La loi HPSPT modifie les SROS en apportant les deux innovations suivantes :

- Leur intégration au sein des "Projets Régionaux de Santé" (PRS),
- L'extension de leur champ d'application à l'offre de soins ambulatoire.

Afin de marquer la spécificité des SROS issus de la loi HPST, ceux-ci sont dénommés "SROS-PRS".

La loi HPST définit deux formes de coopération : les Communautés Hospitalières de Territoire qui intègrent uniquement des établissements publics et les Groupements de Coopération Sanitaire qui permettent d'associer les établissements publics aux établissements privés.

## 1.3 Réseaux de soin et coopérations multi-sites

Plusieurs formes juridiques de réseaux de soin existent à ce jour. Au sein de ces réseaux, les coopérations multi-sites sont réglementées. Les coopérations les plus fréquentes sont citées dans cette partie.

### 1.3.1 Communauté Hospitalière de Territoire

La notion de Communauté Hospitalière de Territoire (CHT) a été introduite en France en 2008. La CHT est une coopération entre plusieurs Centres Hospitaliers publics dans le but d'accomplir des objectifs communs à ces centres. À ce jour, des CHT sont toujours en cours de formation.

#### 1.3.1.1 Définition

La définition donnée par le site du ministère de la santé [www.sante.gouv.fr](http://www.sante.gouv.fr) est la suivante : "la Communauté Hospitalière de Territoire (CHT) est une innovation de la loi Hôpital, Patients, Santé, Territoires (HPST). Sa finalité est la recherche de la meilleure utilisation des ressources à disposition des établissements ainsi que la complémentarité entre les acteurs. Elle permet aux établissements publics de santé de conclure une convention afin de mettre en œuvre une stratégie commune et de gérer en commun certaines fonctions."

La CHT est un outil efficace qui permettra d'assurer la filière de soin. Elle accompagne les rapprochements publics en se fondant sur un projet d'activités médicales commun. La CHT a des compétences obligatoires, notamment en matière de coopération avec la médecine de proximité.



Les deux principaux objectifs de la création des CHT sont :

- Conjuguer au mieux proximité, qualité et sécurité, grâce à une gradation des soins et à la recherche de complémentarités entre établissements.
- Renforcer la démarche de performance dans laquelle doit s’inscrire l’ensemble des établissements.

Les enjeux stratégiques sont la garantie de l’accessibilité et de la continuité du service public, le renforcement de la position stratégique de l’offre, l’optimisation des investissements, le maintien de la motivation des acteurs et la meilleure valorisation des ressources médicales pour garantir une situation financière équilibrée dans la durée.

Une CHT n’a pas de personnalité morale, ce qui pose problème au niveau du pilotage de recrutements inter-hôpitaux. Un praticien hospitalier ne peut pas être recruté communément à plusieurs sites, et donc par une CHT. Les CHT ne sont pas adaptées à la territorialisation des recrutements médicaux. La nécessité d’une nouvelle gouvernance dans les CHT se fait sentir.

D’une façon schématique, la CHT peut être représentée par la Figure 1.2. Cette CHT est composée de trois Centres Hospitaliers. Chaque centre est situé sur un site distinct. Le premier et le troisième sont pourvus de ressources matérielles et humaines alors que le second n’est équipé qu’en ressources matérielles. Les cercles autour du Centre Hospitalier représentent le rayon à cinquante kilomètres autour de ce centre.

La supposition est faite que le site 2 est moins occupé que les autres par manque de moyens humains. En mutualisant les ressources humaines au sein de ce territoire, c’est-à-dire en déplaçant ponctuellement des ressources humaines des sites 1 et 3 vers le site 2, on peut espérer augmenter les taux d’occupation de ce site et désengorger les deux autres, sous-réserve que les sites 1 et 3 soient sur-occupés.

De plus, la mutualisation des ressources humaines peut aider au confort du patient. Le patient qui vit dans le rayon englobant le site 2 ne peut pas être traité sur le site 2 parce qu’il n’y a pas les ressources humaines nécessaires. Il doit se déplacer sur le site 1 qui lui dispose de toutes les ressources nécessaires. Si certaines des ressources du site 1 se déplacent vers le site 2, le patient pourrait être traité sur le site 2 et n’aurait pas à se déplacer.

Une liste non exhaustive des objectifs de la CHT peut être :

- Aider les CH de proximité et créer un partenariat gagnant-gagnant qui ne les pénalise pas mais conserve leur autonomie.
- Rechercher des mutualisations et des complémentarités dans le champ du médico-technique, notamment de l’imagerie.
- Promouvoir des actions de coopération dans les domaines logistiques mais aussi administratifs.
- Renforcer l’égalité des soins des habitants dans le territoire.
- Améliorer la fluidité du parcours des patients en organisant des filières de soin pour l’ensemble des disciplines cohérentes entre les établissements de la CHT.
- Identifier les différentes étapes du parcours patient afin d’organiser la coordination entre les différents acteurs.
- Mettre en place des outils logiciels standardisés et communs pour fluidifier le parcours de soin en travaillant sur l’aval et la régulation. Ces outils doivent permettre

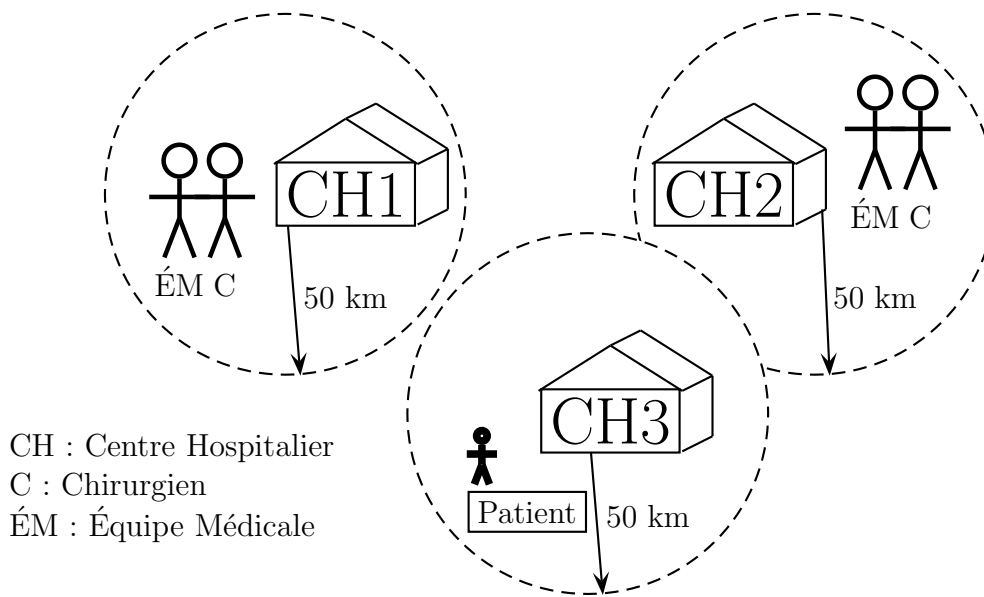


FIGURE 1.2 – Exemple d’une CHT

d’orienter le patient vers la bonne structure.

- Mettre en commun les différentes ressources et coordonner la gestion des personnels médicaux.

Par exemple, concernant la filière gériatrie, la CHT peut se donner pour objectif de privilégier les entrées directes et programmées en ciblant mieux le profil des personnes âgées. Supposons que les différents services sont répartis sur plusieurs sites de la manière suivante : les services de spécialités situés au CHU et le service gériatrique dans un hôpital local. En fonction de la pathologie du patient, qui est déterminée grâce à un suivi régulier du patient, le médecin traitant pourrait orienter le malade directement vers le bon site. À l’heure actuelle, les personnes âgées sont souvent dirigées vers le service des urgences sans distinction. À travers la CHT, les urgences peuvent être désengorgées par la mise en place d’outils communs et une coopération entre les différents sites.

### 1.3.1.2 Chronologie de la mise en place des CHT

La chronologie des CHT au travers des réglementations et rapports rendus au gouvernement depuis 2008 est dressée. Ces rapports donnent leur point de vue sur la CHT. Bien que les aspects mutualisation et coopération sont constamment évoqués, le terme CHT n’est cependant pas toujours cité.

La Figure 1.3 fait un récapitulatif des textes législatifs ou des différents rapports rendus par divers organismes depuis la création des CHT.

La notion de Communauté Hospitalière de Territoire a été introduite par Gérard Larher. Il préconise le regroupement d’établissements pour mutualiser les moyens.

**Juillet 2009 : Loi HPST** D’après la loi Hôpital, Patients, Santé, Territoires du 21 juillet 2009, des établissements publics de santé peuvent conclure une convention de Communauté Hospitalière de Territoire afin de mettre en œuvre une stratégie commune et de

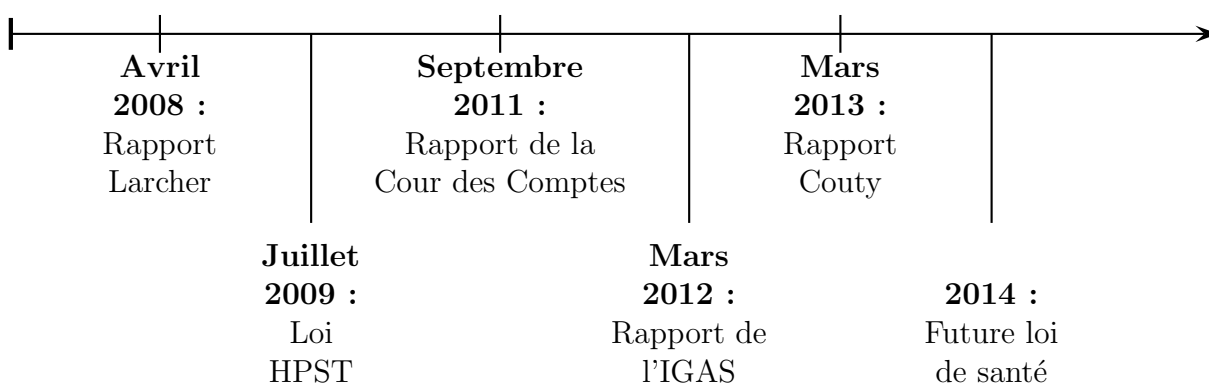


FIGURE 1.3 – Chronologie depuis la création des CHT

gérer en commun certaines fonctions et activités grâce à des délégations ou des transferts de compétences entre les établissements et grâce à la télémédecine. Un établissement public de santé ne peut prendre partie qu'à une seule convention de CHT.

La convention de CHT définit :

- Le projet médical commun de la CHT et les compétences et activités qui seront déléguées ou transférées entre les établissements partenaires ainsi que, le cas échéant, les cessions ou échanges de biens meubles et immeubles liés à ces délégations ou transferts ;
- Les modalités de mise en cohérence des contrats pluriannuels d'objectifs et de moyens, des projets d'établissement, des plans globaux de financement pluriannuels et des programmes d'investissement des établissements ;
- Les modalités de coopération entre les établissements en matière de gestion et les modalités de mise en commun des ressources humaines et des systèmes d'information hospitaliers ;
- Les modalités de fixation des frais pour services rendus acquittés par les établissements en contrepartie des missions assumées pour leur compte par certains d'entre eux ;
- Le cas échéant, les modalités d'articulation entre les établissements publics de santé signataires de la convention et les établissements médico-sociaux publics participant aux actions menées dans le cadre de la convention de CHT ;
- La composition du conseil de surveillance, du directoire et des organes représentatifs du personnel de l'établissement siège de la CHT, qui comprennent chacun des représentants des établissements faisant partie de la convention.

La loi du 10 août 2011 modifie certaines dispositions de la loi du 21 juillet 2009 mais ces dispositions ne touchent pas à la notion de CHT.

**Septembre 2011 : Rapport de la Cour des Comptes** En septembre 2011, la Cour des Comptes a rédigé un rapport sur la Sécurité Sociale (Cours des comptes, 2011). La coopération entre établissements par le biais de CHT y est évoquée de la manière suivante.

Les établissements poursuivent à travers ces coopérations certaines finalités comme la répartition des ressources rares, qu'elles soient humaines, scientifiques, matérielles ou financières :

- Afin de partager et rationaliser la charge de la permanence des soins,

- Ou au contraire pour intensifier l'utilisation de ressources surdimensionnées, notamment pour le secteur de l'imagerie médicale.

La CHT est un outil de rationalisation territoriale du système. Elle permet d'assurer la pérennité de l'offre de proximité en évitant :

- La fermeture de services voire d'établissements,
- La suppression des doublons,
- La mutualisation des ressources rares.

La complémentarité, qu'elle s'entende horizontalement entre établissements de même niveau ou verticalement au long de filières, est un objectif affiché de nombreuses coopérations. Un modèle fréquent de coopération consiste en le partage du temps des médecins entre établissements, notamment sous la forme de consultations avancées des praticiens d'un hôpital dans un hôpital partenaire. Les CHT mettront en œuvre une stratégie commune aux établissements participants et géreront en commun des fonctions et des activités grâce à des délégations ou des transferts de compétences. Les coopérations peuvent aboutir au rapprochement de l'offre de soin vers le patient.

**Mars 2012 : Rapport de l'IGAS** En mars 2012, un rapport intitulé "Fusions et regroupements hospitaliers : quel bilan pour les quinze dernières années?", a été rédigé par l'Inspection Générale des Affaires Sociales (IGAS).

Ce rapport préconise une incitation au développement de complémentarités entre hôpitaux publics sous la forme de CHT avec un projet médical commun intégrant le court séjour. Une incitation forte à des rapprochements volontaires doit être maintenue. La mise en commun des ressources médicales rares ou pour constituer des équipes permet de répondre aux nouvelles conditions techniques de fonctionnement.

En effet, c'est pour offrir un cadre stratégique englobant l'ensemble des activités que la CHT a été créée. Elle permet grâce à une stratégie commune de gérer certaines activités et fonctions :

- L'organisation des délégations ou transferts de compétences entre établissements partenaires, avec éventuellement des transferts d'actifs.
- La mise en cohérence de leurs orientations stratégiques au niveau des investissements et financements.
- La définition de leurs modalités de coopération en matière de gestion et de mise en commun des ressources humaines et des systèmes d'information hospitaliers.
- L'organisation de la gradation des soins, et la répartition des activités entre les CH d'un territoire.

Une fois encore, le rapport souligne que la CHT n'a pas la personnalité morale et qu'un établissement ne peut pas être membre de plusieurs CHT.

Pour favoriser l'ancrage territorial de la coopération au sein de la CHT, il faudrait un instrument de planification au service de la recomposition de l'offre hospitalière, de l'efficacité, et de l'amélioration du parcours du patient. Le rapport déplore qu'il n'existe pas à ce jour d'outil de coopération permettant cette solidarité financière entre les membres de la CHT.

**Décembre 2012 - Février 2013** De mars 2012 à mars 2013, le terme CHT n'est plus employé par le gouvernement. Pourtant des aspects tels que la coopération semblent toujours au cœur de leur préoccupation.

Fin 2012, la ministre de la santé Marisol Touraine met en place le Pacte Territoire Santé. Elle constitue un plan global de lutte contre les déserts médicaux. Un de ses engagements est l'investissement dans les territoires isolés par l'adaptation des hôpitaux de proximité et la responsabilisation des Centres Hospitaliers de niveau régional à l'égard de leur territoire. Pour cela, seront développées des vacations de spécialistes. Le parcours de soins sera articulé et amélioré. Il faut mobiliser la responsabilité territoriale des Centres Hospitaliers de niveau régional pour appuyer l'action des centres de proximité.

En février 2013, le premier ministre Jean-Marc Ayrault fait deux discours. Le premier sur la stratégie nationale de santé porte en partie sur l'organisation du parcours de soin. La médecine ne doit plus être conçue comme une succession d'actes ponctuels. Il faut créer une médecine de parcours qui repose sur la coopération de tous les professionnels et sur l'implication des patients. Dans son second discours, il préconise un territoire de soins numériques permettant notamment l'amélioration de l'organisation, du processus de prise en charge et de soins, de l'utilisation des ressources ainsi que l'amélioration de la coordination des opérateurs de santé dans la production de soins et dans la prise en charge des patients.

**Mars 2013 : Rapport Couty** En parallèle du Pacte Territoire Santé, la ministre de la santé Marisol Touraine a engagé, le 7 septembre 2012, les travaux du pacte de confiance pour l'hôpital. En mars 2013, Édouard Couty remet son rapport à la ministre (Couty, 2013). L'objectif principal est de réaffirmer et de reconstruire le service public hospitalier centré sur le parcours du patient ou de la personne prise en charge, ouvert sur son environnement intégré dans un dispositif plus large sur un territoire. Le rapport vise à la création d'un service public territorial de santé.

En vue d'une simplification de la création de coopérations, le rapport conseille de ne pas inviter ni inciter les établissements à entrer dans des CHT. Parfois ces coopérations sont constituées dans un climat de tension, sinon de défiance, ce qui n'arrange pas leur fonctionnement ultérieur. Il vaut mieux fixer le cap de ces coopérations, leur indiquer l'objectif en matière de coordination des parcours, de répartition de l'offre de soin sur le territoire. Les coopérations ne doivent pas être une obligation mais une proposition naturelle des établissements entre eux.

Le fait de donner la personnalité morale aux CHT n'est toujours pas envisagé.

**Avril 2014 : Rénovation du service public hospitalier** Suite au rapport Couty, le gouvernement commande trois nouveaux rapports, dont celui sur la rénovation du service public hospitalier. En avril 2014, Bernadette Devictor remet ainsi son rapport intitulé "le SPTS (Service Public Territorial de Santé), le SPH (Service Public Hospitalier) : développer l'approche territoriale et populationnelle de l'offre en santé". Elle conseille d'introduire une obligation de coopération à plusieurs dimensions pour les acteurs du Service Public Hospitalier :

- Coopération (mutualisation des compétences, optimisation des équipements sanitaires et des plateaux techniques) entre les établissements hospitaliers d'un même territoire ou au-delà selon les besoins d'orientation du patient,
- Coopération avec les acteurs de santé des secteurs sanitaire, médico-social et social au sein des SPTS.

Dans le même temps, la FHF demande que l'intégration de tous les Centres Hospitaliers à une CHT devienne obligatoire.

**Juin 2014 : Future loi de santé** En juin 2014, Marisol Touraine présente sa future loi de santé, tenant compte de tous les rapports qu'elle a précédemment reçus. Cette loi est présentée au conseil d'état l'été de la même année, puis au conseil des ministres en septembre 2014 pour être débattue à l'Assemblée Nationale début de l'année 2015.

À la grande surprise, les Communautés Hospitalières de Territoire n'y sont pas présentes. Les CHT sont explicitement remplacées par les Groupements Hospitaliers de Territoire. Dans le code de la santé publique, tous les termes CHT sont ainsi remplacés par GHT. Ce changement de terminologie permet au gouvernement de prendre un nouveau départ sur la notion de coopération entre Centres Hospitaliers. Certains le considèrent comme un aveu de la part du gouvernement de l'échec des CHT.

L'un des objectifs de la création des GHT est le renforcement de la coopération hospitalière. Chaque Centre Hospitalier a l'obligation d'intégrer un GHT avant le 31 décembre 2015. Sinon, il verra la fin de ses dotations de financement.

Le GHT aura en charge la gestion du Système d'Information Hospitalier (SIH), le Département d'Information Médicale (DIM) et la facturation des actes médicaux, la politique d'amélioration continue de la qualité et de la sécurité des soins, la gestion de la Pharmacie à Usage Intérieur (PUI), le contrôle de gestion, l'organisation de la permanence des soins et la politique d'achats. Par ailleurs, un des membres du GHT peut gérer les activités administratives et logistiques du groupe. Un établissement membre peut déléguer à un autre établissement membre du même GHT la gestion des équipements d'intérêts communs, les interventions communes de professionnels de santé, la mise en commun de services et plateaux médico-techniques, l'exploitation d'un laboratoire de biologie médicale unique, l'organisation des activités d'enseignements et de recherche et la délégation ou le transfert d'activités entre ses membres.

Désormais, sur un même territoire, les établissements porteront un projet médical commun.

Des professionnels de santé soulignent une difficulté de ce nouveau texte : quid du secret médical dans le cadre de la mutualisation des DIM dans les GHT. Se pose également l'interrogation sur la pérennité des coopérations existantes et celle des coopérations futures. Une autre interrogation concerne le système de gouvernance à mettre en place : il faut doter ces GHT de véritables organes de décision (conseil exécutif, commission médicale, organisation de surveillance).

### 1.3.1.3 Exemples de CHT

En juin 2010, les ARS déclaraient 29 CHT en France. Au dernier recensement en date d'août 2013, il y en avait 76. Des CHT voient régulièrement le jour. Leur principe est toujours évoqué dans les actualités récentes. Nous pouvons citer les deux CHT suivantes :

**La CHT 36** regroupe sept établissements :

- Centre Hospitalier de Châteauroux,
- Centre Hospitalier du Blanc,
- Centre Hospitalier de Buzançais,

- Centre Hospitalier de Châtillon-sur-Indre,
- EHPAD d’Argenton-sur-Creuse,
- EHPAD de Clion-sur-Indre,
- EHPAD de Mézière-en-Brenne.

Elle s’appuie sur le déploiement de projets existants, particulièrement en matière d’organisation de la permanence des soins dans des disciplines critiques telles que l’imagerie, les urgences et l’obstétrique. Elle s’inscrit également dans un projet communautaire innovant autour d’activités structurantes notamment dans la prise en charge de la personne âgée.

Par ailleurs, l’élaboration d’un projet médical commun aux établissements membres inclut des dispositifs de télémedecine, des mutualisations d’équipes et d’équipements.

Enfin, la création de cette communauté facilite la mise en cohérence des systèmes d’information des établissements partenaires facilitant ainsi les échanges de données médicales.

Les bénéfices directs attendus consistent à faciliter l’accès à des soins performants dans le respect des tarifs remboursables et en un partage de compétences, d’expertises et de technologies au sein des équipes médicales, contribuant ainsi au maintien et à l’amélioration de la qualité des soins sur le territoire.

Les établissements s’engagent ainsi à œuvrer ensemble en inscrivant leurs actions dans le cadre d’une stratégie partagée au service de la population du département de l’Indre.

**La CHT du Grand Clermont** est composée de six établissements :

- Centre Hospitalier Universitaire de Clermont-Ferrand,
- Centre Hospitalier de Riom,
- Centre Hospitalier d’Issoire,
- Centre Hospitalier Étienne-Clémentel d’Enval,
- Hôpital Local du Mont-Dore,
- Hôpital Local de Billom.

Elle fédère des établissements exerçant des activités différentes et complémentaires, qu’il s’agisse de disciplines universitaires (la neurochirurgie, par exemple) ou d’activités médicales de proximité, sans exclure un volet médico-social important, avec des Établissements d’Hébergement de Personnes Âgées Dépendantes (EHPAD) notamment, mais aussi un Service de Soins Infirmiers À Domicile (SSIAD).

Le secteur de l’imagerie médicale a été concerné par la mise en place et le développement de scanners au plus près des patients, aux CH de Riom et d’Issoire, grâce à une équipe construite autour de professionnels qui réalisent des temps partagés entre ces établissements et le CHU. Ce travail a été parachevé par des équipements utilisant le PACS (système d’archivage et de transmission d’images, ou Picture Archiving and Communication System) qui permettent d’échanger les images réalisées entre ces différents établissements et ainsi garantissent la continuité de la prise en charge.

### 1.3.1.4 Conclusion sur les CHT

Les CHT ont soulevé beaucoup de questionnements depuis leur création. Elles sont la preuve d’une volonté de mutualisation des ressources pour aller vers une meilleure efficacité du système hospitalier mais elles n’ont pas les moyens nécessaires : en particulier elles n’ont pas la personnalité morale. Finalement, le terme de CHT n’est pas stable en vu de la dernière proposition du ministère de la santé qui suggère de renommer les Com-

munautés Hospitalières de Territoire en Groupements Hospitaliers de Territoire, même si le fond semble être le même. La section suivante présente un exemple de coopération multi-site : la Fédération d’Imagerie Médicale. Ce n’est pas une CHT à proprement parler mais ses objectifs sont les mêmes. D’autres formes de coopération sont ensuite présentées.

### 1.3.2 Fédération d’Imagerie Médicale

La Fédération d’Imagerie Médicale a été créée dans le cadre de la CHT du Grand Clermont. Une convention a été signée par les différents services d’imagerie médicale de la CHT. Cette coopération entre services pourrait se généraliser à d’autres établissements et/ou services.

La FIM consiste en la mise en commun de ressources hospitalières et médicales, humaines et matérielles sur un territoire donné pour le passage d’examens médicaux. Ces examens médicaux peuvent relever de la radiologie conventionnelle, de la radiologie interventionnelle, du scanner, de l’IRM, de la mammographie ou encore de l’échographie. Cette mise en commun a pour objectif l’augmentation de la permanence des soins en homogénéisant l’occupation des équipements et des ressources humaines sur un territoire.

La FIM a les caractéristiques suivantes.

#### 1.3.2.1 L’examen

Par examen médical, il est question uniquement du passage de l’examen en tant que tel, c’est-à-dire lors de l’utilisation de l’équipement d’imagerie médicale, que ce soit pour une prise de cliché, ou un examen interventionnel. N’est considérée que la manipulation radiologique. Il n’est pas question ici de la partie interprétation ultérieure des clichés.

Un examen est la réunion d’une date, d’un équipement et de ressources humaines.

Un examen est réalisé sur un patient. Un examen a une durée connue à l’avance, prenant en compte la durée de nettoyage post-examen si nécessaire. Le temps de nettoyage n’est pas à prendre en compte pour tous les examens. Par exemple, pour les examens utilisant un équipement de type scanner, le temps de nettoyage n’est à considérer qu’à la fin du dernier examen de chaque demi-journée. Si l’examen utilise un équipement de type radiologie interventionnelle, le temps de nettoyage est systématique. La durée de l’examen peut dépendre de l’état de l’équipement et de la compétence des ressources humaines.

La date de prise de rendez-vous est connue. Un examen a une date de disponibilité, date à partir de laquelle il peut être réalisé, et une date au plus tard, date avant laquelle il doit avoir lieu.

Le statut d’un examen est défini de la façon suivante. Un examen est soit :

- Programmé s’il est prévu à l’avance. Auquel cas il est soit :
  - Interne s’il concerne un patient déjà hospitalisé.
  - Externe dans le cas contraire.
- Non-programmé, dans le cas où il n’est pas prévu, mais ce qui ne signifie pas pour autant que ce soit une urgence. Un examen non programmé s’insère dans le planning prévu sans le modifier.
- Urgent. S’il s’agit d’une urgence, il s’insère obligatoirement dans le planning, quitte à perturber le planning initial ou même déprogrammer des examens programmés.



### 1.3.2.2 Le patient

Le patient peut passer un ou plusieurs examens. Dans le cas où le patient doit passer plusieurs examens, des contraintes de précedence peuvent exister entre ces examens. Le patient a un site de référence, c'est le site où il se fera examiner dans la mesure du possible. Le patient est soit un adulte, soit un enfant. Il est soit valide soit handicapé, c'est-à-dire à mobilité réduite. Il peut également être obèse ou claustrophobe.

### 1.3.2.3 Les sites et services

La fédération est composée de plusieurs sites géographiques. Les sites sont éloignés deux à deux d'une distance connue. Un site peut être composé de plusieurs services. Dans chaque service sont présents des équipements. Chaque site emploie des ressources humaines.

### 1.3.2.4 Les équipements

Chaque équipement est présent dans un seul service. Un équipement a un planning d'ouverture, par exemple le nombre d'heures disponibles pour cet équipement à une certaine date. Ce planning tient compte entre autres des opérations de maintenance préventive pendant lesquelles l'équipement n'est pas disponible.

Il existe plusieurs types d'équipement : par exemple les types radiologie conventionnelle, radiologie interventionnelle (cardio, neuro, vasculaire, etc.), scanner, IRM (Imagerie par Résonance Magnétique), mammographie ou échographie.

Il existe des équipements mobiles : l'ampli bloc qui se déplace entre les blocs opératoires et la radiologie mobile qui se déplace au chevet du patient au sein d'un même service. Un équipement mobile ne peut être utilisé que pour des examens internes.

Un équipement peut avoir des spécificités. Il peut notamment permettre de réaliser des examens sur des enfants et/ou des adultes. Il peut également traiter des cas particuliers comme les personnes obèses et les personnes claustrophobes.

### 1.3.2.5 Les ressources humaines

Dans la fédération, plusieurs ressources humaines sont disponibles. Elles peuvent ou non se déplacer entre les différents sites avec une certaine mobilité. Certaines ressources humaines peuvent se déplacer sur tous les sites ou services, alors que d'autres ne peuvent travailler que sur leur site employeur. Le site employeur est le site qui les rémunère. La fédération n'ayant pas la personnalité morale, elle ne peut pas avoir son propre personnel. Un temps de déplacement entre les différents sites doit être pris en compte si la ressource humaine doit se déplacer d'un site à un autre. Les ressources humaines ont des compétences sur certains types d'équipement, elles ne peuvent travailler que sur ces équipements. À la sortie de leur formation, les ressources humaines de type manipulateur sont polyvalentes sur tous les types d'équipement et tous les équipements. Elles se spécialisent au fur et à mesure de leur carrière et finissent souvent par ne travailler que sur un type ou un équipement particulier. Les ressources humaines peuvent avoir la capacité de réaliser des examens sur les enfants et/ou les adultes.

Les ressources humaines sont de plusieurs types : manipulateur, médecin (radiologue, anesthésiste, interventionnel, etc.), brancardier ou Infirmier Anesthésiste Diplômé d'État.

Chacune de ces ressources humaines a un planning de travail, prenant en compte ses temps de pause réglementaires et ses congés. Concernant les temps de pauses, soit ils sont donnés et fixés, soit seules les contraintes réglementaires sont données, auquel cas des contraintes du type "une ressource humaine ne peut pas travailler plus de six heures d'affilée", "une ressource humaine doit avoir une heure de pause-déjeuner" seront données.

En fonction de l'examen et de la spécificité du patient, la composition de l'équipe médicale est différente. Par exemple, les compositions d'équipe en fonction de l'examen peuvent être les suivantes :

- Radiologie conventionnelle : un manipulateur ayant la compétence requise sur ce type d'équipement si le patient est valide et un manipulateur supplémentaire si le patient est à mobilité réduite.
- Radiologie interventionnelle : deux médecins radiologues (un en salle et un aide), deux manipulateurs (un sur la partie commande et un en salle pour servir le matériel), un IADE et un médecin anesthésiste.
- Scanner : deux manipulateurs, un médecin radiologue et un brancardier supplémentaire dans le cas d'un examen interne.
- IRM : deux manipulateurs et un médecin radiologue.
- Mammographie : un médecin radiologue.
- Échographie : un médecin radiologue et un manipulateur commun à tous les examens d'échographie pour diriger les patients vers les équipements.
- Ampli bloc : un manipulateur.
- Radiologie mobile : un manipulateur.

Un des problèmes étudiés peut être un problème de planification d'examens et d'affectation de ressources : chaque examen doit être associé à des ressources humaines et à un équipement. Les contraintes de temps, de lieu et de compatibilité sont à respecter.

### 1.3.3 Autres exemples de coopérations

D'autres formes de coopération existent. Les définitions suivantes sont issues du site du ministère de la santé [www.sante.gouv.fr](http://www.sante.gouv.fr) (dernière consultation le 26 septembre 2014).

#### 1.3.3.1 Groupement de Coopération Sanitaire

Le Groupement de Coopération Sanitaire (GCS) est l'outil de coopération privilégié dans le cadre des coopérations entre le secteur public et privé, mais également entre la médecine de ville et l'hôpital. Il permet d'associer des établissements de santé publics comme privés, des centres de santé, des maisons de santé et des professionnels médicaux libéraux à titre individuel ou collectif, ou encore les acteurs du secteur médico-social. Il est doté, selon le cas, de la personnalité morale de droit public ou de droit privé.

Il s'agit d'un outil bien connu des professionnels puisque soixante-cinq GCS étaient recensés au 31 décembre 2006. Quatre cent quatre-vingt onze étaient recensés au 30 juin 2012.

La loi HPST a refondu le dispositif juridique des GCS de deux manières afin de différencier le GCS de moyens du GCS-Établissement de santé :

- D'une part, en simplifiant et en clarifiant la formule du GCS de moyens :

- Les objets possibles de coopérations sont précisés : mise en commun de moyens de toute nature, humains, mobiliers, immobiliers, financiers, matériels ou immatériels ;
- Sa gouvernance est assouplie par la possibilité de mettre en place un comité restreint auquel l'assemblée générale du GCS peut déléguer une partie de ses compétences ;
- Le GCS en charge de Missions d'Intérêt Général et d'Aide à la Contractualisation (MIGAC) peut percevoir directement des financements.
- D'autre part, en confirmant la possibilité pour certains GCS, extrêmement aboutis et intégrés, d'aller plus loin dans leur coopération autour du soin et d'être érigés en établissements de santé.

Cette éventuelle transformation des GCS en établissements de santé constitue une des forces innovantes du dispositif de la loi HPST. Elle ne concerne toutefois qu'une minorité de projets, extrêmement aboutis et intégrés en termes de prise en charge médicale et soignante.

### **1.3.3.2 Groupement d'Intérêt Économique**

Le Groupement d'Intérêt Économique (GIE) est doté d'une personnalité morale de droit privé. S'adressant à des personnes physiques (professionnels libéraux) ou à des personnes morales de droit public ou de droit privé, il a pour objectif de faciliter ou de développer l'activité économique de ses membres. Il s'agit d'un outil juridique et économique non dédié au secteur sanitaire, même si l'on trouve un certain nombre de GIE portant sur l'exploitation d'équipements matériels lourds comme des scanners.

### **1.3.3.3 Groupement d'Intérêt Public**

Le Groupement d'Intérêt Public (GIP) est doté d'une personnalité morale de droit public. Il s'adresse exclusivement à des personnes morales, de droit public ou de droit privé. Permettant notamment de gérer des équipements ou des activités d'intérêt commun, il peut détenir une autorisation d'équipements matériels lourds mais pas d'autorisation d'activités de soin. Il existe des GIP spécifiques au domaine de l'action sanitaire et sociale, par exemple pour la gestion en commun d'activités supports telles que la fonction linge ou la fonction restauration.

### **1.3.3.4 Groupement de Coopération Sociale ou Médico-Sociale**

Les dispositions du Groupement de Coopération Sociale ou Médico-Sociale (GCS-MS) sont calquées sur celles du Groupement de Coopération Sanitaire. Il s'agit d'un outil de coopération doté, selon le cas, de la personnalité morale de droit public ou de droit privé. Ses membres peuvent être des personnes morales ou physiques, de droit public ou de droit privé. Son champ est large puisqu'il a notamment pour objet de permettre des interventions communes des professionnels des secteurs sociaux, médico-sociaux et sanitaires, d'exercer une mission ou une prestation exercée par l'un de ses membres et de détenir, à ce titre, les autorisations afférentes, de préparer les opérations de fusion ou de regroupements, etc.

### 1.3.3.5 Syndicat InterHospitalier

L'article 23 de la loi HPST indique que les Syndicats InterHospitaliers (SIH) doivent se transformer, sans dissolution ni création d'une personne morale nouvelle, dans des conditions fixées par décret en Conseil d'État, soit en GCS, soit en GIP.

Étant donné l'étendue des coopérations existantes, dans la suite du mémoire nous développons le cas de la CHT. Mais nos travaux peuvent aisément s'étendre à tout type de coopération.

## 1.4 Recherche opérationnelle pour les réseaux de soin

Depuis les dix dernières années, de nombreuses méthodes de recherche opérationnelle ont été utilisées pour résoudre les problèmes de réseaux de soin. En 2011, (Rais and Viana, 2011) citent près de deux cent cinquante articles sur la recherche opérationnelle pour les systèmes de soin en général. L'année précédente, (Cardoen et al., 2010) en citaient plus de cent vingt sur la planification et l'ordonnancement des blocs opératoires. (Van den Bergh et al., 2013) réalisent une revue de la littérature à propos de l'ordonnancement des ressources humaines et citent près de trois cents articles.

Un échantillon des problèmes étudiés est présenté dans cette partie. Trois problèmes sont principalement étudiés : les problèmes de dimensionnement, de planification et d'ordonnancement.

### 1.4.1 Dimensionnement

Le **dimensionnement** d'un système est le fait de dimensionner le système et les ressources nécessaires pour son bon fonctionnement. Par exemple, dans le cas de la Fédération d'Imagerie Médicale, l'objectif du dimensionnement de système est de déterminer le nombre de sites appartenant à la FIM, le nombre d'équipements sur chaque site et le nombre de ressources humaines. Cela permet d'anticiper le système avant même sa conception. L'étude du dimensionnement se fait sur un horizon temporel de quelques années, afin de voir l'évolution du système à long terme. La détermination des types d'équipement et de ressources humaines est également un problème de dimensionnement. Dans ce cas, l'horizon temporel peut être plus court.

(Cassagne et al., 2008) s'intéressent au bloc opératoire car il représente l'un des principaux postes de dépenses. L'objectif est de proposer un outil d'aide à la décision pour le dimensionnement et la planification des blocs opératoires d'un système à concevoir comme pour un système existant. Les auteurs proposent un outil permettant de dimensionner le nombre de salles à construire pour absorber l'ensemble de l'activité prévue sur la nouvelle structure. Ainsi, en faisant varier la charge du système, la spécificité des salles et le planning d'ouverture des salles, l'outil permet de déterminer le nombre de salles à construire, les taux d'occupation de chacune des salles ainsi que le nombre d'équipes médicales nécessaires à tout instant. L'outil étant destiné à être utilisé par les chirurgiens, les résultats permettant de déterminer le nombre de salles doivent être obtenus rapide-

ment et simplement. Cet outil doit facilement être transposable à d'autres niveaux de décision ou de modélisation.

Le cas d'étude porte sur le déménagement de l'Hôtel-Dieu vers le CHU Estaing à Clermont-Ferrand. Le travail de recueil de données porte sur l'activité du premier semestre 2007 de l'Hôtel-Dieu afin d'évaluer la charge future du CHU Estaing et de dimensionner son futur bloc opératoire. Cette activité recense 3677 interventions effectuées.

Cet outil a été développé de façon générique et a servi de base au dimensionnement du CHU Estaing à Clermont-Ferrand. Ces travaux ont fait l'objet d'une thèse (Rodier, 2010). Un chapitre y est dédié au dimensionnement, à la planification et à l'ordonnancement de l'activité opératoire. Les solutions proposées permettent de faire varier la charge du système (nombre d'interventions et nombre de spécialités); la spécificité des salles (spécialisées, partagées, polyvalentes) ainsi que le planning d'ouverture de chaque salle. L'outil développé a été validé par les hospitaliers et a servi de base au dimensionnement du futur bloc opératoire en apportant les informations nécessaires concernant le nombre de salles à construire et à ouvrir pour absorber différents niveaux d'activité.

Au sein d'un bloc opératoire, plusieurs stratégies d'allocation de spécialité sont possibles :

- Chaque salle opératoire est dédiée à une spécialité,
- Chaque salle opératoire est dite polyvalente et peut accueillir toutes les spécialités,
- Le bloc opératoire est composé d'un mélange de ces deux types de salles : dédiées et polyvalentes.

Les travaux de (Choi and Wilhelm, 2014) s'intéressent à l'allocation de spécialités chirurgicales à des journées de salles opératoires. Ils étudient les décisions de planification de capacité qui allouent des spécialités chirurgicales aux salles opératoires et à des journées avec l'objectif de minimiser les coûts relatifs aux patients non traités et à la sous ou sur-utilisation de la capacité des salles opératoires. La sur-utilisation se traduit par des heures supplémentaires. À partir de la planification réalisée, les chirurgiens peuvent ordonnancer leurs activités, tout en ayant la possibilité d'équilibrer leur temps alloué aux actes chirurgicaux avec leur temps alloué à d'autres responsabilités.

### 1.4.2 Planification

La **planification** est le fait de planifier les actes. La notion de temps est à prendre en compte. Quels sont les actes qui seront planifiés à telle date, connaissant les ressources nécessaires? Par exemple, une planification peut-être l'affectation à chaque acte chirurgical des ressources adéquates pendant une demi-journée ou une journée. La planification est généralement étudiée sur quelques semaines, voire quelques mois. Ceci permet notamment de générer le calendrier global de disponibilité de chaque ressource à moyen terme ce qui détermine la charge de travail quotidienne ou hebdomadaire.

La réalisation de la planification des salles opératoires est le fruit de négociation entre les différents acteurs au bloc : les chirurgiens, les anesthésistes, les infirmières, l'équipe managériale, etc., dont les contraintes et intérêts sont souvent différents. Dans ce contexte, une situation gagnant gagnant nécessite une négociation constructive. (Marcon et al., 2003) créent un outil de planification des salles opératoires. Leur objectif est la

minimisation du risque de non réalisation de la planification réalisée tout en stabilisant le temps d'utilisation des salles opératoires. Le risque de non réalisation traduit le fait que la planification proposée peut ne pas être intégralement réalisée à cause de certains aléas.

Les hôpitaux génèrent de façon répétitive la planification des salles, ce qui est consommateur de temps. La durée des actes chirurgicaux est souvent stochastique, ce qui complique la construction du programme opératoire. (Van Oostrum et al., 2008) proposent une planification cyclique d'opérations, en anglais Master Surgical Schedule (MSS), qui permet de déterminer le programme chirurgical principal afin de palier à ce problème. Dans un MSS, les types d'actes chirurgicaux fréquemment réalisés sont planifiés de façon cyclique. Un MSS spécifie pour chaque salle opératoire par jour une liste d'actes chirurgicaux qui doivent être réalisés. Pour contrer l'aspect stochastique, les auteurs proposent une planification avec une marge de sécurité. L'outil proposé permet également un nivellement des exigences pour les lits en amont, tel que les unités de soins intensifs. Si l'occupation de ces lits est trop élevée, cela entraîne à terme des retards.

Deux approches de planification peuvent être utilisées :

- "Open scheduling". Un planning vide est considéré et rempli par les actes chirurgicaux selon leur ordre d'arrivée. La construction du planning est faite en deux étapes. La première concerne le placement des actes chirurgicaux à moyen terme. La seconde permet un ordonnancement détaillé des actes à court terme.
- "Block scheduling". La planification est faite à partir d'une pré-allocation de salles affectées à des chirurgiens, des spécialités ou des groupes de chirurgien pendant des plages horaires prédéfinies. Cette pré-allocation est réalisée grâce au Master Surgical Schedule. Les actes chirurgicaux sont ensuite affectés à ces salles pendant ces plages horaires.

(Chaabane et al., 2008) comparent les deux approches : la première est dite orientée salle opératoire alors que la seconde est orientée chirurgien. Les deux approches donnent finalement des résultats similaires. Les approches sont expérimentées sur un hôpital belge. Les auteurs fournissent un outil d'aide à la décision aux hospitaliers qui peuvent utiliser l'une ou l'autre des deux approches. (Kharraja et al., 2006) comparent aussi ces deux approches pour la planification des opérations dans un bloc opératoire. Dans le même temps, l'outil permet une affectation des chirurgiens aux salles opératoires et un ordonnancement des interventions. L'approche block scheduling permet une meilleure utilisation des ressources que l'approche open scheduling.

Lors de la planification, des heures supplémentaires peuvent être allouées aux différentes ressources, dont les salles opératoires. (Hans et al., 2008) proposent une planification robuste ayant pour objectifs la maximisation des taux d'utilisation des salles, la minimisation des heures supplémentaires, ainsi que la non annulation d'actes chirurgicaux. Des marges de sécurité déterminées par des données historiques sont affectées à chaque salle opératoire pendant chaque journée en supplément de son temps d'ouverture. Les auteurs déplorent un manque de méthodes robustes dans la littérature et en proposent ainsi une. La robustesse de la méthode assure une limitation des heures supplémentaires.

Dans beaucoup d'hôpitaux, le bloc opératoire sert également à traiter les cas arrivant du service des urgences. Pour prendre en compte cet aspect lors de la planification, certains papiers intègrent un facteur stochastique à leur étude. Par exemple, (Lamiri et al., 2009) prennent en compte les opérations urgentes dans leur problème de planification chirurgicale : les salles opératoires sont partagées entre les actes programmés et le service des urgences. Le modèle intègre donc une partie stochastique. L'objectif est la minimisation du dépassement des coûts.

Dans un bloc opératoire, la ressource goulot n'est pas nécessairement au sein du bloc à proprement parler ni des ressources utilisées pendant l'acte. (VanBerkel and Blake, 2007) montrent que la ressource goulot peut se situer au niveau des lits post-interventionnels. (Iser et al., 2008) proposent une planification des actes qui tient compte de ces lits. Les ressources post-interventionnelles impliquent des incertitudes pour la création de la planification. En effet, la salle de réveil est une ressource partagée et est donc alimentée par plusieurs salles opératoires simultanément. La salle de réveil est la ressource critique : s'il n'y a pas de lit en salle de réveil disponible alors que l'acte chirurgical est terminé, le patient attend en salle opératoire, ce qui retarde les opérations suivantes. Les auteurs considèrent ainsi deux niveaux pour leur étude : le block opératoire et la salle de réveil.

(Kharraja et al., 2006) ajoutent un troisième niveau. Ils considèrent trois étages : les blocs opératoires, les Salles de Soins Post-Interventionnel ainsi que l'unité de Soins Intensifs. En fonction de l'état du patient, la sortie de la salle opératoire est soit une SSPI ou l'unité de SI. Si la sortie est une SSPI, le patient attend dans la salle opératoire jusqu'à libération d'une place en SSPI. Si la sortie est l'unité de SI, le patient va dans cette unité sans délai.

### 1.4.3 Ordonnancement

L'**ordonnancement** intervient au niveau le plus fin. Il consiste à ordonnancer, c'est-à-dire placer les actes dans l'ordre d'exécution. La liste des actes planifiés sur une journée avec leurs ressources nécessaires est ordonnancée pendant cette journée. Chaque acte est placé sur une échelle de temps plus précise tout au long de la journée, par rapport aux autres actes, avec affectation des ressources adéquates. L'ordonnancement permet de fixer précisément les rendez-vous d'actes chirurgicaux.

Une approche en deux étapes est considérée par (Jebali et al., 2006). Dans une première étape, les actes chirurgicaux sont affectés à des salles opératoires pendant une journée. Dans une seconde étape, ils sont ordonnancés au sein de chaque salle les uns par rapport aux autres tout en tenant compte des contraintes de relation entre les différentes ressources utilisées et les spécifications des processus. Deux stratégies sont testées : dans la première stratégie, l'ordonnancement ne remet pas en question l'affectation aux salles ; dans la seconde stratégie, certaines affectations de salle peuvent être corrigées pour améliorer l'ordonnancement. Les objectifs sont l'amélioration de l'utilisation des salles opératoires. Les contraintes liées à la salle post-opératoire sont prises en compte dans l'ordonnancement. La stratégie qui ne remet pas en question l'affectation aux salles donne de bons résultats et est moins longue à mettre en place, ce qui montre l'intérêt de faire une planification d'actes avant de les ordonnancer.

Un temps de traitement plus long que la moyenne entraîne un retard de l'acte suivant ainsi que pour tous les autres actes de la journée. Ceci a un coût, notamment via les heures supplémentaires des ressources qu'il faut rémunérer. (Denton et al., 2007) mettent en avant l'intérêt d'une règle de séquençage basée sur la variance des durées chirurgicales. Un simple séquençage d'activité par durée de traitement peut affecter le temps d'attente des équipes chirurgicales, l'occupation des salles opératoires et les heures supplémentaires des salles opératoires. Deux types de patient sont considérés : les patients internes qui sont hospitalisés et les patients externes. De part l'aspect stochastique de leur modèle, les auteurs prennent en compte la durée aléatoire des actes chirurgicaux pour anticiper les potentiels retards. Les modèles stochastiques qui considèrent l'incertitude des durées chirurgicales offrent la possibilité d'améliorations significatives dans l'ordonnancement journalier des salles opératoires.

Les cas d'incertitudes peuvent également toucher les ressources. Les ressources se raréfient mais elles peuvent aussi être ponctuellement en pénurie, dans le cas d'absence de ressources humaines ou de panne des ressources matérielles. La planification des flux de patients dans des hôpitaux subissant des pénuries de ressources est traitée par (Gartner and Kolisch, 2014). Les patients sont regroupés par pathologie. Le parcours patient est déterminé : quelles sont les activités qui doivent être subies par le patient. Chaque activité est ensuite ordonnancée. Leur test sur des données réelles montre leur apport par rapport à l'ordonnancement courant. Les temps d'attente entre admission et opération sont réduits.

#### 1.4.4 Réseaux hospitaliers multi-sites

Les articles traitant de problèmes multi-sites dans le secteur hospitalier sont rares, ou alors leur cadre d'étude est assez limité.

La planification de vacations chirurgicales par spécialité est traitée par (Santibáñez et al., 2007). La disponibilité des salles opératoires, la capacité en lits, les préférences des chirurgiens et les listes d'attentes sont prises en compte. Les auteurs proposent un modèle permettant d'allouer des spécialités aux salles opératoires. Il prend également en compte les contraintes en ressources post-chirurgicales. Ce cas est appliqué à une autorité de santé en Colombie-Britannique. Huit hôpitaux sont étudiés, sur quatre semaines d'activité, avec des données réelles. L'allocation de spécialités est commune aux huit hôpitaux, ces hôpitaux étant considérés comme un ensemble. Ainsi, en considérant les ressources comme un ensemble, le nombre de lits utilisés est réduit, mais l'effet n'est pas impressionnant d'après les auteurs. L'objectif est la minimisation du nombre maximum de lits de réveil et de soins intensifs utilisés par jour. Ces travaux montrent qu'en affectant les spécialités différemment, sans augmenter les capacités des ressources critiques, les listes d'attentes sont réduites et l'utilisation des ressources optimisée.

(Everett, 2002) développe un outil d'aide à la décision pour la gestion des files d'attente des patients nécessitant un acte chirurgical. Le système est composé de plusieurs hôpitaux travaillant en coopération. Une file d'attente est commune à plusieurs hôpitaux.



Les patients sont caractérisés par leur traitement et un degré d'urgence. Chaque patient est affecté à l'un des sites : chaque jour, à chaque hôpital sont affectés les patients en fonction de la disponibilité de l'hôpital. Les patients sont affectés le jour suivant si aucun hôpital ne peut les accueillir. L'affectation des ressources sur les différents sites n'est pas traitée. L'auteur ne détaille pas ses règles d'affectation des patients sur les différents sites. Chaque hôpital est considéré comme une entité individuelle mais toutes ces entités ont les mêmes caractéristiques.

La mutualisation des ressources humaines sur un système multi-site est étudiée par (Trilling, 2006) dans son manuscrit intitulé "Aide à la décision pour le dimensionnement et le pilotage de ressources humaines mutualisées en milieu hospitalier". Elle traite de la mise en commun, du partage et de l'échange de ressources pour la réalisation d'actes communs. Les ressources concernées sont notamment les brancardiers et les aides soignantes qui sont des ressources "universelles" utilisées par tous les services et hôpitaux.

(VanBerkel and Blake, 2007) développent un outil pour la réduction des temps d'attente et la planification des capacités en lits pour la chirurgie. Leur cadre d'étude est réparti sur plusieurs sites. L'outil décide dans quel hôpital la capacité doit être préservée. C'est un problème d'allocation de ressource fixe. Le cas d'étude concerne la division de chirurgie générale à la Capital District Health Authority à Halifax, Nouvelle-Écosse, Canada. Cette division est composée de plusieurs sites. L'objectif est de fournir un outil d'aide à la décision pour la planification de capacités et une analyse de performances. Sont étudiées les conséquences d'une redistribution de lits post-opératoires entre les différents sites. En effet, les auteurs rappellent que le système hospitalier doit faire face à une demande croissante avec des moyens limités. L'horizon temporel est de plusieurs années.

Les auteurs s'intéressent aux listes d'attente. Le temps d'attente des patients est le critère d'étude. Les données correspondent à un an d'activité. Le temps d'attente a augmenté sur cette dernière année. Grâce à la simulation, les auteurs prévoient les temps d'attente sur les quatre prochaines années. Ils comparent différentes stratégies qu'ils évaluent en fonction de ce critère. Les résultats leur permettent de dresser une liste de recommandations quant à la répartition des lits post-opératoires sur les différents sites afin de réduire au mieux ces temps d'attente sur les prochaines années.

Des problèmes de capacité des unités de soin intensif peuvent résulter en une annulation d'actes programmés, une surcharge des équipes médicales ou un rejet des patients urgents. Ainsi, les patients urgents peuvent finalement être transportés dans des sites éloignés. Une solution coopérative est étudiée par (Litvak et al., 2008) qui traitent du cas de l'unité de soin intensif aux Pays-Bas. Plusieurs hôpitaux appartenant à la même région mettent en place un nombre de lits partagés pour les patients urgents de la région. Cette solution permet une amélioration du service aux patients. Cette étude est faite dans le cas de débordement, de surplus de la demande.

### 1.4.5 Conclusion

Les problèmes concernant le système hospitalier étudiés dans la littérature concernent trois problèmes : les problèmes de dimensionnement, de planification et d'ordonnancement. La majorité des travaux référencés se focalise sur un problème en particulier. Peu de travaux s'intéressent au problème de dimensionnement. Dans la suite de ce chapitre, nous proposons une approche permettant de résoudre tous ces problèmes.

Beaucoup de travaux traitant les systèmes de soin sont dédiés. Ils s'intéressent à un cas particulier, un service ou un hôpital donné. Leurs modèles ne sont pas facilement réutilisables. Les modèles que nous proposons dans le chapitre suivant sont le plus générique possible, afin d'être utilisables le plus souvent possible.

Les aspects mutualisation de ressources n'ont pas été très étudiés. On les retrouve dans le cas de ressources communes à plusieurs services, comme les brancardiers ou les aides soignantes, ou dans le cas de gestion de situation de crise. Nous proposons d'approfondir cette notion de mutualisation dans une approche quotidienne : une mutualisation de ressources au quotidien, ces ressources pouvant être des ressources spécialisées.

## 1.5 Objectifs de la thèse

L'objectif est la mise en place d'un outil d'aide à la décision permettant d'étudier et de résoudre un large spectre de problèmes.

### 1.5.1 Niveaux de décision

Pour situer les différents problèmes qui peuvent être rencontrés dans les systèmes industriels, trois niveaux de décision ont été définis (Ballou, 1992), (Giard, 2003), (Rodier, 2010). Les trois niveaux dépendent de l'horizon de planification utilisé pour appliquer les décisions. Les objectifs de chacun de ces niveaux ne sont pas les mêmes.

Pour le système hospitalier, les différents niveaux de décision sont définis de la façon suivante (Gourgand, 2008) :

- Le **niveau stratégique** correspond aux problèmes de conception du système,
- Le **niveau tactique** correspond à l'utilisation du système et à l'adéquation ressources/besoins,
- Le **niveau opérationnel** concerne le pilotage du système à court terme.

Les trois objectifs principaux peuvent donc être les suivants :

- Concevoir ou adapter le système en le dimensionnant,
- Configurer le système en planifiant les activités et les ressources,
- Piloter le système en évaluant ses performances afin de pouvoir ajuster et optimiser son fonctionnement.

### 1.5.2 Niveaux de modélisation

Trois niveaux peuvent être utilisés pour la modélisation de système (Rodier, 2010) :

**Le niveau macroscopique** désigne le niveau supérieur d'observation. Le système est appréhendé dans sa globalité, l'ensemble des patients forme un tout, de même que l'ensemble des ressources humaines et des équipements. L'activité est également fortement agrégée. Une modélisation macroscopique est généralement utilisée pour résoudre des problèmes de planification stratégiques voire tactiques.

**Le niveau mesoscopique** permet de regrouper une à plusieurs entités individuelles ayant certaines caractéristiques communes et représentant un sous-groupe de la vue macroscopique. Ce niveau de modélisation est le plus fréquent pour l'étude du système hospitalier. Il est fréquemment utilisé pour résoudre des problèmes de planification au niveau tactique et opérationnel mais également de manière plus générale pour l'évaluation de la performance des systèmes.

**Au niveau microscopique**, chaque élément est considéré dans son individualité comme entité atomique, c'est-à-dire indécomposable. Une modélisation microscopique peut également être utilisée pour résoudre des problèmes de planification le plus souvent au niveau tactique et opérationnel. Ce niveau de modélisation sera nécessaire si l'on souhaite agir sur le comportement de chaque entité de manière individualisée (par exemple : l'affectation d'une ressource à un planning ou à une zone géographique particulière) et si l'on veut obtenir des détails au niveau le plus fin (par exemple : le temps d'occupation de chaque ressource, détail des ressources affectées à chaque activité, etc.).

Ainsi nous pouvons définir les trois termes suivants :

Une **activité** est une activité élémentaire, non décomposable. Par exemple, dans le cas de l'imagerie médicale, le passage de l'examen est une activité, c'est-à-dire lorsque le patient est déjà installé sur l'équipement et que les clichés sont train d'être réalisés. Une autre activité peut être l'installation du patient sur l'équipement.

Un **processus** est un groupe d'activités. Pour passer une radiologie, le patient passe à travers plusieurs activités : prise en charge par le personnel d'accueil, installation sur l'équipement de radiologie, passage de l'examen, etc.

Un **système** est un groupe de processus. Le système de la CHT appliqué à l'imagerie médicale est composé de tous les processus définissant chacun des examens médicaux.

### 1.5.3 Définition des problèmes

En reprenant les travaux de (Rodier, 2010) qui elle-même s'inspire du couplage des niveaux de décision et des niveaux de modélisation de (Chabrol et al., 2005), nous pouvons dresser le Tableau 1.1 qui définit les neuf problèmes qui peuvent être considérés en fonction du niveau de décision et du niveau de modélisation dans le cas de la CHT.

		MACROSCOPIQUE Vue Globale	MESOSCOPIQUE Vue Processus	MICROSCOPIQUE Vue Activité
STRATÉGIQUE (> 1 an)	Concevoir	<b>Dimensionner le système</b> <i>Définir les quantité et répartition :</i> Quantité de sites appartenant à la CHT, répartition des sites sur la CHT, répartition des ressources par site, etc.	<b>Dimensionner les processus</b> <i>Définir les processus de prise en charge du patient :</i>  Activités internes, externes, dans le service des urgences, entre les différents sites, etc.	<b>Dimensionner les activités</b> <i>Définir la composition de chaque activité :</i>  Quantité et types d'équipement, de ressource humaine différents paramètres, etc.
TACTIQUE (semaine/mois)	Configurer	<b>Planifier les flux du système</b> <i>Déterminer les différents flux traversant le système :</i>  Origine(s), destination(s), caractéristiques, etc.	<b>Planifier les processus</b> <i>Déterminer les ressources nécessaires aux processus :</i> Processus d'imagerie médicale, prise en charge des urgences, transport des patients, etc.	<b>Planifier les activités</b> <i>Déterminer les ressources nécessaires aux activités :</i> Nombre et type de ressources nécessaires en fonction de l'activité et du patient, etc.
OPÉRATIONNEL (jour/semaine)	Piloter	<b>Évaluer et ajuster le système</b> <i>Adapter les flux aux aléas :</i> Fermeture temporaire d'un site appartenant à la CHT, prise en charge exceptionnelle des flux patients d'un autre site, etc.	<b>Évaluer et ajuster les processus</b> <i>Adapter les processus aux aléas :</i> Modification de la planification des activités, modification du traitement des patients aux urgences, etc.	<b>Évaluer et ajuster les activités</b> <i>Adapter les activités aux aléas :</i> Ajustement des ressources prenant en charge l'activité, modification des plannings journaliers des ressources, etc.

TABLEAU 1.1 – Matrice 3\*3 des problèmes de la CHT

### 1.5.4 Outil d'aide à la décision

L'outil d'aide à la décision doit permettre de résoudre chacun de ces neuf problèmes. Les entrées, sorties et critères ne sont pas les mêmes en fonction du problème considéré.

**Dimensionner le système :** à partir d'une planification type annuelle d'activités, l'outil doit permettre la détermination de la configuration de la CHT, c'est-à-dire est-il utile de créer une CHT avec les sites concernés? Quel est le nombre de sites qui doivent appartenir à la CHT? Le cas d'un site particulier qui veut rejoindre une CHT existante peut être étudié : quel est l'intérêt pour le site et la CHT de cette intégration? L'achat de nouveaux équipements est-il justifié? etc. Le dimensionnement des ressources peut également être fait. Par exemple, au sein d'un réseau hospitalier, il s'agit de déterminer le nombre de ressources nécessaires sur chaque site et le type des équipements ou les compétences des ressources humaines.

Pour cela, les critères déterminants sont l'occupation des sites et des ressources, le nombre de déplacements réalisés dans le cas de ressources mobiles, etc. Si beaucoup de déplacements sont réalisés, il peut être intéressant de déterminer une nouvelle localisation des ressources mobiles.

**Dimensionner les processus :** à partir d'une planification type annuelle d'activités, l'outil doit déterminer la configuration des processus. Par exemple, dans un processus donné, quel est le rôle des brancardiers, à quel moment interviennent-ils? Comment sont gérés les patients internes ou externes dans chacun des processus? Comment sont gérés les patients urgents par rapport aux patients programmés? Le dimensionnement de processus permet d'organiser les activités les unes par rapport aux autres au sein d'un même processus.

Les critères déterminants sont l'occupation des ressources ainsi que leurs déplacements. La bonne gestion des activités au sein des processus est traduite par le bon enchaînement des activités entre elles, le fait qu'il n'y ait pas de temps morts entre chaque activité, ainsi que la date de fin de chaque activité au sein d'un processus.

**Dimensionner les activités :** à partir d'une planification type annuelle d'activités, les activités sont déterminées vis-à-vis du nombre et du type de ressources nécessaires à leur réalisation. En fonction de ces ressources, une activité est réalisée plus ou moins rapidement. Le compromis doit être trouvé entre un grand nombre de ressources nécessaires à sa réalisation et son temps de traitement, à supposer que plus de ressources sont présentes, moins le temps de traitement est long. Faut-il privilégier un temps de traitement court quitte à nécessiter beaucoup de ressources? La polyvalence des ressources peut être testée. Une ressource polyvalente est compétente sur de nombreux processus, elle est plus réactive en cas d'aléa de système mais est potentiellement moins compétente, le temps de traitement de l'activité est plus long. À l'inverse, une ressource spécialisée est plus compétente mais peut traiter un spectre d'activités plus réduit.

Les critères déterminants sont l'occupation des ressources et les temps de traitement théoriques pour chaque activité.

**Planifier les flux du système :** à partir du dimensionnement de système, quels sont les flux qui sont traités sur chacun des sites? Chaque flux est affecté sur chaque site. Chaque ressource humaine est-elle au préalable affectée sur un site et la planification des activités qu'elle traite en est déduite, ou est-ce que l'outil peut déterminer l'affectation sur site de chaque ressource en fonction de la charge de chacun des sites?

Pour cela, le critère le plus pertinent est le taux d'occupation de chacune des ressources sur chaque site. Il faut répartir équitablement les flux sur chacun des sites.

**Planifier les processus :** à partir du dimensionnement de processus, il faut planifier les processus en leur affectant effectivement les ressources humaines et les équipements en fonction de leur disponibilité. Si les heures supplémentaires des ressources sont une solution envisagée, elles sont testées à ce niveau. En fonction de leur coût, est-il plus intéressant d'autoriser les heures supplémentaires pour réaliser plus d'activités au sein d'un processus dans un intervalle de temps donné? Les équipes médicales sont également composées à ce niveau.

Le taux d'occupation de chacune des ressources est là aussi un critère important.

**Planifier les activités :** à partir du dimensionnement des activités, c'est-à-dire le nombre et type de ressources nécessaires pour chaque activité, l'outil permet la planification des activités entrantes dans le système. La planification consiste à affecter à chaque activité la ou les ressources adéquates identifiées précisément ainsi qu'une période. Il ne s'agit plus de dire un chirurgien cardiaque est nécessaire mais monsieur X qui est chirurgien cardiaque est affecté. La planification d'activités détermine la planification des ressources.

Les critères pertinents pour cette planification sont la date de fin de la dernière activité, le taux d'occupation de chacune des ressources, etc.

**Évaluer et ajuster le système :** il s'agit d'évaluer et d'ajuster le système face aux aléas. Par exemple, face à une situation de crise (épidémie, catastrophe naturelle), à partir des sites disponibles et en tenant compte des sites manquants, comment répartir les activités et processus sur les sites restants pour répondre à la demande initiale. À partir de de la planification préalablement établie des flux du système, l'outil doit réaffecter les flux sur les sites encore opérationnels.

**Évaluer et ajuster les processus :** dans le cas de ressources manquantes, ressources humaines absentes ou équipements défaillants, l'outil doit pouvoir ajuster les processus. Une ressource qui était prévue pour un processus peut elle être remplacée par une autre ressource? Une activité peut elle être substituée à une autre? La planification des activités au sein d'un processus avec l'affectation des ressources adéquates est connue. En fonction des aléas, l'outil doit adapter cette planification. Par exemple, si les brancardiers sont en grève, il faut réaffecter les ressources au mieux, certaines ressources pouvant remplacer les brancardiers quitte à pénaliser les activités qu'ils étaient sensés effectuer.

**Évaluer et ajuster les activités :** il s'agit d'évaluer le système dans son état actuel, afin notamment de pouvoir adapter le système aux aléas. Ceci permet l'ordonnancement des activités, c'est-à-dire ordonnancer les activités les unes par rapport aux autres en

tenant compte des aléas au niveau temporel le plus fin. À partir de la planification des activités avec affectation de ressources, l'outil fournit l'ordonnancement des activités en tenant compte des différents aléas.

L'outil d'aide à la décision proposé doit pouvoir traiter chacun de ces problèmes. Le décideur, qui sera aux commandes de cet outil, pourra en fonction de ses attentes, configurer l'outil afin de tester différentes stratégies. Par exemple, pour le problème de dimensionnement de système, pour une demande donnée, il pourra tester l'impact du nombre de sites et de ressources disponibles sur la planification. Grâce à cet outil, il sera à même de prendre les bonnes décisions et de les justifier.

### 1.6 Conclusion

Le contexte hospitalier a été présenté dans ce chapitre. À cause de restrictions budgétaires et d'une augmentation démographique, le système hospitalier doit traiter plus de patients avec moins de moyens. De plus, les déserts médicaux sont de plus en plus nombreux. Pour répondre à ces problématiques, les instances gouvernementales développent de nouvelles stratégies comme par exemple la coopération multi-site. Plusieurs sites, sur un territoire donné, vont coopérer et travailler ensemble dans le but d'atteindre des objectifs communs à chacun de ces sites. Parmi les différentes formes de coopération existantes, la Communauté Hospitalière de Territoire est une solution. Elle a été introduite en 2008. Elle a pour but d'améliorer l'efficacité globale des Centres Hospitaliers qui en font partie. Un des moyens utilisés peut être la mutualisation de certaines ressources.

Pour favoriser l'ancrage territorial de la coopération au sein de la CHT, il faut un instrument de planification au service de la recomposition de l'offre hospitalière, de l'efficacité, et de l'amélioration du parcours du patient. Il n'existe pas à ce jour d'outil de coopération permettant cette solidarité. Nos travaux permettent de justifier ces coopérations entre Centres Hospitaliers et visent à créer cet instrument de planification. De plus, cet outil pourra servir comme d'un outil d'aide à la décision pour la création de nouvelle coopération.

La recherche opérationnelle a été mise au service des systèmes hospitaliers depuis de nombreuses années. De nombreux chercheurs ont tenté d'optimiser le fonctionnement de ces systèmes en répondant à des problématiques particulières, en proposant des outils de dimensionnement, planification ou ordonnancement.

Nous proposons un outil d'aide à la décision permettant de répondre à la fois à des problèmes de dimensionnement, de planification et d'ordonnancement. Dans le prochain chapitre, la spécification et la formalisation de ces problèmes sont réalisées. Par abus de langage, dans la suite de ce manuscrit, le terme CHT est utilisé pour décrire n'importe quelle forme de coopération multi-site.

# Spécification et formalisation de la Communauté Hospitalière de Territoire

## Sommaire

---

<b>2.1</b>	<b>Introduction</b>	<b>36</b>
<b>2.2</b>	<b>Spécification</b>	<b>36</b>
2.2.1	Processus de modélisation	36
2.2.2	Sous-système physique	37
2.2.3	Sous-système logique	38
2.2.4	Sous-système décisionnel	38
2.2.4.1	Critères	39
2.2.4.2	Contraintes	39
<b>2.3</b>	<b>Problèmes considérés</b>	<b>41</b>
2.3.1	Hypothèses	41
2.3.2	Définition des problèmes considérés	41
<b>2.4</b>	<b>Formalisation</b>	<b>44</b>
2.4.1	Modèle 1 : CHT	44
2.4.2	Modèle 2 : CHT_RH non mobiles	46
2.4.3	Modèle 3 : CHT_RH mobiles	49
2.4.4	Récapitulatif des modèles	51
<b>2.5</b>	<b>Le problème du bin packing</b>	<b>56</b>
2.5.1	Présentation du problème	56
2.5.2	Analogies avec notre problème	57
2.5.2.1	Problème 1 : sans ressources humaines	57
2.5.2.2	Problèmes 2 et 3 : avec ressources humaines	58
<b>2.6</b>	<b>Conclusion</b>	<b>60</b>

---



## 2.1 Introduction

Le problème de la Communauté Hospitalière de Territoire a été défini dans le Chapitre 1. Ce problème peut être étudié à différents niveaux de décision : stratégique, tactique et opérationnel, ainsi qu'à différents niveaux de modélisation : macroscopique, mésoscopique et microscopique. La Section 2.2 fait une spécification complète du domaine selon la méthodologie ASCI (Analyse, Spécification, Conception et Implémentation) en le décomposant en trois sous-systèmes : le sous-système physique, le sous-système logique et le sous-système décisionnel.

L'accent est mis sur le niveau tactique microscopique : nous définissons le problème de planification d'activités avec affectation de ressources. À partir de certaines hypothèses, plusieurs sous-problèmes sont définis dans la Section 2.3. La formalisation de tous les modèles correspondants est réalisée dans la Section 2.4 : les données, les variables, les contraintes et enfin les critères considérés par chaque modèle sont décrits. La Section 2.4.1 présente le modèle le plus basique dans lequel les ressources humaines sont considérées à capacité infinie. La Section 2.4.2 présente le modèle avec affectation des ressources humaines. Le modèle présenté dans la Section 2.4.3 considère que les ressources humaines sont mobiles. Un récapitulatif de tous les modèles est fait dans la Section 2.4.4. La spécification et formalisation de ces modèles sont en partie publiées dans (Gourgand et al., 2015).

Le problème de planification d'activités avec affectation de ressources peut être comparé à un problème de bin packing. La présentation du problème du bin packing est faite dans la Section 2.5. L'analogie entre les différents problèmes considérés et le problème du bin packing est faite dans cette même section. Ainsi, une résolution du problème du bin packing sera le point de départ de la résolution des problèmes de planification d'activités avec affectation de ressources.

## 2.2 Spécification

### 2.2.1 Processus de modélisation

La résolution d'un problème nécessite la réalisation d'un modèle plus ou moins complexe du système étudié pour l'évaluation de ses performances. Le processus de modélisation défini par (Gourgand, 1984) est appliqué. Ce processus est itératif et se compose de quatre étapes :

- La construction d'un modèle de connaissance à partir de l'observation du système,
- L'élaboration du modèle d'action à partir du modèle de connaissance,
- L'exploitation du modèle d'action pour obtenir les valeurs des critères de performance et l'analyse de ces valeurs en vue d'appréhender et d'améliorer le comportement du système,
- La modification éventuelle des valeurs des caractéristiques du système dans le but de modifier les valeurs des critères de performance ou le comportement du modèle par rapport au système réel.

La décomposition systémique proposée par la méthodologie ASCI est utilisée. La méthodologie ASCI (Analyse, Spécification, Conception et Implémentation) a été

développée pour la conception et l'implémentation d'un environnement de simulation. Elle peut être appliquée à un système qui existe déjà ou à un système qui doit encore être conçu (Gourgand and Kellert, 1991). Pour analyser un système complexe, ASCI conseille une séparation du système en trois sous-systèmes : le sous-système physique, le sous-système logique et le sous-système décisionnel :

- Le sous-système physique est composé des entités physiques utilisées pour réaliser l'ensemble des activités, leur distribution géographique et leurs interconnexions.
- Le sous-système logique est composé des flux que le système doit traiter, toutes les activités concernant le traitement de ces flux et toutes les entités dans le système qui y sont relatives.
- Le sous-système décisionnel est le centre de décision qui contient toutes les règles de gestion, d'affectation de ressources et de fonctionnement du système.

Chaque sous-système interagit avec les deux autres tel que décrit dans la Figure 2.1 (Chabrol et al., 2008).

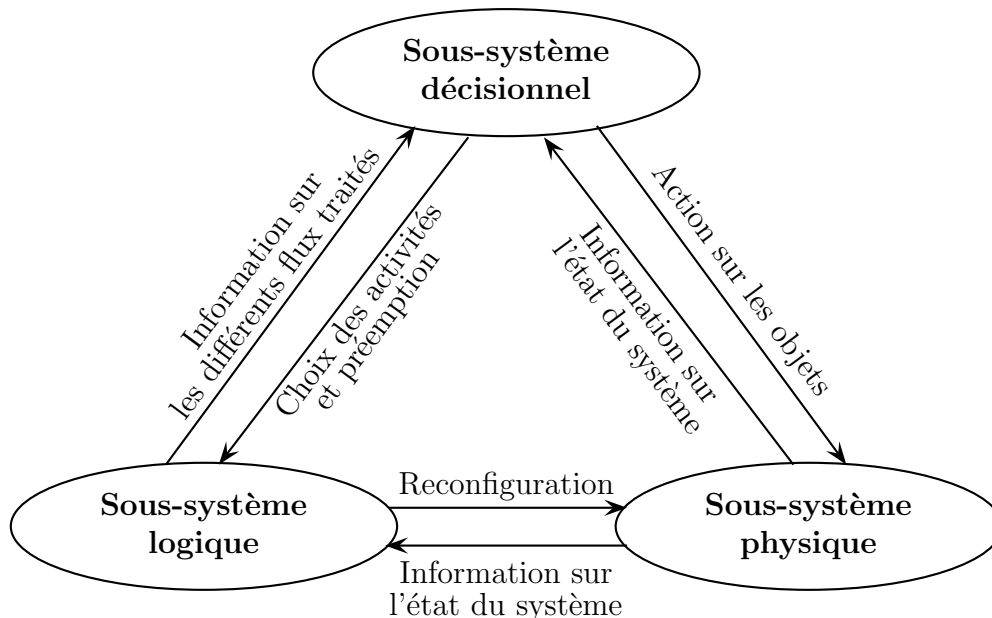


FIGURE 2.1 – Communication entre les trois sous-systèmes

Dans la suite, nous utilisons cette méthodologie afin de réaliser la spécification du problème étudié.

### 2.2.2 Sous-système physique

La CHT est composée de plusieurs sites. Il y a une distance connue entre ces sites.

Sur chaque site, il y a un ou plusieurs équipements. Un équipement appartient à un type. Chaque équipement a un planning d'ouverture qui définit ses heures de disponibilité par période. Par exemple, un équipement donné peut être disponible uniquement cinq heures le lundi parce qu'il nécessite une opération de maintenance ou parce qu'un médecin

extérieur l'a réservé. Des heures d'ouverture supplémentaires peuvent être autorisées.

Des ressources humaines composent une équipe médicale. La composition de cette équipe dépend de l'activité considérée. Cette équipe peut avoir un nombre spécifique de brancardiers, de médecins spécialistes, d'infirmiers, etc.

Les ressources humaines appartiennent à un site donné, c'est le site sur lequel elles sont employées, mais elles peuvent travailler sur d'autres sites appartenant à la même CHT. Ainsi, elles peuvent être amenées à se déplacer entre les différents sites. Par exemple, une ressource humaine peut travailler sur le site A le lundi matin, puis sur le site B le lundi après-midi, puis encore sur le site B le mardi matin et de nouveau sur le site A le mardi après-midi. Un temps est donné aux ressources humaines pour se déplacer d'un site à un autre.

Une ressource humaine peut utiliser un type d'équipement ou plusieurs en fonction de ses compétences. Une ressource humaine polyvalente, c'est-à-dire qui a la compétence de travailler sur plusieurs types d'équipement, est potentiellement moins performante qu'une ressource humaine ayant la compétence de ne travailler que sur un type d'équipement, voire un équipement en particulier. Cette performance peut se traduire à travers les temps de traitement des activités concernées.

Chaque ressource humaine a un planning qui définit son temps de travail réglementaire, tenant compte des temps de pause et des congés.

### 2.2.3 Sous-système logique

Une activité nécessite des ressources humaines et un équipement. Toutes les ressources requises doivent être compatibles. L'activité est compatible avec certains équipements. L'équipement affecté doit être compatible avec l'activité. Cet équipement appartient à un type, la ressource humaine affectée doit avoir les compétences nécessaires pour utiliser ce type d'équipement.

Une activité doit être réalisée avant une période au plus tard, appelée période limite. Chaque activité a un temps de traitement connu. Ce temps de traitement peut dépendre de la ressource humaine affectée et de l'équipement utilisé. Les deux peuvent ne pas avoir les mêmes compétences sur l'activité considérée. Chaque activité commence pendant une période et termine pendant cette même période.

Chaque activité a un site de référence, là où elle doit être réalisée, si possible.

### 2.2.4 Sous-système décisionnel

L'objectif est la création d'un modèle qui, à partir d'un ensemble d'activités, construit un planning associant le triplet {activité, ressources humaines, équipement} à une période. L'étude est faite dans une approche prédictive, toutes les activités peuvent être traitées depuis le début de l'horizon de planification. L'objectif de ce modèle n'est pas de faire un ordonnancement précis des activités mais d'attribuer une période à chaque activité. Cette planification doit respecter des contraintes et optimiser des critères.

### 2.2.4.1 Critères

Trois classes de critères peuvent être définies : les critères économiques, les critères relatifs au confort du patient ou les critères hospitaliers relatifs au bon fonctionnement du système.

À propos des aspects économiques, les critères sont relatifs aux coûts. Les taux d'occupation de chaque site, chaque équipement et de chaque ressource humaine permettent de s'assurer de la bonne utilisation de ces entités. Afin d'être le plus économique possible, il faut que ces taux d'occupation soient maximaux. Il est cependant préférable de définir une marge de sécurité afin que le système puisse être réactif en cas d'aléas. Si un retard survient en début de journée et que la planification n'est réalisée qu'à 95% de la capacité maximale, le retard pourra être absorbé par cette marge de sécurité. Si la planification est faite à 100%, un retard risquerait l'annulation de la dernière activité prévue en fin de journée ou la réalisation d'heures supplémentaires si cela est possible.

Toutes les activités sont planifiées pendant l'horizon de planification considéré. La période affectée à la dernière activité est un critère pertinent permettant de s'assurer que toutes les activités sont réalisées au plus tôt. Cette période sera appelée *makespan* par la suite. Plus le *makespan* est petit, plus il reste du temps disponible en fin d'horizon temporel pour potentiellement réaliser les activités suivantes. De la même façon, la somme des périodes affectées à toutes les activités est aussi un critère pertinent traduisant le fait que toutes les activités sont réalisées au plus tôt.

Le nombre de déplacements des ressources humaines au sein de la CHT peut être considéré. De la même façon, le nombre d'heures supplémentaires du temps de travail des ressources humaines ou du temps d'ouverture des équipements peut être pris en compte. Les heures supplémentaires et les déplacements ont un coût pour le système hospitalier. Il est préférable de les minimiser. Mais il peut être intéressant d'autoriser certaines heures supplémentaires et certains déplacements dans le but d'augmenter le nombre d'activités pendant la période considérée. Le bon compromis entre ces critères est à définir.

À propos du confort du patient, le critère est le nombre d'activités réalisées sur leur site de référence. Si une activité ne peut pas être affectée sur son site de référence, un autre critère peut être la distance entre le site de référence et le site effectif pour la réalisation de chaque activité.

Concernant les critères hospitaliers, le nombre d'activités réalisées avant leur période limite doit être maximisé. Les activités doivent être affectées avant leur période limite. Ainsi si le patient considéré nécessite d'autres activités, l'activité suivante pourra être réalisée à temps.

Les critères peuvent ne pas être les mêmes en fonction du niveau utilisé. Le Tableau 2.1 récapitule les différents critères utilisables par niveau de décision.

### 2.2.4.2 Contraintes

Les contraintes peuvent être divisées en deux catégories :

Les contraintes qui doivent être respectées :

	Niveau Stratégique	Niveau Tactique	Niveau Opérationnel
Taux d'occupation de chaque site	×	×	
Taux d'occupation de chaque équipement	×	×	
Taux d'occupation de chaque ressource humaine	×	×	
Nombre de déplacements effectués par les ressources humaines	×	×	
Nombre d'heures supplémentaires effectuées par les ressources humaines	×	×	
Nombre d'heures supplémentaires effectuées par les équipements	×	×	
Période affectée à la dernière activité (makespan)	×	×	×
Sommes des périodes affectées à toutes les activités	×	×	×
Nombre d'activités réalisées avant leur date limite		×	×
Nombre d'activités réalisées sur leur site de référence		×	×
Somme des distances entre le site affecté et le site de référence		×	×

TABLEAU 2.1 – Les critères rencontrés à chaque niveau de décision

- La compatibilité entre l'équipement et l'activité : l'affectation doit satisfaire à la liste donnée d'incompatibilités entre les activités et les équipements.
- La compatibilité entre les compétences de la ressource humaine et le type de l'équipement : pour une même activité, la ressource humaine affectée doit être capable de travailler sur le type de l'équipement considéré.
- Chaque activité doit être affectée à une ressource humaine, un équipement et une période.
- Lors de l'affectation, la ressource humaine considérée doit être affectée pendant la période considérée sur le site où l'équipement est localisé.
- Si une ressource humaine est autorisée à se déplacer pendant l'horizon de planification, ses déplacements sont contraints : une ressource humaine ne peut travailler que sur un site pendant une période.

Les contraintes qu'il est souhaitable de respecter :

- Les activités doivent être réalisées avant leur période limite.
- Les activités doivent être planifiées sur leur site de référence.
- Les équipements doivent être utilisés pendant leur temps d’ouverture. Sinon, le temps additionnel est considéré comme des heures supplémentaires. Les heures supplémentaires sont limitées en temps.
- Les ressources humaines doivent travailler pendant leur planning prédéfini. Sinon, le temps additionnel est considéré comme des heures supplémentaires. Les heures supplémentaires sont limitées en temps.

## 2.3 Problèmes considérés

Le problème considéré est le problème de planification d’activités avec affectation des ressources humaines et équipements requis. Ainsi, nos travaux se concentrent au niveau tactique microscopique. Le diagramme de classes UML du problème considéré est donné par la Figure 2.2. Des hypothèses sont maintenant faites afin de diviser le problème complet en plusieurs sous-problèmes.

### 2.3.1 Hypothèses

Les hypothèses suivantes sont faites :

- Un seul type de ressource humaine est considéré. Les ressources humaines sont compatibles avec un ou des types d’équipement.
- Les temps de traitement des activités sont donnés et fixes.
- Pour chaque activité, la date de disponibilité (release date  $r_i$ ) est égale à la date de prise de rendez-vous. Ces dates sont nulles : toutes les activités sont connues au début de la planification.
- Chaque patient n’a qu’une activité à réaliser. Finalement, la notion de patient n’est pas considérée, seule l’activité est prise en compte.
- Les notions de site et service sont confondues. Les distances entre sites sont prises en compte dans les temps alloués aux ressources humaines pour se déplacer d’un site à un autre. Ici, ces temps sont supposés constants, donc tous les sites sont supposés être à la même distance les uns des autres.

### 2.3.2 Définition des problèmes considérés

Le problème complet est divisé en trois sous-problèmes de difficulté croissante :

- Le problème 1 est le plus basique, l’hypothèse est faite que les ressources humaines sont toujours disponibles. Seuls les équipements sont considérés. Le diagramme de classe du problème 1 est représenté par la Figure 2.3.
- Dans le problème 2, les ressources humaines sont à capacité limitée. Elles peuvent travailler sur un ou plusieurs types d’équipement. Elles ne peuvent pas se déplacer, elles travaillent toujours sur le même site. L’affectation sur site des ressources humaines est donnée.

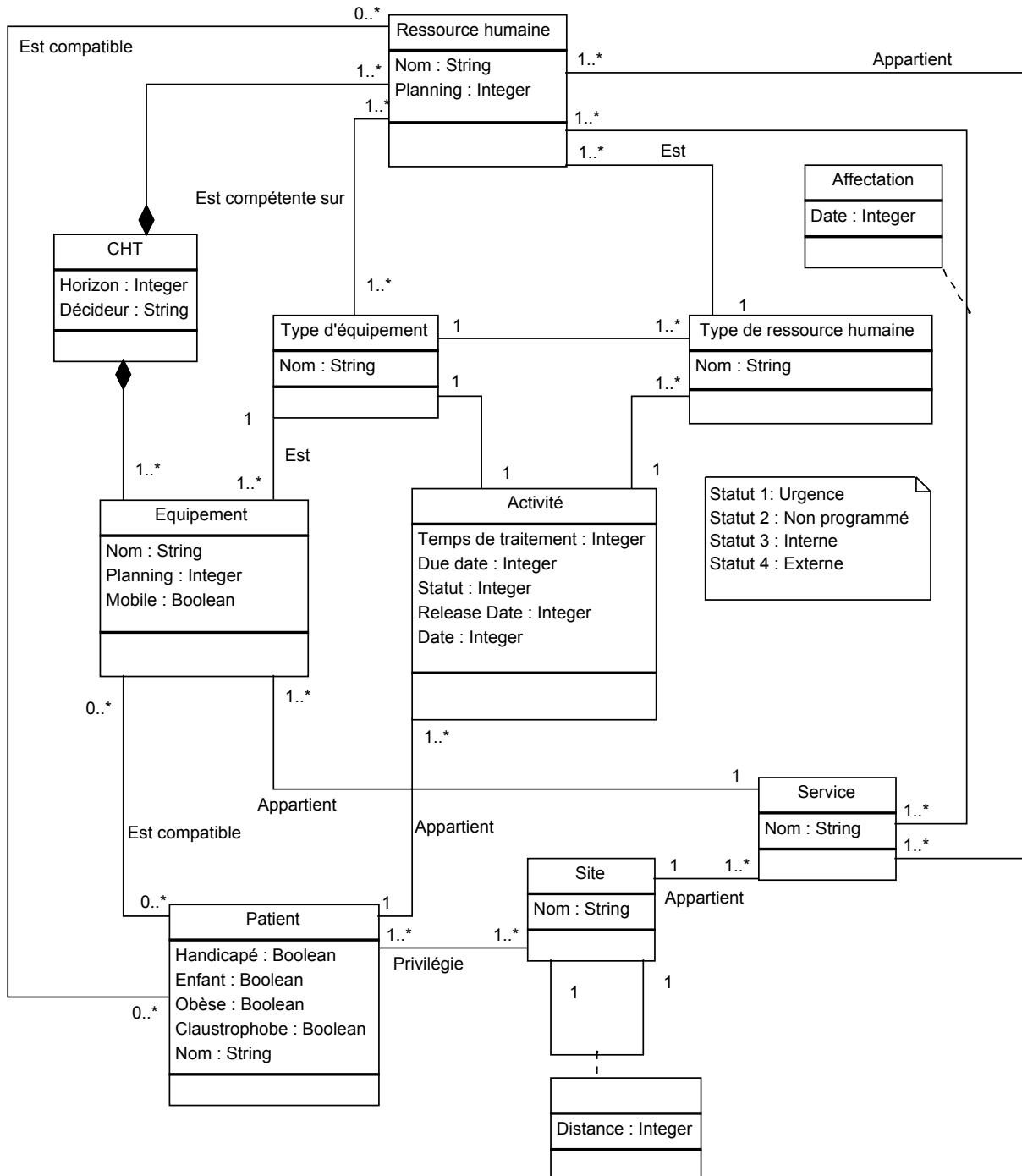


FIGURE 2.2 – Diagramme de classes du problème complet de la CHT

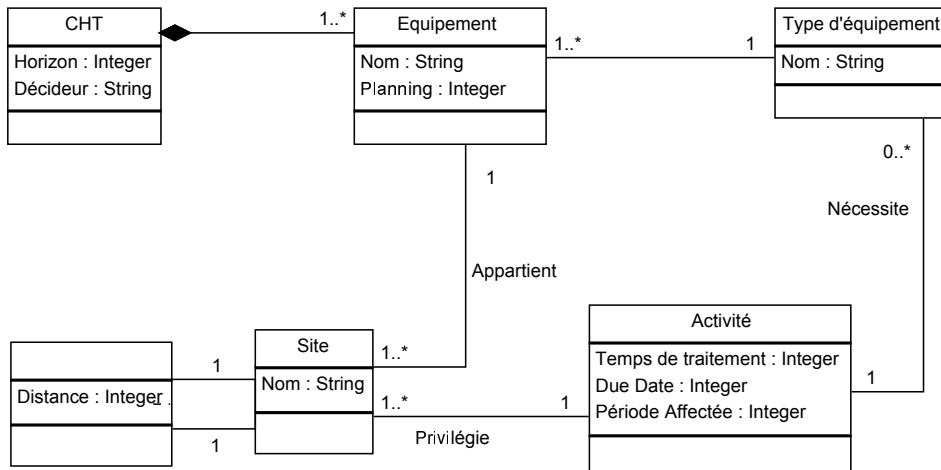


FIGURE 2.3 – Diagramme de classes du problème 1

- Dans le problème 3, les ressources humaines sont mobiles. Les ressources humaines peuvent travailler sur plusieurs sites, elles peuvent se déplacer d'un site à l'autre. L'affectation sur site des ressources humaines est à déterminer par le modèle. Le diagramme de classe du problème 3 est représenté par la Figure 2.4.

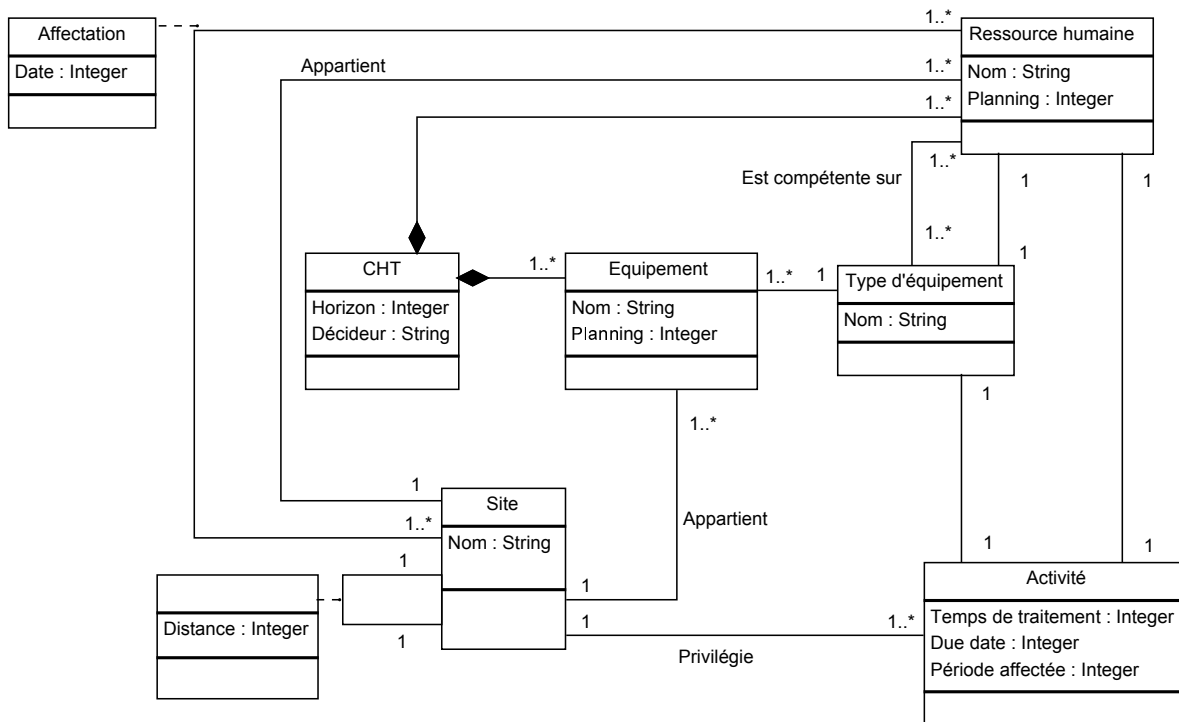


FIGURE 2.4 – Diagramme de classes du problème 3



## 2.4 Formalisation

Ces trois problèmes sont formalisés mathématiquement, en commençant par le problème 1 le plus basique jusqu'au problème 3 le plus complexe. Au fur et à mesure de la formalisation des problèmes, les données, variables, contraintes et critères supplémentaires sont ajoutés. Finalement, la Section 2.4.4 fait un récapitulatif des données, variables, contraintes et critères utilisés par chacun des modèles définis.

Les notations suivantes sont utilisées dans chacun des modèles :

- L'indice  $i$  correspond à l'activité.
- L'indice  $k$  correspond au site.
- L'indice  $l$  correspond à l'équipement.
- L'indice  $t$  correspond à la période.
- L'indice  $m$  correspond à la ressource humaine.

### 2.4.1 Modèle 1 : CHT

#### Données

Les données utilisées dans ce modèle sont les suivantes.

$N$  : l'ensemble des activités à planifier dans la CHT pendant l'horizon de planification.

$K$  : l'ensemble des sites distincts de la CHT.

$L$  : l'ensemble des équipements disponibles au sein de la CHT.

$T$  : l'ensemble des périodes.

$t_i$  : le temps de traitement de l'activité  $i \in N$ .

$d_i$  : la période limite de l'activité  $i \in N$ .

$$r_{i,k} = \begin{cases} 1 & \text{si le site } k \in K \text{ est le site de référence de l'activité } i \in N, \\ 0 & \text{sinon.} \end{cases}$$

$$c_{i,l} = \begin{cases} 1 & \text{si l'activité } i \in N \text{ peut être réalisée par l'équipement } l \in L, \\ 0 & \text{sinon.} \end{cases}$$

$$s_{k,l} = \begin{cases} 1 & \text{si l'équipement } l \in L \text{ est situé sur le site } k \in K, \\ 0 & \text{sinon.} \end{cases}$$

$p_{l,t}$  : le temps d'ouverture de l'équipement  $l \in L$  pendant la période  $t \in T$ .

$L_{max}$  : le temps d'ouverture maximal des équipements par période.

#### Variables

Dans le modèle 1, la variable considérée est l'affectation d'une activité à un équipement pendant une période.

$$x_{i,l,t} = \begin{cases} 1 & \text{si l'activité } i \in N \text{ est affectée à l'équipement } l \in L \\ & \text{pendant la période } t \in T, \\ 0 & \text{sinon.} \end{cases}$$

La variable  $g_{l,t}$  définit le temps d'ouverture supplémentaire alloué à l'équipement  $l \in L$  pendant la période  $t \in T$ .

### Contraintes

Chaque activité est affectée à un équipement et une période, selon l'Équation (2.1).

$$\sum_{l \in L} \sum_{t \in T} x_{i,l,t} = 1, \forall i \in N \quad (2.1)$$

Chaque activité est affectée à un équipement compatible, selon l'Équation (2.2).

$$\sum_{l \in L} \sum_{t \in T} x_{i,l,t} \cdot (1 - c_{i,l}) = 0, \forall i \in N \quad (2.2)$$

Un équipement peut être imposé pour réaliser une activité. La contrainte "l'équipement  $l_2$  doit être utilisé pour réaliser l'activité  $i$ " est définie par l'Équation (2.3).

$$\sum_{t \in T} x_{i,l_2,t} = 1 \quad (2.3)$$

L'activité est réalisée avant sa date limite, selon l'Équation (2.4).

$$\sum_{l \in L} \sum_{\substack{t \in T \\ t \leq d_i}} x_{i,l,t} = 1, \forall i \in N \quad (2.4)$$

L'activité est réalisée sur son site de référence, selon l'Équation (2.5).

$$\sum_{k \in K} \sum_{l \in L} \sum_{t \in T} x_{i,l,t} \cdot s_{k,l} \cdot r_{i,k} = 1, \forall i \in N \quad (2.5)$$

Pour chaque période, pour chaque équipement, le temps d'ouverture supplémentaire est défini par l'Équation (2.6).

$$\sum_{i \in N} t_i \cdot x_{i,l,t} - p_{l,t} \leq g_{l,t}, \forall l \in L, \forall t \in T \quad (2.6)$$

Pour chaque équipement et pour chaque période, le temps d'ouverture supplémentaire est limité, selon l'Équation (2.7).

$$p_{l,t} + g_{l,t} \leq L_{max}, \forall l \in L, \forall t \in T \quad (2.7)$$

Si les temps d'ouverture supplémentaires des équipements ne sont pas autorisés, le nombre d'heures supplémentaires est nul, selon l'Équation (2.8).

$$\sum_{l \in L} \sum_{t \in T} g_{l,t} = 0 \quad (2.8)$$

Les contraintes binaires sont définies par l'Équation (2.9).

$$x_{i,l,t} \in \{0, 1\}, \forall i \in N, \forall l \in L, \forall t \in T \quad (2.9)$$

Les contraintes de non négativité sont définies par l'Équation (2.10).

$$g_{l,t} \geq 0, \forall l \in L, \forall t \in T \quad (2.10)$$

## Critères

Le modèle peut minimiser trois critères.

La somme des périodes affectées à toutes les activités, selon l'Équation (2.11).

$$\sum_{i \in N} \sum_{l \in L} \sum_{t \in T} t \cdot x_{i,l,t} \quad (2.11)$$

La période affectée à la dernière activité planifiée, selon l'Équation (2.12).

$$\max_{t \in T} \sum_{i \in N} \sum_{l \in L} t \cdot x_{i,l,t} \quad (2.12)$$

Le temps d'ouverture supplémentaire des équipements s'il est autorisé, selon l'Équation (2.13).

$$\sum_{l \in L} \sum_{t \in T} g_{l,t} \quad (2.13)$$

Les contraintes définies par les Équations (2.4) et (2.5) peuvent être relâchées. Elles deviennent les critères suivants.

Le nombre d'activités réalisées après leur période limite, selon l'Équation (2.14).

$$\sum_{i \in N} \sum_{l \in L} \sum_{\substack{t \in T \\ t > d_i}} x_{i,l,t} \quad (2.14)$$

Le nombre d'activités non réalisées sur leur site de référence, selon l'Équation (2.15).

$$|N| - \sum_{i \in N} \sum_{k \in K} \sum_{l \in L} \sum_{t \in T} x_{i,l,t} \cdot s_{k,l} \cdot r_{i,k} \quad (2.15)$$

### 2.4.2 Modèle 2 : CHT\_RH non mobiles

#### Données

Les mêmes données que dans le modèle 1 sont utilisées. Les données supplémentaires prises en compte par le modèle 2 sont les suivantes.

$A$  : l'ensemble des types d'équipement.

$M$  : l'ensemble des ressources humaines.

$$b_{m,a} = \begin{cases} 1 & \text{si la ressource humaine } m \in M \text{ peut travailler sur le type} \\ & \text{d'équipement } a \in A, \\ 0 & \text{sinon.} \end{cases}$$

$$d_{l,a} = \begin{cases} 1 & \text{si l'équipement } l \in L \text{ appartient au type } a \in A, \\ 0 & \text{sinon.} \end{cases}$$

$$v_{m,k,t} = \begin{cases} 1 & \text{si la ressource humaine } m \in M \text{ travaille sur le site } k \in K \\ & \text{pendant la période } t \in T, \\ 0 & \text{sinon.} \end{cases}$$

$q_{m,t}$  : le temps de travail de la ressource humaine  $m \in M$  pendant la période  $t \in T$ .

$M_{max}$  : le temps maximal travaillé par une ressource humaine pendant une période.

## Variables

Dans le modèle 2, la variable est l'affectation d'un équipement et d'une ressource humaine à une activité pendant une période.

$$x_{i,l,t,m} = \begin{cases} 1 & \text{si l'activité } i \in N \text{ est affectée à l'équipement } l \in L \\ & \text{et à la ressource humaine } m \in M \text{ pendant la période } t \in T, \\ 0 & \text{sinon.} \end{cases}$$

$h_{m,t}$  calcule le nombre d'heures supplémentaires de la ressource humaine  $m \in M$  pendant la période  $t \in T$ .

## Contraintes

Chaque activité est affectée à un équipement, une ressource humaine et une période, selon l'Équation (2.16).

$$\sum_{l \in L} \sum_{t \in T} \sum_{m \in M} x_{i,l,t,m} = 1, \forall i \in N \quad (2.16)$$

Une activité est affectée à un équipement compatible, selon l'Équation (2.17).

$$\sum_{l \in L} \sum_{t \in T} \sum_{m \in M} x_{i,l,t,m} \cdot (1 - c_{i,l}) = 0, \forall i \in N \quad (2.17)$$

L'équipement et la ressource humaine affectés à une activité sont compatibles, selon l'Équation (2.18).

$$\sum_{l \in L} \sum_{t \in T} \sum_{m \in M} x_{i,l,t,m} \cdot (1 - \sum_{a \in A} d_{l,a} \cdot b_{m,a}) = 0, \forall i \in N \quad (2.18)$$

La ressource humaine et l'équipement utilisés doivent être affectés au même site, selon l'Équation (2.19).

$$x_{i,l,t,m} - \sum_{k \in K} s_{k,l} \cdot v_{m,k,t} \leq 0, \forall i \in N, \forall l \in L, \forall m \in M, \forall t \in T \quad (2.19)$$

Un équipement peut être imposé pour réaliser une activité. La contrainte "l'équipement  $l_2$  doit être utilisé pour réaliser l'activité  $i$ " est définie par l'Équation (2.20).

$$\sum_{t \in T} \sum_{m \in M} x_{i,l_2,t,m} = 1 \quad (2.20)$$

Une ressource humaine peut être imposée pour réaliser une activité. La contrainte "la ressource humaine  $m_2$  doit être utilisée pour réaliser l'activité  $i$ " est définie par l'Équation (2.21).

$$\sum_{l \in L} \sum_{t \in T} x_{i,l,t,m_2} = 1 \quad (2.21)$$

Une ressource humaine peut ne travailler que sur un équipement. Par exemple, l'équipement  $l_1$  peut être imposé à la ressource humaine  $m_1$ , définie par l'Équation (2.22).

$$\sum_{i \in N} \sum_{\substack{l \in L \\ l \neq l_1}} \sum_{t \in T} x_{i,l,t,m_1} = 0 \quad (2.22)$$

L'activité est réalisée avant sa date limite, selon l'Équation (2.23).

$$\sum_{l \in L} \sum_{\substack{t \in T \\ t \leq d_i}} \sum_{m \in M} x_{i,l,t,m} = 1, \forall i \in N \quad (2.23)$$

L'activité est réalisée sur son site de référence, selon l'Équation (2.24).

$$\sum_{k \in K} \sum_{l \in L} \sum_{t \in T} \sum_{m \in M} x_{i,l,t,m} \cdot s_{k,l} \cdot r_{i,k} = 1, \forall i \in N \quad (2.24)$$

Les temps d'ouverture supplémentaires des équipements sont définis par l'Équation (2.25).

$$\sum_{i \in N} \sum_{m \in M} t_i \cdot x_{i,l,t,m} - p_{l,t} \leq r_{l,t}, \forall l \in L, \forall t \in T \quad (2.25)$$

Pour chaque équipement et pour chaque période, le temps d'ouverture supplémentaire est limité, selon l'Équation (2.7). Si les temps d'ouverture supplémentaires des équipements ne sont pas autorisés, il faut s'assurer que le nombre d'heures supplémentaires est nul, selon l'Équation (2.8). Pour chaque période, pour chaque ressource humaine, le temps supplémentaire travaillé par la ressource humaine est calculé par l'Équation (2.26).

$$\sum_{i \in N} \sum_{l \in L} t_i \cdot x_{i,l,t,m} - q_{m,t} \leq h_{m,t}, \forall m \in M, \forall t \in T \quad (2.26)$$

Pour chaque ressource humaine et pour chaque période, les heures supplémentaires sont contraintes en temps, selon l'Équation (2.27).

$$q_{m,t} + h_{m,t} \leq M_{max}, \forall m \in M, \forall t \in T \quad (2.27)$$

Si les ressources humaines ne sont pas autorisées à effectuer des heures supplémentaires, le nombre d'heures supplémentaires est nul, selon l'Équation (2.28).

$$\sum_{m \in M} \sum_{t \in T} h_{m,t} = 0 \quad (2.28)$$

Les contraintes binaires sont définies par l'Équation (2.29).

$$x_{i,l,t,m} \in \{0, 1\}, \forall i \in N, \forall l \in L, \forall m \in M, \forall t \in T \quad (2.29)$$

Les contraintes de non négativité relatives aux temps d'ouverture supplémentaires des équipements sont définies par l'Équation (2.10). Celles relatives aux heures supplémentaires des ressources humaines sont définies par l'Équation (2.30).

$$h_{m,t} \geq 0, \forall m \in M, \forall t \in T \quad (2.30)$$

## Critères

Les mêmes critères que pour le modèle 1 sont à minimiser pour le modèle 2. Cependant, la variable  $x_{i,l,t}$  est à remplacer par la variable  $x_{i,l,t,m}$ . La somme des périodes affectées à toutes les activités est donc définie par l'Équation (2.31). La période affectée à la dernière activité planifiée est définie par l'Équation (2.32).

$$\sum_{i \in N} \sum_{l \in L} \sum_{t \in T} \sum_{m \in M} t \cdot x_{i,l,t,m} \quad (2.31)$$

$$\max_{t \in T} \sum_{i \in N} \sum_{l \in L} \sum_{m \in M} t \cdot x_{i,l,t,m} \quad (2.32)$$

Les temps d'ouverture supplémentaires des équipements sont définis par l'Équation (2.13). Les heures supplémentaires des ressources humaines sont définies par l'Équation (2.33).

$$\sum_{m \in M} \sum_{t \in T} h_{m,t} \quad (2.33)$$

Les contraintes définies par les Équations (2.23) et (2.24) peuvent être relâchées. Elles deviennent les critères suivants.

Le nombre d'activités réalisées après leur période limite est défini par l'Équation (2.34).

$$\sum_{i \in N} \sum_{l \in L} \sum_{\substack{t \in T \\ t > d_i}} \sum_{m \in M} x_{i,l,t,m} \quad (2.34)$$

Le nombre d'activités non réalisées sur leur site de référence est défini par l'Équation (2.35).

$$|N| - \sum_{i \in N} \sum_{k \in K} \sum_{l \in L} \sum_{t \in T} \sum_{m \in M} x_{i,l,t,m} \cdot S_{k,l} \cdot T_{i,k} \quad (2.35)$$

### 2.4.3 Modèle 3 : CHT\_RH mobiles

Dans ce modèle, les ressources humaines sont mobiles, leur affectation sur site n'est pas donnée mais est à déterminer par le modèle.

#### Données

Les mêmes données que celles utilisées dans le modèle 2 sont utilisées par le modèle 3. Les différences suivantes relatives aux données sont à noter.

L'affectation  $v_{m,k,t}$  de la ressource humaine  $m$  sur le site  $k$  pendant la période  $t$  est supprimée de la liste des données parce que l'affectation sur site des ressources humaines est à déterminer par le modèle.

Le site employeur de la ressource humaine est défini par la donnée suivante :

$$v_{m,k,0} = \begin{cases} 1 & \text{si la ressource humaine } m \in M \text{ est employée par le site } k \in K, \\ 0 & \text{sinon.} \end{cases}$$

Le temps alloué à une ressource humaine pour se déplacer d'un site à un autre est donné par  $tt$ .

## Variables

Les mêmes variables que pour le modèle 2 sont utilisées dans le modèle 3. Deux nouvelles variables sont nécessaires.

Une variable affecte une ressource humaine à un site pendant une période.

$$y_{m,k,t} = \begin{cases} 1 & \text{si la ressource humaine } m \in M \text{ travaille sur le site } k \in K \text{ pendant} \\ & \text{la période } t \in T, \\ 0 & \text{sinon.} \end{cases}$$

Au début de l'horizon de planification, la ressource humaine commence depuis le site employeur :  $y_{m,k,0} = v_{m,k,0}$ .

La seconde variable supplémentaire est le déplacement de la ressource humaine au début de la période considérée.

$$w_{m,t} = \begin{cases} 1 & \text{si la ressource humaine } m \in M \text{ se déplace au début de} \\ & \text{la période } t \in T, \\ 0 & \text{sinon.} \end{cases}$$

## Contraintes

Toutes les contraintes à respecter dans le modèle 2 sont à respecter dans le modèle 3. La formulation de la contrainte considérée est précisée si elle n'est pas la même que dans le modèle précédent.

Chaque activité est affectée à un équipement, une ressource humaine et une période, selon l'Équation (2.16). Une activité est affectée à un équipement compatible, selon l'Équation (2.17). L'équipement et la ressource humaine affectés à une activité sont compatibles, selon l'Équation (2.18).

La ressource humaine et l'équipement utilisés doivent être affectés au même site, selon l'Équation (2.36).

$$x_{i,l,t,m} - \sum_{k \in K} s_{k,l} \cdot y_{m,k,t} \leq 0, \forall i \in N, \forall l \in L, \forall m \in M, \forall t \in T \quad (2.36)$$

Un équipement peut être imposé pour réaliser une activité. La contrainte "l'équipement  $l_2$  doit être utilisé pour réaliser l'activité  $i$ " est définie par l'Équation (2.20). Une ressource humaine peut être imposée pour réaliser une activité. La contrainte "la ressource humaine  $m_2$  doit être utilisée pour réaliser l'activité  $i$ " est définie par l'Équation (2.21). Une ressource humaine peut ne travailler que sur un équipement. Par exemple, l'équipement  $l_1$  peut être imposé à la ressource humaine  $m_1$ , définie par l'Équation (2.22).

Les temps d'ouverture supplémentaires des équipements sont définis par l'Équation (2.25). Pour chaque équipement et pour chaque période, le temps d'ouverture supplémentaire est limité, selon l'Équation (2.7). Si les temps d'ouverture supplémentaires des équipements ne sont pas autorisés, le nombre d'heures supplémentaires est nul, selon l'Équation (2.8). Pour chaque période, pour chaque ressource humaine, le calcul du temps supplémentaire travaillé par la ressource humaine est défini par l'Équation (2.37).

$$\sum_{i \in N} \sum_{l \in L} t_i \cdot x_{i,l,t,m} - q_{m,t} + w_{m,t} \cdot t \leq h_{m,t}, \forall m \in M, \forall t \in T \quad (2.37)$$

Pour chaque ressource humaine et pour chaque période, les heures supplémentaires sont contraintes en temps, selon l'Équation (2.27). Si les ressources humaines ne sont pas autorisées à effectuer des heures supplémentaires, les heures supplémentaires sont nulles, selon l'Équation (2.28). Une ressource humaine est affectée à un seul site pendant une période, selon l'Équation (2.38).

$$\sum_{k \in K} y_{k,m,t} \leq 1, \forall m \in M, \forall t \in T \quad (2.38)$$

La détermination des déplacements de la ressource humaine est définie par l'Équation (2.39).

$$y_{m,k,t} - y_{m,k,(t-1)} \leq w_{m,t}, \forall k \in K, \forall m \in M, \forall t \in T \setminus \{1\} \quad (2.39)$$

La réalisation de l'activité avant sa date limite est formalisée par l'Équation (2.34). La réalisation de l'activité sur son site de référence est définie par l'Équation (2.35).

Les contraintes binaires sont définies par l'Équation (2.29). De nouvelles contraintes binaires relatives à l'affectation sur site des ressources humaines sont définies par l'Équation (2.40).

$$y_{m,k,t} \in \{0, 1\}, \forall k \in K, \forall m \in M, \forall t \in T \quad (2.40)$$

Les contraintes de non négativité relatives aux temps d'ouverture supplémentaires des équipements sont définies par l'Équation (2.10), celles relatives aux heures supplémentaires des ressources humaines par l'Équation (2.30). De nouvelles contraintes de non négativité relatives aux déplacements sont définies par l'Équation (2.41).

$$w_{m,t} \geq 0, \forall m \in M, \forall t \in T \quad (2.41)$$

## Critères

En plus des critères minimisés dans le modèle 2, un nouveau critère est à prendre en compte dans le modèle 3 : le nombre de déplacements des ressources humaines, selon l'Équation (2.42).

$$\sum_{m \in M} \sum_{t \in T} w_{m,t} \quad (2.42)$$

### 2.4.4 Récapitulatif des modèles

Les données, variables, contraintes et critères considérés dans chaque modèle sont résumés dans la suite, avec les numéros d'équation associés.

Dans le Tableau 2.2, les données sont associées à chacun des modèles. Une case vide signifie que la donnée n'est pas utilisée par le modèle considéré.

Le Tableau 2.3 résume les variables utilisées par chaque modèle.

Le Tableau 2.4 résume les contraintes qui doivent être respectées par chaque modèle. Les contraintes relatives aux temps d'ouverture des équipements et aux temps de travail des ressources humaines ne sont pas les mêmes si les heures d'ouverture supplémentaires des équipements ou les heures supplémentaires des ressources humaines sont autorisées ou interdites.

Les critères qui peuvent être optimisés par chacun des modèles sont résumés dans le Tableau 2.5.



Modèle	1	2	3
Activités à planifier	$N$	$N$	$N$
Sites	$K$	$K$	$K$
Équipements	$L$	$L$	$L$
Types d'équipement		$A$	$A$
Ressources humaines		$M$	$M$
Horizon de planification	$T$	$T$	$T$
Temps de traitement de l'activité	$t_i$	$t_i$	$t_i$
Temps donné à une ressource humaine pour aller d'un site à un autre			$tt$
Période limite de l'activité	$d_i$	$d_i$	$d_i$
Site de référence de l'activité	$r_{i,k}$	$r_{i,k}$	$r_{i,k}$
Type d'équipement sur lequel la ressource humaine peut travailler		$b_{m,a}$	$b_{m,a}$
Équipement qui peut réaliser l'activité	$c_{i,l}$	$c_{i,l}$	$c_{i,l}$
Type de l'équipement		$d_{l,a}$	$d_{l,a}$
Site de l'équipement	$s_{k,l}$	$s_{k,l}$	$s_{k,l}$
Site sur lequel la ressource humaine travaille pendant la période		$v_{m,k,t}$	
Site sur lequel la ressource humaine est employée			$v_{m,k,0}$
Temps d'ouverture de l'équipement	$pl,t$	$pl,t$	$pl,t$
Temps de travail de la ressource humaine		$q_{m,t}$	$q_{m,t}$
Temps d'ouverture maximal de l'équipement	$L_{max}$	$L_{max}$	$L_{max}$
Temps de travail maximal de la ressource humaine		$M_{max}$	$M_{max}$

TABLEAU 2.2 – Données considérées par chaque modèle

Modèle	1	2	3
Affectation d'une activité à un équipement pendant une période	$x_{i,l,t}$		
Affectation d'une activité à un équipement et une ressource humaine pendant une période		$x_{i,l,t,m}$	$x_{i,l,t,m}$
Nombre d'heures supplémentaires d'un équipement pendant une période	$g_{l,t}$	$g_{l,t}$	$g_{l,t}$
Nombre d'heures supplémentaires d'une ressource humaine pendant une période		$h_{m,t}$	$h_{m,t}$
Affectation d'une ressource humaine sur un site pendant une période			$y_{m,k,t}$
Déplacement d'une ressource humaine au début d'une période			$w_{m,t}$

TABLEAU 2.3 – Variables considérées par chaque modèle

Modèle	1	2	3
Affectation de chaque activité à un équipement et une période	(2.1)		
Affectation de chaque activité à un équipement, une période et une ressource humaine		(2.16)	(2.16)
Compatibilité activité équipement	(2.2)	(2.17)	(2.17)
Une activité affectée à un équipement ou une ressource humaine en particulier	(2.3)	(2.20), (2.21)	(2.20), (2.21)
Une ressource humaine travaille sur un équipement en particulier		(2.22)	(2.22)
Équipement et ressource humaine de même type		(2.18)	(2.18)
Équipement et ressource humaine sur le même site		(2.19)	(2.36)
Activité réalisée avant sa date limite	(2.4)	(2.23)	(2.23)
Activité réalisée sur son site de référence	(2.5)	(2.24)	(2.24)
Temps d'ouverture des équipements	((2.6) & (2.8))    ((2.6) & (2.7))	((2.25) & (2.8))    ((2.6) & (2.7))	((2.25) & (2.8))    ((2.6) & (2.7))
Temps de travail des des ressources humaines		((2.26) & (2.28))    ((2.26) & (2.27))	((2.37) & (2.28))    ((2.37) & (2.27))
Affectation d'un seul site par période à chaque ressource humaine			(2.38)
Calcul des déplacements			(2.39)
Contraintes binaires	(2.9)	(2.29)	(2.29) & (2.40)
Contraintes de non négativité	(2.10)	(2.10) & (2.30)	(2.10) & (2.30) & (2.41)

TABLEAU 2.4 – Contraintes utilisées par chaque modèle

Modèle	1	2	3
Somme des périodes affectées	(2.11)	(2.31)	(2.31)
Dernière période affectée	(2.12)	(2.32)	(2.32)
Heures d'ouverture supplémentaires des équipements	(2.13)	(2.13)	(2.13)
Nombre d'activités réalisées après leur date limite	(2.14)	(2.34)	(2.34)
Nombre d'activités réalisées sur leur site de référence	(2.15)	(2.35)	(2.35)
Heures supplémentaires des ressources humaines		(2.33)	(2.33)
Nombre de déplacements des ressources humaines			(2.42)

TABLEAU 2.5 – Critères pris en compte par chacun des modèles

## 2.5 Le problème du bin packing

Le problème de planification d'activités avec affectation de ressources peut être vu comme un problème de bin packing. Le problème du bin packing est présenté ainsi que les analogies entre les deux problèmes.

### 2.5.1 Présentation du problème

Le problème du bin packing considère  $N$  objets ayant chacun une taille connue et plusieurs boîtes de même capacité. L'objectif est de placer tous les objets dans un nombre minimum de boîtes en s'assurant que la somme des tailles des objets placés dans chaque boîte est inférieure à la capacité de la boîte en affectant chaque objet dans une et une seule boîte.

Ce problème existe aussi en deux et trois dimensions. Seul le cas à une dimension est considéré.

Par exemple, dans la Figure 2.5, les objets 1 et 2 sont rangés dans la boîte  $m_1$ , les objets 3 et 5 dans la boîte  $m_2$  et l'objet 4 dans la boîte  $m_3$ .

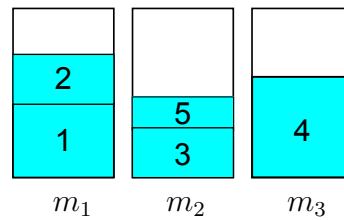


FIGURE 2.5 – Une solution du problème du bin packing 1D

Le problème peut être formalisé de la façon suivante.

#### Données

- $N$  le nombre d'objets à ranger.
- $w_i$  la taille de l'objet  $i$ .
- $N$  le nombre de boîtes disponibles.
- $C$  la capacité de chaque boîte.

#### Variables

- $x_{i,j} = \begin{cases} 1 & \text{si l'objet } i \in \{1, N\} \text{ est affecté à la boîte } j \in \{1, N\}, \\ 0 & \text{sinon.} \end{cases}$
- $y_j = \begin{cases} 1 & \text{si la boîte } j \in \{1, N\} \text{ est utilisée,} \\ 0 & \text{sinon.} \end{cases}$

### Contraintes

La capacité de chaque boîte doit être respectée (2.43). Chaque objet doit être affecté à une et une seule boîte (2.44). Les contraintes définies par les Équations (2.45) et (2.46) sont les contraintes d'intégrité.

$$\sum_{i=1}^N w_i \cdot x_{i,j} \leq C \cdot y_j, \quad \forall j \in \{1, N\} \quad (2.43)$$

$$\sum_{j=1}^N x_{i,j} = 1, \quad \forall i \in \{1, N\} \quad (2.44)$$

$$x_{i,j} \in \{0, 1\}, \quad \forall i \in \{1, N\}, \forall j \in \{1, N\} \quad (2.45)$$

$$y_j \in \{0, 1\}, \quad \forall j \in \{1, N\} \quad (2.46)$$

### Objectif

L'objectif est la minimisation du nombre de boîtes utilisées (2.47).

$$\min z = \sum_{j=1}^N y_j \quad (2.47)$$

Ce problème peut être rencontré dans des applications industrielles ou informatiques. (Bortfeldt and Wäscher, 2012) font un état de l'art sur le problème d'emballage : le problème du bin packing peut être utilisé pour ranger un ensemble de colis dans un nombre minimum de containers. Il peut également être utilisé pour résoudre les problèmes d'équilibrage de ligne d'assemblages (Wee and Magazine, 1982) ou d'ordonnancement multi-processeurs (Coffman Jr et al., 1978). (Garey et al., 1976) proposent une extension du problème du bin packing pour résoudre un problème d'ordonnancement avec des contraintes de ressources.

## 2.5.2 Analogies avec notre problème

Les analogies suivantes entre le problème de planification d'activités avec affectation de ressources et le problème du bin packing peuvent être faites.

### 2.5.2.1 Problème 1 : sans ressources humaines

Le problème 1 consiste en l'affectation des activités à un équipement pendant une période. Les ressources humaines sont supposées de capacité illimitée. Si nous considérons que l'horizon de planification est composé de couples  $(l, t)$  avec l'équipement  $l \in L$  et la période  $t \in T$ , l'objectif du problème est d'affecter les activités aux couples  $(l, t)$ . Les activités doivent être affectées le plus tôt possible, l'objectif est donc de minimiser le nombre de couples  $(l, t)$ , c'est-à-dire de minimiser le nombre de boîtes.

Une illustration de notre problème est proposée par la Figure 2.6. Dans cette illustration, l'affectation d'examens médicaux à des équipements de type radio, IRM ou scanner est considérée.

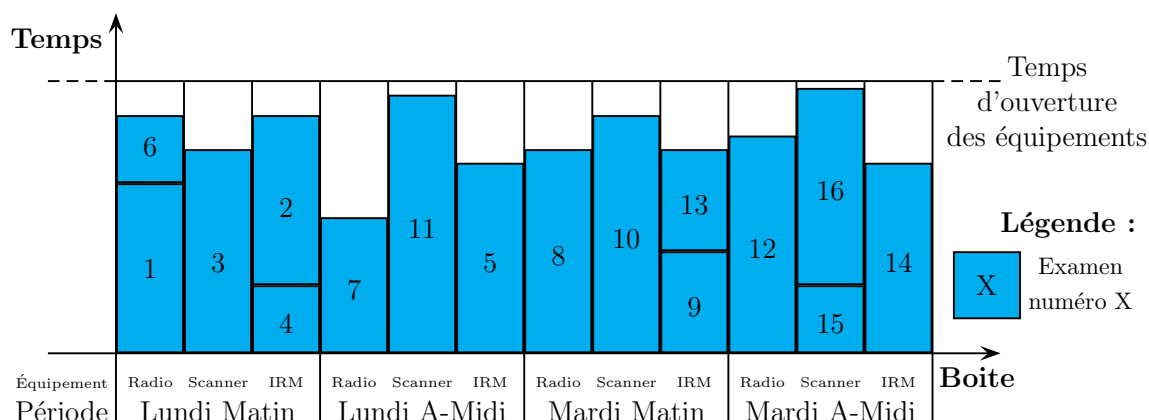


FIGURE 2.6 – Une représentation du problème de planification d’activités avec affectation de ressources

Le Tableau 2.6 résume les analogies entre le problème du bin packing et le problème de planification d’activités avec affectation de ressources. Notre problème considère une contrainte supplémentaire : avant l’affectation d’un équipement  $l$  à une activité  $i$ , il faut vérifier la contrainte de compatibilité, il faut vérifier que  $c_{i,l} = 1$ .

L’objectif principal du problème considéré est différent de celui du problème du bin packing. Alors que le problème du bin packing vise à minimiser le nombre de boîtes utilisées, notre problème vise à minimiser la somme des périodes affectées.

De plus, dans notre problème, les temps d’ouverture peuvent être différents en fonction de l’équipement et de la période. En reprenant l’analogie avec le problème du bin packing, cela signifie que les boîtes peuvent avoir des tailles différentes, ce qui n’est pas le cas pour le problème du bin packing de base.

### 2.5.2.2 Problèmes 2 et 3 : avec ressources humaines

Les problèmes 2 et 3 tiennent compte des ressources humaines. En plus d’être affectées à des équipements et des périodes, les activités doivent être affectées à des ressources humaines. L’analogie est faite entre ces problèmes et le problème du bin packing interdépendant.

Le problème du bin packing interdépendant a été introduit par (Boutevin, 2003). Soient  $p_1$  et  $p_2$  deux problèmes du bin packing. Chacun de ces problèmes contient un nombre donné de boîtes. Ce nombre peut différer entre les deux problèmes. Des groupes de boîtes sont définis à l’intérieur de chaque problème, ces groupes peuvent contenir une à plusieurs boîtes. Le nombre de groupes est le même dans les deux problèmes.

Chaque objet est affecté à une et une seule boîte dans chacun des problèmes. En l’absence de contraintes supplémentaires, la résolution de chacun de ces deux problèmes pourrait être traitée séparément. La notion d’interdépendance provient de l’ajout de la contrainte suivante : si un objet est affecté à une boîte du groupe  $g$  du problème  $p_1$ , alors il est obligatoirement affecté à une boîte du groupe  $g$  du problème  $p_2$ , et vice versa. L’objectif est de répartir les objets dans les boîtes des deux problèmes de manière à minimiser le nombre de boîtes utilisées dans un des deux problèmes, tout en satisfaisant les contraintes de capacité des boîtes et d’interdépendance entre les problèmes.

	Problème du bin packing	Problème de planification d'activités avec affectation de ressources
<b>Données</b>	Objet	Activité
	Boite	Couple équipement-période
	Taille d'un objet	Temps de traitement d'une activité
	Taille d'une boite	Temps d'ouverture d'un équipement
	-	Période limite
	-	Site de référence
<b>Problème</b>	Affecter les objets aux boites	Affecter les activités à une période et à un équipement
<b>Contraintes</b>	Contrainte de la taille des boites	Contrainte de la durée d'ouverture des équipements
	-	Contrainte de compatibilité
<b>Critères</b>	Minimiser le nombre de boites utilisées	-
	-	Minimiser la somme des périodes affectées

TABLEAU 2.6 – Tableau d'analogies entre les deux problèmes

Pour le problème de planification d'activités avec affectation de ressources, les deux problèmes peuvent être définis de la manière suivante :

- Problème  $p_1$  : affectation de chaque activité à un équipement pendant une période, tout en respectant la durée d'ouverture des équipements pendant la période et la compatibilité entre l'activité et l'équipement.
- Problème  $p_2$  : affectation de chaque activité à une ressource humaine pendant une période, tout en respectant le temps de travail de la ressource humaine pendant la période et la compatibilité entre l'activité et la ressource humaine.

La notion de compatibilité entre l'activité et la ressource humaine n'est pas directement définie mais peut-être déduite de la façon suivante : une activité  $i$  est compatible avec la ressource humaine  $m$  si et seulement si l'activité  $i$  est traitée par l'équipement  $l$  qui est de type  $a$  et que la ressource humaine  $m$  a la capacité de travailler sur le type  $a$ .

Dans notre cas, le groupe  $g$  correspond à la combinaison [période  $t \times$  type  $a$ ]. Dans les deux problèmes, l'activité doit être affectée à la même période et au même type.

Chaque couple (équipement, période) du problème  $p_1$  appartient à une période  $t$  et un type  $a$ , via le type de l'équipement. Chaque couple (ressource humaine, période) du problème  $p_2$  appartient lui aussi à une période  $t$  et un type  $a$  via la capacité de la ressource humaine à travailler sur ce type. Ainsi, si une activité est affectée à un couple (équipement, période) du groupe  $[t \times a]$  du problème  $p_1$ , alors elle est obligatoirement affectée à un couple (ressource humaine, période) du même groupe  $[t \times a]$  du problème  $p_2$ , et vice versa. Ceci



peut être illustré par la Figure 2.7.

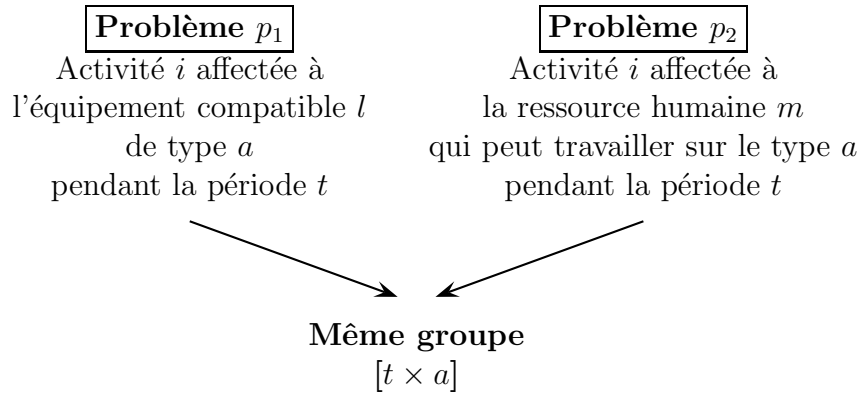


FIGURE 2.7 – Illustration du problème du bin packing interdépendant

La Figure 2.8 donne un exemple de représentation du problème de planification d'activités avec affectation de ressources tenant compte des ressources humaines, sous la forme d'un problème de bin packing interdépendant. Le problème  $p_1$  est l'affectation des activités aux couples (équipement  $\times$  période) comme pour le problème 1 sans les ressources humaines. Le problème  $p_2$  est l'affectation des activités aux couples (ressource humaine  $\times$  période). Un groupe est défini par sa période et son type. Le problème  $p_1$  est composé de deux groupes et trois boîtes par période. Le second groupe est composé de deux boîtes : deux équipements sont de type scanner. Le problème  $p_2$  est composé de ces deux mêmes groupes par période, il y a une boîte par groupe, seule une ressource humaine peut travailler sur un type d'équipement. Une activité affectée au groupe 1 du problème  $p_1$  doit être affectée au groupe 1 du problème  $p_2$ . Par exemple, l'activité 1 est affectée au groupe [lundi matin  $\times$  radio] dans les deux problèmes interdépendants. Plus précisément, elle est affectée le lundi matin à l'équipement  $l_3$  et à la ressource humaine  $m_2$  qui sont tous les deux de type radio.

## 2.6 Conclusion

Ce chapitre a présenté la spécification complète du domaine. À partir d'hypothèses, trois sous-problèmes ont été définis. La formalisation des modèles considérés dans la suite de ce manuscrit a été réalisée.

Ces modèles se focalisent sur le niveau tactique. Mais ils peuvent facilement être utilisés sur les deux autres niveaux stratégique et opérationnel en fonction de la durée de l'horizon temporel considéré ainsi que du découpage en périodes.

Les trois problèmes définis peuvent se ramener à un problème de bin packing. Le problème de planification d'activités avec affectation de ressources est donc vu comme un problème de bin packing avec incompatibilités. Un état de l'art sur le problème du bin packing est rédigé dans le chapitre suivant. Les méthodes que nous proposons sont par la suite testées sur des instances du bin packing avant d'être adaptées à nos problèmes.

À cause de la taille des modèles étudiés, une résolution par méthodes approchées est

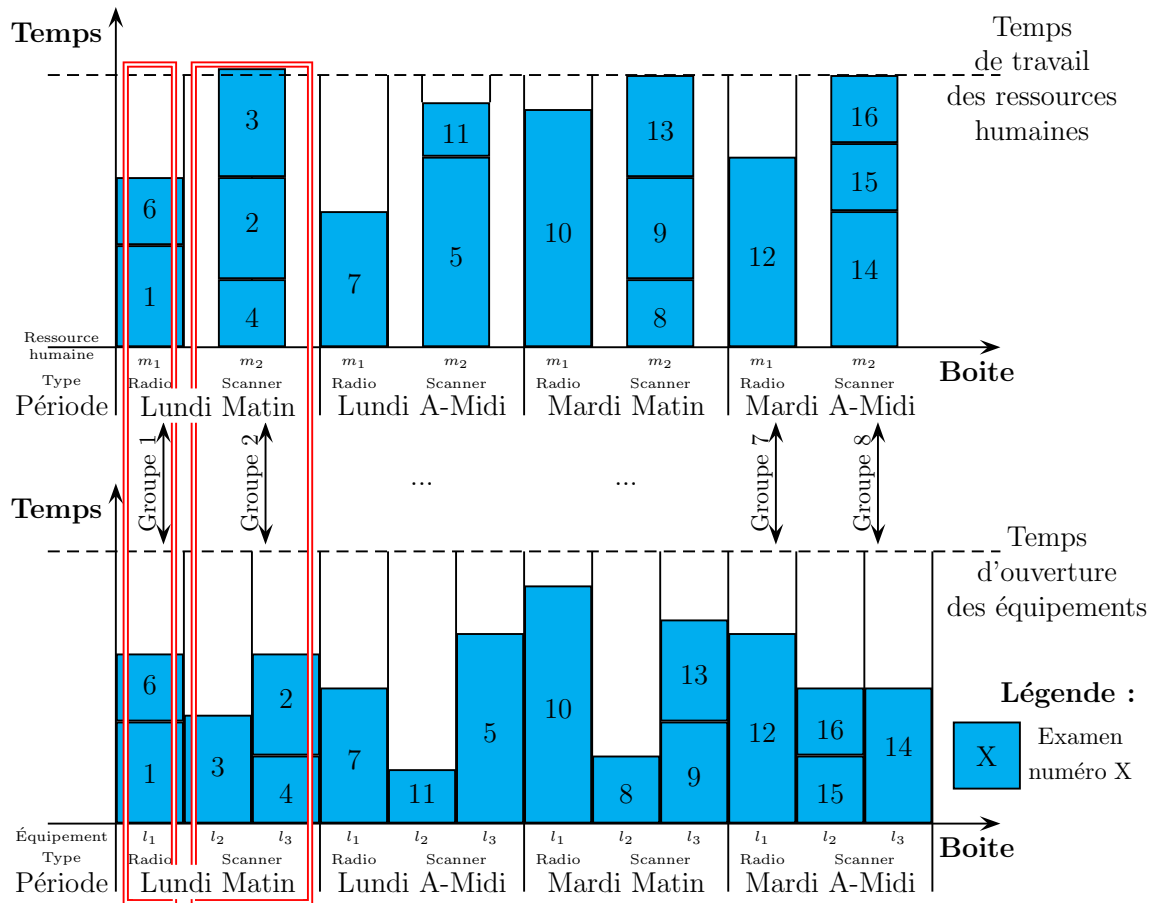


FIGURE 2.8 – Une représentation du problème de la CHT avec les ressources humaines

envisagée. Le chapitre suivant fait également un état de l'art sur les méthodes de résolution approchée qui sont utilisées dans la suite de ce manuscrit.



# État de l'art

## Sommaire

---

<b>3.1</b>	<b>Introduction</b>	<b>64</b>
<b>3.2</b>	<b>Méthodes de résolution pour l'optimisation combinatoire</b>	<b>64</b>
3.2.1	Méthodes exactes pour la résolution des problèmes linéaires	64
3.2.2	Méthodes approchées	66
3.2.2.1	Heuristique gloutonne	67
3.2.2.2	Métaheuristiques basées individu	68
3.2.2.3	Métaheuristiques basées population	72
3.2.3	Approches d'hybridation	78
3.2.3.1	Chainage entre deux méthodes	79
3.2.3.2	Couplage entre deux méthodes	79
<b>3.3</b>	<b>Le problème du bin packing</b>	<b>80</b>
3.3.1	Méthodes exactes	81
3.3.2	Heuristiques	82
3.3.3	Métaheuristiques	88
3.3.3.1	Fonctions objectifs	88
3.3.3.2	Métaheuristiques basées individu	89
3.3.3.3	Métaheuristiques à population	91
3.3.4	Conclusion de l'état de l'art sur le problème du bin packing	94
<b>3.4</b>	<b>Différentes utilisations de la PSO pour la résolution de problèmes d'optimisation combinatoire</b>	<b>95</b>
3.4.1	Le problème des $n$ dames	96
3.4.2	Le problème du voyageur de commerce	96
3.4.3	Le RCPSP	98
3.4.4	Les problèmes d'ordonnancement	99
3.4.5	Le problème du bin packing	101
3.4.6	Conclusion sur la PSO	102
<b>3.5</b>	<b>Conclusion</b>	<b>102</b>

---

## 3.1 Introduction

Les problèmes étudiés tout au long de ce manuscrit ont été formalisés mathématiquement dans le Chapitre 2 sous la forme de problèmes linéaires à variables entières. De nombreuses méthodes de résolution existent pour aborder ce genre de problème. La Section 3.2 présente un état de l'art des méthodes de résolution des problèmes d'optimisation combinatoire. Dans un premier temps, les méthodes exactes sont référencées. Mais à cause de la complexité et la taille des problèmes étudiés, il n'existe pas d'algorithme de résolution en un temps polynomial. C'est pourquoi de nombreuses méthodes approchées existent. Un panel de ces méthodes est détaillé dans la suite de cette section, en différenciant les métaheuristiques basées individus de celles basées population. Plus récemment, d'autres méthodes ont été proposées : les approches d'hybridation couplent plusieurs des méthodes précédemment présentées.

L'analogie entre le problème de planification d'activités avec affectation de ressources et le problème du bin packing a été faite dans le Chapitre 2. Des méthodes exactes, des heuristiques et des métaheuristiques ont été proposées pour la résolution du problème du bin packing, présentées dans la Section 3.3.

La Section 3.4 se concentre sur l'optimisation par essaim particulière. Cette métaheuristique basée population est une méthode récente pleine de promesses. Après avoir été développée pour des problèmes d'optimisation continue, elle a été appliquée à la résolution de nombreux problèmes d'optimisation combinatoire. Des exemples de problèmes d'optimisation combinatoire résolus grâce à l'optimisation par essaim particulière sont présentés dans cette dernière section.

## 3.2 Méthodes de résolution pour l'optimisation combinatoire

Les problèmes réels tels que les problèmes de dimensionnement ou de planification pour les systèmes hospitaliers peuvent être formalisés de façon théorique. L'optimisation combinatoire permet la résolution de ces modèles théoriques ainsi conçus.

Un problème d'optimisation combinatoire est un couple  $(\Omega, H)$  où  $\Omega$  est l'ensemble des solutions du problème et  $H : \Omega \rightarrow \mathbb{R}$  est une fonction coût ou un critère de performance. L'objectif est de trouver une ou plusieurs solutions  $X^*$  qui minimisent la fonction coût :

$$X^* \in \Omega / H(X^*) = \min_{X \in \Omega} H(X)$$

### 3.2.1 Méthodes exactes pour la résolution des problèmes linéaires

L'objectif du problème à atteindre, les contraintes à respecter et les variables de décisions sont spécifiés de façon mathématique. La Programmation Linéaire en Nombres Entiers (PLNE) permet de modéliser mathématiquement le problème dans le cas où les objectifs et les contraintes sont linéaires et que les variables sont toutes entières. La Programmation Linéaire (PL) est dite mixte si certaines variables sont entières et d'autres réelles.

Si les variables ne sont pas toutes entières, on parle de programmation mixte, en anglais Mixed Integer Programming (MIP), si les objectifs et contraintes ne sont pas linéaires. On parle de Mixed Integer Linear Programming (MILP) si le problème est linéaire. Si toutes les variables sont entières, on parle d'Integer Programming (IP) ou d'Integer Linear Programming (ILP) si le problème est linéaire. Nous ne nous intéressons dans la suite qu'aux problèmes linéaires.

Le Tableau 3.1 résume les différents problèmes en fonction de leurs caractéristiques.

	Variables entières	Variables réelles
Problème linéaire	PLNE ou ILP	PL ou MILP
Problème non linéaire	IP	MIP

TABLEAU 3.1 – Définition des problèmes

La formalisation du problème du bin packing donnée par les Équations (2.43) à (2.47) est un exemple de PLNE. La PLNE est difficile alors que la programmation linéaire réelle est polynomiale. L'algorithme du simplexe est le plus utilisé pour sa rapidité moyenne. Cependant, il ne garantit pas l'accès à une solution optimale en un temps polynomial.

Une solution est admissible si elle vérifie toutes les contraintes au problème. Une façon intuitive de résoudre un problème serait l'énumération de toutes les solutions admissibles afin de trouver celle qui optimise la fonction objectif. Mais chaque problème possède une grande combinatoire de solutions admissibles. Des méthodes de résolution exactes ont été développées pour optimiser la résolution de ces problèmes.

**Algorithme du simplexe** Pour résoudre un problème linéaire dont les variables sont réelles, l'algorithme du simplexe a été proposé (Dantzig, 1990). C'est un algorithme itératif permettant de rechercher une solution voisine à partir d'une solution admissible qui améliore la fonction objectif.

Les méthodes suivantes s'appliquent aux programmes linéaires lorsqu'il y a des variables entières :

**Génération de colonnes** La génération de colonnes est une amélioration de l'algorithme du simplexe. Elle vise à générer les variables au fur et à mesure de la résolution du problème (Desaulniers et al., 2005). Elle est basée sur la décomposition de Danzig-Wolfe.

**Algorithme de Benders** À l'inverse, la décomposition de Benders génère les contraintes au fur et à mesure de la résolution du problème (Benders, 1962). C'est une approche par génération de lignes.

**Relaxation linéaire** La relaxation linéaire relâche la contrainte d'intégrité des variables. Ainsi, l'algorithme du simplexe peut notamment être utilisé sur le problème relaxé. Si la solution optimale de la relaxation linéaire est entière, alors c'est aussi une solution optimale pour le PLNE. Si elle n'est pas entière, on a une borne inférieure au problème.

**Branch-and-bound** La méthode de recherche arborescente par séparation et évaluation est basée sur l'algorithme du branch-and-bound, en anglais connu sous l'acronyme B&B. C'est une méthode énumérative intelligente, qui permet grâce à des propriétés du problème, de ne pas parcourir toutes les solutions réalisables pour trouver une solution optimale. Une fonction d'évaluation est utilisée sur la relaxation linéaire, qui permet de dire si la solution partielle est candidate à l'optimalité ou pas. Sinon, toutes les solutions composées de cette solution partielle peuvent être écartées. Pour un PLNE à maximiser, une borne supérieure est donnée par sa relaxation linéaire. Ainsi, dans le cas du problème de maximisation, l'algorithme est le suivant :

- Découper l'ensemble des solutions  $X$ ,
- Sur chaque nœud  $Y \subseteq X$ , calculer une borne supérieure  $B(Y)$  de l'optimum  $H(Y^*)$ ,
- Si  $B(Y)$  est inférieure ou égale à la meilleure solution trouvée, alors on élague  $Y$ ,
- Sinon on découpe successivement  $Y$ .

Il est facile de se ramener à un problème de minimisation à partir d'un problème de maximisation car maximiser  $H$  revient à minimiser son opposé. L'algorithme de branch-and-bound découpe le problème en plusieurs problèmes. (Lawler and Wood, 1966) font un état de l'art sur la méthode du branch-and-bound.

**Cutting planes** La méthode de coupes (cutting planes method) est basée sur l'algorithme de plans sécants (Kelley, 1960). Elle permet d'ajouter des contraintes au problème afin de réduire son ensemble de solutions admissibles. Cette méthode fonctionne à partir de la relaxation linéaire du problème. L'algorithme est le suivant :

- Tant que la solution au problème non relaxé n'a pas été trouvée,
- Résoudre la formulation relaxée, l'optimum est noté  $X^*$ ,
- Si  $X^*$  est entier, alors c'est la solution au problème non relaxé.
- Sinon, trouver une inégalité valide violée par  $X^*$  et l'ajouter au problème relaxé. Une inégalité est valide si elle est respectée par toutes les solutions admissibles discrètes.

L'algorithme de plans sécants affine la description du problème autour de l'optimal.

**Branch-and-cut** La méthode branch-and-cut combine l'algorithme de plans sécants et l'algorithme de branch-and-bound (Padberg and Rinaldi, 1991). Elle consiste à appliquer l'algorithme de plans sécants sur chaque nœud avant de brancher.

**Branch-and-price** La méthode branch-and-price combine l'algorithme du branch-and-bound et la génération de colonnes après relaxation du problème (Johnson, 1989). La génération de colonnes est appliquée à chaque nœud du branch-and-bound.

La Figure 3.1 positionne les différentes méthodes les unes par rapport aux autres.

Des solveurs permettent de résoudre ces problèmes de façon exacte, comme le solveur CPLEX de IBM. Les solveurs modernes utilisent la méthode branch-and-cut.

### 3.2.2 Méthodes approchées

L'utilisation de méthodes exactes est parfois rendue impossible, soit quand le nombre de données est trop grand, soit à cause des extensions qui rendent le problème trop complexe pour le résoudre dans des temps convenables. L'utilisation de méthodes

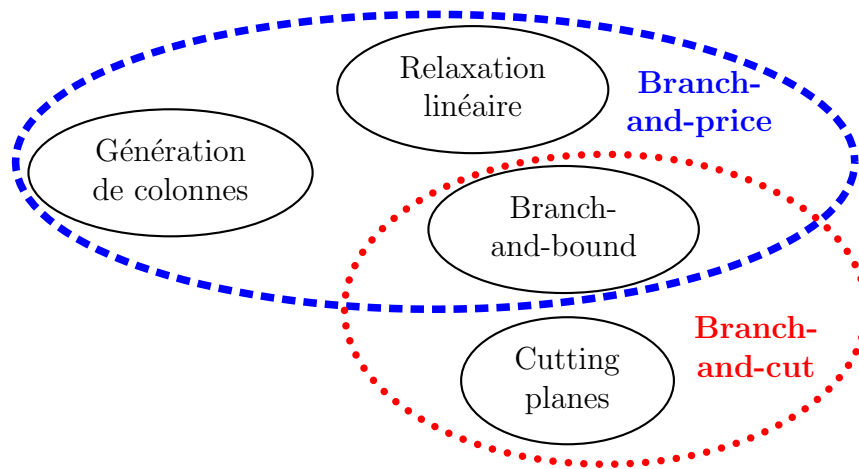


FIGURE 3.1 – Méthodes de résolution exactes

approchées devient alors indispensable pour la résolution du problème. On appelle ces méthodes approchées des heuristiques ou des métaheuristiques. Une heuristique est une méthode spécifique à un problème, alors que les métaheuristiques sont des méthodes génériques.

Deux types de métaheuristiques sont distingués :

- Les méthodes de recherche locale qui consistent à faire évoluer une seule solution à la fois, appelées les métaheuristiques basées individu :
  - La méthode de la descente,
  - Le recuit simulé,
  - L’algorithme du kangourou,
  - La recherche à voisinage variable,
  - La recherche tabou.
- Les algorithmes évolutionnaires qui consistent à faire évoluer un ensemble de solutions simultanément, appelées les métaheuristiques basées population :
  - L’algorithme génétique,
  - L’optimisation par colonie de fourmis,
  - L’optimisation par essaim particulaire.

### 3.2.2.1 Heuristique gloutonne

Une heuristique est dite gloutonne si une fois qu’elle a pris une décision, elle ne revient jamais dessus. L’algorithme glouton est également appelé greedy algorithm. C’est une méthode très rapide et qui peut parfois donner de bons résultats.

À titre d’exemple, les méthodes Next Fit, First Fit et Best Fit développées par (Johnson, 1973), qui sont des heuristiques de résolution du problème du bin packing, sont des heuristique gloutonnes.



### 3.2.2.2 Métaheuristiques basées individu

Pour traiter un problème d'optimisation avec une méthode de recherche locale, il faut définir au préalable un ou plusieurs systèmes de voisinage. L'application du système de voisinage sur la solution courante définit une nouvelle solution candidate, appelée également voisin. Ce procédé permet de parcourir l'ensemble des solutions. La solution courante et la solution candidate peuvent être comparées au regard de la fonction coût ou du critère de performance.

Les notations utilisées sont les suivantes :

- $X$  est la solution courante,
- $RX$  est la solution record,
- $X'$  est la solution candidate (ou voisin),
- $V, W$  sont deux systèmes de voisinage,
- $H$  est la fonction coût (ou critère de performance).

Le critère d'arrêt peut être un nombre maximal d'itérations, un temps de calcul maximal ou l'égalité entre la fonction coût courante et une valeur à atteindre.

#### La méthode de la descente

La méthode de la descente est une des méthodes de recherche locale les plus anciennes. Son principe consiste à générer une solution  $X'$  voisine d'une solution courante  $X$  selon le système de voisinage  $V$ . Si la fonction objectif  $H(X')$  est inférieure ou égale à  $H(X)$  alors la solution  $X'$  est acceptée. La méthode de la descente converge vers un minimum local. L'avantage de cette méthode réside dans sa simplicité et sa rapidité. Mais les résultats obtenus sont souvent de qualité médiocre.

Deux types de descente existent, ils diffèrent quant à leur stratégie d'exploration, c'est-à-dire le choix de  $X'$  dans  $V(X)$ .

**La descente déterministe :** le mécanisme d'exploration est déterministe. Le voisin  $X'$  est choisi dans  $V(X)$  tel que  $H(X') < H(X)$ . Plusieurs stratégies de choix sont possibles : choix du meilleur voisin ou choix du premier voisin qui améliore la solution courante par exemple. L'Algorithme 1 décrit l'algorithme de principe de la descente déterministe.

---

#### Algorithme 1 : Algorithme de principe de la descente déterministe

---

**Données :** Solution initiale  $X$

```

1 tant que  $\exists Z \in V(X) / H(Z) < H(X)$  faire
2   | Soit un voisin  $X' \in V(X)$  tel que  $H(X') < H(X)$ 
3   |  $X := X'$ 

```

---

**La descente stochastique :** le mécanisme d'exploration est stochastique. Le voisin  $X'$  est choisi uniformément et aléatoirement dans  $V(X)$ . L'Algorithme 2 décrit l'algorithme de principe de la descente stochastique.

**Algorithme 2** : Algorithme de principe de la descente stochastique

**Données** : Solution initiale  $X$

```

1 tant que Critère d'arrêt non atteint faire
2   Choisir uniformément et aléatoirement  $X' \in V(X)$ 
3   si  $H(X') \leq H(X)$  alors
4      $X := X'$ 

```

**Algorithme du recuit simulé**

Le recuit simulé est une méthode inspirée d'un processus utilisé en métallurgie qui consiste à alterner les cycles de refroidissement lent et de réchauffage. En particulier, le recuit simulé inhomogène a été utilisé par (Metropolis et al., 1953) pour simuler le refroidissement physique en métallurgie. Appliqué au domaine de l'optimisation, cela revient à exécuter une descente avec une probabilité non nulle de choisir une solution moins bonne que la solution courante. Cette probabilité d'accepter une solution défavorable diminue au fur et à mesure des itérations.

Le recuit simulé converge en probabilité vers l'ensemble des solutions optimales si le système de voisinage  $V$  satisfait les propriétés d'accessibilité et de réversibilité (Aarts and van Laarhoven, 1987). L'Algorithme 3 décrit l'algorithme de principe du recuit simulé.

**Algorithme 3** : Algorithme de principe du recuit simulé inhomogène

**Données** : température  $T_0$ , solution initiale  $X$ , facteur de décroissance  $\alpha$

```

1  $T := T_0$ 
2 tant que Critère d'arrêt non atteint faire
3   Choisir uniformément et aléatoirement  $X' \in V(X)$ 
4   si  $\text{rand}[0, 1] \leq e^{-\frac{H(X')-H(X)}{T}}$  alors
5      $X := X'$ 
6   Générer une nouvelle température  $T := \alpha \times T$ 

```

Deux paramètres doivent être choisis :

- La température initiale  $T_0$  est choisie telle que toutes les transitions sont acceptées au début, c'est-à-dire  $e^{-\frac{H(X')-H(X)}{T_0}} \simeq 1, \forall(X, X')$ .
- Le facteur de décroissance  $\alpha$  est choisi de telle façon que la température finale est proche de zéro.

**Algorithme du kangourou**

Une simple descente reste dans le premier minimum local trouvé. La recherche locale itérée permet de sortir de ce minimum local (Lourenço et al., 2002). Après application de la recherche locale, une perturbation est appliquée à la solution courante afin de sortir du minimum local. La recherche locale est ensuite appliquée à partir de la solution perturbée.

L'algorithme du kangourou est une recherche locale itérée. La recherche locale utilisée est une descente stochastique. S'il n'y a pas d'amélioration de la solution après un certain nombre  $A$  d'itérations, un saut est fait. Pour faire ce saut, une solution est choisie dans un système de voisinage  $W$  différent du système  $V$ .

L'algorithme du kangourou converge en probabilité vers l'ensemble des solutions optimales si et seulement si le système de voisinage  $W$  satisfait la propriété d'accessibilité (Fleury, 1993). L'Algorithme 4 décrit l'algorithme du kangourou.

---

**Algorithme 4** : Algorithme de principe de l'algorithme du kangourou

---

**Données** : Nombre d'itérations  $A$  avant un saut, solution initiale  $X$

```

1  $c := 0, RX := X$ 
2 tant que Critère d'arrêt non atteint faire
3   si  $c < A$  alors
4     Choisir uniformément et aléatoirement  $X' \in V(X)$ 
5     si  $H(X') \leq H(X)$  alors
6       si  $H(X') < H(X)$  alors
7          $c := 0$ 
8         si  $H(X') < H(RX)$  alors
9            $RX := X'$ 
10         $X := X'$ 
11   sinon
12     Choisir uniformément et aléatoirement  $X' \in W(X)$ 
13     si  $H(X') \neq H(X)$  alors
14       si  $H(X') < H(RX)$  alors
15          $RX := X'$ 
16        $c := 0$ 
17      $X := X'$ 
18    $c := c + 1$ 

```

---

Deux réglages doivent être effectués :

- Le choix du système de voisinage  $W$  : il consiste souvent en l'application un certain nombre de fois du système de voisinage  $V$ . Il doit vérifier  $\text{card}(W) > \text{card}(V)$ .
- Le nombre d'itérations  $A$  avant un saut : (Fleury, 1993) définit  $A$  tel que  $A \geq \text{card}(V) \cdot \ln(2)$ .

### Recherche à voisinage variable

Le principe de la recherche à voisinage variable ou Variable Neighborhood Search (VNS) repose sur plusieurs systèmes de voisinage (Hansen and Mladenović, 2001). En effet, un minimum local par rapport à un système de voisinage n'en est pas nécessairement un par rapport à un autre système de voisinage et un minimum global est un minimum global par rapport à tous les voisinages possibles. Soient  $K$  systèmes de voisinage  $V_k$  et une recherche locale  $V$ . À partir de la solution courante, une solution voisine est générée

dans le système de voisinage  $V_1$ . La recherche locale  $V$  est effectuée à partir de cette solution. Si la recherche locale n'apporte pas d'amélioration de la solution courante, le système de voisinage suivant est utilisé, jusqu'à avoir utilisé tous les systèmes de voisinage. L'algorithme de principe de la recherche à voisinage variable est donné dans l'Algorithme 5.

---

**Algorithme 5** : Algorithme de principe de la recherche à voisinage variable

---

**Données** : solution initiale  $X$

```

1  $k := 1, RX := X$ 
2 tant que  $k \leq K$  faire
3   Choisir uniformément et aléatoirement  $X' \in V_k(X)$ 
4   tant que Critère d'arrêt non atteint faire
5     Choisir uniformément et aléatoirement  $X'' \in V(X')$ 
6     si  $H(X'') \leq H(X')$  alors
7        $X' := X''$ 
8   si  $H(X') < H(X)$  alors
9      $X := X'$ 
10     $k := 1$ 
11    si  $H(X') < H(RX)$  alors
12       $RX := X'$ 
13  sinon
14     $k := k + 1$ 

```

---

## Recherche tabou

La recherche tabou a été développée par (Glover, 1986) et indépendamment par (Hansen, 1986). Cette méthode fait appel à un ensemble de règles et de mécanismes pour guider la recherche au travers de l'espace des solutions de manière à éviter la convergence vers un optimum local.

La recherche tabou examine un échantillon de solutions dans  $V(X)$  et retient la meilleure solution  $X'$  au sens de la fonction coût même si  $X'$  est moins bonne que  $X$ . La recherche tabou ne s'arrête donc pas au premier optimum local trouvé. Cependant, cette stratégie peut entraîner des cycles. Par exemple,  $X \rightarrow X' \rightarrow X \rightarrow X' \dots$  est un cycle de longueur 2. Pour les éviter, on mémorise les  $k$  dernières solutions visitées dans une mémoire à court terme de type FIFO et on interdit tout mouvement qui conduit à une de ces solutions. Cette mémoire est appelée liste tabou. La valeur  $k$  détermine la longueur de cette mémoire et permet d'éviter tous les cycles de longueur inférieure ou égale à  $k$ . Cette valeur dépend du problème à résoudre et peut éventuellement évoluer au cours de la recherche.

L'Algorithme 6 décrit le fonctionnement de la recherche tabou.

---

**Algorithme 6** : Algorithme de principe de la recherche tabou

---

**Données** : solution initiale  $X$

```

1  $RX := X$ 
2 tant que Critère d'arrêt non atteint faire
3   Choisir la meilleure solution  $X'$  non taboue parmi les solutions candidates
4   si  $H(X') < H(RX)$  alors
5      $RX := X'$ 
6   Mettre à jour la liste des mouvements tabous
7    $X := X'$ 

```

---

### 3.2.2.3 Métaheuristiques basées population

La principale caractéristique des métaheuristiques précédemment citées est qu'elles ne possèdent qu'une solution courante à un moment donné, c'est-à-dire un individu. Le principe des métaheuristiques à population est de générer plusieurs solutions courantes et de faire en sorte qu'elles s'influencent entre elles afin de tendre vers une solution optimale du problème.

Les métaheuristiques à population sont des algorithmes évolutionnistes qui sont basés sur le principe du processus de comportement naturel des espèces. Ils doivent leur nom à l'analogie entre leur déroulement et le mécanisme de sélection naturelle et de croisement des individus d'une population à l'autre.

#### Algorithme génétique

Les algorithmes génétiques ont été introduits par (Holland, 1962) puis (Goldberg et al., 1989). L'algorithme de base s'appuie sur un codage binaire. Un individu est un chromosome, un gène est la composante de base du chromosome et une allèle est la valeur effective d'un gène (0 ou 1). L'algorithme s'appuie sur un ensemble d'opérateurs génétiques, qui opèrent stochastiquement sur un chromosome sans aucune connaissance du problème. Les principaux opérateurs sont les suivants :

- La sélection consiste à choisir les individus suivant la valeur de leur fonction objectif.
- Le croisement permet, grâce au croisement entre deux individus appelés parents, de générer une ou plusieurs nouvelles solutions appelées enfants. Par exemple, le croisement bi-points consiste à choisir aléatoirement deux points de croisement et à échanger les segments des deux parents déterminés par ces deux points.
- La mutation consiste à changer aléatoirement la valeur de certains gènes. Elle réintroduit de nouvelles valeurs pour des gènes alors que le croisement réalise uniquement des recombinaisons de valeurs existantes.

L'Algorithme 7 décrit le principe général de l'algorithme génétique.

#### Optimisation par colonie de fourmis

L'optimisation par colonie de fourmis s'inspire du comportement des fourmis recherchant un chemin entre leur colonie et une source de nourriture. Les fourmis sèment une piste de phéromones derrière leur passage. Les autres fourmis peuvent sentir cette

---

**Algorithme 7 :** Algorithme de principe des algorithmes génétiques

---

**Données :**  $P_0$  une population initiale de  $P$  individus

- 1  $i := 0$
- 2 **tant que** *Critère d'arrêt non atteint* **faire**
- 3      $i := i + 1$
- 4     Soit  $P_i$  une population vide
- 5     **tant que**  $P_i$  *contient moins de*  $P$  *individus* **faire**
- 6         Choisir aléatoirement  $I_1$  et  $I_2$  deux individus de  $P_{i-1}$
- 7         Appliquer l'opérateur de croisement sur  $I_1$  et  $I_2$  pour obtenir les enfants  $O_1$  et  $O_2$
- 8         Ajouter  $O_1$  et  $O_2$  à  $P_i$
- 9     Appliquer l'opérateur de mutation sur certains individus de  $P_i$
- 10      $P_i := P_i \cup P_{i-1}$
- 11     Sélectionner  $P$  individus de  $P_i$  et supprimer les autres

---

phéromone et la suivre. Quand une colonie de fourmis est présente devant deux chemins possibles, chaque fourmi en choisit initialement un aléatoirement, résultant en une probabilité de 50% pour chacun des chemins. Étant donné que les fourmis font des aller-retour entre leur colonie et la source de nourriture en utilisant la même piste à l'aller qu'au retour, les fourmis utilisant le plus court chemin seront revenues le plus rapidement. Donc, immédiatement après leur retour il y a plus de phéromones sur le plus court chemin, influençant les autres fourmis à suivre ce chemin. Ainsi, toute la colonie suit le plus court chemin à partir d'un instant donné. (Dorigo, 1992) propose une méthode pour l'optimisation combinatoire qui utilise ce principe.

La méthode est illustrée par le problème de voyageur de commerce d'après une application faite par (Dorigo et al., 1996). L'Algorithme 8 décrit son algorithme de principe. L'espace des solutions est représenté sous la forme d'un graphe composé de  $m$  sommets et d'arêtes. Les sommets sont les villes et les arêtes les chemins entre les villes. Les fourmis doivent parcourir toutes les villes une et une seule fois. Elles déposent des pistes de phéromone dont la concentration varie en fonction de la distance qu'elles ont parcourue.

La colonie est composée de  $K$  fourmis. Chaque fourmi  $k$  explore le graphe en fonction des informations qu'elle rencontre, c'est-à-dire les phéromones. Chaque fourmi peut être vue comme une méthode de construction, capable de générer une solution pour un problème donné. À chaque étape de la construction, chaque fourmi doit décider de la marche à suivre pour compléter la solution partielle. Une telle décision est basée sur les phéromones qui guident les fourmis vers les choix qui ont donné de bons résultats dans les constructions précédentes. Soit  $\tau_{ij}(t)$  l'intensité de phéromones sur l'arête  $(i, j)$  à l'itération  $t$ . À l'itération  $t$ , chaque fourmi doit choisir le sommet où elle sera à l'instant  $t + 1$ . Un déplacement vers un sommet est effectué selon une probabilité qui dépend de l'intensité de phéromones. Plus l'intensité sur une arête est élevée, plus la probabilité d'emprunter l'arête est forte. La règle de déplacement est régie par l'Équation (3.1).

$$p_{ij}^k(t) = \begin{cases} \frac{(\tau_{ij}(t))^\alpha \cdot (\eta_{ij})^\beta}{\sum_{l \in J_i^k} (\tau_{il}(t))^\alpha \cdot (\eta_{il})^\beta} & \text{si } j \in J_i^k \\ 0 & \text{si } j \notin J_i^k \end{cases} \quad (3.1)$$

avec  $J_i^k$  la liste des déplacements possibles pour la fourmi  $k$ ,  $\alpha$  et  $\beta$  deux paramètres contrôlant l'importance relative de l'intensité de la piste  $\tau_{ij}$  et de l'intensité de la visibilité  $\eta_{ij}$ .  $\eta_{ij}$  est calculé comme étant l'inverse de la distance entre les villes  $i$  et  $j$ .

À chaque itération de l'algorithme,  $m-1$  déplacements sont effectués par les  $K$  fourmis.

Une fois que chaque fourmi a construit une solution, l'intensité des phéromones est mise à jour. L'intensité sur l'arête  $(i, j)$  est modifiée selon l'Équation (3.2).

$$\tau_{ij}(t+1) = \rho \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (3.2)$$

où  $\rho$  est un coefficient d'évaporation et  $\Delta\tau_{ij}(t)$  représente l'intensité de phéromones déposée par toutes les fourmis qui sont passées par l'arête  $(i, j)$  à l'itération  $t$ . Un choix conduisant à une bonne solution entraîne une augmentation du taux de phéromone correspondant, alors qu'un mauvais choix entraîne une diminution. La meilleure solution obtenue est une conséquence de l'interaction subtile entre renforcement et évaporation de la phéromone, qui fait que seules les meilleures liaisons subsistent.

---

**Algorithme 8** : Algorithme de principe de l'optimisation par colonie de fourmis

---

```

1 Initialiser les phéromones et choisir un nombre  $K$  de fourmis
2  $t := 0$ 
3 tant que Critère d'arrêt non atteint faire
4    $t := t + 1$ 
5   pour toutes les Fourmis  $k$  de 1 à  $K$  faire
6     Choisir uniformément et aléatoirement une ville
7     pour toutes les Villes non visitées  $i$  faire
8       Choisir une ville  $j$  parmi les villes restantes, selon l'Équation (3.1)
9       Déposer l'intensité de phéromones  $\Delta\tau_{ij}(t)$  sur chaque arête  $(i, j)$ 
10  Évaporer les pistes selon l'Équation (3.2)

```

---

### Optimisation par Essaim Particulaire

L'Optimisation par Essaim Particulaire (OEP), ou Particle Swarm Optimization (PSO), est une métaheuristique à population qui a été introduite par James Kennedy, un psychologue social, et Russel Eberhart, un ingénieur électricien, en 1995 (Eberhart and Kennedy, 1995). La PSO est originaire de la simulation du comportement des vols d'oiseaux (Reynolds, 1987). L'observation montre qu'au début, chaque oiseau vole sans destination particulière. Tous les oiseaux vont finalement spontanément se regrouper en un essaim jusqu'à ce qu'un des oiseaux se pose sur le perchoir. De la même façon, en optimisation, les oiseaux sont des particules allant vers un objectif à optimiser. Les particules parcourent l'ensemble des solutions pour trouver la meilleure solution, en fonction de leurs propres découvertes et des expériences passées de l'essaim.

Cette métaheuristique a d'abord été utilisée en optimisation continue avant d'être adaptée en optimisation combinatoire.

**En optimisation continue** L'essaim est un ensemble de  $P$  particules.

À l'instant  $t$ , chaque particule  $p \in \{1, P\}$  a une position  $X_{p,t}$  et une vitesse  $V_{p,t}$ . Chaque particule connaît sa meilleure position  $P_{p,t}$  et la valeur de sa fonction objectif. Chaque particule connaît la meilleure position précédente de l'essaim  $G_t$ , la meilleure position atteinte par une particule depuis l'instant initial, et la valeur de sa fonction objectif.

Chaque particule se déplace de sa position courante  $X_{p,t}$  à l'instant  $t$  à sa nouvelle position  $X_{p,t+1}$  à l'instant  $t + 1$  par l'application de sa vitesse  $V_{p,t}$  à sa position courante.

À l'instant  $t + 1$ , une particule est influencée par la combinaison entre les trois comportements suivants, pondérés par des coefficients socio-cognitifs :

- suivre son propre chemin,
- aller vers sa meilleure position précédente,
- aller vers la meilleure position précédente de l'essaim.

La Figure 3.2 illustre le comportement d'une particule avec ses différentes influences.

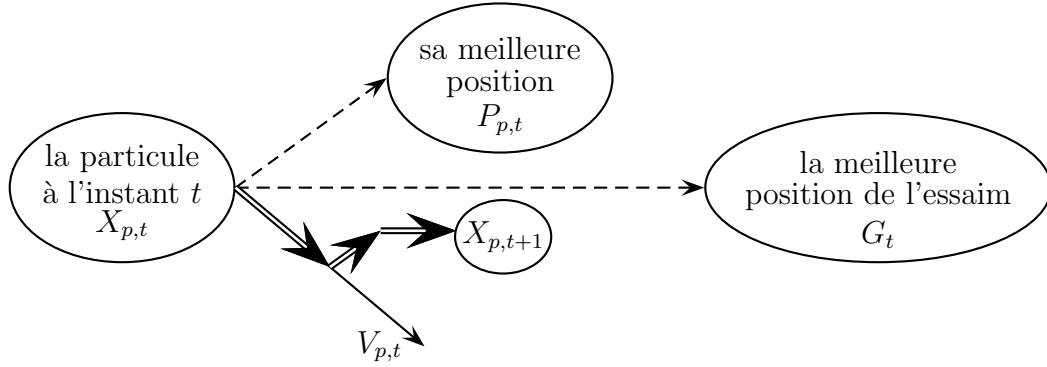


FIGURE 3.2 – Comportement d'une particule

Cette combinaison est modélisée par le Système d'Équations (3.3).

$$\begin{cases} X_{p,t+1} &= X_{p,t} + V_{p,t+1} \\ V_{p,t+1} &= \underbrace{c_1 \cdot V_{p,t}}_{\text{inertie}} + \underbrace{c_2 \cdot r_2 \cdot (P_{p,t} - X_{p,t})}_{\text{influence personnelle}} + \underbrace{c_3 \cdot r_3 \cdot (G_t - X_{p,t})}_{\text{influence de la meilleure position de l'essaim}} \end{cases} \quad (3.3)$$

Avec :

- $p$  la particule,
- $V_{p,t}$  la vitesse de la particule  $p$  au temps  $t$ ,
- $X_{p,t}$  la position de la particule  $p$  au temps  $t$ ,
- $P_{p,t}$  la meilleure position de la particule  $p$  connue au temps  $t$ ,
- $G_t$  la meilleure position de l'essaim connue au temps  $t$ ,
- $c_1, c_2, c_3 \in [0, 2[$  les coefficients socio-cognitifs,
- $r_2, r_3$  des nombres générés uniformément et aléatoirement  $\in [0, 1[$ .

Les opérateurs peuvent être interprétés de la façon suivante :

- $P_{p,t} - X_{p,t}$  : différence entre deux positions, qui caractérise une vitesse pour passer d'une position à l'autre,



- $c_1.V_{p,t}$  : produit entre un coefficient et une vitesse, ce qui correspond à la portion de la vitesse à conserver,
- $V_{p,t}^{(1)} + V_{p,t}^{(2)}$  : somme entre des vitesses, la vitesse résultat est une composition des vitesses sommées,
- $X_{p,t} + V_{p,t+1}$  : somme entre une position et une vitesse, c'est-à-dire l'application de la vitesse à la position.

Au lieu d'être influencée par la meilleure position de l'essaim (position globale), la particule peut être influencée par la meilleure particule appartenant à un voisinage. Dans ce cas, la PSO est définie comme étant locale (Kennedy, 2006). Il existe plusieurs moyens de définir ce voisinage (Kennedy, 1999), mais deux types sont distingués :

- le voisinage physique, qui tient compte des distances. En pratique, les distances sont recalculées à chaque étape, ce qui est assez coûteux.
- le voisinage social, qui tient seulement compte des relations. En pratique, pour chaque particule, son voisinage est défini par une liste de particules au tout début, et ne change pas.

Il faut noter que quand le processus converge, un voisinage social devient un voisinage physique.

Dans la suite, nous identifions aussi bien la troisième influence comme étant l'influence globale ou une influence locale. Nous verrons à travers les exemples présentés dans la Section 3.4. que chacune des deux influences a autant été utilisée même s'il s'avère que l'influence locale était plus utilisée au début du développement de la PSO. D'après (Eberhart et al., 1996), l'influence globale converge plus rapidement que l'influence locale, mais l'influence locale est moins susceptible de tomber dans un minimum local.

L'algorithme de principe de la PSO est donné par l'Algorithme 9. Les positions et vitesses initiales sont générées aléatoirement. Les particules sont comparées via la valeur de la fonction objectif.

La PSO a notamment été utilisée pour la minimisation de la fonction de Rosenbrock. (Kemmoé Tchomté and Gourgand, 2009) étudient les effets de la taille de l'essaim, ainsi que la valeur des différents coefficients sur la fonction objectif en appliquant la PSO aux fonctions suivantes : Sphere, Rosenbrock, Rastrigin, Griewank et Schaffer. D'autres fonctions ont été résolues par (Shahzad et al., 2009) : Weighted Sphere, Quartic, Schwefel, Ackley. (Yan et al., 2012) résolvent les fonctions Shubert, Hansen et Camel.

**En optimisation combinatoire** Pour utiliser la PSO en optimisation combinatoire, il faut définir les opérateurs suivants :

- $X_{p,t} \bar{\oplus} V_{p,t+1}$  : addition entre une vitesse et une position,
- $c_1 \otimes V_{p,t}$  : multiplication entre un coefficient et une vitesse,
- $P_{p,t} \ominus X_{p,t}$  : différence entre deux positions,
- $V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$  : addition entre deux vitesses.

Le passage de  $t$  à  $t + 1$  est donc défini par le Système d'Équations (3.4).

$$\begin{cases} X_{p,t+1} &= X_{p,t} \bar{\oplus} V_{p,t+1} \\ V_{p,t+1} &= c_1 \otimes V_{p,t} \oplus c_2.r_2 \otimes (P_{p,t} \ominus X_{p,t}) \oplus c_3.r_3 \otimes (G_t \ominus X_{p,t}) \end{cases} \quad (3.4)$$

---

**Algorithme 9** : Algorithme de principe de la PSO en optimisation continue
 

---

```

1 Paramétrage des coefficients de la PSO, taille de l'essaim
2 Initialisation aléatoire des positions et vitesses des particules
3 Détermination de la meilleure position de chaque particule
4 Détermination de la meilleure position de l'essaim
5 tant que Critère d'arrêt non atteint faire
6   pour toutes les Particules faire
7     Mise à jour de la vitesse et de la position en appliquant le Système
     d'Équations (3.3)
8     Détermination de la meilleure position de la particule :
9     si  $H(X_{p,t+1}) \leq H(P_{p,t})$  alors
10    |    $P_{p,t+1} := X_{p,t+1}$ 
11   Détermination de la meilleure particule de l'essaim :
12   pour toutes les Particules faire
13   |   si  $H(P_{p,t+1}) \leq H(G_t)$  alors
14   |   |    $G_t := P_{p,t+1}$ 
    
```

---

Le Tableau 3.2 fait un comparatif des éléments à définir en optimisation continue et en optimisation combinatoire.

Élément ou opérateur	Optimisation continue	Optimisation combinatoire
Position	$X_{p,t}, P_{p,t}, G_t$	$X_{p,t}, P_{p,t}, G_t$
Vitesse	$V_{p,t}$	$V_{p,t}$
Addition vitesse position	$X_{p,t} + V_{p,t+1}$	$X_{p,t} \oplus V_{p,t+1}$
Multiplication coefficient vitesse	$c_1 \cdot V_{p,t}$	$c_1 \otimes V_{p,t}$
Différence position position	$P_{p,t} - X_{p,t}$	$P_{p,t} \ominus X_{p,t}$
Addition vitesse vitesse	$V_{p,t}^{(1)} + V_{p,t}^{(2)}$	$V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$

TABLEAU 3.2 – Comparatif éléments PSO

Une version binaire de la PSO discrète a été définie et utilisée avec succès par (Kennedy and Eberhart, 1997). Dans leur adaptation binaire, les auteurs définissent la position  $X_{p,t}$  de la particule  $p$  à l'instant  $t$  comme étant une suite de 0 ou de 1. La vitesse  $V_{p,t}$  de la particule  $p$  à l'instant  $t$  est le nombre de bits changés à chaque itération. Ainsi, une particule avec zéro bit changé ne bouge pas tandis qu'elle se déplace le "plus loin" si tous ses bits sont inversés. Les auteurs décident que pour une fonction binaire, la vitesse est traduite en terme de probabilités. Pour un bit donné, la différence entre deux positions  $P_{p,t} \ominus X_{p,t}$  est égale à 0, 1 ou -1. Le calcul de  $V_{p,t+1}$  est régi par la deuxième équation du Système d'Équations (3.4). Toujours pour un bit donné, le déplacement de la particule est ensuite régi par l'Équation (3.5), avec  $\Psi(V_{p,t+1})$  une transformation qui contraint  $V_{p,t+1}$  à

l'intervalle  $[0, 1]$  et qui peut donc être considérée comme une probabilité.

$$X_{p,t+1} = \begin{cases} 1 & \text{si } rand < \Psi(V_{p,t+1}), \\ 0 & \text{sinon} \end{cases} \quad (3.5)$$

Les auteurs testent leur version de la PSO sur cinq fonctions tests. L'essaim est composé de vingt particules.

(Clerc, 2004) s'inspire des travaux de (Kennedy and Eberhart, 1997) pour définir le cadre d'utilisation de la PSO en optimisation combinatoire. Chacun des points suivants doit être défini :

- Un espace de recherche des positions :  $\Omega$ . La Figure 3.3 décrit l'ensemble  $\Omega$  des positions possibles, un essaim de six particules à l'instant  $t$  ( $X_{p,t}, p = 1, \dots, 6$ ) et ce même essaim à l'instant  $t + 1$  ( $X_{p,t+1}, p = 1, \dots, 6$ ).

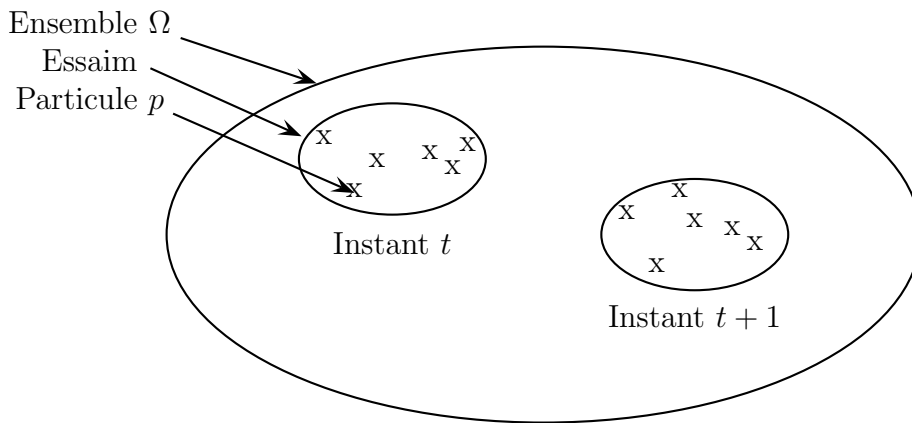


FIGURE 3.3 – Ensemble  $\Omega$  des particules  $p$

- Une fonction coût,  $H$  sur  $\Omega$ , qui à  $X_{p,t}$  dans  $\Omega$  lui associe une valeur dans  $C = \{c\}$ .  

$$H : \Omega \rightarrow C$$

$$X_{p,t} \mapsto H(X_{p,t}) = c$$
- Un ordre sur  $C$ , ou plus généralement un ordre partiel, tel que pour chaque paire d'éléments  $(c_i, c_j) \in C$  on ait  $c_i < c_j$  ou  $c_i \geq c_j$ .
- Si l'on veut utiliser un voisinage physique, il faut définir une distance  $d$  dans l'espace de recherche.

### 3.2.3 Approches d'hybridation

L'hybridation consiste à combiner deux méthodes  $A$  et  $B$  pour résoudre un problème d'optimisation. Une méthode peut être une méthode de résolution exacte ou approchée (heuristique ou métaheuristique) ou un modèle d'évaluation des performances.

(Talbi, 2002) présente une taxonomie des métaheurstiques hybrides. Cette taxonomie est reprise par (Raidl, 2006) qui propose une classification complète des métaheurstiques hybrides. Les deux auteurs différencient les méthodes hybrides dites de chaînage de celles de couplage.

(Puchinger and Raidl, 2005) font un état de l'art sur l'hybridation entre les méthodes exactes et les métaheuristiques. L'objectif de ces méthodes est de diminuer les temps de calcul ou d'améliorer les solutions trouvées.

Les principales approches d'hybridation sont les suivantes (Deroussi et al., 2014).

### 3.2.3.1 Chainage entre deux méthodes

Le chainage entre les méthodes  $A$  et  $B$  est illustré par la Figure 3.4. La méthode  $A$  sert à générer une solution initiale pour la méthode  $B$ . La méthode  $A$  peut être une heuristique de construction, la méthode  $B$  peut être une métaheuristique basée individu.

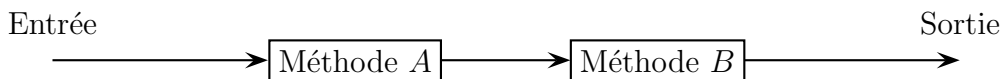


FIGURE 3.4 – Chainage entre deux méthodes

Pour le problème d'équilibrage de lignes d'assemblages, (Norre, 2005) compare plusieurs chainages d'heuristiques et plusieurs chainages d'heuristiques et de métaheuristiques. Deux types d'heuristique ont été proposés : les heuristiques basées sur les priorités et les heuristiques basées sur la charge. Les deux ordres de chainage sont testés. Les chainages (heuristiques basées sur la charge + heuristiques basées sur les priorités) sont d'assez bonne qualité. Les heuristiques sont également utilisées pour construire la ou les solutions initiales des métaheuristiques. L'une des métaheuristiques utilisées est le recuit simulé.

### 3.2.3.2 Couplage entre deux méthodes

Le couplage entre deux méthodes peut être séquentiel ou hiérarchique.

**Approche séquentielle :** Les méthodes  $A$  et  $B$  collaborent pour résoudre un même problème. Cette approche est illustrée par la Figure 3.5. Les méthodes  $A$  et  $B$  sont utilisées de manière séquentielle et itérative. Le résultat de la méthode  $B$  est réinjecté en entrée de la méthode  $A$ , ce qui permet d'itérer le processus de résolution. Par exemple, la méthode  $A$  est une métaheuristique et la méthode  $B$  une méthode de recherche locale. La métaheuristique génère une solution. Cette solution devient l'entrée de la méthode de recherche locale. Après application de la recherche locale, la solution améliorée devient la nouvelle entrée de la métaheuristique. Ce processus est itéré un certain nombre de fois.

Cette approche est aussi qualifiée d'approche haut-niveau (Talbi, 2002) : le fonctionnement interne des méthodes n'est pas impacté, chaque méthode agit indépendamment. Il n'y a pas de relation directe dans les fonctionnements internes des méthodes.

(Scholl, 1999) propose un couplage séquentiel pour la résolution du problème d'équilibrage de ligne d'assemblage. Sa méthode est basée sur un couplage entre la méthode tabou et des méthodes de séparation/évaluation tronquées. Ses résultats servent de références aux travaux de (Norre, 2005) qui propose d'autres hybridations de méthodes.

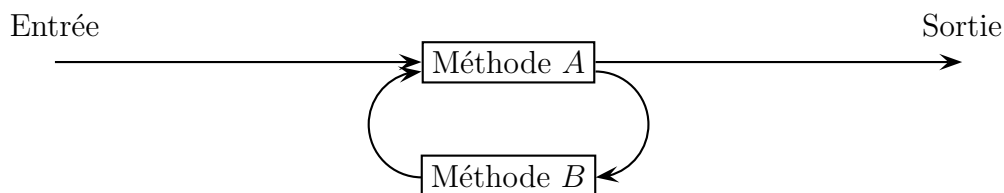


FIGURE 3.5 – Couplage séquentiel entre deux méthodes

**Approche hiérarchique :** On parle d'approche hiérarchique si la méthode  $A$  est considérée comme maître et la méthode  $B$  comme esclave (Deroussi et al., 2014). Cette approche est illustrée par la Figure 3.6. Une fonction de la méthode maître est remplacée par la méthode esclave. C'est le cas si la méthode  $A$  est une métaheuristique et la méthode  $B$  un modèle d'évaluation des performances. Cette approche est utile lorsque les critères d'évaluation des performances sont difficiles à calculer. Cette approche est également qualifiée d'approche bas-niveau.

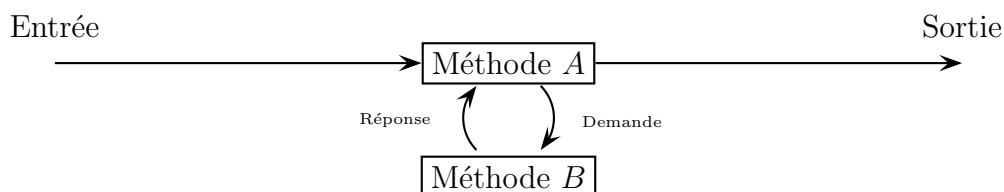


FIGURE 3.6 – Couplage hiérarchique entre deux méthodes

(Norre, 2005) compare des couplages heuristiques - modèle d'évaluation des performances à des couplages métaheuristiques - modèle d'évaluation des performances pour la résolution du problème de flow-shop de permutation stochastique. La méthode  $A$  est donc soit une heuristique soit une métaheuristique, la méthode  $B$  est le modèle d'évaluation des performances. Les heuristiques proposées sont des adaptations d'heuristiques existantes pour le problème du flow-shop en contexte déterministe. Les métaheuristiques utilisées sont la descente stochastique, le recuit simulé ou l'algorithme du kangourou. Le modèle d'évaluation des performances est un modèle markovien ou un modèle de simulation.

### 3.3 Le problème du bin packing

Depuis son introduction en 1973 par Johnson (Johnson, 1973), le problème du bin packing a beaucoup été étudié. Ce problème est un problème de décision NP-Complet (Garey and Johnson, 1979). Le problème d'optimisation associé est NP-difficile : quel est le nombre minimum de boîtes nécessaires pour affecter chaque objet à une et une seule boîte, tout en respectant les contraintes de capacité des boîtes ? Étant donnée la taille des problèmes réels traités, de nombreuses méthodes approchées ont été développées pour le résoudre. Des heuristiques ont été introduites, ainsi que des méthodes de résolution par

métaheuristiques.

Le problème du bin packing est toujours étudié, que ce soit concernant l'étude de bornes inférieures (Balogh et al., 2012) ou la proposition de nouvelle résolution par métaheuristique (Bin et al., 2012).

### 3.3.1 Méthodes exactes

Malgré la complexité du problème étudié, des papiers s'intéressent à la résolution du problème du bin packing grâce à des méthodes exactes. (Valério de Carvalho, 1999) propose un algorithme de résolution utilisant la génération de colonnes et la méthode du branch-and-bound. (Vanderbeck, 1999) présente un algorithme de résolution basé sur la génération de colonnes pour le problème de cutting stock et montre qu'il fonctionne également sur certaines instances du bin packing. (Valério de Carvalho, 2002) étudie les modèles de programmation linéaire pour les problèmes du bin packing et de cutting stock.

Dans cette partie, nous insistons particulièrement sur les deux méthodes suivantes : *MTP* et *BISON* qui ont souvent servi de références pour des travaux ultérieurs de recherche.

#### *MTP* ou *MTRP* : procédure de réduction

(Martello and Toth, 1990) introduisent des bornes inférieures, une procédure de réduction, ainsi qu'un algorithme exact issu de cette procédure.

Leur méthode est basée sur le critère de dominance qui est illustré par la Figure 3.7. Étant donné le contenu de deux boîtes  $B_1$  et  $B_2$ , s'il existe un sous-ensemble  $\{i_1, \dots, i_n\}$  d'objets dans  $B_2$  et une partition  $P = \{P_1, \dots, P_n\}$  de  $B_1$  tels que pour chaque objet  $i_i$  il existe un correspondant  $P_i$  de taille inférieure ( $w_{i_h} \geq \sum_{k \in P_h} w_k, \forall h \in \{1, n\}$ ), alors on dit que  $B_2$  domine  $B_1$  parce qu'une solution ayant  $B_2$  parmi ses boîtes ne requiert pas plus de boîtes qu'une solution ayant  $B_1$ .

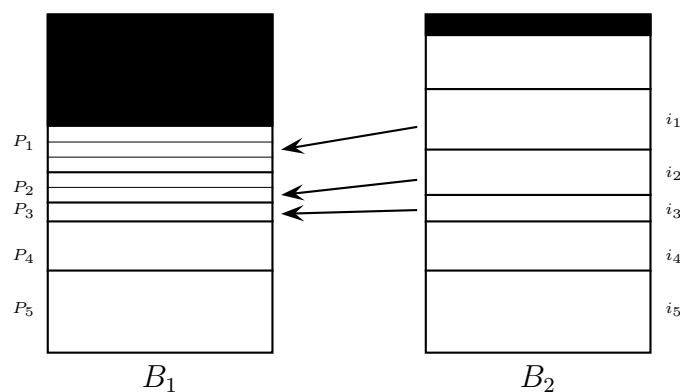


FIGURE 3.7 – Critère de dominance :  $B_2$  domine  $B_1$

La procédure de réduction de Martello et Toth (*MTRP*) cherche des boîtes qui dominent toutes les autres. Quand une telle boîte est trouvée, le problème est réduit en retirant de la liste d'objets les objets de la boîte dominante. Cette procédure réduit l'espace de recherche auquel est appliquée une méthode énumérative telle que l'algorithme de branch-and-bound. La procédure de réduction est notée *MTRP* alors que la méthode de résolution exacte en découlant est notée *MTP*.

*MTP* est une méthode de résolution exacte. Mais afin d'éviter que l'algorithme ne tourne en un temps exponentiel, la recherche doit être limitée à une certaine taille d'ensemble d'objets considérés. Le nombre de trois objets revient régulièrement. Parfois, c'est le nombre d'applications de la procédure de réduction qui est limité. *MTP* est une méthode efficace mais pour un nombre limité de données.

Cette méthode a été reprise et adaptée dans la littérature dans des procédures de recherche locale, notamment dans les papiers (Falkenauer, 1996) et (Levine and Ducatelle, 2004). (Schwerin and Wäscher, 1997) présentent un générateur de données étudiées et testées grâce à *MTP*.

### *BISON*

(Scholl et al., 1997) proposent une procédure hybride exacte, appelée *BISON*, qui combine la recherche tabou et une procédure de branch-and-bound qui utilise la méthode de borne inférieure locale. Cette procédure est composée de six étapes :

- Application de la procédure de réduction *MTRP*.
- Calcul de bornes inférieures *LB*.
- Application d'heuristiques (*FFD*, *B2F*, *BFD*, *WFD*, *FFD – B2F*) donnant une borne supérieure *UB*. Si  $UB = LB$  alors *stop*.
- Calculs de nouvelles bornes inférieures et mise à jour de *LB*. Si  $UB = LB$  alors *stop*.
- Application de la recherche tabou et actualisation de *UB*. Si  $UB = LB$  alors *stop*.
- Application de la méthode de la borne inférieure locale.

Pour la même raison que *MTP*, les calculs sont limités en temps. (Scholl et al., 1997) comparent leur méthode avec celle de (Martello and Toth, 1990) et obtiennent de meilleurs résultats. La méthode *BISON* ne permet pas de trouver une solution optimale pour de grandes instances en peu de temps.

Nous nous intéressons maintenant aux méthodes approchées.

### 3.3.2 Heuristiques

Les algorithmes de liste les plus connus ont été développés par (Johnson, 1973) : Next Fit, First Fit et Best Fit. First Fit et Best Fit ont été publiés dès 1974 (Johnson, 1974). Leur garantie de performance a été précisée la même année (Johnson et al., 1974). De nouvelles heuristiques ont depuis été proposées mais ces trois heuristiques restent les plus utilisées.

Deux catégories d'heuristiques sont distinguées :

- Off-line où les objets doivent être tous connus au début de l'utilisation de l'heuristique pour être affectés aux boîtes,
- On-line où l'affectation peut être faite au fur et à mesure de l'arrivée des objets.

#### Next Fit

L'heuristique Next Fit, notée *NF*, est la méthode de résolution la plus intuitive. Elle consiste à placer un maximum d'objets dans la boîte courante. Les objets sont parcourus dans un ordre donné. Si l'espace disponible dans cette boîte n'est pas suffisant pour l'objet courant, la boîte est définitivement fermée et une nouvelle boîte est ouverte. Cette

nouvelle boîte devient la boîte courante. L'heuristique est donnée par l'Algorithme 10. Cet algorithme est linéaire en temps (Coffman Jr et al., 1996). La garantie de performance de l'heuristique est définie par l'Équation (3.6) (Johnson, 1973).

$$NF(L) \leq 2.OPT(L) + 1, \forall L \quad (3.6)$$

où

- $L$  désigne un ensemble d'objets de taille quelconque à placer dans des boîtes toutes de même capacité,
- $NF(L)$  est le résultat retourné par l'heuristique  $NF$  en nombre de boîtes utilisées,
- $OPT(L)$  représente le nombre optimal de boîtes requises.

---

**Algorithme 10** : Algorithme de principe de l'heuristique Next Fit

---

**Données** : liste d'objets  $(\sigma_i)_{i \in \{1, N\}}$

```

1  $F_j := 0, \forall j \in \{1, N\}$ 
2  $j := 1$ 
3 pour tous les  $i \in \{1, N\}$  faire
4   si l'objet  $\sigma_i$  tient dans la boîte  $j$  ( $F_j + w_{\sigma_i} \leq C$ ) alors
5     Affecter l'objet  $\sigma_i$  à la boîte  $j$  :
6      $x_{\sigma_i, j} := 1$ 
7      $F_j := F_j + w_{\sigma_i}$ 
8   sinon
9      $j := j + 1$ 

```

---

### First Fit

L'heuristique First Fit, notée  $FF$ , diffère de l'heuristique précédente dans le sens où aucune boîte n'est définitivement fermée. Les objets sont toujours parcourus dans un ordre donné. Chaque objet est placé dans la première boîte rencontrée pouvant le contenir. Dès qu'un objet ne rentre dans aucune boîte, une nouvelle boîte est considérée. L'Algorithme 11 décrit cette heuristique.

La garantie de performance de l'heuristique est définie par l'Équation (3.7) (Johnson et al., 1974), avec  $FF(L)$  le nombre de boîtes utilisées pour un ensemble d'objets  $L$  en appliquant l'heuristique  $FF$ .

$$FF(L) \leq \frac{17}{10}.OPT(L) + 2, \forall L \quad (3.7)$$

### Best Fit

L'heuristique Best Fit, notée  $BF$ , consiste à placer les objets dans la meilleure boîte ayant assez d'espace disponible. La meilleure boîte est celle avec le moins d'espace encore disponible après l'affectation de cet objet. Les objets sont toujours parcourus dans un ordre donné. L'Algorithme 12 présente cette heuristique.



---

**Algorithme 11** : Algorithme de principe de l'heuristique First Fit

---

**Données** : liste d'objets  $(\sigma_i)_{i \in \{1, N\}}$

- 1  $F_j := 0, \forall j \in \{1, N\}$
- 2 **pour tous les**  $i \in \{1, N\}$  **faire**
- 3     *affecté* := *faux*
- 4     **tant que** *affecté* = *faux* **faire**
- 5          $j := 1$
- 6         **si** l'objet  $\sigma_i$  tient dans la boîte  $j$  ( $F_j + w_{\sigma_i} \leq C$ ) **alors**
- 7             Affectation de l'objet  $\sigma_i$  à la boîte  $j$  :
- 8              $x_{\sigma_i, j} := 1$
- 9              $F_j := F_j + w_{\sigma_i}$
- 10            *affecté* := *vrai*
- 11         **sinon**
- 12              $j := j + 1$

---

(Johnson et al., 1974) montrent que la garantie de performance de l'heuristique est définie par l'Équation (3.8), avec  $BF(L)$  le nombre de boîtes utilisées pour un ensemble d'objets  $L$  en appliquant l'heuristique  $BF$ . Les garanties de performance des heuristiques  $FF$  et  $BF$  sont les mêmes.

$$BF(L) \leq \frac{17}{10} \cdot OPT(L) + 2, \forall L \quad (3.8)$$

---

**Algorithme 12** : Algorithme de principe de l'heuristique Best Fit

---

**Données** : liste d'objets  $(\sigma_i)_{i \in \{1, N\}}$

- 1  $F_j := 0, \forall j \in \{1, N\}$
- 2  $j_{max} := 1$
- 3 **pour tous les**  $i \in \{1, N\}$  **faire**
- 4      $j_{best} := 0$
- 5     Recherche de la meilleure boîte :
- 6     **pour tous les**  $j \in \{1, j_{max}\}$  **faire**
- 7         **si**  $(F_j + w_{\sigma_i} \leq C) \wedge (F_j \leq F_{j_{best}})$  **alors**
- 8              $j_{best} := j$
- 9     **si** Aucune boîte ne peut recevoir l'objet  $\sigma_i$  ( $j_{best} = 0$ ) **alors**
- 10          $j_{max} := j_{max} + 1$
- 11          $j_{best} := j_{max}$
- 12     Affectation de l'objet  $\sigma_i$  à la boîte  $j_{best}$  :
- 13      $x_{\sigma_i, j_{best}} := 1$
- 14      $F_{j_{best}} := F_{j_{best}} + w_{\sigma_i}$

---

### Algorithmes décroissants

Les trois algorithmes précédemment détaillés pour la résolution du problème du bin packing sont des algorithmes on-line. Ils permettent d'affecter les objets aux boîtes au fur et à mesure que les objets sont connus. À l'inverse, il existe également des algorithmes off-line, qui eux nécessitent que l'intégralité des objets soit connue au début de l'utilisation de l'algorithme.

C'est le cas des algorithmes décroissants. Les objets peuvent être considérés selon un certain ordre de priorité. L'ordre de priorité consistant à trier les objets par taille décroissante permet d'obtenir les algorithmes décroissants définis par (Johnson, 1973) :

- Next Fit Decreasing : *NFD*,
- First Fit Decreasing : *FFD*,
- Best Fit Decreasing : *BFD*.

L'auteur montre que les garanties de performance de *BFD* et *FFD* sont toutes deux égales à celle donnée par l'Équation (3.10), où  $FFD(L)$  (respectivement  $BFD(L)$ ) désigne le nombre de boîtes utilisées pour un ensemble d'objets  $L$  en utilisant l'heuristique *FFD* (respectivement *BFD*). Finalement, (Garey and Johnson, 1979) prouvent l'égalité pour la garantie de performance de *FFD* définie par l'Équation (3.9).

$$FFD(L) = \frac{11}{9} \cdot OPT(L) + 4, \forall L \quad (3.9)$$

$$BFD(L) \leq \frac{11}{9} \cdot OPT(L) + 4, \forall L \quad (3.10)$$

### Autres algorithmes précurseurs

D'autres heuristiques ont été proposées par (Johnson, 1973) :

- Next-2 Fit, notée *N2F*. Si l'objet courant tient dans la boîte précédente, l'y placer. Sinon, le placer dans la boîte courante si possible. S'il ne tient dans aucune de ces deux boîtes, en ouvrir une nouvelle. Cette heuristique considère uniquement deux boîtes pour chaque objet. D'une façon générale, on parle d'heuristique *NkF* qui considère  $k$  boîtes pour chaque objet.
- Worst Fit, notée *WF*, consiste à placer chaque objet dans la boîte partiellement remplie ayant le plus petit niveau, le niveau d'une boîte désignant la somme des tailles des objets placés à l'intérieur. S'il n'y tient pas, ouvrir une nouvelle boîte.
- Worst Fit Decreasing, notée *WFD*, consiste à appliquer la logique de *WF* à la liste d'objets triée par ordre décroissant de taille.
- Almost Worst Fit, notée *AWF*. Placer l'objet courant dans la boîte partiellement remplie ayant le second plus petit niveau s'il y tient. Sinon le placer dans la boîte partiellement remplie ayant le plus petit niveau. S'il ne tient dans aucune de ces deux boîtes, en ouvrir une autre.
- Almost Worst Fit Decreasing, notée *AWFD*, consiste à appliquer la logique de *AWF* à la liste d'objets triée par ordre décroissant de taille.

### ***MFFD* : First Fit Decreasing modifié**

En 1985, (Garey and Johnson, 1985) proposent une modification de l'heuristique *FFD* afin d'améliorer sa garantie de performance. Le départ de l'heuristique est le même que celui de *FFD* : les objets sont classés par ordre décroissant de taille. Des groupes d'objets en fonction de leur taille sont créés, ce qui génère un second classement des objets. Cet algorithme a une garantie de performance de  $\frac{71}{60} \approx 1.1833$  au lieu de  $\frac{11}{9} \approx 1.222$  pour *FFD*. Mais c'est un algorithme en cinq étapes qui est plus compliqué à mettre en œuvre que *FFD*.

### ***B2F* : Best Two Fit**

Dans l'heuristique *B2F*, chaque objet de la liste triée est affecté à la boîte courante si possible. Lorsque plus aucun objet ne peut être placé dans la boîte courante et que cette boîte contient plus d'un objet, l'algorithme vérifie si le plus petit objet de la boîte peut être remplacé par les deux plus petits objets parmi ceux qui ne sont pas encore rangés et qui ont une taille de  $\frac{1}{6}$  ou plus de la taille de la boîte. Si oui, le plus petit objet est remplacé par une telle paire avec la taille totale la plus grande qui convienne au placement. La garantie de performance de *B2F* est donnée par l'Équation (3.11) avec  $B2F(L)$  le nombre de boîtes utilisées pour un ensemble d'objets  $L$  en appliquant l'heuristique *B2F*.

$$B2F(L) \leq \frac{5}{4} \cdot OPT(L) + 4, \forall L \quad (3.11)$$

(Friesen and Langston, 1991) proposent une composition des algorithmes *FFD* et *B2F* pour donner l'algorithme *CFB*. Dans l'algorithme *CFB*, *FFD* et *B2F* sont appliqués tous les deux et la meilleure affectation est sélectionnée. Les résultats retournés par les algorithmes *FFD* et *B2F* sont au pire respectivement  $\frac{11}{9} \approx 1.222$  et  $\frac{5}{4} = 1.25$ . Celui retourné par *CFB* est  $\frac{6}{5} = 1.2$ .

### ***MBS* : Minimum Bin Slack**

(Gupta and Ho, 1999) proposent une heuristique nommée Minimum Bin Slack. Cette heuristique considère les boîtes une à une et non pas les objets. De la même façon que pour les heuristiques décroissantes, la liste d'objets considérée est rangée par ordre décroissant de tailles d'objets.

À chaque étape, un ensemble d'objets qui respecte au mieux la capacité d'une boîte est cherché. Une procédure de recherche teste tous les sous-ensembles d'objets possibles de la liste qui respectent la capacité de la boîte. Le sous-ensemble qui approche au mieux la capacité de la boîte est adopté. Les objets concernés sont supprimés de la liste considérée. Si l'algorithme trouve un sous-ensemble qui remplit exactement une boîte, la recherche est arrêtée parce qu'il n'y aura pas de meilleure affectation possible. La recherche commence par les objets de plus grande taille, parce qu'ils sont habituellement plus difficiles à ranger dans des boîtes. Le processus est arrêté lorsque la liste d'objets considérée est vide. (Gupta and Ho, 1999) montrent que *MBS* est supérieure en terme de qualité de solutions que les heuristiques *FFD* et *BFD*.

Cette heuristique est reprise par (Fleszar and Hindi, 2002) pour donner l'heuristique *MBS'*. Pour chaque boîte, seuls les ensembles contenant le plus gros objet non encore affecté sont considérés. Cette dernière version est ensuite reprise par (Singh and Gupta,

2007), puis par (Fleszar and Charalambous, 2011). Dans les trois cas, les heuristiques sont adaptées pour des métaheuristiques, détaillé dans la Section 3.3.3.

Les garanties de performance de *MBS* et *MBS'* ont été étudiées par (Caprara and Pferschy, 2004) et (Caprara and Pferschy, 2005). Ces deux heuristiques ne sont pas aussi performantes que les heuristiques déjà connues bien que leurs résultats expérimentaux soient meilleurs.

### ***SS* : Sum of Squares algorithm**

L'algorithme *SS* est un algorithme on-line introduit par (Csirik et al., 1999). L'avantage de *SS* est qu'il est mieux adapté à différentes distributions d'objets.

Soit  $N(g)$  le nombre de boites dans l'affectation courante avec un espace libre  $g$ ,  $1 \leq g < C$ , où l'espace libre  $g$  est la différence entre la capacité  $C$  et la somme des tailles des objets affectés à la boite considérée. Le prochain objet  $a_i$  est mis dans la boite qui produit la plus petite valeur mise à jour de  $\sum_{1 \leq g < C} N(g)^2$ . En cas de plusieurs boites correspondantes, l'objet est placé dans la boite ayant le moins d'espace libre. (Csirik et al., 2006) font une étude théorique de cet algorithme et proposent des extensions.

### ***L2F* : Last two Fit**

En 2010, (Kim and Wy, 2010) proposent une amélioration des heuristiques décroissantes par une amélioration appelée Last two Fit (*L2F*). Cette heuristique vérifie si un objet supplémentaire peut être ajouté à la boite courante. Si non, l'objet le plus petit de la boite est remplacé par une paire de deux objets non encore affectés dont la taille combinée est plus grande que celle de l'objet remplacé. Cette amélioration est appliquée sur plusieurs heuristiques décroissantes existantes : *NFD*, *FFD* et *BFD*.

Une extension est proposée : *BBB\_FFD\_L2F* (Bin By Bin First Fit Decreasing with Last Two Fit). Le plus gros objet non affecté de la liste triée démarre une nouvelle boite et tous les objets suivants qui peuvent y être affectés le sont. L'algorithme essaie ensuite de remplacer le plus petit objet de la boite par une paire d'objets non affectés dont la taille combinée est plus grande que celle de l'objet considéré et dont cette taille combinée est la plus grande de toutes les tailles combinées de tous les objets non affectés. La procédure est répétée jusqu'à ce que tous les objets soient affectés.

Cette procédure ressemble à *B2F* mais pour *B2F* la sélection de la paire d'objets est limitée aux deux objets les plus petits dont la taille combinée est au moins  $\frac{1}{6}$  de la taille de la boite. *L2F* n'a pas cette limitation.

## **Conclusion des heuristiques**

Depuis 1973, année où les premières heuristiques ont été développées, de nombreuses propositions ont été faites, jusqu'à encore récemment en 2010. Ces heuristiques ont donné de meilleurs résultats au fur et à mesure de leur développement, mais sont aussi de plus en plus complexes à mettre en place. Des métaheuristiques ont alors récemment été utilisées.

### 3.3.3 Métaheuristiques

L'utilisation des métaheuristiques pour la résolution du problème du bin packing débute dans les années 1990. Les métaheuristiques basées individu sont distinguées des métaheuristiques basées population. Le descriptif général de la méthode considérée est présenté, ainsi que le système de voisinage ou les opérateurs de croisement et de mutation le cas échéant.

Le codage le plus couramment rencontré utilise la solution  $X = (X_i)_{i \in \{1, N\}}$ , avec  $X_i$  le numéro de la boîte où l'objet  $i$  est affecté. Ce codage est qualifié de classique. Le codage utilisé par chacun des travaux référencés n'est spécifié que s'il est différent du codage classique.

#### 3.3.3.1 Fonctions objectifs

Trois fonctions objectifs ont été utilisées.

La plus intuitive est le nombre de boîtes utilisées, donné par l'Équation (2.47). Elle est qualifiée de fonction objectif classique.

Plusieurs solutions peuvent amener à la même valeur de la fonction objectif classique. L'espace de recherche est donc réduit à un ensemble de solutions dont le critère vaut une unité au dessus de l'optimal. La capacité de guidage de l'algorithme dans la recherche est perdue. Pour contrer ce phénomène, une fonction objectif a été introduite par (Falkenauer and Delchambre, 1992). Elle caractérise le remplissage des boîtes utilisées. Pour un nombre donné de boîtes, elle privilégie des solutions où des boîtes sont pleines et des boîtes sont presque vides à des solutions où toutes les boîtes sont remplies de façon homogène. Il semble ainsi plus facile de vider une boîte. Cette fonction est définie par l'Équation (3.12). L'objectif du problème du bin packing est la maximisation de cette fonction.

$$f_{BPP}(X) = \frac{\sum_{j=1}^{M(X)} (F_j(X)/C)^k}{M(X)} \quad (3.12)$$

Avec :

- $M(X)$  le nombre de boîtes utilisées par la solution courante  $X$ ,
- $F_j$  la somme des tailles des objets placés dans la boîte  $j$ , telle que définie dans l'Équation (3.13), avec  $\delta_{j, X_i} = 1$  si  $X_i = j$ , c'est-à-dire si l'objet  $i$  est affecté à la boîte  $j$ , 0 sinon.

$$F_j(X) = \sum_{i=1}^N w_i \cdot \delta_{j, X_i}, \forall j \in \{1, N\} \quad (3.13)$$

- $C$  la capacité d'une boîte,
- $k > 1$  une constante.

Plus  $k$  est grand, plus on privilégie des boîtes bien remplies aux boîtes remplies de façon égales. (Falkenauer, 1996) prouve algébriquement que pour les valeurs de  $k$  supérieures à 2, une solution de  $M + 1$  boîtes avec  $M_F$  boîtes pleines pourrait obtenir une fonction coût supérieure à une solution avec  $M$  boîtes remplies de façon égale. Des valeurs supérieures à 2 amènent à une convergence trop rapide de la solution courante

vers un optimum local. La meilleure valeur est donc  $k = 2$ .

(Fleszar and Hindi, 2002) décident eux aussi de ne pas utiliser la fonction objectif classique. Puisqu'une bonne solution a toujours des boites presque pleines, ils décident d'utiliser la fonction objectif définie par l'Équation (3.14) avec  $F_j$  la somme des tailles des objets  $i$  placés dans la boite  $j$  définie par l'Équation (3.13).

$$H(X) = \sum_{j=1}^N F_j(X)^2 \quad (3.14)$$

Le Tableau 3.3 résume les fonctions objectifs utilisées dans chacun des papiers présentés dans cette section.

	Classique	Falkenauer	Fleszar
Métaheuristiques basées individu			
(Fleszar and Hindi, 2002)			✓
(Fleszar and Charalambous, 2011)			
(Alvim et al., 2004)	✓		
(Loh et al., 2008)			✓
Métaheuristiques basées population			
(Reeves, 1996)		✓	
(Falkenauer, 1996)		✓	
(Levine and Ducatelle, 2004)		✓	
(Brugger et al., 2004)		✓	
(Bhatia and Basu, 2004)		✓	
(Singh and Gupta, 2007)		✓	
(Stawowy, 2008)		✓	

TABLEAU 3.3 – Fonctions objectifs du problème du bin packing optimisées par la littérature

### 3.3.3.2 Métaheuristiques basées individu

#### *PMBS'* + *VNS*

(Fleszar and Hindi, 2002) développent un algorithme hybride (*PMBS'* + *VNS*) qui combine une version modifiée de l'heuristique Minimum Bin Slack de (Gupta and Ho, 1999) et la recherche à voisinage variable (Variable Neighborhood Search). La solution initiale est construite via la version modifiée de l'heuristique *MBS*, notée *PMBS'*. Si *PMBS'* n'améliore pas la solution courante pendant 1000 itérations, la métaheuristique *VNS* est exécutée à partir de la dernière solution générée par *PMBS'*.

Le système de voisinage utilisé dans *PMBS'* est le suivant : un objet dans la boite la moins pleine est sélectionné et utilisé comme base d'application de *MBS'*. *MBS'*

fonctionne de la même façon que *MBS*, mais un objet (le premier de la liste ordonnée par taille décroissante des objets non encore affectés) est fixé dans la nouvelle affectation. Tous les objets autres que celui précédemment sélectionné sont considérés. Si une boîte est vidée lors de cette opération, elle est retirée de la solution.

Le système de voisinage utilisé dans *VNS* est composé de deux étapes :

- "Shaking" :  $k$  déplacements d'objets sont effectués. Pour effectuer un déplacement, sont considérés le déplacement d'un objet choisi aléatoirement dans une nouvelle boîte, ainsi que l'échange entre un objet choisi aléatoirement et un autre objet si les tailles de ces deux objets sont différentes. L'objet choisi aléatoirement est choisi parmi les objets n'ayant pas encore subi de déplacement. Le déplacement est possible s'il ne conduit pas à un dépassement de contrainte relatif à la capacité des boîtes. Un déplacement est choisi aléatoirement parmi tous les déplacements considérés et est effectué.
- Recherche locale : le voisin ainsi créé subit une recherche locale. Seuls les objets appartenant à des boîtes non pleines sont considérés. Tous les déplacements suivants sont considérés : un transfert d'un objet d'une boîte à une autre ou un échange entre deux objets. On effectue le meilleur déplacement, celui correspondant à la meilleure amélioration de la solution courante.

Si le voisin ainsi généré est meilleur que la solution courante, la solution courante devient le voisin. Sinon,  $k$  est incrémenté et un nouveau voisinage est appliqué à la solution courante. La métaheuristique est arrêtée si la borne inférieure est atteinte ou si  $k > k^{max}$ ,  $k^{max}$  étant déterminé expérimentalement.

### ***HI\_BP* : Métaheuristique hybride basée sur une recherche tabou**

(Alvim et al., 2004) proposent une métaheuristique basée sur une recherche tabou, notée *HI\_BP*. Cette méthode a plusieurs caractéristiques : l'incorporation de stratégies de bornes inférieures, la génération de solutions initiales en référence au problème dual min-max, l'utilisation de redistribution de charge basée sur la dominance, la différenciation et l'équilibrage, ainsi qu'un processus d'amélioration utilisant la recherche tabou. La stratégie de sélection de déplacement utilisée par la recherche tabou est une contribution majeure de ce travail et peut être appliquée à d'autres problèmes dans des situations similaires.

Le système de voisinage utilisé dans la méthode tabou est le déplacement d'objets au sein des boîtes.

### ***WABP* : Recuit pondéré**

(Loh et al., 2008) appliquent une nouvelle approche métaheuristique, le recuit pondéré (*WA* : Weight Annealing) de (Ninio and Schneider, 2005). Un algorithme glouton est utilisé. L'espace de recherche est pondéré. Les différentes solutions possibles sont ainsi plus ou moins pénalisées. La pondération de l'espace de recherche évolue à chaque itération jusqu'à atteindre zéro. Ainsi à la fin de l'expérimentation, l'espace de recherche est le même que l'espace de recherche original.

Le système de voisinage est l'échange d'objets entre tous les couples possibles de boîtes. Quatre systèmes de voisinage en particulier sont utilisés. Les deux premiers ont

été proposés par (Fleszar and Hindi, 2002). Les échanges suivants sont considérés :

- (1, 0) : déplacement d'un objet de la boîte  $j_1$  vers la boîte  $j_2$ .
- (1, 1) : échange de l'objet  $i_1$  de la boîte  $j_1$  avec l'objet  $i_2$  la boîte  $j_2$ .
- (1, 2) : échange de l'objet  $i_1$  de la boîte  $j_1$  avec les objets  $i_2$  et  $i_3$  la boîte  $j_2$ .
- (2, 2) : échange des objets  $i_1$  et  $i_2$  de la boîte  $j_1$  avec les objets  $i_3$  et  $i_4$  la boîte  $j_2$ .

### **SAW : Sufficient Average Weight**

(Fleszar and Charalambous, 2011) proposent un mécanisme de contrôle de la taille moyenne des objets affectés dans des boîtes pour éviter de trop grands espaces vides dans les dernières boîtes. Ce mécanisme est basé sur le principe de la moyenne suffisante de tailles (Sufficient Average Weight). Un ensemble d'objets a une SAW si la moyenne des tailles de ses objets est au moins aussi grande que la moyenne des tailles de tous les objets non affectés. L'heuristique sélectionne un des ensembles maximaux qui ont une SAW, ou si aucun n'existe, l'ensemble maximal qui viole le moins la condition de suffisance. Trois versions de l'heuristique sont proposées. Elle sont ensuite combinées à une méthode de réduction.

La version métaheuristique de l'application de cette heuristique consiste en des perturbations successives de la solution courante. Le système de voisinage est une adaptation de la perturbation utilisée par (Fleszar and Hindi, 2002). Les auteurs ne précisent pas la fonction objectif utilisée.

### **3.3.3.3 Métaheuristicques à population**

#### **Algorithme Génétique Hybride**

(Reeves, 1996) propose une version hybride de l'algorithme génétique avec les algorithmes de liste  $NF$ ,  $FF$  et  $BF$ . Avec un codage classique, l'opérateur de croisement entre deux parents peut amener à un enfant dégradé, utilisant plus de boîtes que chacun des deux parents. L'auteur propose donc l'hybridation avec une heuristique de construction comme amélioration de son utilisation.

Une réduction de la taille du problème est utilisée afin de réduire la taille de l'espace de recherche (Reeves Reduction Rules). Elle consiste en la suppression d'un sous-ensemble  $I'$  d'objets du problème tel que  $\sum_{i \in I'} \omega_i \geq (1 - \epsilon) \cdot C$ , avec  $\epsilon$  un nombre réel très petit. Les résultats avec réduction de l'espace sont meilleurs mais rien n'assure que la solution optimale se trouve encore dans l'espace de recherche après application de la réduction.

Le codage utilisé est un codage type permutation. Une solution représente une liste d'objets à laquelle un algorithme de liste est appliqué.

Un système de croisement classique est utilisé pour le passage d'une génération à la suivante. Un opérateur mutation est également utilisé.

#### **HGGA : Algorithme Génétique de Groupement Hybride**

L'algorithme proposé par (Falkenauer, 1996) utilise une approche de groupement pour résoudre le problème du bin packing. L'algorithme génétique considère les boîtes



entières et non pas les objets individuels. Un chromosome est un ensemble de boîtes, avec dans chaque boîte les objets qui y sont rangés. La métaheuristique est composée de trois étapes dont deux opérateurs génétiques :

- Croisement. À partir de deux parents, l'opérateur de croisement crée le chromosome enfant en intégrant des boîtes du second parent dans le premier parent. Les boîtes du chromosome enfant ayant des objets dupliqués sont détruites. Les objets qui y étaient présents mais n'étaient pas présents dans d'autres boîtes sont maintenant libres. Les boîtes générées par l'opérateur de croisement qui ne respectent pas la contrainte de capacité sont également détruites, leurs objets sont libérés.
- Recherche locale. Elle est inspirée du critère de dominance de (Martello and Toth, 1990). Elle permet d'échanger ces objets libres avec d'autres objets plus gros déjà rangés dans des boîtes. Ainsi, les boîtes sont mieux remplies.
- Mutation. Des boîtes sont ouvertes aléatoirement : leurs objets sont libérés.

À chaque étape, les objets libérés sont réintégrés aux boîtes via l'heuristique *FFD*.

Pour une meilleure utilisation de l'algorithme génétique de groupement, l'auteur utilise le codage classique augmenté d'une partie de groupement, qui correspond aux boîtes. L'opérateur de croisement est ainsi appliqué sur la seconde partie du chromosome, ce qui évite le cas particulier où le chromosome enfant utilise plus de boîtes que les chromosomes parents.

### **HACO : Optimisation par colonie de fourmis hybride**

(Levine and Ducatelle, 2004) adaptent l'optimisation par colonie de fourmis au problème du bin packing. La piste de phéromones traduit la probabilité d'avoir un objet de taille  $w_i$  et un objet de taille  $w_j$  dans la même boîte. L'heuristique utilisée est une adaptation de *FFD*.

L'optimisation par colonie de fourmis est hybridée à un algorithme de recherche locale. Cette recherche locale est inspirée de celle de (Falkenauer, 1996). Elle est itérée avec la nouvelle solution jusqu'à ce qu'il n'y ait plus d'amélioration de la fonction objectif.

Le codage est en deux parties : un objet  $i$  est affecté à une boîte  $j$ . Les objets ne sont pas individualisés, ils sont désignés par leur taille et non pas par leur numéro :  $S = (w_i)_{i \in \{1, N\}}$  avec  $w_i$  la taille de l'objet  $i$ , associé à la somme des tailles des objets déjà affectés à la boîte considérée.

Le système de voisinage est défini par l'utilisation de l'optimisation par colonie de fourmis. La construction d'une solution voisine est régie par la piste de phéromones. La stratégie de mise à jour de la phéromone est une stratégie globale où seule la meilleure fourmi actualise la mémoire.

### **AntPacking**

(Brugger et al., 2004) utilisent également l'optimisation par colonie de fourmis pour résoudre ce problème. La construction d'une solution est basée sur l'heuristique *FFD* et la piste de phéromones. La règle de décision est basée sur une probabilité d'exploiter la meilleure option disponible sur une étape de décision. Une fois qu'une boîte est remplie, une recherche locale est appliquée, inspirée par l'heuristique *FFD* et le critère de dominance de (Martello and Toth, 1990). Les objets rangés dans la boîte

courante sont remplacés un à un par un objet non rangé. La recherche locale s'arrête quand il n'y a pas d'amélioration possible.

Un prétraitement des données est réalisé pour diminuer l'espace de recherche. Les couples d'objets remplissant exactement une boîte sont retirés de la liste des objets à ranger.

Le codage utilisé est semblable à celui de (Levine and Ducatelle, 2004). La taille de l'objet courant est codée, ainsi que le degré de remplissage de la boîte considérée qui est traduit par l'espace restant dans chaque boîte.

Le système de voisinage est défini par l'utilisation de l'optimisation par colonie de fourmis. La stratégie de mise à jour de la phéromone est une stratégie binaire locale : la fonction est égale à 1 si la boîte considérée est pleine, 0 sinon.

### ***MGGA : Multi-chromosomal Genetic Representation and Better-Fit Heuristic***

(Bhatia and Basu, 2004) proposent un codage génétique multi-chromosome et des opérateurs génétiques pour la résolution du problème du bin packing. Cette méthode est hybridée avec une heuristique. L'application d'un opérateur génétique libère un ou des objets qui sont réaffectés par application d'une heuristique.

Une heuristique appelée Better-Fit est proposée. Un objet non affecté remplace un objet affecté s'il améliore le remplissage de la boîte. L'heuristique tente ensuite de la même façon d'affecter l'objet remplacé, en commençant par la première boîte. Le processus continue jusqu'à ce qu'un objet remplacé ne puisse plus être affecté. Il est alors affecté avec l'heuristique *BF*.

Le codage utilisé est le suivant : un individu est composé de chromosomes. Un chromosome représente la boîte. Un gène correspond au numéro de l'objet affecté au chromosome.

Les opérateurs génétiques utilisés sont redéfinis : croisement entre deux individus, mutation et translocation.

### ***C\_BP : Approche combinée***

(Singh and Gupta, 2007) proposent une approche combinée *C\_BP* qui utilise l'heuristique *H\_SGGA* qu'ils détaillent dans ce même papier et une amélioration de *PMBS'* proposé par (Fleszar and Hindi, 2002). Dans l'approche combinée, la solution générée par l'heuristique amélioration de *PMBS'* devient une des solutions initiales utilisées par *H\_SGGA*.

L'approche hybride *H\_SGGA* consiste à affecter grâce à une heuristique les objets libérés. L'heuristique est dérivée de l'heuristique Better-Fit (Bhatia and Basu, 2004), qui devient l'heuristique Complete/Better-Fit. Avant application de l'heuristique Better-Fit, si un objet non affecté peut remplir une boîte, il y est affecté. L'heuristique Better-Fit considère dans un premier temps uniquement les échanges (1-1). Lorsque l'échange (1-1) n'est plus possible, une amélioration consiste à considérer l'échange (2-1).

Pour le codage d'une solution, un chromosome est un ensemble de boîtes. Il n'y a pas d'ordre entre les boîtes.

Les opérateurs génétiques utilisés sont les suivants :

- Mutation : cinq objets déjà affectés sont libérés aléatoirement.

- Croisement : l'enfant construit est l'héritage aléatoire successif des boîtes les mieux remplies de chacun des deux parents. Entre deux héritages de boîte, les objets affectés à l'enfant sont supprimés des boîtes des parents.

### **ES : Algorithme évolutionnaire**

(Stawowy, 2008) propose une version simplifiée de l'algorithme génétique : il n'utilise ni hybridation ni spécialisation.

Le codage utilisé est une permutation avec séparateurs : c'est une liste de  $n$  objets et  $l$  séparateurs de boîtes, les entiers de rang  $\{1, \dots, n\}$  représentent les objets, les entiers  $\{n + 1, \dots, n + l\}$  représentent les séparateurs.

L'opérateur de mutation est utilisé :  $\lambda$  enfants sont générés à partir des parents par simples mutations. Les enfants générés sont issus de permutation et/ou d'insertion depuis la solution parent. Les meilleurs enfants sont sélectionnés. Si après un certain nombre d'itérations, il n'y a pas d'enfant meilleur que la meilleure solution courante, le pire enfant de la génération précédente devient le nouveau parent. Cette procédure de déstabilisation permet de sortir des minimums locaux.

Après évaluation du parent, des opérations de réduction de l'espace de recherche sont possibles. Sans celles-ci, dans la plupart des cas, la solution optimale n'est pas atteignable. Les méthodes de réduction de l'espace de recherche n'assurent pas toujours l'accessibilité de la solution optimale. Trois méthodes de réduction ont été testées :

- Martello and Toth Reduction Procedure (Martello and Toth, 1990),
- Reeves Reduction Rule (Reeves, 1996),
- Separators removal, proposée par l'auteur. À un instant donné, les séparateurs de boîtes ne sont plus considérés. En effet, beaucoup de séparateurs redondants sont présents, ce qui peut produire des déplacements par mutation improductifs.

### **BPSO : Binary Particle Swarm Optimization**

(Bin et al., 2012) proposent une amélioration de la *BPSO* avec structure hiérarchique : les notions de leader et de suiveur sont introduites. À chaque itération, si un suiveur trouve une meilleure solution qu'un leader actuel, il deviendra un leader à l'itération suivante, ce qui permet de sortir des minimums locaux. De plus, la mutation est introduite afin d'éviter une convergence trop rapide de la solution vers un optimum local. Le système de voisinage utilisé est régi par une marche aléatoire des suiveurs vers les leaders.

Cette méthode est principalement développée pour la résolution du problème du voyageur de commerce. Peu de détails sur l'application au problème du bin packing sont donnés par les auteurs.

### **3.3.4 Conclusion de l'état de l'art sur le problème du bin packing**

Un large spectre de méthodes de résolution a été utilisé pour résoudre le problème du bin packing. Des méthodes exactes, des heuristiques et des métaheuristiques ont été

proposées. Les méthodes les plus récurrentes sont les métaheuristiques basées population, en particulier l'algorithme génétique.

Les méthodes développées dans la littérature sont des méthodes globalement complexes. La plupart des papiers propose par exemple un couplage entre une métaheuristique et une recherche locale, suivi d'une reconstruction de solution grâce à des heuristiques, utilisant au préalable une réduction de l'espace de recherche. La réduction de l'espace de recherche peut potentiellement conduire à une perte de la solution optimale.

Dans la suite de ce manuscrit, nous concentrons nos travaux sur le développement de méthodes approchées simples pour la résolution du problème du bin packing. L'objectif principal de cette thèse est la résolution du problème de planification d'activités avec affectation de ressources. Nous nous devons donc de proposer des méthodes facilement utilisables, notamment par le personnel hospitalier.

Une extension du problème du bin packing est le problème du bin packing avec conflits (Bin Packing Problem with Conflicts *BPPC* ou *BPC*). Certains objets ne peuvent pas être affectés dans la même boîte dû à des restrictions de compatibilités entre ces objets. Le problème du bin packing avec conflits a été résolu par (Gendreau et al., 2004), (Khanafar et al., 2010) et (Muritiba et al., 2010) :

- (Gendreau et al., 2004) développent des heuristiques adaptées de celles du problème du bin packing de base et du problème de coloration de graphe. Une borne inférieure est proposée.
- (Khanafar et al., 2010) travaillent sur des méthodes de réduction afin d'améliorer les bornes inférieures connues.
- (Muritiba et al., 2010) proposent une résolution du *BPPC* par une méthode approchée qui intègre des calculs de borne inférieure, des heuristiques (algorithme évolutionnaire) et des métaheuristiques (méthode tabou). Les solutions générées sont utilisées pour l'initialisation d'un algorithme Branch & Price.

Dans le cas de conflits entre objets, il y a  $N^2$  compatibilités ou incompatibilités à définir. Dans le cas d'incompatibilités entre objets et boîtes, il n'y a que  $N \times M$  incompatibilités à définir, avec  $M$  le nombre de boîtes. C'est pourquoi nous avons opté pour la formalisation de notre problème comme un problème du bin packing avec incompatibilités entre objets et boîtes. Les méthodes que nous proposerons pour la résolution du problème du bin packing de base seront facilement applicables au problème du bin packing avec incompatibilités entre objets et boîtes.

### 3.4 Différentes utilisations de la PSO pour la résolution de problèmes d'optimisation combinatoire

Des problèmes d'optimisation combinatoire ont été résolus avec la PSO. Certains de ces problèmes sont présentés ci-dessous avec quelques exemples de résolution par la PSO. Les définitions des opérateurs proposées par chaque papier pour chacun des problèmes sont détaillées.

### 3.4.1 Le problème des $n$ dames

Le problème des  $n$  dames ( $n$ -queens problem) consiste à placer  $n$  dames sur un échiquier de dimensions  $N \times N$  de façon à ce qu'elles ne peuvent pas s'attaquer les unes les autres. Ainsi, il doit y avoir une et une seule dame sur chaque ligne, chaque colonne et chaque diagonale.

En 2003, (Hu et al., 2003) s'intéressent à ce problème comme à un problème de permutation. Chaque particule est une liste ordonnée d'éléments, chaque élément ne pouvant être présent qu'une et une seule fois dans la liste. Un élément représente une case de l'échiquier occupée par une dame : le  $i$ -ème numéro de la liste représente le numéro de colonne occupée sur la  $i$ -ème ligne de l'échiquier. Par exemple, la liste  $(1, 3, 2)$  représente une solution où les trois dames sont placées aux intersections suivantes : première ligne et première colonne, deuxième ligne et troisième colonne, troisième ligne et deuxième colonne.

L'application de la composante de la vitesse  $V_{p,t+1}$  influencée par la meilleure position personnelle  $P_{p,t}$ , sur une position  $X_{p,t}$  est définie comme la probabilité d'appliquer une permutation. Cette permutation est l'une des permutations à effectuer pour passer de la position  $X_{p,t}$  à la position  $P_{p,t}$ .

De plus, afin que l'essaim ne converge pas trop rapidement vers la meilleure position globale  $G_t$ , un opérateur de mutation est introduit. Si la particule est identique à  $G_t$ , elle échange aléatoirement une paire d'éléments.

Le réglage de paramètres choisi est le suivant :

- Nombre de particules : 10
- Influence locale composée de deux voisines.
- Coefficients :  $c_1 = 0.5 + \frac{rand}{2}$ ,  $c_2 = c_3 = 1.49445$

### 3.4.2 Le problème du voyageur de commerce

Le problème du voyageur de commerce est connu sous le nom de Travelling Salesman Problem. Étant donné un nombre de villes séparées d'une distance connue, il s'agit de déterminer une tournée qui passe une et une seule fois par chaque ville et qui se termine au point de départ. L'objectif est la minimisation de la distance totale parcourue par le voyageur de commerce.

(Clerc, 2004) définit le cadre général de la méthode de résolution de la manière suivante :

- La position d'une particule est définie comme un  $N$ -cycle non orienté :  $X_{p,t} = (X_{i1}, X_{i2}, \dots, X_{iN})$  parce que les cycles Hamiltoniens sont recherchés. L'espace de recherche est défini comme l'ensemble fini de tous les  $N$ -cycles.
- La vitesse d'une particule appliquée à une position à un instant donne une autre position. C'est une liste des permutations de deux éléments. La vitesse est représentée par  $V_{p,t} = ((X_k, Y_k)_{k=1, \dots, |V_{p,t}|})$  avec  $|V_{p,t}|$  la longueur de la liste. La vitesse avec une liste vide est représentée par  $V_{p,t} = \emptyset$ .

Les opérateurs suivants ont été définis :

- L'addition  $X_{p,t} \oplus V_{p,t+1}$  entre une position et une vitesse permet de déplacer une particule dans l'espace de recherche. Par exemple, si on considère  $X_{p,t} = (1, 2, 3, 4)$  et  $V_{p,t+1} = ((12)(34))$ , alors  $X_{p,t} \oplus V_{p,t+1} = X_{p,t+1} = (2, 1, 4, 3)$ .
- Soient deux vitesses  $V_{p,t}^{(1)}$  et  $V_{p,t}^{(2)}$ , l'addition  $V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$  est la liste des permutations qui contient d'abord celles de  $V_{p,t}^{(1)}$  suivies par celles de  $V_{p,t}^{(2)}$ .
- La multiplication externe d'une vitesse  $V_{p,t}$  par un scalaire  $c_1$  :
  - Si  $c_1 = 0$ , alors  $c_1 \otimes V_{p,t} = \emptyset$ .
  - Si  $0 < c_1 < 1$ , alors on tronque la liste de permutations de la vitesse  $V_{p,t}$  à  $E(c_1 \cdot \|V_{p,t}\|)$  avec  $E(x) \leq x < E(x) + 1$ .
  - Si  $c_1 \geq 1$  alors on duplique la liste de permutations de la vitesse  $V_{p,t}$ . Soit  $c_1 = k + c'_1$  avec  $k$  valeur entière non-nulle et  $c'_1$  le reste de la division entière,  $c_1 \otimes V_{p,t} = \underbrace{V_{p,t} \oplus V_{p,t} \oplus \dots \oplus V_{p,t}}_{k \text{ fois}} \oplus c'_1 \otimes V_{p,t}$ .
- La soustraction  $P_{p,t} \ominus X_{p,t}$  définit une vitesse obtenue par un algorithme donné, celui déterminant les permutations nécessaires pour le passage de  $X_{p,t}$  à  $P_{p,t}$ .
- La distance entre deux positions  $P_{p,t}$  et  $X_{p,t}$  est la longueur de la vitesse  $V_{p,t}$ , vitesse obtenue en calculant  $P_{p,t} \ominus X_{p,t}$  ou  $X_{p,t} \ominus P_{p,t}$ .

Le réglage de paramètres choisi est le suivant :

- Nombre de particules :  $N - 1$
- Influence locale composée de quatre voisines, incluant la particule elle-même.
- Coefficients :  $c_1 \cdot r_1 \in ]0, 1[$ ,  $(c_2 \cdot r_2, c_3 \cdot r_3) \in [0, 2]^2$

(Shi et al., 2007) résolvent le TSP en proposant une nouvelle façon de calculer la vitesse de la particule. L'originalité de la méthode est l'ajout d'une stratégie incertaine. Chaque particule est influencée par sa propre trajectoire, sa meilleure position et la meilleure position globale. L'application d'une heuristique 2-opt permet de supprimer les croisements de ligne dans la tournée générée qui correspond à la meilleure position de l'essaim. Les auteurs notent de meilleurs résultats que les travaux précédents utilisant la PSO pour résoudre le TSP. Les papiers plus anciens résolvent des instances composées de 17 villes. Les auteurs proposent une méthode dont la qualité des solutions est au pire des cas 10% moins bonne que la solution optimale pour des instances de 50 à 80 villes.

(Goldberg et al., 2008) résolvent le TSP en proposant de nouveaux opérateurs de vitesse. Là aussi l'influence globale est utilisée. Plusieurs types de déplacement sont testés :  $v_1$  est un voisinage inversion (échange entre deux positions) ou voisinage Lin-Kernighan sur la position courante (échange d'un nombre donné d'arêtes),  $v_2$  et  $v_3$  sont le voisinage path-relinking (échanges successifs entre deux positions adjacentes) pour l'influence de la meilleure position personnelle et l'influence de la meilleure position globale. Chacun des trois éléments  $v_1, v_2$  et  $v_3$  définissent des déplacements à effectuer sur la position courante. La composition entre chacun des trois éléments est régie par des probabilités. Des combinaisons entre les différents opérateurs vitesse sont également implémentées.

Le réglage de paramètres choisi est le suivant :

- Nombre de particules : 20
- Coefficients :  $c_1 \cdot r_1, c_2 \cdot r_2, c_3 \cdot r_3 \in ]0, 1[$ . Au début des calculs,  $c_1 = 0.95, c_2 = 0.05, c_3 = 0.05$ . Tout au long des itérations,  $c_1$  diminue alors que  $c_2$  et  $c_3$  augmentent.

Les auteurs considèrent des instances composées jusqu'à 7397 villes.

(Fan, 2010) utilise la PSO sur le TSP avec l'influence globale. Son approche est une approche hybridée avec trois opérations : un opérateur de croisement, un facteur heuristique et un facteur de perturbation adaptative. Une position est une liste ordonnée de villes. Un changement de position est effectué par la succession des trois opérations. La nouvelle position de la particule est le résultat d'un croisement entre sa position courante, sa meilleure position et la meilleure position de l'essaim. Une fois le déplacement de toutes les particules effectué, les meilleures positions de chaque particule et la meilleure position de l'essaim sont actualisées. Ensuite, chaque particule subit une mise à jour de sa position grâce au facteur heuristique et au facteur de perturbation adaptative. La taille de l'essaim dépend de la taille des instances. Les résultats sont meilleurs que ceux des précédents travaux.

(Labeled et al., 2012) hybrident la PSO pour la résolution du TSP avec l'heuristique d'amélioration 2-opt. L'approche est globale. L'approche proposée permet d'éviter le réglage de paramètres. Une position est une liste ordonnée de villes. Le déplacement d'une particule est défini par l'opérateur de croisement : un croisement des positions  $X_{p,t}$  et  $P_{p,t}$  et un croisement des positions  $X_{p,t}$  et  $G_t$  sont générés. La position ayant la meilleure valeur de la fonction objectif devient la nouvelle particule. L'opérateur de croisement permet de ne pas tomber dans un optimum local. L'heuristique 2-opt est ensuite appliquée sur cette nouvelle position pour l'améliorer. L'essaim est composé de cinquante particules.

### 3.4.3 Le RCPSP

Le RCPSP (Resource Constrained Project Scheduling Problem) peut être défini de la façon suivante. Un projet nécessite la réalisation de plusieurs activités. Les activités interagissent entre elles par le biais de deux contraintes : il existe des précédences entre des activités, une activité doit être réalisée avant une autre, et les activités nécessitent des ressources pour être réalisées. Un certain nombre de ressources renouvelables appartenant à un type sont disponibles. Chaque activité nécessite un nombre donné de ressources d'un type particulier. Chaque activité a un temps de traitement. L'objectif est l'ordonnancement des activités au plus tôt avec affectation de ressources.

(Kemmoé Tchomté and Gourgand, 2009) proposent une adaptation de la PSO pour la résolution du RCPSP. Le cadre général défini par (Clerc, 2004) a été repris par les auteurs. Il a cependant été adapté pour s'assurer de la validité des séquences. En effet, une permutation entre deux activités pour le problème du RCPSP peut amener une solution non valide si la précedence entre les activités n'est plus respectée. Pour cela, la somme entre une position et une vitesse est redéfinie dans le but d'éviter les séquences non valides. Une détection de particules clones est ajoutée au processus. Une recherche locale est réalisée sur la meilleure particule pour tenter de l'améliorer.

L'approche globale est utilisée. Les coefficients utilisés sont  $c_1 = 0.047$ ,  $c_2 = 0.378$  et  $c_3 = 1.494$ . L'essaim est composé de  $N$  particules, avec  $N$  le nombre de dimensions de l'espace de recherche, c'est-à-dire le nombre d'activités à réaliser. La méthode proposée

permet d'obtenir des résultats comparables aux méthodes existantes.

(Lemamou et al., 2012) traitent le problème du RCPSP avec une approche globale. Ils proposent une nouvelle définition des opérateurs qui respecte les contraintes de précédence. Une particule est une liste ordonnée d'activités. Un algorithme série ou parallèle est appliqué à cette liste pour affecter les activités. L'ordonnement peut se faire dans l'ordre normal de la liste ou à l'envers. Deux paramètres sont donc ajoutés à la particule : l'algorithme et l'ordre. La vitesse de la particule est un ensemble de mouvements à appliquer aux activités de la liste. La différence entre deux positions est la liste des déplacements d'activités à réaliser pour passer d'une position à l'autre. Le produit d'une vitesse par un réel correspond à l'arrondi de la composante de la vitesse. L'addition entre deux vitesses est la somme des déplacements définis par chacune de ces vitesses. L'application de la vitesse sur la particule donne une pseudo-séquence qu'il faut corriger. En effet, les déplacements peuvent amener à une pseudo-séquence où certains emplacements sont vides ou d'autres sont occupés par plusieurs activités. Une procédure de réparation lui est ensuite appliquée pour respecter les contraintes de précédence. Une procédure de recherche locale est ajoutée à chaque itération.

Les instances testées sont composées de trente, soixante ou cent vingt activités. L'essaim est composé de cinquante particules. Les coefficients utilisés sont  $c_1 = 0.5$ ,  $c_2 = 1.5$  et  $c_3 = 2$ . Les résultats sont comparables aux résultats existants.

### 3.4.4 Les problèmes d'ordonnement

(Ponnambalam et al., 2009) utilisent la PSO avec l'approche globale pour la résolution du problème d'ordonnement du flowshop. Il est caractérisé par un nombre de machines et un nombre de jobs ayant un temps de traitement sur chaque machine. Une machine ne peut traiter qu'un job à la fois et un job n'est traité que par une machine à la fois. Les jobs sont non préemptifs, un job ne peut pas être achevé avant son traitement complet. Chaque job est traité successivement par chacune des machines. Lorsqu'il a été traité par une machine, il est traité par la suivante. Dès que le job libère la machine courante, le job suivant y est traité. L'objectif de ce problème est d'ordonner tous les jobs. Les deux critères couramment utilisés sont :

- le makespan, c'est-à-dire la date de fin de traitement du dernier job dans le système,
- le flowtime, c'est-à-dire la somme des dates de fin de tous les jobs, communément appelé la somme des  $C_i$ .

Une particule est une liste ordonnée de jobs. La vitesse représente une liste de permutations de jobs. La différence entre deux positions est la liste des permutations à appliquer sur une position pour atteindre l'autre position. Après la mise à jour des vitesses et positions, une procédure de recherche locale est appliquée sur chaque particule à chaque itération.

L'essaim est composé de quatre-vingt particules pour toutes les tailles d'instances traitées (vingt jobs et cinq, dix ou vingt machines). Les valeurs des coefficients sont  $c_1 = 1$ ,  $c_2 = 2$  et  $c_3 = 2$ .



(Izakian et al., 2010) traitent le problème d'ordonnancement de grid job scheduling (tâches informatiques). Soit un nombre de jobs à effectuer et un nombre de machines parallèles disponibles. Chaque job a un temps de traitement pour chaque machine. Les jobs sont indépendants entre eux et non préemptifs. L'objectif est d'affecter les jobs aux machines, tout en minimisant deux critères : le makespan et le flowtime.

Les auteurs utilisent la PSO binaire pour résoudre leur problème. Ils s'inspirent de la définition d'opérateurs de (Kennedy and Eberhart, 1997) où une particule est un vecteur binaire et la vitesse est la probabilité pour chaque composante de la particule de prendre la valeur 0 ou 1.

L'approche globale est utilisée même si le détail de la méthode est faite sur une approche locale. Deux codages sont utilisés : un codage indirect et un codage direct.

Avec le codage indirect, une particule est une matrice binaire ayant autant de lignes que de machines et autant de colonnes que de jobs. La composante vaut 1 si le job correspondant est affecté à la machine correspondante. La vitesse est une matrice dont les valeurs sont comprises dans un intervalle. L'équation classique de calcul de la vitesse est utilisée. Lors de la mise à jour de la position de la particule, pour chaque job, la valeur 1 est associée à la machine dont la valeur est la plus grande dans la matrice vitesse.

Avec le codage direct, une particule est un vecteur, chaque indice du vecteur correspondant au job et la valeur de la composante correspondant à la machine affectée au job. La vitesse est la même que dans le codage indirect. Une transformation mathématique permet de calculer rapidement la nouvelle vitesse et la nouvelle position de la particule. Le codage direct est utilisé en pratique parce qu'il est plus rapide.

L'essaim est composé de cinquante particules. Les valeurs des coefficients sont  $c_1 = 0$ ,  $c_2 = 2$  et  $c_3 = 2$ . Chaque composante de la matrice vitesse est bornée par la valeur 40. Les instances sont composées de cinq cent douze jobs et seize machines. Leur méthode donne de bons résultats quant au makespan mais de moins bons quant au flowtime. Ceci est dû à la hiérarchisation des critères choisie par les auteurs, ils privilégient la minimisation du makespan en dépit de celle du flowtime.

(Kashan and Karimi, 2009) adaptent la PSO sur le problème d'ordonnancement à des machines parallèles. L'objectif est d'affecter des tâches à un nombre donné de machines, tout en minimisant la durée totale des tâches affectées. Le critère utilisé est le makespan, la date à laquelle est effectuée la dernière tâche.

L'approche est globale. Une particule est un vecteur, chaque indice du vecteur correspondant à la tâche, la composante associée correspondant à la machine affectée à cette tâche. Les auteurs redéfinissent leurs opérateurs. Pour l'opérateur différence entre deux positions, les tâches ayant deux affectations différentes dans chacune de ces deux positions sont affectées successivement à une machine dès qu'elle est disponible. L'opérateur multiplication consiste en une multiplication terme à terme. Les coefficients sont en réalité des vecteurs de taille  $N$  dont chaque composante vaut 0 ou 1. L'opérateur somme entre deux positions est l'application du croisement entre ces deux positions. Une heuristique de recherche locale est appliquée à chaque itération sur 10% des particules.

Les coefficients relatifs à la meilleure position personnelle et la meilleure position globale sont générées par la distribution de Bernouilli, pour laquelle la probabilité d'être égal à 1 est 0.3. L'essaim est composé de dix particules.

### 3.4.5 Le problème du bin packing

L'objectif du bin packing est d'affecter des objets dans un minimum de boîtes tout en respectant la capacité des boîtes et la taille des objets. Dans le cas multi-objectif, les deux critères à minimiser peuvent être le nombre de boîtes utilisées ainsi que les déviations par rapport aux centres de gravité de chaque boîte.

(Liu et al., 2008) utilisent la PSO pour résoudre ce problème. Les auteurs incluent un facteur évolutionnaire à leur approche grâce à l'opérateur mutation. L'approche globale est utilisée. Les auteurs proposent une influence de chaque particule par sa meilleure position atteinte ou la meilleure position globale, contrairement à l'utilisation classique où la particule est influencée par les deux en même temps. La métaheuristique PSO est hybridée avec une heuristique de construction.

Chaque particule contient le nombre de boîtes utilisées et l'ordre dans lequel les objets sont rangés dans les boîtes par l'heuristique. La meilleure position personnelle et la meilleure position globale sont ici remplacées par la meilleure boîte de la meilleure position personnelle et la meilleure boîte de la meilleure position globale. Le déplacement de la particule est déterminé par l'insertion de la meilleure boîte personnelle ou la meilleure globale en première position de la particule. Tous les objets dupliqués sont supprimés.

Un opérateur mutation est ensuite appliqué à la particule. Avec une certaine probabilité, soit il échange des objets entre boîtes ou concatène les deux boîtes les moins remplies en une seule ou sépare une boîte en deux, soit il applique une rotation d'objets dans une boîte, soit il mélange des objets dans une boîte. Si les boîtes ainsi modifiées ne vérifient plus les contraintes au problème, elles sont détruites, ainsi que les deux boîtes les moins remplies.

L'heuristique de construction permet de réaffecter les objets non affectés dans un ordre aléatoire, afin de s'assurer que toutes les solutions sont des particules valides.

Les tailles de population varient entre cent et cinq cents particules. Les instances testées sont composées de cinq cents objets.

(Krichen and Dahmani, 2010) s'intéressent au problème du bin packing bi-objectif uni-dimensionnel dans le cas d'un chargement d'avion. Leurs travaux sont inspirés de (Kashan and Karimi, 2009) qui traitent du problème d'ordonnement de machines parallèles précédemment détaillé. En effet, le problème du job scheduling peut être vu comme une adaptation du problème du bin packing. Un objet est défini par son poids et sa priorité. Les objets doivent être rangés dans des boîtes qui elles-mêmes sont rangées dans des compartiments de l'avion. Les deux critères à maximiser sont le poids et la priorité de chargement total des boîtes, tout en minimisant le nombre de boîtes et en équilibrant les charges dans l'avion.

L'approche est une approche globale. Une particule est une liste de couples. Chaque couple correspond à un objet, le premier élément indiquant le numéro de la boîte, le second le numéro du compartiment. Le déplacement d'une particule se fait en appliquant directement le Système d'Équations (3.4). Les auteurs ne détaillent pas leurs opérateurs. Les contraintes peuvent être violées lors d'un déplacement de particules. Les dépassements de contraintes sont pénalisés dans la fonction objectif. Le cadre expérimental n'est pas détaillé.

### 3.4.6 Conclusion sur la PSO

D'abord développée pour la résolution de problèmes d'optimisation continue, la PSO a rapidement et largement été adaptée à la résolution de problèmes d'optimisation combinatoire (TSP, RCPSP, Bin Packing, Flowshop, etc.). Pour cela, une redéfinition des opérateurs est nécessaire. Chaque problème nécessite sa propre redéfinition, et plusieurs travaux traitant du même problème ont pu l'aborder différemment. Des adaptations ont également été proposées (hybridation avec une recherche locale, hybridation avec une heuristique de construction, etc.). Chacun des travaux présentés obtient de bons résultats dans son domaine. Cette méthode étant récente, elle attire notre attention pour la résolution de notre problème.

## 3.5 Conclusion

Ce chapitre a présenté un état de l'art en trois parties.

Le problème relatif à la Communauté Hospitalière de Territoire, problème de planification d'activités avec affectation de ressources, a été formalisé comme une Programmation Linéaire en Nombres Entiers. La première partie de ce chapitre a présenté un état de l'art sur les méthodes de résolution des problèmes d'optimisation combinatoire. Des méthodes exactes existent, mais étant donnée la complexité de notre problème, il n'existe pas de méthode exacte pour le résoudre en un temps polynomial. Un état de l'art sur les méthodes de résolution approchées a donc été présenté.

Notre problème peut être vu comme un problème du bin packing. De nombreux travaux se sont penchés sur la résolution de ce problème par des méthodes exactes ainsi que par des méthodes approchées. Des heuristiques ont été proposées mais elles ne garantissent pas l'obtention d'une solution optimale. L'utilisation de métaheuristiques pour la résolution du problème du bin packing s'est répandue depuis les années 1990. Des métaheuristiques basées individu ainsi que des métaheuristiques basées population ont été proposées. Beaucoup de ces méthodes combinent l'utilisation de ces métaheuristiques à des méthodes de recherche locale ou de réduction de l'espace de recherche. Ces méthodes sont souvent compliquées à mettre en place. Dans la suite de ce manuscrit, nous proposons des méthodes simples à appliquer.

Une des méthodes que nous allons proposer utilise une métaheuristique récemment développée : l'optimisation par essaim particulaire. Cette métaheuristique a été utilisée pour la résolution de nombreux problèmes d'optimisation combinatoire, comme le problème des  $n$ -dames, le problème du voyageur de commerce, le problème de gestion de projet à contraintes de ressources, les problèmes d'ordonnancement. Peu de travaux ont porté sur l'application de l'optimisation par essaim particulaire à la résolution du problème du bin packing.

Le chapitre suivant propose des méthodes de résolution du problème du bin packing. Nous utiliserons notamment un couplage entre l'optimisation par essaim particulaire et des algorithmes de liste. Les méthodes proposées seront étendues pour la résolution du problème du bin packing avec incompatibilités. Nous avons montré dans le chapitre précédent que le problème de planification d'activités avec affectation de ressources peut être considéré comme tel.

# Proposition de méthodes pour la résolution de problèmes de bin packing

## Sommaire

---

<b>4.1</b>	<b>Introduction</b>	<b>105</b>
<b>4.2</b>	<b>Métaheuristiques pour le bin packing de base</b>	<b>105</b>
4.2.1	Codage d'une solution	105
4.2.2	Métaheuristiques basées individu	106
4.2.2.1	Exploration de $C$	106
4.2.2.2	Exploration de $\Omega$	107
4.2.2.3	Système de voisinage	107
4.2.2.4	Algorithme du recuit simulé	108
4.2.2.5	Algorithme du kangourou	108
4.2.3	Métaheuristique basée population	108
4.2.3.1	Position et espace de recherche	109
4.2.3.2	Fonction objectif	109
4.2.3.3	Principe général	109
4.2.3.4	Définition des opérateurs	110
4.2.4	Conclusion du codage classique	111
<b>4.3</b>	<b>Couplage heuristiques métaheuristiques pour le bin packing de base</b>	<b>112</b>
4.3.1	Codage d'une solution	112
4.3.2	Métaheuristiques basées individu	114
4.3.2.1	Solution initiale	114
4.3.2.2	Systèmes de voisinage	114
4.3.2.3	Algorithme du recuit simulé	114
4.3.2.4	Algorithme du kangourou	116
4.3.3	Métaheuristique basée population	116
4.3.3.1	Principe général	118
4.3.3.2	Déplacement 1	121
4.3.3.3	Déplacement 2	125

4.3.4	Conclusion du couplage hiérarchique . . . . .	125
<b>4.4</b>	<b>Extension des méthodes pour le bin packing avec incompatibilités . . . . .</b>	<b>125</b>
4.4.1	Codage . . . . .	127
4.4.2	Proposition d'algorithmes de liste . . . . .	127
4.4.2.1	Next Fit CHT . . . . .	127
4.4.2.2	First Fit CHT . . . . .	127
4.4.2.3	Best Fit CHT . . . . .	127
4.4.2.4	Worst Fit CHT . . . . .	129
4.4.2.5	Algorithmes décroissants . . . . .	129
4.4.3	Système de voisinage . . . . .	129
4.4.4	Pénalisation de contraintes . . . . .	131
4.4.5	Méthode des critères pondérés . . . . .	132
<b>4.5</b>	<b>Conclusion . . . . .</b>	<b>133</b>

---

## 4.1 Introduction

Le problème du bin packing a été présenté dans le Chapitre 2. Ce problème consiste à ranger des objets dans un nombre minimum de boîtes en respectant la contrainte de capacité. Le problème considéré dans ce manuscrit consiste en la planification d'activités avec affectation de ressources. L'analogie entre ces deux problèmes a été faite. Il existe des incompatibilités dans le cas de notre problème : toutes les activités ne peuvent pas être affectées à toutes les ressources. Ce problème est finalement considéré comme un problème de bin packing avec incompatibilités, à l'inverse du problème du bin packing de base.

Notre objectif est la mise en place d'un outil de planification pour la CHT. Cela passe par la résolution des modèles mathématiques formalisés dans le Chapitre 2. Puisque nos problèmes sont une extension du problème du bin packing et que le problème du bin packing est NP-difficile, les problèmes de planification d'activités avec affectation de ressources sont également NP-difficiles. Nous utilisons donc des méthodes approchées pour les résoudre.

Un état de l'art sur le problème du bin packing a été fait dans le Chapitre 3. De nombreuses méthodes de résolution ont été proposées. Beaucoup d'entre elles sont fastidieuses à appliquer. L'utilisation finale de notre outil sera faite par le personnel hospitalier. Nous devons donc penser des méthodes les plus simples possible. Dans ce chapitre, nous proposons de nouvelles méthodes de résolution qui sont simples à développer et à utiliser.

La Section 4.2 présente une application de métaheuristiques sur le problème du bin packing de base. Les métaheuristiques utilisées sont le recuit simulé, l'algorithme du kangourou et l'optimisation par essaim particulaire. Nous proposons ensuite dans la Section 4.3 un nouveau codage inspiré des problèmes de permutation. Ce codage est utilisé par un couplage entre des heuristiques et des métaheuristiques. Les mêmes métaheuristiques que dans la section précédente sont appliquées. Toutes ces méthodes sont adaptées au problème du bin packing avec incompatibilités dans la Section 4.4. Pour cela, les éléments ayant besoin d'une redéfinition sont précisés.

## 4.2 Métaheuristiques pour le bin packing de base

Dans cette section, certaines des métaheuristiques présentées dans le Chapitre 3 sont appliquées à la résolution du problème du bin packing. Le codage utilisé, le calcul de la solution initiale et les systèmes de voisinage sont détaillés. Pour les métaheuristiques basées individu, le recuit simulé ainsi que l'algorithme du kangourou sont appliqués. Dans la catégorie métaheuristique basée population, nous étudions l'optimisation par essaim particulaire. Le paramétrage de chacune des méthodes est donné dans cette section.

### 4.2.1 Codage d'une solution

Le codage classique utilisé pour la résolution du bin packing est utilisé. Une solution est représentée par le vecteur  $X = (X_i)_{i \in \{1, N\}}$ . L'objet  $i$  est affecté à la boîte  $X_i$ . Par exemple, en considérant le vecteur  $X = (3, 2, 2, 1, 3)$ , l'objet 1 est affecté à la boîte 3, l'objet 2 à la boîte 2, l'objet 3 à la boîte 2, l'objet 4 à la boîte 1 et l'objet 5 à la boîte 3.

L'ensemble des solutions est noté  $\Omega$ . Le cardinal de  $\Omega$  est égal à  $N^N$ . Un second ensemble  $C \subseteq \Omega$  est défini, c'est l'ensemble des solutions admissibles. Une solution est admissible si elle vérifie la contrainte de capacité des boites.

La Figure 4.1 représente les deux ensembles considérés.

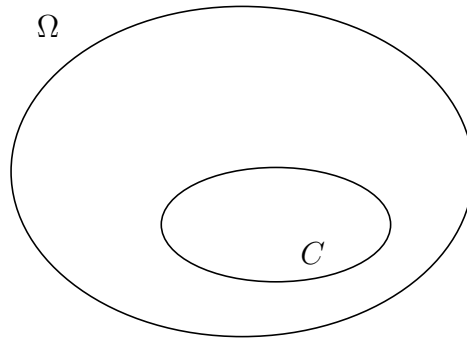


FIGURE 4.1 – Ensembles de solution  $C \subseteq \Omega$

## 4.2.2 Métaheuristiques basées individu

Dans cette partie, nous proposons deux approches en fonction de la stratégie d'exploration des solutions :

- La première approche se concentre sur l'exploration des solutions admissibles, c'est-à-dire  $C$ .

- La seconde approche s'intéresse à l'ensemble de toutes les solutions, c'est-à-dire  $\Omega$ .

Dans ces deux approches, nous utilisons  $Cont(X)$  qui permet d'évaluer le respect ou non de la contrainte de capacité des boites. Elle est définie par l'Équation (4.1). Si  $Cont(X) = 0$ , toutes les capacités de boites sont respectées par l'affectation de la solution  $X$ . Ainsi, la solution  $X$  est admissible, c'est-à-dire  $X \in C$ .

$$Cont(X) = \sum_{j=1}^N \max(0; F_j(X) - C) \quad (4.1)$$

### 4.2.2.1 Exploration de $C$

La première approche, qui explore l'ensemble  $C$  des solutions admissibles, est définie par l'Algorithme 13.

Une solution initiale admissible est déterminée par l'application de l'une des trois heuristiques de base de résolution du problème du bin packing : Next Fit, First Fit ou Best Fit. Ces heuristiques ont été présentées dans le Chapitre 3.

Chaque solution voisine doit également être admissible. À chaque génération d'une solution voisine, on vérifie donc qu'aucune capacité de boite n'est dépassée. Si ce n'est pas le cas, une nouvelle solution est générée jusqu'à trouver une solution admissible.

La fonction objectif  $f_{BPP}$  proposé par (Falkenauer and Delchambre, 1992) défini par l'Équation (3.12) est utilisé pour le calcul de  $H$ .

**Algorithme 13** : Première approche : exploration de  $C$

```

1 Construction d'une solution initiale  $X$  par une heuristique
2 tant que Critère d'arrêt non atteint faire
3   | répéter
4   |   | Construction d'une solution voisine  $X' \in V(X)$ 
5   |   | jusqu'à Les contraintes sont respectées :  $Cont(X') = 0$ ;
6   |   | si  $H(X') \leq H(X)$  alors
7   |   |   |  $X := X'$ 

```

**4.2.2.2 Exploration de  $\Omega$**

La seconde approche, qui explore l'ensemble  $\Omega$  de toutes les solutions, est définie par l'Algorithme 14.

Une solution initiale est générée aléatoirement : chaque objet est affecté de façon aléatoire à une boîte.

Toutes les solutions voisines construites sont acceptées. Pour chaque solution voisine, la somme des tailles d'objets affectés dépassant la capacité des boîtes est calculée par l'Équation (4.1). La fonction objectif  $H$  est une hiérarchisation entre les critères  $f_{BPP}$  et  $Cont$ . Ces critères sont à minimiser. Ainsi, le nombre de contraintes dépassées peut être nul à la fin des expérimentations. Dans ce cas, la solution finale est admissible.

**Algorithme 14** : Seconde approche : exploration de  $\Omega$

```

1 Génération d'une solution initiale  $X$  aléatoirement
2 tant que Critère d'arrêt non atteint faire
3   | Construction d'une solution voisine  $X' \in V(X)$ 
4   | si  $(Cont(X') < Cont(X))$  OU  $((Cont(X') = Cont(X))$  ET  $(H(X') \leq H(X)))$ 
5   |   | alors
6   |   |   |  $X := X'$ 

```

**4.2.2.3 Système de voisinage**

Le système de voisinage  $V$  permet de passer de la solution courante  $X$  à la solution voisine  $X'$ . Le système de voisinage utilisé génère aléatoirement une solution voisine. Un objet  $i$  est choisi aléatoirement. Il est affecté à une autre boîte  $X'_i \neq X_i$ . L'algorithme de ce système de voisinage est donné par l'Algorithme 15.

**Algorithme 15** : Système de voisinage

**Entrées** :  $X$  solution et  $M$  le nombre de boîtes utilisées

```

1 Choisir uniformément et aléatoirement  $i \in \{1, N\}$ 
2 Choisir uniformément et aléatoirement  $j \in \{1, M + 1\}$  tel que  $j \neq X_i$ 
3  $X_i := j$ 

```



#### 4.2.2.4 Algorithme du recuit simulé

Les deux paramètres à régler pour l'algorithme du recuit simulé sont la température initiale  $T_0$  et le facteur de décroissance  $\alpha$ . La température initiale est choisie de telle façon que toutes les transitions sont autorisées au début, c'est-à-dire  $e^{-\frac{H(X')-H(X)}{T_0}} \simeq 1, \forall(X, X')$ . (Aarts and van Laarhoven, 1987) proposent un algorithme qui calcule la température initiale, donné par l'Algorithme 16. (Kirkpatrick, 1984) donne  $L_{T_0}$  égal au nombre de variables du problème,  $N$  dans notre cas.

---

#### Algorithme 16 : Détermination de la température initiale $T_0$

---

**Données** :  $X$  une solution quelconque

- 1  $\Delta H := 0$
  - 2 **pour tous les  $l$  de 1 à  $L_{T_0}$  faire**
  - 3     Choisir uniformément et aléatoirement  $X'$  dans  $V(X)$
  - 4      $\Delta H := \Delta H + \max\{0; H(X') - H(X)\}$
  - 5      $X := X'$
  - 6  $\overline{\Delta H}^{(+)} := \frac{\Delta H}{L_{T_0}}$
  - 7  $x_0 := 0.8$
  - 8  $T_0 := \frac{\overline{\Delta H}^{(+)}}{\ln(x_0^{-1})}$
- 

La formule utilisée pour calculer le facteur de décroissance  $\alpha$  est définie par l'Équation (4.2). Elle permet une décroissance lente de la température.  $T_a$  est la dernière température calculée dans l'algorithme du recuit simulé, proche de zéro.

$$\alpha = \text{IterMax} \sqrt{\left(\frac{T_a}{T_0}\right)} \quad (4.2)$$

#### 4.2.2.5 Algorithme du kangourou

Le paramètre à régler dans l'algorithme du kangourou est le palier  $A$ . La formule utilisée pour calculer  $A$  est donnée par l'Équation (4.3) (Fleury, 1993).

$$A \geq \text{card}(V) \cdot \ln(2) \quad (4.3)$$

Dans le cas de notre système de voisinage,  $\text{card}(V) = N.M$  donc le calcul de  $A$  est donné par l'Équation (4.4).

$$A \geq N.M \cdot \ln(2) \quad (4.4)$$

Le système de voisinage  $W$  utilisé consiste en l'application huit fois du système de voisinage  $V$ .

### 4.2.3 Métaheuristique basée population

La métaheuristique basée population utilisée est la PSO. Pour rappel, les notations utilisées sont les suivantes. À l'instant  $t$ ,

- Chaque particule  $p$  a
  - une position courante  $X_{p,t}$ ,
  - la valeur de sa fonction objectif  $H(X_{p,t})$ ,
  - sa meilleure position  $P_{p,t}$  connue à l'instant  $t$ ,
  - la valeur de la fonction objectif de sa meilleure position  $H(P_{p,t})$ .
- La meilleure position de l'essaim est notée  $G_t$ , la valeur de sa fonction objectif est notée  $H(G_t)$ .

Le passage de  $t$  à  $t + 1$  est défini par le Système d'Équations (3.4), qui est rappelé ci-dessous.

$$\begin{cases} X_{p,t+1} &= X_{p,t} \oplus V_{p,t+1} \\ V_{p,t+1} &= c_1 \otimes V_{p,t} \oplus c_2 \cdot r_2 \otimes (P_{p,t} \ominus X_{p,t}) \oplus c_3 \cdot r_3 \otimes (G_t \ominus X_{p,t}) \end{cases}$$

Pour faciliter le codage, l'étude est faite dans le cadre de la première approche, où  $X_{p,t} \in \Omega$ , c'est-à-dire une solution peut ne pas être admissible. Pour s'assurer de l'admissibilité des solutions à partir d'un certain nombre d'itérations, il faut pénaliser les contraintes non respectées.

#### 4.2.3.1 Position et espace de recherche

Comme précédemment, le codage classique utilisé pour la résolution du bin packing est utilisé. Une solution  $X_{p,t}$  est l'affectation d'un objet  $i \in \{1, N\}$  à une boîte  $X_{p,t}(i)$ . L'ensemble des solutions est noté  $\Omega$ . Le cardinal de  $\Omega$  est égal à  $N^N$ .

#### 4.2.3.2 Fonction objectif

Le dépassement de contraintes est pénalisé par la fonction objectif. Un objet peut être affecté à une boîte fictive et les capacités de boîte peuvent être dépassées. L'utilisation de boîtes fictives est une conséquence de la définition des opérateurs. Ces dépassements sont représentés par  $Cont$ . L'algorithme de calcul de  $Cont$  est défini par l'Algorithme 17. La fonction objectif est une hiérarchisation entre les critères  $f_{BPP}$  et  $Cont$ .

---

#### Algorithme 17 : Évaluation de contraintes pour la PSO

---

**Entrées :**  $X_{p,t}$  solution courante

- 1  $Cont(X_{p,t}) := \sum_{j=1}^N \max(0; F_j(X) - C)$
  - 2 **pour tous les**  $j \in \{1, N\}$  **faire**
  - 3     **pour tous les**  $i \in \{1, N\}$  **faire**
  - 4         **si**  $(X_{p,t}(i) < 1) \vee X_{p,t}(i) > N$  **alors**
  - 5              $Cont(X_{p,t}) := Cont(X_{p,t}) + w_i$
- 

#### 4.2.3.3 Principe général

L'algorithme de l'application de la PSO est donné par l'Algorithme 18.

---

**Algorithme 18** : Pseudo-code de la PSO pour le BPP

---

Données :  $P$  nombre de particules,  $T_{max}$ ,  $c_1$ ,  $c_2$ ,  $c_3$

- 1  $t := 0$
- 2 **pour toutes les Particules  $p$  de 1 à  $P$  faire**
- 3     *Génération aléatoire de la position initiale*
- 4     À chaque objet  $i \in \{1, N\}$  est affectée une boite  $X_{p,0}(i) \in \{1, N\}$
- 5     *Enregistrement de la meilleure position personnelle*
- 6      $P_{p,0} := X_{p,0}$
- 7     *Détermination de la meilleure particule de l'essaim*
- 8      $G_{t=0} := P_{p',t=0}$  tel que  $H(P_{p',t=0}) = \min_{p \in \{1, P\}} H(P_{p,t=0})$
- 9     **tant que Critères d'arrêt non atteint faire**
- 10     **pour toutes les Particules  $p$  de 1 à  $P$  faire**
- 11         Tirage aléatoire de  $r_2, r_3 \in [0, 1] \times [0, 1]$
- 12         *Détermination de la nouvelle position  $X_{p,t+1}$*
- 13          $V_{p,t+1} := c_1 \otimes V_{p,t} \oplus c_2 \cdot r_2 \otimes (P_{p,t} \ominus X_{p,t}) \oplus c_3 \cdot r_3 \otimes (G_t \ominus X_{p,t})$
- 14          $X_{p,t+1} := X_{p,t} \oplus V_{p,t+1}$
- 15         *Détermination de la meilleure position personnelle*
- 16          $P_{p,t+1} := X_{p,t+1}$  si  $H(X_{p,t+1}) \leq H(P_{p,t+1})$
- 17         *Détermination de la meilleure particule de l'essaim*
- 18          $G_{t+1} := P_{p',t+1}$  tel que  $H(P_{p',t+1}) = \min_{p \in P} H(P_{p,t+1})$
- 19          $t := t + 1$

---

La définition des opérateurs nous permet de détailler le déplacement d'une particule défini par la Ligne 12 de l'Algorithme 18.

#### 4.2.3.4 Définition des opérateurs

##### Vitesse

L'opérateur vitesse  $V_{p,t+1}$  est un vecteur de la même dimension que le vecteur position. Une fois appliquée à une position  $X_{p,t}$ , la vitesse donne une autre position  $X_{p,t+1}$ . À chaque objet  $i$ , elle associe la différence entre les numéros de boîtes qui lui sont affectés dans les deux positions.

Si  $X_{p,t} = (3, \dots)$  et  $X_{p,t+1} = (5, \dots)$ , alors  $V_{p,t+1} = (2, \dots)$ . En effet,  $X_{p,t+1} = X_{p,t} + V_{p,t+1}$ , c'est-à-dire pour la première composante,  $5 = 3 + 2$ .

La position et la vitesse sont des vecteurs de même dimension. Toutes les opérations définies ci-dessous sont définies grâce aux opérateurs mathématiques classiques : addition "+" ou soustraction "-".

##### Déplacement (addition) "position $\oplus$ vitesse"

L'addition d'une position et d'une vitesse déplace la particule dans l'espace de recherche. L'application de la vitesse à la position est définie par la somme du vecteur

position  $X_{p,t}$  et du vecteur vitesse  $V_{p,t+1}$  dont le résultat est le vecteur position  $X_{p,t+1}$ . Le déplacement  $X_{p,t} \oplus V_{p,t+1}$  est défini par l'Équation (4.5).

$$X_{p,t+1}(i) = X_{p,t}(i) + V_{p,t+1}(i), \forall i \in \{1, N\} \quad (4.5)$$

Par exemple, si  $X_{p,t} = (1, 2, 3, 4, 5, 6)$  et  $V_{p,t+1} = (0, 1, 0, -1, 0, 0)$  alors  $X_{p,t+1} = X_{p,t} \oplus V_{p,t+1} = (1, 3, 3, 3, 5, 6)$ .

### Soustraction "position $\ominus$ position"

Soient  $X_{p,t}^{(1)}$  et  $X_{p,t}^{(2)}$  deux positions. La différence  $X_{p,t}^{(2)} \ominus X_{p,t}^{(1)}$  est définie telle que la vitesse  $V_{p,t}$  appliquée à la position  $X_{p,t}$  donne la position  $X_{p,t}^{(2)}$ . L'algorithme calculant la vitesse est celui déterminant les déplacements de boîtes nécessaires pour passer de la position  $X_{p,t}^{(1)}$  à la position  $X_{p,t}^{(2)}$ . La soustraction  $X_{p,t}^{(2)} \ominus X_{p,t}^{(1)}$  est donc la différence entre ces deux vecteurs, composante par composante, définie par l'Équation (4.6).

$$V_{p,t}(i) = X_{p,t}^{(2)}(i) - X_{p,t}^{(1)}(i), \forall i \in \{1, N\} \quad (4.6)$$

Par exemple, si  $X_{p,t}^{(1)} = (1, 2, 3, 4, 5, 6)$  et  $X_{p,t}^{(2)} = (1, 3, 3, 3, 5, 6)$  alors  $V_{p,t} = (0, 1, 0, -1, 0, 0)$ .

### Addition "vitesse $\oplus$ vitesse"

Soient  $V_{p,t}^{(1)}$  et  $V_{p,t}^{(2)}$  deux vitesses. La vitesse  $V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$  est la somme des différences d'affectation de chaque objet  $i$  dans chacune des vitesses, c'est à dire la somme des composantes des deux vecteurs. L'addition  $V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$  est définie par l'Équation (4.7).

$$V_{p,t}(i) = V_{p,t}^{(1)}(i) + V_{p,t}^{(2)}(i), \forall i \in \{1, N\} \quad (4.7)$$

Par exemple, si  $V_{p,t}^{(1)} = (0, 1, 0, -1, 0, 0)$  et  $V_{p,t}^{(2)} = (2, 0, 0, -2, 0, 0)$  alors  $V_{p,t} = (2, 1, 0, -3, 0, 0)$ .

### Multiplication "coefficient $\otimes$ vitesse"

Soient  $c$  un coefficient réel et  $V_{p,t}$  une vitesse. La multiplication  $c \otimes V_{p,t}$  est égale pour chaque composante  $i \in \{1, N\}$  à l'arrondi à l'entier le plus proche du produit  $c \cdot V_{p,t}(i)$ .

## 4.2.4 Conclusion du codage classique

Deux approches ont été proposées pour l'application des métaheuristiques : mener la recherche dans l'ensemble  $C$  des solutions admissibles ou dans l'ensemble  $\Omega$  de toutes les solutions. Dans les deux approches, l'évaluation des contraintes non respectées doit être faite à chaque étape.

Si la recherche est effectuée dans l'ensemble  $C$  des solutions admissibles, l'évaluation des contraintes permet de s'assurer de l'admissibilité de la solution générée : une nouvelle solution est générée tant qu'elle n'est pas admissible.

Si la recherche est effectuée dans l'ensemble  $\Omega$  de toutes les solutions, toutes les solutions sont acceptées mais le dépassement de contraintes doit être pénalisé. Ainsi, à la fin de la recherche, la solution finale est admissible.

Dans les deux cas, ces vérifications sont coûteuses en calculs.

Un nouveau codage est proposé, utilisant une hybridation entre les métaheuristiques précédemment utilisées et les algorithmes de construction classique. L'utilisation des heuristiques nous assure l'admissibilité de toutes les solutions à moindre coût.

### 4.3 Couplage heuristiques métaheuristiques pour le bin packing de base

Les méthodes présentées dans cette partie utilisent une hybridation entre une métaheuristique et un algorithme de liste. Parmi les couplages présentés dans le Chapitre 3, notre proposition correspond à un couplage avec approche hiérarchique.

#### 4.3.1 Codage d'une solution

Le codage proposé est inspiré par celui utilisé dans les problèmes d'ordonnement de permutation. Soit  $\Omega'$  l'ensemble des listes d'objets comme dans un problème de permutation. Une solution  $(Y_i)_{i \in \{1, N\}} \in \Omega'$  est donc une liste avec  $Y_i$  un numéro d'objet.

Soit  $L$  un algorithme de liste.  $L$  est appliqué à la liste d'objets  $Y$  pour déterminer l'affectation des objets aux boîtes :  $(X_i)_{i \in \{1, N\}} \in S$ . Les algorithmes de liste utilisés peuvent être  $NF$ ,  $FF$  ou  $BF$ . Ainsi, comme dans le codage classique,  $X_i$  est le numéro de boîte affecté à l'objet  $i$ .

Une fonction coût  $H$  est ensuite appliquée à l'affectation  $X$ . Le critère  $f_{BPP}$  est utilisé. Le schéma général du codage est le suivant :

$$Y \in \Omega' \xrightarrow[\text{Heuristique } L]{} L(Y) = X \in S \xrightarrow[\text{Critère } H]{} H(X)$$

La Figure 4.2 représente les ensembles de solution considérés, avec

- $\Omega'$  l'ensemble de toutes les listes d'objets.  $\text{card}(\Omega') = N!$
- $\Omega$  l'ensemble de toutes les affectations possibles d'objets aux boîtes, sans vérifier la contrainte de capacité.  $\text{card}(\Omega) = N^N$
- $C$  l'ensemble des solutions admissibles : une solution est admissible si elle respecte la contrainte de capacité.  $C \subseteq \Omega$ .
- $S$  l'ensemble des solutions construites par application d'un algorithme de liste sur une liste d'objets.  $S \subseteq \Omega$  et  $S \subseteq C$ .

**Preuve que  $S$  contient l'ensemble des solutions optimales.** Soit  $X^*$  une solution optimale quelconque.  $X^* = (X_1^*, \dots, X_N^*)$  avec  $X_i^*$  le numéro de la boîte affectée à l'objet  $i$ . On en déduit la liste des objets rangés dans chaque boîte. Par concaténation de ces listes, on retrouve une liste ordonnée de tous les objets :  $Y^* \in \Omega'$ . En appliquant

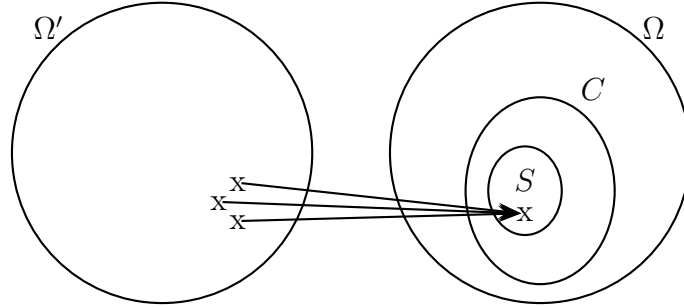


FIGURE 4.2 – Ensembles de solutions utilisés par l’hybridation

l’heuristique  $FF$  ou  $NF$  à  $Y^*$ , on retrouve  $X^*$ , donc  $X^* \in S$ . Donc l’ensemble  $C^*$  des solutions optimales est inclus dans  $S$ .

L’exemple suivant considère l’affectation de huit objets. Deux solutions différentes dans  $\Omega'$  donnent le même élément dans  $S$ . Au total,  $2 \times 4! \times 2 = 96$  éléments de  $\Omega'$  donnent le même élément dans  $S$ .

	Solution 1		Solution 2																								
	<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td style="width: 33%; text-align: center;">5</td><td style="width: 33%; text-align: center;">8</td><td style="width: 33%; text-align: center;">2</td></tr> <tr><td style="width: 33%; text-align: center;">3</td><td style="width: 33%; text-align: center;">4</td><td style="width: 33%; text-align: center;">7</td></tr> <tr><td style="width: 33%; text-align: center;">1</td><td style="width: 33%; text-align: center;">6</td><td style="width: 33%; text-align: center;">1</td></tr> <tr><td style="width: 33%; text-align: center;">2</td><td style="width: 33%; text-align: center;">1</td><td style="width: 33%; text-align: center;">3</td></tr> </table>	5	8	2	3	4	7	1	6	1	2	1	3		<table border="1" style="border-collapse: collapse; width: 100%;"> <tr><td style="width: 33%; text-align: center;">5</td><td style="width: 33%; text-align: center;">8</td><td style="width: 33%; text-align: center;">7</td></tr> <tr><td style="width: 33%; text-align: center;">3</td><td style="width: 33%; text-align: center;">6</td><td style="width: 33%; text-align: center;">2</td></tr> <tr><td style="width: 33%; text-align: center;">1</td><td style="width: 33%; text-align: center;">4</td><td style="width: 33%; text-align: center;">1</td></tr> <tr><td style="width: 33%; text-align: center;">2</td><td style="width: 33%; text-align: center;">1</td><td style="width: 33%; text-align: center;">3</td></tr> </table>	5	8	7	3	6	2	1	4	1	2	1	3
5	8	2																									
3	4	7																									
1	6	1																									
2	1	3																									
5	8	7																									
3	6	2																									
1	4	1																									
2	1	3																									
$Y \in \Omega' =$	(3, 5, 1, 6, 4, 8, 7, 2)		(3, 5, 1, 4, 6, 8, 2, 7)																								
$X \in S =$	(2, 3, 1, 2, 1, 2, 3, 2)		(2, 3, 1, 2, 1, 2, 3, 2)																								

Soit  $S_B$  l’ensemble des solutions  $\in S$  construites par l’heuristique  $BF$ ,  $S_F$  l’ensemble construit par  $FF$  et  $S_N$  l’ensemble construit par  $NF$ . Nous avons le fait suivant :  $C^* \subseteq S_B \subseteq S_F \subseteq S_N \subseteq C \subseteq \Omega$ . En exploitant ce codage, l’espace de recherche est réduit et il est certain qu’une solution optimale est atteignable.

Cette proposition présente certains avantages :

- L’admissibilité des solutions est assurée par les algorithmes de liste.
- Beaucoup de métaheuristiques existent déjà pour résoudre les problèmes d’ordonnement de permutation, avec leurs systèmes de voisinage pour les métaheuristiques basées individu, et leurs opérateurs de croisement et/ou de mutation pour les métaheuristiques basées population.
- La métaheuristique est appliquée sur un plus petit ensemble  $S$  que l’original  $\Omega$ .
- L’ensemble  $S$  contient toutes les solutions optimales.
- Cette méthode est une combinaison de méthodes existantes et simples qui sont faciles à implémenter.

Le principe de fonctionnement de la méthode hybride est détaillé dans l’Algorithme 19 sur l’exemple de la descente stochastique.

Trois métaheuristiques sont utilisées :

---

**Algorithme 19** : Algorithme de principe de l'hybridation métaheuristique basée individu - algorithme de liste

---

**Données** : Solution initiale  $Y \in \Omega'$

- 1  $X := L(Y)$  : un algorithme de liste est appliqué à la liste  $Y$
- 2 **tant que** *Critère d'arrêt non atteint* **faire**
- 3     Choisir uniformément et aléatoirement  $Y' \in V(Y)$
- 4      $X' := L(Y')$
- 5     **si**  $H(X') \leq H(X)$  **alors**
- 6          $Y := Y'$
- 7          $X := X'$

---

- L'algorithme du recuit simulé,
- L'algorithme du kangourou,
- L'optimisation par essaim particulaire.

Bien que déjà détaillés dans le Chapitre 3, les algorithmes de principe sont redonnés pour illustrer le passage d'un espace de recherche à l'autre dans l'utilisation des métaheuristicues.

### 4.3.2 Métaheuristicues basées individu

L'application du couplage des métaheuristicues basées individu et des heuristiques a été présentée dans (Gourgand et al., 2014b).

#### 4.3.2.1 Solution initiale

Une solution initiale est obtenue de façon aléatoire. Une liste de  $N$  objets est générée aléatoirement et uniformément. Une heuristique est ensuite appliquée à cette liste ordonnée d'objets.

#### 4.3.2.2 Systèmes de voisinage

Plusieurs systèmes de voisinage  $V$  peuvent être utilisés :

- $P_i$  : l'objet à la position  $i$  permute avec celui à la position  $i + 1$ . Il est défini par l'Algorithme 20. Son cardinal vaut  $|P_i| = N$
- $P_{i,j}$  : l'objet à la position  $i$  permute avec celui à la position  $j \neq i$ . Il est défini par l'Algorithme 21. Son cardinal vaut  $|P_{i,j}| = N \cdot (N - 1)$
- $I_{i,j}$  : l'objet à la position  $i$  est inséré à la position  $j \neq i$ . Il est défini par l'Algorithme 22. Son cardinal vaut  $|I_{i,j}| = \frac{N \cdot (N-1)}{2}$

Tous ces systèmes de voisinage satisfont les propriétés d'accessibilité et de réversibilité.

#### 4.3.2.3 Algorithme du recuit simulé

L'algorithme du recuit simulé utilisé avec l'hybridation est décrit par l'Algorithme 23. La détermination de la température initiale et du facteur de décroissance est faite de la même façon que précédemment grâce à l'Algorithme 16 et à l'Équation (4.2).

---

**Algorithme 20** : Système de voisinage  $P_i$

---

**Données** : Liste  $Y$  des  $N$  objets

- 1 Choisir uniformément et aléatoirement  $i \in N$
  - 2  $aux := Y_i$
  - 3  $Y_i := Y_{i+1}$
  - 4  $Y_{i+1} := aux$
- 

---

**Algorithme 21** : Système de voisinage  $P_{i,j}$

---

**Données** : Liste  $Y$  des  $N$  objets

- 1 Choisir uniformément et aléatoirement  $i \in N$
  - 2 Choisir uniformément et aléatoirement  $j \in N$
  - 3 **tant que**  $i = j$  **faire**
  - 4     └ Choisir uniformément et aléatoirement  $j \in N$
  - 5  $aux := Y_i$
  - 6  $Y_i := Y_j$
  - 7  $Y_j := aux$
- 

---

**Algorithme 22** : Système de voisinage  $I_{i,j}$

---

**Données** : Liste  $Y$  des  $N$  objets

- 1 Choisir uniformément et aléatoirement  $i \in N$
  - 2 Choisir uniformément et aléatoirement  $j \in N$
  - 3 **tant que**  $i = j$  **faire**
  - 4     └ Choisir uniformément et aléatoirement  $j \in N$
  - 5  $aux := Y_i$
  - 6 **si**  $j > i$  **alors**
  - 7     └ **pour**  $k$  de  $i$  à  $j - 1$  **faire**
  - 8         └  $Y_k := Y_{k+1}$
  - 9 **si**  $j < i$  **alors**
  - 10     └ **pour**  $k$  de  $i$  à  $j + 1$ ,  $pas = -1$  **faire**
  - 11         └  $Y_k := Y_{k-1}$
  - 12  $Y_j := aux$
-



---

**Algorithme 23** : Algorithme de principe l'algorithme hybride du recuit simulé

---

**Données** : Température  $T_0$ , Facteur de décroissance  $\alpha$ , Nombre maximal d'itérations  $IterMax$ , Solution initiale  $Y \in \Omega'$

```

1  $iter := 0, T := T_0$ 
2  $X := L(Y)$  : un algorithme de liste est appliqué à la liste  $Y$ 
3 Solution record  $RY := Y, RX := X$ 
4 tant que  $iter < IterMax$  faire
5   Choisir uniformément et aléatoirement  $Y' \in V(Y)$ 
6    $X' := L(Y')$ 
7   si  $H(X') < H(RX)$  alors
8      $RY := Y'$ 
9      $RX := X'$ 
10  si  $H(X') \leq H(X)$  alors
11     $Y := Y'$ 
12     $X := X'$ 
13  sinon
14     $Y := Y'$  et  $X := X'$  avec la probabilité  $e^{-\frac{H(X')-H(X)}{T}}$ 
15   $iter := iter + 1$ 
16  Générer une nouvelle température  $T := \alpha \times T$ 

```

---

#### 4.3.2.4 Algorithme du kangourou

L'Algorithme 24 décrit l'algorithme du kangourou. Le palier  $A$  avant le saut est déterminé de la même façon que précédemment grâce à l'Équation (4.4). Le système de voisinage  $W$  consiste ici aussi en l'application un certain nombre de fois du système de voisinage utilisé  $V$ , il respecte donc la propriété d'accessibilité.

### 4.3.3 Métaheuristique basée population

Cette méthode hybride est testée avec l'application de la PSO combinatoire. Le principe général de la méthode est présenté dans un premier temps. Deux méthodes pour le déplacement de la particule ont été développées et sont détaillées par la suite.

Les notations utilisées pour la PSO sont complétées, dû au fait que la méthode s'applique dans les deux espaces  $\Omega'$  et  $S$ . À l'instant  $t$ ,

- Chaque particule  $p$  a
  - une position courante  $Y_{p,t} \in \Omega'$ ,
  - la valeur de sa fonction objectif  $H(X_{p,t})$ , déterminée après application de l'algorithme de liste  $L$  à la position  $Y_{p,t}$ ,  $X_{p,t} = L(Y_{p,t})$ ,
  - sa meilleure position  $P_{p,t} \in \Omega'$ ,
  - la valeur de la fonction objectif de sa meilleure position  $H(L(P_{p,t}))$ , déterminée après application de l'algorithme de liste  $L$  à la position  $P_{p,t}$ .
- La meilleure position de l'essaim est notée  $G_t \in \Omega'$ , la valeur de sa fonction objectif est notée  $H(L(G_t))$ .

---

**Algorithme 24 :** Algorithme de principe de l'algorithme hybride du kangourou

---

**Données :** Nombre d'itérations  $A$  avant un saut, Nombre maximal d'itérations  
 $IterMax$ , Solution initiale  $Y \in \Omega'$

```

1   $iter := 0, c := 0$ 
2   $X := L(Y)$  : un algorithme de liste est appliqué à la liste  $Y$ 
3  Solution record  $RY := Y, RX := X$ 
4  tant que  $iter < IterMax$  faire
5      si  $c < A$  alors
6          Choisir uniformément et aléatoirement  $Y' \in V(Y)$ 
7           $X' := L(Y')$ 
8          si  $H(X') \leq H(X)$  alors
9              si  $H(X') < H(X)$  alors
10                  $c := 0$ 
11                 si  $H(X') < H(RX)$  alors
12                      $RY := Y'$ 
13                      $RX := X'$ 
14              $Y := Y'$ 
15              $X := X'$ 
16         sinon
17             Choisir uniformément et aléatoirement  $Y' \in W(Y)$ 
18              $X' := L(Y')$ 
19             si  $H(X') \neq H(X)$  alors
20                 si  $H(X') < H(RX)$  alors
21                      $RY := Y'$ 
22                      $RX := X'$ 
23                  $c := 0$ 
24              $Y := Y'$ 
25              $X := X'$ 
26          $c := c + 1$ 
27      $iter := iter + 1$ 

```

---

Les différents ensembles, fonctions et variables sont résumés ci-dessous :

	$\Omega'$		$S$		$\mathbb{R}$
Position courante	$Y_{p,t}$	$\longrightarrow$	$X_{p,t} = L(Y_{p,t})$	$\longrightarrow$	$H(X_{p,t})$
Meilleure position personnelle	$P_{p,t}$	$\longrightarrow$	$L(P_{p,t})$	$\longrightarrow$	$H(L(P_{p,t}))$
Meilleure particule de l'essaim	$G_t$	$\longrightarrow$	$L(G_t)$	$\longrightarrow$	$H(L(G_t))$

Ainsi, dans le cas hybride, le passage de  $t$  à  $t + 1$  est défini par le Système d'Équations (4.8).

$$\begin{cases} Y_{p,t+1} &= Y_{p,t} \oplus V_{p,t+1} \\ V_{p,t+1} &= c_1 \otimes V_{p,t} \oplus c_2 \cdot r_2 \otimes (P_{p,t} \ominus Y_{p,t}) \oplus c_3 \cdot r_3 \otimes (G_t \ominus Y_{p,t}) \end{cases} \quad (4.8)$$

### 4.3.3.1 Principe général

L'initialisation de l'essaim de particules est défini par l'Algorithme 25. À l'état initial  $t = 0$ , pour chaque particule  $p$ , la liste d'objets  $Y_{p,0}$  est générée aléatoirement. L'heuristique de construction  $L$  affecte les objets de la liste  $Y_{p,0}$  aux boîtes, cette affectation est représentée par la solution  $X_{p,0}$ . La valeur de la fonction objectif  $H(X_{p,0})$  est déterminée. La meilleure position de la particule  $P_{p,0}$  est enregistrée. La meilleure position de l'essaim  $G_0$  est ensuite déterminée.

L'Algorithme 26 donne l'algorithme de principe de la PSO combinatoire hybridée. Tant que le critère d'arrêt n'est pas atteint, à chaque itération, un déplacement de particule est réalisé : la nouvelle position  $Y_{p,t+1}$  de chaque particule de l'essaim est déterminée ; la valeur de la fonction coût  $H(X_{p,t+1})$  est évaluée après l'application de l'algorithme de liste  $L$  ; les nouvelles meilleures positions de chaque particule  $P_{p,t+1}$  sont déterminées ; la position  $G_{t+1}$  est déterminée comme étant la meilleure des  $P_{p,t+1}, \forall p \in \{1, P\}$ .

---

#### Algorithme 25 : Initialisation de l'essaim de particules

---

**Données :**  $P$  nombre de particules

1 **pour toutes les Particules  $p$  de 1 à  $P$  faire**

- |   |  |
|---|--|
| 2 | Initialisation aléatoire de la position $Y_{p,0}$ et vitesse $V_{p,0}$ de la particule $p$ |
| 3 | Application de l'algorithme de liste à la position courante $X_{p,0} := L(Y_{p,0})$        |
| 4 | Enregistrement de la meilleure position de la particule : $P_{p,0} := Y_{p,0}$             |

5 **Détermination de la meilleure particule de l'essaim :**

$$G_0 := Y_{p,0} / H(L(G_0)) = \min_{p'=1,P} \{H(L(Y_{p',0}))\}$$


---

Afin d'éviter la convergence trop rapide de l'essaim vers un minimum local, deux perturbations sont proposées.

**Mutation et dispersion d'essaim (PSOM & PSOMD)** Afin d'éviter une convergence trop rapide de l'essaim, nous proposons d'adapter l'opérateur mutation à l'optimisation par essaim particulaire. Avec une probabilité donnée, la particule est mutée, c'est-à-dire un objet de la liste  $Y_{p,t}$  est déplacée à une position aléatoire de la liste. L'algorithme de principe est donné par l'Algorithme 27.

---

**Algorithme 26** : Principe de la PSO combinatoire hybride

---

**Données** :  $P$  nombre de particules

- 1 Initialisation de l'essaim de particules
  - 2 **tant que** *Critère d'arrêt non atteint* **faire**
  - 3     **pour toutes les** *Particules*  $p$  **de** 1 à  $P$  **faire**
  - 4          $Y_{p,t+1}$  : Déplacement de la particule
  - 5         Application de l'algorithme de liste à la position courante :  
 $X_{p,t+1} := L(Y_{p,t+1})$
  - 6         Détermination de la meilleure position de la particule :  $P_{p,t+1} := Y_{p,t+1}$  si  
 $H(X_{p,t+1}) \leq H(L(P_{p,t+1}))$
  - 7     Détermination de la meilleure particule de l'essaim :  
 $G_{t+1} := Y_{p,t+1} / H(L(G_{t+1})) = \min_{p'=1,P} \{H(L(Y_{p',t+1}))\}$
  - 8      $t := t + 1$
- 

---

**Algorithme 27** : PSOM Mutation

---

**Données** :  $P$  nombre de particules

- 1 Initialisation de l'essaim de particules
  - 2 **tant que** *Critère d'arrêt non atteint* **faire**
  - 3     **pour toutes les** *Particules*  $p$  **de** 1 à  $P$  **faire**
  - 4          $Y_{p,t+1}$  : Déplacement de la particule
  - 5         **si** *Probabilité de mutation* **alors**
  - 6             ┌ Un objet de  $Y_{p,t+1}$  est déplacé à une position aléatoire de la liste
  - 7             └ Application de l'algorithme de liste à la position courante
  - 8             └ Détermination de la meilleure position de la particule
  - 9     Détermination de la meilleure particule de l'essaim
  - 10      $t := t + 1$
-

En plus de subir une mutation, l'essaim peut être dispersé. La dispersion d'essaim est inspirée des tests de NoHope et des facteurs de Rehope défini par (Clerc, 1999). Après un nombre donné d'itérations sans amélioration de la solution courante, l'évolution des particules est guidée. En particulier, le dernier point du test de NoHope stipule que lorsqu'après un certain nombre d'itérations fixé à l'avance, l'algorithme de la PSO ne parvient plus à améliorer la solution courante, une partie de l'essaim est réinitialisée aléatoirement via sa position et sa vitesse.

Malgré la mutation de particule, s'il n'y a pas d'amélioration de la solution courante après un certain nombre d'itérations, l'essaim est réinitialisé aléatoirement. L'algorithme de principe est donné par l'Algorithme 28.

---

**Algorithme 28** : PSOMD Mutation et Dispersion

---

**Données** :  $P$  nombre de particules,  
 Nombre d'itérations  $A$  avant une réinitialisation

```

1  $c := 0$ 
2 Initialisation de l'essaim de particules
3 tant que Critère d'arrêt non atteint faire
4   si  $c < A$  alors
5     pour toutes les Particules  $p$  de 1 à  $P$  faire
6        $Y_{p,t+1}$  : Déplacement de la particule
7       si Probabilité de mutation alors
8          $\lfloor$  Un objet de  $Y_{p,t+1}$  est déplacé à une position aléatoire de la liste
9         Application de l'algorithme de liste à la position courante
10         $\lfloor$  Détermination de la meilleure position de la particule
11        Détermination de la meilleure particule de l'essaim
12         $t := t + 1$ 
13        si  $G_{t+1} \neq G_t$  alors
14           $\lfloor$   $c := 0$ 
15      sinon
16        Initialisation de l'essaim de particules
17         $c := 0$ 
18     $c := c + 1$ 
    
```

---

**Dispersion avec influence du Record (PSOR)** Dans cette nouvelle perturbation, la dispersion d'essaim permet de générer une quatrième influence : la position record. La position record est enregistrée avant chaque nouvelle réinitialisation d'essaim. En plus de subir les trois influences définies par sa propre vitesse, sa meilleure position et la meilleure position de l'essaim, la particule subit l'influence définie par la meilleure position record  $R$  enregistrée lors de la dispersion de l'essaim.

L'algorithme de la dispersion de l'essaim est décrit par l'Algorithme 29. Lorsque le nombre d'itérations sans amélioration atteint un palier défini, la meilleure solution record est enregistrée dans la position  $R_t$  puis l'essaim est généré à nouveau aléatoirement. À l'instant  $t + 1$ , le déplacement de la particule  $p$  est ainsi influencé par sa vitesse  $V_{p,t}$ , sa

meilleure position  $P_{p,t}$ , la meilleure position de l'essaim  $G_t$  et la meilleure position record  $R_t$  actualisée à chaque dispersion d'essaim. Le déplacement de particule est ainsi défini par le Système d'Équations (4.9).

$$\begin{cases} Y_{p,t+1} &= Y_{p,t} \oplus V_{p,t+1} \\ V_{p,t+1} &= c_1 \otimes V_{p,t} \oplus c_2 \cdot r_2 \otimes (P_{p,t} \ominus Y_{p,t}) \oplus c_3 \cdot r_3 \otimes (G_t \ominus Y_{p,t}) \oplus c_4 \cdot r_4 \otimes (R_t \ominus Y_{p,t}) \end{cases} \quad (4.9)$$

---

**Algorithme 29** : PSOR dispersion avec influence du Record

---

**Données** :  $P$  nombre de particules,  
Nombre d'itérations  $A$  avant une réinitialisation

```

1  $c := 0$ 
2 Initialisation de l'essaim de particules
3 tant que Critère d'arrêt non atteint faire
4   si  $c < A$  alors
5     pour toutes les Particules  $p$  de 1 à  $P$  faire
6        $Y_{p,t+1}$  : Déplacement de la particule
7       Application de l'algorithme de liste à la position courante
8       Détermination de la meilleure position de la particule
9     Détermination de la meilleure particule de l'essaim
10     $t := t + 1$ 
11    si  $G_{t+1} \neq G_t$  alors
12       $c := 0$ 
13  sinon
14    Enregistrement de la meilleure position  $R = G_t$ 
15    Initialisation de l'essaim de particules
16     $c := 0$ 
17   $c := c + 1$ 

```

---

Deux déplacements de particule sont proposés et détaillés :

- Le premier passe par la définition des opérateurs,
- Le second fait une interprétation de l'influence de la particule par les trois facteurs, de façon à simplifier les calculs.

#### 4.3.3.2 Déplacement 1

Le premier déplacement, noté *PSO1*, consiste en l'application à chaque particule du Système d'Équations (4.8). Chaque opérateur est défini de la façon suivante.

##### Vitesse

Le vecteur vitesse  $V_{p,t+1}$ , une fois appliqué à une position  $Y_{p,t}$ , donne une autre

position  $Y_{p,t+1}$ . Il correspond à la liste des déplacements d'objets à effectuer pour passer de la liste définie par  $Y_{p,t}$  à celle définie par  $Y_{p,t+1}$ .

Plusieurs déplacements d'objets dans la liste sont envisageables :

- $P_i$  : l'objet au rang  $i$  est permuté avec l'objet au rang  $i + 1$ .
- $P_{i,j}$  : l'objet au rang  $i$  est permuté avec celui se trouvant au rang  $j$ .
- $I_{i,j}$  : l'objet au rang  $i$  est inséré au rang  $j$ .

Par abus de langage, on appelle longueur du vecteur  $V_{p,t}$  le nombre de couples de ce vecteur, notée  $|V_{p,t}|$ .

### Déplacement (addition) "position $\oplus$ vitesse"

L'addition d'une position et d'une vitesse déplace la particule dans l'espace de recherche. C'est l'application des déplacements définis par la vitesse sur la position courante.

Soient la position  $Y_{p,t} = (1, 2, 3, 4, 5, 6)$ , une vitesse  $V_{p,t+1}$  et la nouvelle position  $Y_{p,t+1} = Y_{p,t} \oplus V_{p,t+1}$ .

- Si la vitesse est une liste de permutations  $P_i$ , par exemple  $V_{p,t+1} = (1, 3, 4)$  alors  $Y_{p,t+1} = (2, 1, 4, 5, 3, 6)$ .
- Si la vitesse est une liste de permutations  $P_{i,j}$ , par exemple  $V_{p,t+1} = ((1, 3), (2, 3), (4, 5))$  alors  $Y_{p,t+1} = (3, 1, 2, 5, 4, 6)$ .
- Si la vitesse est une liste d'insertions  $I_{i,j}$ , par exemple  $V_{p,t+1} = ((1, 3), (2, 3), (4, 5))$  alors  $Y_{p,t+1} = (2, 1, 3, 5, 4, 6)$ .

L'Algorithme 30 définit le déplacement  $Y_{p,t+1} = Y_{p,t} \oplus V_{p,t+1}$ .

---

#### Algorithme 30 : Déplacement $PSO1$ ( $Y_{p,t} \oplus V_{p,t+1}$ )

---

**Données** :  $Y_{p,t}, V_{p,t+1}$

**Résultat** :  $Y_{p,t+1} = Y_{p,t} \oplus V_{p,t+1}$

```

1  $Y_{p,t+1} := Y_{p,t}$ 
2 pour  $k$  de 1 à  $|V_{p,t+1}|$  faire
3   si  $V$  est une liste de permutations  $P_k$  alors
4      $L$ 'objet au rang  $k$  de  $Y_{p,t+1}$  est permuté avec celui au rang  $k + 1$  de  $Y_{p,t+1}$ 
5   sinon
6     Soit  $(i, j) = V_{p,t+1}(k)$ 
7     si  $V$  est une liste de permutations  $P_{i,j}$  alors
8        $L$ 'objet au rang  $i$  de  $Y_{p,t+1}$  est permuté avec celui au rang  $j$  de  $Y_{p,t+1}$ 
9     si  $V$  est une liste d'insertions  $I_{i,j}$  alors
10     $L$ 'objet au rang  $i$  de  $Y_{p,t+1}$  est inséré au rang  $j$  de  $Y_{p,t+1}$ 

```

---

### Soustraction "position $\ominus$ position"

Soient  $Y_{p,t}^{(1)}$  et  $Y_{p,t}^{(2)}$  deux positions. La différence  $Y_{p,t}^{(2)} \ominus Y_{p,t}^{(1)}$  est définie telle que

la vitesse  $V_{p,t}^{(1)}$ , trouvée par un algorithme donné, appliquée à la position  $Y_{p,t}^{(1)}$  donne la position  $Y_{p,t}^{(2)}$ . L'algorithme utilisé est celui déterminant les déplacements nécessaires pour passer de la position  $Y_{p,t}^{(1)}$  à la position  $Y_{p,t}^{(2)}$ . Dans le cas de la permutation  $P_{i,j}$ , l'objet courant au rang  $i$  dans  $Y_{p,t}^{(1)}$  est permuté avec celui se trouvant au rang  $j$  de  $Y_{p,t}^{(1)}$ ,  $j$  étant le rang occupé par l'objet courant dans  $Y_{p,t}^{(2)}$ . Les déplacements  $P_i$  et  $I_{i,j}$  sont aussi testés.

Par exemple, avec la permutation  $P_{i,j}$ , si  $Y_{p,t}^{(2)} = (3, 1, 2, 5, 4, 6)$  et  $Y_{p,t}^{(1)} = (1, 2, 3, 4, 5, 6)$  alors  $Y_{p,t}^{(2)} \ominus Y_{p,t}^{(1)} = ((1, 2), (1, 3), (4, 5))$  car il faut appliquer les permutations successives  $(1, 2)$  puis  $(1, 3)$  puis  $(4, 5)$  pour passer de la position  $(1, 2, 3, 4, 5, 6)$  à la position  $(3, 1, 2, 5, 4, 6)$ .

La condition "trouvée par un algorithme donné" est nécessaire. En effet, deux vitesses différentes appliquées à la même position peuvent donner la même nouvelle position. En particulier, l'algorithme est choisi de telle façon que si  $Y_{p,t}^{(1)} = Y_{p,t}^{(2)} \Rightarrow V_{p,t}^{(1)} = Y_{p,t}^{(2)} \ominus Y_{p,t}^{(1)} = \emptyset$ .

Par exemple, la vitesse  $V_{p,t}^{(2)} = ((1, 3), (2, 3), (4, 5))$  appliquée à la même position  $Y_{p,t}^{(1)}$  donne la même position  $Y_{p,t}^{(2)}$ .

L'Algorithme 31 définit la soustraction  $Y_{p,t}^{(2)} \ominus Y_{p,t}^{(1)}$ .

---

**Algorithme 31** : Soustraction ( $Y_{p,t}^{(2)} \ominus Y_{p,t}^{(1)}$ )
 

---

**Données** :  $Y_{p,t}^{(1)}, Y_{p,t}^{(2)}$

**Résultat** :  $V_{p,t} = Y_{p,t}^{(2)} \ominus Y_{p,t}^{(1)}$

```

1  k := 1
2  pour tous les i ∈ N faire
3      Trouvé := faux
4      j := 1
5      tant que Trouvé = faux faire
6          si Yp,t(1)(i) = Yp,t(2)(j) alors
7              Trouvé := vrai
8              si i ≠ j alors
9                  Enregistrement de la permutation correspondante dans Vp,t
10                 Vp,t(k) := (i, j)
11                 L'objet au rang i de Yp,t(1) est permuté avec celui au rang j de Yp,t(1)
12                 k := k + 1
13             j := j + 1
    
```

---

**Addition "vitesse  $\oplus$  vitesse"**

Soient  $V_{p,t}^{(1)}$  et  $V_{p,t}^{(2)}$  deux vitesses. La vitesse  $V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$  est l'application des déplacements d'objets associés à  $V_{p,t}^{(1)}$  puis l'application des déplacements associés à  $V_{p,t}^{(2)}$ .

Par exemple, si  $V_{p,t}^{(1)} = ((1, 3), (2, 3), (4, 5))$  et  $V_{p,t}^{(2)} = (5, 6)$  alors  $V_{p,t}^{(1)} \oplus V_{p,t}^{(2)} = ((1, 3), (2, 3), (4, 5), (5, 6))$ .



L'Algorithme 32 définit l'addition entre les vitesses  $V_{p,t}^{(1)}$  et  $V_{p,t}^{(2)}$ .

---

**Algorithme 32** : Addition ( $V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$ )

---

**Données** :  $V_{p,t}^{(1)}, V_{p,t}^{(2)}$   
**Résultat** :  $V_{p,t}^{total} = V_{p,t}^{(1)} \oplus V_{p,t}^{(2)}$

- 1 **pour tous les**  $k$  **de** 1 **à**  $|V_{p,t}^{(1)}|$  **faire**
- 2      $V_{p,t}^{total}(k) := V_{p,t}^{(1)}(k)$
- 3 **pour tous les**  $k$  **de**  $|V_{p,t}^{(1)}| + 1$  **à**  $|V_{p,t}^{(2)}| + |V_{p,t}^{(1)}|$  **faire**
- 4      $V_{p,t}^{total}(k) := V_{p,t}^{(2)}(k - |V_{p,t}^{(1)}|)$

---

**Multiplication "coefficient  $\otimes$  vitesse"**

Soient  $c$  un coefficient réel et  $V_{p,t}$  une vitesse. En fonction de la valeur de  $c$ , la multiplication  $c \otimes V_{p,t}$  est définie par :

- Si  $c = 0$ , alors  $c \otimes V_{p,t} = \emptyset$ .
- Si  $c \in ]0, 1[$ , alors  $V_{p,t}$  est tronqué à l'entier  $E(c \cdot |V_{p,t}|)$  tel que  $E(x) \leq x < E(x) + 1$ . Les  $E(c \cdot |V_{p,t}|)$  premiers déplacements du vecteur  $V_{p,t}$  sont appliqués.
- Si  $c \geq 1$ , alors tous les déplacements définis par  $V_{p,t}$  plus  $E((c - 1) \cdot |V_{p,t}|)$  fois les premiers déplacements du vecteur  $V_{p,t}$  sont effectués.

Par exemple, avec  $V_{p,t} = ((1, 3), (2, 3), (4, 5), (5, 6))$ ,  $|V_{p,t}| = 4$ .

- Si  $c = 0.4$  alors  $c \otimes V_{p,t} = (1, 3)$ .
- Si  $c = 1.4$  alors  $c \otimes V_{p,t} = ((1, 3), (2, 3), (4, 5), (5, 6), (1, 3))$ .

L'Algorithme 33 décrit la multiplication de la vitesse  $V_{p,t}$  par le coefficient  $c$ .

---

**Algorithme 33** : Multiplication ( $c \otimes V_{p,t}$ )

---

**Données** :  $c, V_{p,t}$   
**Résultat** :  $V'_{p,t} = c \otimes V_{p,t}$

- 1 **si** ( $c = 0$ ) **alors**
- 2      $V'_{p,t} := 0$
- 3 **si**  $c \in ]0, 1[$  **alors**
- 4     **pour tous les**  $k$  **de** 1 **à**  $E(c \cdot |V_{p,t}|)$  **faire**
- 5          $V'_{p,t}(k) := V_{p,t}(k)$
- 6 **si**  $c \in [1, 2]$  **alors**
- 7     **pour tous les**  $k$  **de** 1 **à**  $E(c \cdot |V_{p,t}|)$  **faire**
- 8         **si**  $k \leq |V_{p,t}|$  **alors**
- 9              $V'_{p,t}(k) := V_{p,t}(k)$
- 10         **sinon**
- 11              $V'_{p,t}(k) := V_{p,t}(|V_{p,t}| - k)$

---

### 4.3.3.3 Déplacement 2

Lors du déplacement d'une particule dans la version *PSO1*, chaque objet de la liste définie par la position de la particule est influencé par une combinaison entre :

- la vitesse courante de la particule,
- la meilleure position de la particule,
- la meilleure position de l'essaim.

Dans le but d'alléger les calculs, nous proposons ici un nouveau déplacement de particule, noté *PSO2*, qui n'utilise pas la notion de vitesse. En fonction de la valeur des coefficients  $c_2, c_3$  et d'un nombre choisi aléatoirement et uniformément  $r \in [0, 1[$ , chaque objet de la liste occupera soit le rang occupé dans la position courante, soit le rang occupé dans la meilleure position personnelle, soit le rang occupé dans la meilleure position de l'essaim :

- Si  $r < c_2$ , l'objet courant occupera le rang qu'il occupe dans sa meilleure position,
- Si  $c_2 \leq r < c_2 + c_3$  : l'objet courant occupera le rang qu'il occupe dans la meilleure position de l'essaim,
- Sinon, l'objet courant n'est pas influencé, il reste à son rang actuel.

En reprenant les mêmes notations, le pseudo-code du déplacement d'une particule *PSO2* est donné par l'Algorithme 34.

### 4.3.4 Conclusion du couplage hiérarchique

Un nouveau codage inspiré des problèmes d'ordonnement de permutation permet d'utiliser un couplage hiérarchique entre une métaheuristique et une heuristique. Grâce à ce couplage, la métaheuristique est appliquée sur un espace de recherche réduit qui contient l'ensemble des solutions optimales. L'utilisation d'un algorithme de construction assure l'admissibilité des solutions générées. Cette méthode est une combinaison de méthodes existantes et simples qui sont faciles à implémenter.

Un plus large spectre de problèmes peut être considéré avec la méthode couplée. En effet, les contraintes au problème sont vérifiées par l'heuristique lors de la construction de la solution à chaque étape de la métaheuristique. Ainsi, la méthode couplée peut être adaptée pour la résolution du problème de bin packing avec incompatibilités par une modification de l'algorithme de construction.

## 4.4 Extension des méthodes pour le bin packing avec incompatibilités

Nos méthodes sont maintenant étendues pour la résolution du problème du bin packing avec incompatibilités, c'est-à-dire le problème de planification d'activités avec affectation de ressources. L'application des métaheuristicques à la résolution de ce problème a en partie été publiée dans (Gourgand et al., 2014a). La présence d'incompatibilités entre les activités et les ressources implique quelques modifications dans les méthodes précédemment présentées, notamment concernant les algorithmes de liste, les systèmes de voisinage et le calcul des critères.

---

**Algorithme 34** : Déplacement d'une particule *PSO2*

---

**Données** :  $Y_{p,t}, P_{p,t}, G_t, c_2, c_3$ , instant  $t$   
**Résultat** :  $Y_{p,t+1}$

```

1  pour tous les  $i \in \{1, N\}$  faire
2      Tirage aléatoire et uniforme de  $r \in [0, 1[$ 
3      si  $r < c_2$  alors
4          L'objet au rang  $i$  est influencé par sa meilleure position personnelle
5           $Trouvé := faux$ 
6           $j := 0$ 
7          tant que  $Trouvé = faux$  faire
8               $j := j + 1$ 
9              si  $Y_{p,t}(i) = P_{p,t}(j)$  alors
10                 Recherche dans  $P_{p,t}$  du rang  $j$  occupé par  $Y_{p,t}(i)$ 
11                  $Trouvé := vrai$ 
12             L'objet au rang  $i$  de  $Y_{p,t}$  est permuté avec celui au rang  $j$  de  $Y_{p,t}$ 
13         si  $c_2 \leq r < c_2 + c_3$  alors
14              $Trouvé := faux$ 
15              $j := 0$ 
16             tant que  $Trouvé = faux$  faire
17                  $j := j + 1$ 
18                 si  $Y_{p,t}(i) = G_t(j)$  alors
19                      $Trouvé := vrai$ 
20             L'objet au rang  $i$  de  $Y_{p,t}$  est permuté avec celui au rang  $j$  de  $Y_{p,t}$ 
21         sinon
22              $\lfloor$  L'objet au rang  $i$  ne bouge pas

```

---

### 4.4.1 Codage

Une boîte est un couple (équipement  $l$ , période  $t$ ). Une activité  $i$  est affectée à un équipement  $l$  pendant une période  $t$  par la variable  $x_{i,l,t}$ .

$$x_{i,l,t} = \begin{cases} 1 & \text{si l'activité } i \in N \text{ est affectée à l'équipement } l \in L \\ & \text{pendant la période } t \in T, \\ 0 & \text{sinon.} \end{cases}$$

### 4.4.2 Proposition d'algorithmes de liste

Nous proposons d'appliquer les algorithmes de liste existant pour le problème du bin packing de base au problème du bin packing avec incompatibilités.

#### 4.4.2.1 Next Fit CHT

L'adaptation de Next Fit consiste à affecter le maximum d'activités au couple  $(l, t)$  courant. Si l'espace encore disponible dans ce couple n'est pas suffisant pour l'activité courante, le couple est définitivement fermé, il est impossible de placer une activité dans ce couple par la suite. Un nouveau couple est alors ouvert et devient le couple courant. La compatibilité entre l'activité et le couple doit être respectée. On souhaite réaliser les activités le plus tôt possible donc on examine tous les équipements d'une période avant de passer à la période suivante.

L'Algorithme 35 présente cette heuristique. On place l'activité courante dans le couple  $(l, t)$  courant à condition que le temps disponible dans ce couple soit supérieur au temps de traitement  $t_i$  de l'activité  $i$  et que l'activité  $i$  soit compatible avec l'équipement  $l$ , c'est-à-dire  $c_{i,l} = 1$ .

À cause des incompatibilités, il y a autant de couples ouverts que d'équipements. En effet, une activité peut ne pas être affectée à un couple à cause de sa compatibilité, alors que l'activité suivante pourrait y être affectée. Si un couple était fermé quand l'activité courante n'est pas compatible, beaucoup de couples seraient potentiellement vides.

#### 4.4.2.2 First Fit CHT

L'adaptation de First Fit diffère de l'heuristique précédente dans le sens où les activités sont placées dans le premier couple capable de les recevoir. Un nouveau couple est considéré si et seulement si tous les couples précédents ont été examinés et qu'aucun d'entre eux n'a assez d'espace disponible pour recevoir l'activité courante. Aucun couple n'est définitivement fermé. L'Algorithme 36 présente cette heuristique.

#### 4.4.2.3 Best Fit CHT

L'adaptation de Best Fit consiste à placer l'activité courante dans le meilleur couple ayant assez d'espace disponible. Pour une activité, le meilleur couple est celui avec le moins d'espace encore disponible après l'affectation de cette activité. L'Algorithme 37 présente cette heuristique.

---

**Algorithme 35** : Algorithme de principe de l'heuristique Next Fit CHT

---

**Données** : liste d'activités  $(\sigma_i)_{i \in N}$  ;  $p_{l,t}, \forall l \in L, \forall t \in T$  ;  $t_i, \forall i \in N$

- 1  $F_{l,t} := 0, \forall l \in L, \forall t \in T$
- 2  $Fermé(l, t) := faux, \forall l \in L, \forall t \in T$
- 3 **pour tous les  $i$  de 1 à  $|N|$  faire**
- 4      $t := 1, l := 1, affecté := faux$
- 5     **tant que**  $(t \leq T) \wedge (affecté = faux)$  **faire**
- 6         **tant que**  $(l \leq L) \wedge (affecté = faux)$  **faire**
- 7             **si**  $Fermé(l, t) = faux$  **alors**
- 8                 **si** l'activité  $\sigma_i$  est compatible avec l'équipement  $l$  ( $c_{\sigma_i, l} = 1$ ) **alors**
- 9                     **si** l'activité  $\sigma_i$  tient dans le couple  $(l, t)$  ( $F_{l,t} + t_{\sigma_i} \leq p_{l,t}$ ) **alors**
- 10                         Affecter l'activité  $\sigma_i$  au couple  $(l, t)$  :
- 11                              $x_{\sigma_i, l, t} := 1$
- 12                              $F_{l,t} := F_{l,t} + t_{\sigma_i}$
- 13                              $affecté := vrai$
- 14                     **sinon**
- 15                          $Fermé(l, t) := vrai$
- 16              $l := l + 1$
- 17      $t := t + 1$

---



---

**Algorithme 36** : Algorithme de principe de l'heuristique First Fit CHT

---

**Données** : liste d'activités  $(\sigma_i)_{i \in N}$  ;  $p_{l,t}, \forall l \in L, \forall t \in T$  ;  $t_i, \forall i \in N$

- 1  $F_{l,t} := 0, \forall l \in L, \forall t \in T$
- 2 **pour tous les  $i$  de 1 à  $|N|$  faire**
- 3      $t := 1, l := 1, affecté := faux$
- 4     **tant que**  $(t \leq T) \wedge (affecté = faux)$  **faire**
- 5         **tant que**  $(l \leq L) \wedge (affecté = faux)$  **faire**
- 6             **si** l'activité  $\sigma_i$  est compatible avec l'équipement  $l$  ( $c_{\sigma_i, l} = 1$ ) **alors**
- 7                 **si** l'activité  $\sigma_i$  tient dans le couple  $(l, t)$  ( $F_{l,t} + t_{\sigma_i} \leq p_{l,t}$ ) **alors**
- 8                     Affecter l'activité  $\sigma_i$  au couple  $(l, t)$  :
- 9                          $x_{\sigma_i, l, t} := 1$
- 10                          $F_{l,t} := F_{l,t} + t_{\sigma_i}$
- 11                          $affecté := vrai$
- 12              $l := l + 1$
- 13      $t := t + 1$

---

---

**Algorithme 37** : Algorithme de principe de l'heuristique Best Fit CHT
 

---

**Données** : liste d'activités  $(\sigma_i)_{i \in N}$  ;  $p_{l,t}, \forall l \in L, \forall t \in T$  ;  $t_i, \forall i \in N$

- 1  $F_{l,t} := 0, \forall l \in L, \forall t \in T$
- 2  $F_{0,0} := 0, p_{0,0} := \max_{\forall l \in L, \forall t \in T} p_{l,t}$
- 3 **pour tous les**  $i$  **de** 1 à  $|N|$  **faire**
- 4      $l_{best} := 0, t_{best} := 0$
- 5     **pour tous les**  $t$  **de** 1 à  $|T|$  **faire**
- 6         **pour tous les**  $l$  **de** 1 à  $|L|$  **faire**
- 7             **si** l'activité  $\sigma_i$  est compatible avec l'équipement  $l$  ( $c_{\sigma_i,l} = 1$ ) **alors**
- 8                 **si**  $(F_{l,t} + t_{\sigma_i} \leq p_{l,t}) \wedge ((p_{l,t} - F_{l,t} - t_{\sigma_i}) \leq (p_{l_{best},t_{best}} - F_{l_{best},t_{best}} - t_{\sigma_i}))$
- 9                     **alors**
- 10                          $l_{best} := l, t_{best} := t$
- 11             Affectation de l'activité  $\sigma_i$  au couple  $(l_{best}, t_{best})$  :
- 12                  $x_{\sigma_i, l_{best}, t_{best}} := 1$
- 13                  $F_{l_{best}, t_{best}} := F_{l_{best}, t_{best}} + t_{\sigma_i}$

---

#### 4.4.2.4 Worst Fit CHT

L'adaptation de l'heuristique Worst Fit consiste à l'inverse à placer l'activité courante dans le pire couple, celui avec le moins d'espace encore disponible après l'affectation de cette activité. Il faut veiller à affecter une activité à un couple n'ayant encore aucune activité affectée si et seulement si aucun autre couple ne peut accueillir cette activité, sinon toutes les activités seraient affectées à un couple différent. L'Algorithme 38 présente cette heuristique.

#### 4.4.2.5 Algorithmes décroissants

Les algorithmes décroissants sont appliqués à une liste ordonnée d'activités. En considérant que le critère principal est le nombre d'activités placées avant leur date au plus tard, les activités ayant la plus petite date au plus tard sont les premières à être affectées aux couples, afin qu'elles soient effectuées en priorité. Le second critère de tri est le temps de traitement de chaque activité, ainsi les activités ayant un plus grand temps de traitement sont effectuées en priorité.

### 4.4.3 Système de voisinage

Le système de voisinage permet le passage d'une solution à une autre dans le cas des métaheuristiques basées individu. Le système de voisinage utilisé  $V$  est défini par l'Algorithme 39. Une activité  $i \in N$  est choisie aléatoirement. Son affectation courante est supprimée. Un équipement  $l \in L$  est choisi aléatoirement parmi ceux qui lui sont compatibles, tel que  $c_{i,l} = 1$ . Une période  $t \in T$  est choisie aléatoirement. L'activité  $i$  est affectée à l'équipement  $l$  pendant la période  $t$ . L'Équation (4.10) définit le cardinal du système de voisinage  $V$ . Ce système de voisinage ne garantit pas le respect des contraintes

---

**Algorithme 38** : Algorithme de principe de l'heuristique Worst Fit CHT

---

**Données** : liste d'activités  $(\sigma_i)_{i \in N}$  ;  $p_{l,t}, \forall l \in L, \forall t \in T$  ;  $t_i, \forall i \in N$

```

1  $F_{l,t} := 0, \forall l \in L, \forall t \in T$ 
2  $t_{max} := 1$ 
3 pour tous les  $i$  de 1 à  $|N|$  faire
4      $l_{best} := 0, t_{best} := 0$ 
5     pour tous les  $t$  de 1 à  $t_{max}$  faire
6         pour tous les  $l$  de 1 à  $|L|$  faire
7             si l'activité  $\sigma_i$  est compatible avec l'équipement  $l$  ( $c_{\sigma_i,l} = 1$ ) alors
8                 si  $(F_{l,t} + t_{\sigma_i} \leq p_{l,t}) \wedge ((p_{l,t} - F_{l,t} - t_{\sigma_i}) \leq (p_{l_{best},t_{best}} - F_{l_{best},t_{best}} - t_{\sigma_i}))$ 
9                     alors
10                          $l_{best} := l, t_{best} := t$ 
11         si Aucun couple ne peut recevoir l'activité  $\sigma_i$  ( $l_{best} = 0 \wedge t_{best} = 0$ ) alors
12             tant que ( $l_{best} = 0$ ) faire
13                  $t_{max} := t_{max} + 1$ 
14                  $t_{best} := t_{max}$ 
15                 Recherche d'un équipement  $l$  compatible :
16                  $l := 1$ 
17                 tant que ( $l \leq |L|$ ) faire
18                     si l'activité  $\sigma_i$  est compatible avec l'équipement  $l$  ( $c_{\sigma_i,l} = 1$ ) alors
19                          $l_{best} := l$ 
20                          $l := l + 1$ 
21 Affection de l'activité  $\sigma_i$  au couple  $(l_{best}, t_{best})$  :
22      $x_{\sigma_i, l_{best}, t_{best}} := 1$ 
23      $F_{l_{best}, t_{best}} := F_{l_{best}, t_{best}} + t_{\sigma_i}$ 

```

---

sur le temps d'ouverture.

$$\text{card}(V) \leq N.L.T \quad (4.10)$$

---

**Algorithme 39** : Système de voisinage  $V$

---

**Données** :  $c_{i,l}, \forall i \in N, \forall l \in L$

- 1 Choisir aléatoirement et uniformément  $i \in N$
  - 2 Soit  $l_1$  et  $t_1$  tel que  $x_{i,l_1,t_1} = 1$
  - 3  $x_{i,l_1,t_1} := 0$
  - 4 Choisir aléatoirement et uniformément  $l \in L$  tel que  $l$  est compatible avec  $i$  et  $l \neq l_1$
  - 5 Choisir aléatoirement et uniformément  $t \in T$  tel que  $t \neq t_1$
  - 6  $x_{i,l,t} := 1$
- 

Une illustration du système de voisinage est donnée par la Figure 4.3. L'affectation de l'activité 6 passe du couple  $(l_2, t_2)$  au couple  $(l_2, t_1)$ .

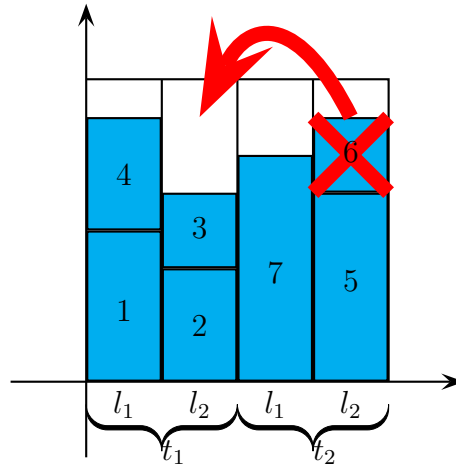


FIGURE 4.3 – Illustration du système de voisinage  $V$

Des systèmes de voisinage guidés peuvent également être envisagés. Par exemple, l'objet  $i$  peut être choisi parmi l'un des objets appartenant au couple  $(l, t)$  le moins plein et peut être affecté au couple compatible le plus plein ayant suffisamment d'espace libre pour le contenir.

#### 4.4.4 Pénalisation de contraintes

Comme pour la résolution du problème du bin packing de base, l'ensemble de toutes les solutions est noté  $\Omega$ . Le cardinal de  $\Omega$  est donné par l'Équation (4.11).

$$\text{card}(\Omega) = L^{N^T} \quad (4.11)$$

$C \in \Omega$  est l'ensemble des solutions admissibles. Une solution est admissible si elle respecte les deux contraintes suivantes :



- Les temps d’ouverture des équipements, selon l’Équation (2.6). L’hypothèse est faite que les temps d’ouverture supplémentaires des équipements ne sont pas autorisés.
- La compatibilité entre les activités et les équipements qui y sont affectées, selon l’Équation (2.2). La compatibilité entre les activités et les équipements est certes respectée par le système de voisinage, mais pas si la solution initiale est générée aléatoirement.

Les dépassements de contraintes sont calculés pour définir le nouveau critère  $Cont$ . Il est égal à la somme des :

- différences entre la somme des temps de traitement des activités affectées à l’équipement pendant la période moins le temps d’ouverture de cet équipement pendant cette période,
- et le nombre d’activités affectées à un équipement non compatible.

$Cont$  est calculé grâce aux contraintes définies par les Équations (2.2) et (2.6). Son calcul est défini par l’Équation (4.12). Si  $Cont$  est égal à zéro, alors toutes les contraintes sont respectées.

$$Cont(X) = \sum_{l \in L} \sum_{t \in T} \max(0, \sum_{i \in N} t_i \cdot x_{i,l,t} - p_{l,t}) + \sum_{i \in N} \sum_{l \in L} \sum_{t \in T} x_{i,l,t} \cdot (1 - c_{i,l}) \quad (4.12)$$

Ce critère est intégré à la fonction objectif en utilisant la méthode des critères pondérés.

#### 4.4.5 Méthode des critères pondérés

La méthode des critères pondérés (Coello, 2000) est utilisée. La fonction objectif est obtenue en sommant les différents critères avec un coefficient  $\omega_c$ . Le problème est écrit de la façon suivante :  $\min \sum_{c=1}^C \omega_c \cdot C_c$  avec  $C_c$  un critère et  $\omega_c$  le coefficient relatif. Les coefficients sont habituellement choisis tels que  $\sum_{c=1}^C \omega_c = 1$ .

De plus, l’ordre lexicographique permet de prendre en compte une préférence entre les objectifs (Ehr Gott, 1999). Ainsi, l’utilisateur peut choisir les critères qui sont les plus importants pour lui en changeant la valeur des différents coefficients.

La fonction objectif est une somme pondérée entre les trois critères à minimiser :

- $C_1$  : la période affectée à la dernière activité planifiée, selon l’Équation (2.12).
- $C_2$  : la somme des périodes affectées à toutes les activités, selon l’Équation (2.11).
- $Cont$  : le dépassement de contraintes défini par l’Équation (4.12) si l’ensemble  $\Omega$  de toutes les solutions est utilisé.

Puisque les critères  $C_1$  et  $C_2$  sont dépendants, ils ne sont pas minimisés conjointement. La fonction objectif peut donc être définie par l’Équation (4.13) ou par l’Équation (4.14).

$$\min Cont \cdot 10^{\omega_0} + C_1 \cdot 10^{\omega_1} \quad (4.13)$$

$$\min Cont \cdot 10^{\omega_0} + C_2 \cdot 10^{\omega_2} \quad (4.14)$$

Pour une meilleure lisibilité des critères,  $\omega_0$  est choisi tel que  $10^{\omega_0} > C_1 \cdot 10^{\omega_1}$  et  $\omega_1$  est choisi égal à 0. Ainsi, le critère  $Cont$  est minimisé en premier afin de donner priorité à

des solutions admissibles. Si  $C_2$  est utilisé à la place de  $C_1$ , le même principe s'applique.

Nous adaptons également la méthode epsilon contrainte (Haimès et al., 1971) en changeant certains critères en contraintes. En effet, les contraintes suivantes peuvent être relaxées :

- $C_3$  : le nombre d'activités réalisées après leur période limite, selon l'Équation (2.14).
- $C_4$  : le nombre d'activités non réalisées sur leur site de référence, selon l'Équation (2.15).

Les deux nouveaux critères sont pondérés de la même façon que les précédents.

Par exemple, la fonction objectif peut s'écrire de la façon suivante :

$$\min Cont.10^{\omega_0} + C_2.10^{\omega_2} + C_3.10^{\omega_3} + C_4.10^{\omega_4}$$

Si le résultat de cette fonction objectif est 120 034 018, en supposant que  $\omega_0 = 9, \omega_2 = 6, \omega_3 = 3, \omega_4 = 0$ , alors les valeurs des critères sont  $Cont = 0, C_2 = 120, C_3 = 34, C_4 = 18$ . Ainsi, la solution correspondante est admissible, la somme des périodes affectées aux activités vaut 120, 34 activités sont affectées après leur date limite et 18 activités sont affectées sur un autre site que leur site de référence.

## 4.5 Conclusion

Plusieurs méthodes de résolution du problème du bin packing ont été proposées dans ce chapitre.

Des métaheuristiques basées individu comme le recuit simulé et l'algorithme du kangourou sont utilisées dans un premier temps. Pour cela deux approches sont envisagées : la recherche peut être faite dans l'ensemble  $C$  des solutions admissibles ou dans l'ensemble  $\Omega$  de toutes les solutions. Une métaheuristique basée population est également appliquée : l'optimisation par essaim particulière. Cette méthode explore exclusivement l'ensemble  $\Omega$  de toutes les solutions. Pour toutes les métaheuristiques, si les solutions non admissibles sont envisagées, il faut cependant pénaliser le dépassement de contraintes pour minimiser les contraintes non respectées.

Un nouveau codage est ensuite proposé, inspiré des problèmes d'ordonnement de permutation. Ce codage nous permet l'application d'un couplage entre une métaheuristique et une heuristique. La métaheuristique intègre l'heuristique dans son processus. La recherche est faite sur une liste. L'heuristique est utilisée pour affecter les objets de la liste aux boîtes, afin de pouvoir calculer la valeur de la fonction objectif qui évalue la solution. Cette approche a plusieurs avantages :

- La recherche est réalisée dans un ensemble plus restreint que l'ensemble original,
- Les solutions générées sont toutes admissibles grâce à l'utilisation des heuristiques,
- Les solutions optimales sont accessibles.

De plus, un avantage non négligeable est la simplicité de la méthode proposée. En effet, elle couple une métaheuristique simple et connue de tous à une heuristique de base de résolution du problème du bin packing.

Ce couplage est appliqué avec plusieurs métaheuristiques : le recuit simulé, l'algorithme du kangourou et l'optimisation par essaim particulière. L'optimisation par essaim

particulière est étudiée plus en détail. Plusieurs stratégies de déplacement d'une particule sont proposées : la première approche utilise une définition classique des opérateurs alors que la seconde se base sur une interprétation du principe même de la PSO. Plusieurs améliorations sont également proposées afin d'éviter une convergence trop rapide vers des optimums locaux. Les notions de dispersion et de mutation sont adaptées à nos méthodes.

Toutes ces méthodes peuvent être appliquées au problème du bin packing avec incompatibilités. En fonction de la méthode utilisée, l'adaptation n'est pas la même. Si les métaheuristiques basées individu sont utilisées, le système de voisinage est redéfini : il faut s'assurer du respect de la contrainte d'incompatibilité lors de la génération du voisin. Si la recherche est effectuée dans l'ensemble  $\Omega$  de toutes les solutions, le calcul du dépassement de contraintes doit intégrer la contrainte d'incompatibilité.

L'application du couplage d'une heuristique avec une métaheuristique au problème du bin packing avec incompatibilités est simple à mettre en œuvre : il suffit d'utiliser une heuristique adéquate au problème. Les contraintes d'incompatibilités sont intégrées à cette heuristique.

Toutes ces méthodes sont testées dans le chapitre suivant. Le problème du bin packing est très étudié par la littérature, beaucoup d'instances sont disponibles. Ainsi, nos méthodes seront aisément testées afin de les appliquer dans un second temps au problème de planification d'activités avec affectation de ressources.

## Mise en œuvre et résultats

### Sommaire

---

<b>5.1</b>	<b>Introduction</b>	<b>136</b>
<b>5.2</b>	<b>Le problème du bin packing</b>	<b>136</b>
5.2.1	Instances	136
5.2.2	Métaheuristiques	140
5.2.3	Couplage heuristiques métaheuristiques	140
5.2.3.1	Métaheuristiques basées individu	140
5.2.3.2	Métaheuristiques basées population	141
5.2.4	Conclusion de la résolution du problème du bin packing	142
<b>5.3</b>	<b>Le modèle 1 de la CHT</b>	<b>142</b>
5.3.1	Création des instances	142
5.3.2	Résultats référence : CPLEX	143
5.3.3	Algorithmes de liste	144
5.3.4	Métaheuristiques	146
5.3.5	Couplage heuristiques métaheuristiques	146
5.3.5.1	Métaheuristiques basées individu	146
5.3.5.2	Métaheuristique basée population	147
<b>5.4</b>	<b>Les modèles avec ressources humaines</b>	<b>147</b>
5.4.1	Résolution par méthode exacte	147
5.4.2	Limitation des méthodes exactes	150
<b>5.5</b>	<b>Conclusion</b>	<b>150</b>

---

## 5.1 Introduction

Le problème du bin packing a beaucoup été étudié depuis son introduction en 1973. Il est encore étudié de nos jours. De nombreuses instances avec leurs résultats sont disponibles. Grâce à ce large benchmark, toute personne proposant de nouvelles méthodes peut se comparer avec les méthodes existantes. Dans ce chapitre, nous appliquons les méthodes proposées dans le chapitre précédent à ces instances afin de pouvoir évaluer nos propositions. Dans un premier temps, les méthodes développées pour le problème du bin de base sont testées. Dans un second temps, ces méthodes adaptées au problème du bin packing avec incompatibilités sont appliquées.

La Section 5.2 se concentre sur la résolution du problème du bin packing. En préambule, les instances de la littérature sont présentées. Les résultats des travaux cités dans le Chapitre 3 sur ces instances sont résumés, afin d’avoir une base de comparaison. Toutes nos méthodes sont ensuite appliquées : les métaheuristiques ainsi que le couplage des heuristiques et des métaheuristiques.

La Section 5.3 présente les résultats de l’application de ces mêmes méthodes à la résolution du problème de la CHT : problème de planification d’activités avec affectation de ressources qui est considéré comme un problème du bin packing avec incompatibilités. Les instances utilisées ont été générées par nos soins, de façon aléatoire mais représentent la réalité hospitalière. Ces instances sont dans un premier temps résolues par un solveur. Ces résultats sont utilisés comme référence pour la suite. Nos algorithmes de liste proposés dans le chapitre précédent sont appliqués à ces instances, ainsi que toutes les autres méthodes proposées et déjà testées sur les instances du bin packing.

La Section 5.4 considère les problèmes de la CHT avec prise en compte des ressources humaines. Ces problèmes, modélisés dans le Chapitre 2, ont été assimilés à des problèmes du bin packing avec interdépendance. Les modèles respectifs à ces problèmes n’ont pas été résolus de manière approchée. Ils ont cependant été étudiés grâce à un solveur, afin de tester leur difficulté et de prouver la nécessité d’adapter les méthodes précédemment proposées à ces modèles.

## 5.2 Le problème du bin packing

### 5.2.1 Instances

Les instances traitées dans la littérature sont utilisées pour tester et valider nos méthodes. Ces instances sont détaillées ci-dessous. Le Tableau 5.1 résume les instances utilisées par chacun des différents papiers de la littérature référencés dans le Chapitre 3.

- (Uniform)  $u_{120}$ ,  $u_{250}$ ,  $u_{500}$ ,  $u_{1000}$ . La capacité de la boîte est  $C = 150$ . Les tailles d’objets sont entières et uniformément distribuées entre 20 et 150. Quatre tailles de jeu de données existent : 120, 250, 500 et 1000 objets. Pour chaque taille de problème, 20 instances sont générées. Ces instances ont été générées par (Falkenauer, 1996) et résolues à l’optimalité.
- (Triplet)  $t_{60}$ ,  $t_{120}$ ,  $t_{249}$ ,  $t_{501}$ . La capacité de la boîte est égale à  $C = 1000$ . Les tailles d’objets sont entières et uniformément distribuées entre 250 et 500. Quatre tailles de jeu de données existent : 60, 120, 249 et 501 objets. Pour chaque taille de problème, 20 instances sont générées. Ces instances ont été générées par (Falkenauer,

- 1996). Les solutions optimales sont connues, il y a exactement trois objets par boîte.
- set\_1, set\_2, set\_3 :
    - set\_1 : 720 instances avec des tailles d’objets uniformément distribuées sur trois intervalles :  $[1, 100]$ ,  $[20, 100]$  et  $[30, 100]$ . La capacité de la boîte est égale à  $C = 100, 120$  et  $150$ . Il y a 50, 100, 200 et 500 objets.
    - set\_2 : 480 instances avec  $C = 1000$  et 50, 100, 200 et 500 objets. Chaque boîte a en moyenne 3.9 objets.
    - set\_3 : 10 instances avec  $C = 100000$  et 200 objets. Les tailles d’objets sont uniformément distribuées sur  $[20000, 35000]$ . Ce set est considéré comme le plus difficile des trois.
- Ces instances ont été générées par (Scholl et al., 1997) qui ont résolu 1184 des instances à l’optimalité. (Alvim et al., 2004) ont résolu les 26 autres instances.
- was\_1, was\_2 :
    - was\_1 : 100 instances avec  $C = 1000$  et 100 objets. La taille minimale d’un objet est 150 et la taille maximale est 200.
    - was\_2 : 100 instances avec  $C = 1000$  et 120 objets. La taille minimale d’un objet est 150 et la taille maximale est 200.
- Ces instances ont été générées par (Schwerin and Wäscher, 1997). Toutes ces instances ont été résolues à l’optimalité.
- gau\_1 : Ces 17 problèmes sont issus de (Wäscher and Gau, 1996). Ils sont réputés difficiles d’après leurs auteurs. Certains ont été résolus à l’optimalité. Pour les instances n’ayant pas été résolues à l’optimalité, nous prenons comme référence les résultats fournis par (Fleszar and Charalambous, 2011).

Peu de papiers ont essayé de résoudre l’intégralité des instances disponibles. Parmi les travaux cités, aucune métaheuristique basée individu n’a été testée sur les instances was\_1, was\_2 et gau\_1.

Le Tableau 5.2 présente les résultats trouvés par chacun des papiers référencés. Chaque valeur correspond au nombre de valeurs optimales trouvées pour chaque taille d’instance par chaque méthode. Les notations de chaque méthode sont les suivantes :

- $P - MBS' + VNS$  : (Fleszar and Hindi, 2002),
- $P - SAWMBS$  : (Fleszar and Charalambous, 2011),
- $HI\_BP$  : (Alvim et al., 2004),
- $WABP$  : (Loh et al., 2008),
- $HGGA$  : (Falkenauer, 1996),
- $HACO$  : (Levine and Ducatelle, 2004),
- AntPacking : (Brugger et al., 2004),
- $MGGA$  : (Bhatia and Basu, 2004),
- $H - SGGA$  : (Singh and Gupta, 2007),
- $ES$  : (Stawowy, 2008).

	u_120, u_250, u_500, u_1000	t_60, t_120, t_249, t_501	set_1, set_2, set_3	was_1, was_2, gau_1
Métaheuristiques basées individu				
(Fleszar and Hindi, 2002)	✓	✓	✓	
(Fleszar and Charalambous, 2011)	✓	✓	✓	✓
(Alvim et al., 2004)	✓	✓	✓	✓
(Loh et al., 2008)	✓	✓	✓	✓
Métaheuristiques basées population				
(Reeves, 1996)				
(Falkenauer, 1996)	✓	✓		
(Levine and Ducatelle, 2004)	✓			
(Brugger et al., 2004)	✓			
(Bhatia and Basu, 2004)	✓			
(Singh and Gupta, 2007)	✓	✓	✓	
(Stawowy, 2008)	✓	✓	✓	

TABLEAU 5.1 – Instances du problème du bin packing testées par la littérature

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1	TOTAL
Nombre	20	20	20	20	20	20	20	20	720	480	10	100	100	17	1587
Métaheuristiques basées individu															
<i>P – MBS' + VNS</i>	20	19	20	20	20	20	20	20	694	474	2	-	-	-	1329
<i>P – SAWMBS</i>	20	19	20	20	20	20	20	20	720	480	10	100	100	17	1586
<i>HI_BP</i>	20	20	20	20	20	20	20	20	720	480	10	100	100	12	1582
<i>WABP</i>	20	20	20	20	20	20	20	20	720	480	10	100	100	14	1584
Métaheuristiques basées population															
<i>HGGA</i>	18	17	20	20	18	20	20	20	-	-	-	-	-	-	153
<i>HACO</i>	20	18	20	20	-	-	-	-	-	-	-	-	-	-	78
AntPacking	20	20	20	20	-	-	-	-	-	-	-	-	-	-	80
<i>MGGA</i>	20	20	20	20	-	-	-	-	-	-	-	-	-	-	80
<i>H – SGGA</i>	20	18	20	20	6	0	1	2	717	479	10	-	-	-	1293
<i>ES</i>	20	20	20	20	20	20	20	20	680	478	10	-	-	-	1328

TABLEAU 5.2 – Résultats de la littérature sur les instances du problème du bin packing



Dans la suite, une analyse de toutes les méthodes est proposée. Les tableaux détaillés de tous les résultats sont donnés en Annexe A. Le paramétrage des méthodes a été présenté dans le Chapitre 4. Ce même paramétrage est utilisé tout au long de ce chapitre.

Pour toutes les méthodes testées sur les instances du bin packing, quatre réplifications ont été réalisées. Le temps de calcul est limité à une heure si une solution optimale n'a pas été trouvée dans cet intervalle de temps. Notre objectif est de proposer des méthodes qui donnent de bons résultats en un temps réduit.

Le critère  $f_{BPP}$  proposé par (Falkenauer and Delchambre, 1992) défini par l'Équation (3.12) est utilisé. L'objectif est la minimisation de l'opposé de ce critère.

## 5.2.2 Métaheuristiques

Les métaheuristiques semblent peu adaptées à la résolution du problème du bin packing. Leur application donne des résultats au pire des cas 10% moins bons que la solution optimale. Sur un total de 1587 instances testées, seules 724 ont été résolues à l'optimalité. Les instances résolues à l'optimalité par les métaheuristiques sont des instances de petite taille parmi les ensembles `u_120`, `set_1`, `set_2`, `was_1`, `was_2` et `gau_1`.

Les méthodes appliquées dans l'ensemble  $C$  des solutions admissibles donnent de meilleurs résultats que celles appliquées dans l'ensemble  $\Omega$  de toutes les solutions en un temps limité. Lorsque l'exploration est faite dans l'ensemble  $\Omega$  de toutes les solutions, la solution initiale est moins bonne, la méthode aurait besoin de plus de temps pour améliorer la solution courante.

## 5.2.3 Couplage heuristiques métaheuristiques

### 5.2.3.1 Métaheuristiques basées individu

Le couplage entre les heuristiques et les métaheuristiques basées individu donne de bons résultats. Sur les 1587 instances testées, 1420 ont été testées à l'optimalité. Les résultats obtenus sont au pire des cas 5% moins bons que la solution optimale. Le Tableau 5.3 résume pour chaque ensemble d'instance le meilleur résultat trouvé par un couplage heuristique métaheuristique basée individu : le nombre d'instances résolues à l'optimalité et, le cas échéant l'écart à l'optimal au pire des cas en pourcentage et en nombre de boîtes supplémentaires. Par exemple, pour l'ensemble `set_2`, la solution optimale pour l'instance "N2W1B2R5.BPP" est 34 mais nos méthodes ne trouvent pas mieux que 35, l'écart est de 2.94% (+1 boîte).

Tous les couplages possibles ont été testés. D'une façon générale, l'utilisation du recuit simulé est meilleure que celle de l'algorithme du kangourou. Les utilisations des heuristiques First Fit et Best Fit sont équivalentes, celle de l'heuristique Next Fit donne les moins bons résultats. Le système de voisinage permutation  $P_{i,j}$  donne de meilleurs résultats que le système de voisinage insertion.

Plus la taille des instances est importante, moins la solution optimale est trouvée. Par exemple, sur les mêmes jeux de données `u_120` à `u_1000`, toutes les instances de taille 120 sont résolues à l'optimalité mais seulement douze sur vingt le sont concernant les instances de taille 1000. Si le critère d'arrêt avait été un temps de calcul plus long, plus d'instances auraient été résolues. En effet, avec un temps limité à dix heures au lieu d'une, nos méthodes trouvent la solution optimale pour toutes les instances de taille 120 mais

Instance	Nombre d'optimaux	Pire écart % (boite supplémentaire)
u_120	20/20	-
u_250	16/20	0.98% (+1)
u_500	15/20	0.49% (+1)
u_1000	00/20	0.25% (+1)
t_60	07/20	5.00% (+1)
t_120	01/20	2.50% (+1)
t_249	00/20	1.20% (+1)
t_501	00/20	0.60% (+1)
set_1	703/720	2.33% (+1)
set_2	476/480	2.94% (+1)
set_3	9/10	1.79% (+1)
was_1	100/100	-
was_2	100/100	-
gau_1	16/17	3.70% (+1)

TABLEAU 5.3 – Application du couplage heuristique métaheuristique basée individu pour la résolution du problème du bin packing

ne la trouvent pas pour une seule instance de taille 250, deux instances de taille 500 et trois instances de taille 1000.

Les résultats pour les instances t\_60 à t\_501, appelées instances *triplets*, sont moins bons. Ces instances sont particulièrement difficiles, la solution optimale est composée de boîtes contenant exactement trois objets. Nos méthodes sont cependant robustes pour résoudre ces instances. En effet, bien que la solution optimale soit rarement atteinte, toutes les solutions ont une seule boîte supplémentaire pour la plus petite taille d'instance, neuf boîtes pour la plus grande taille d'instance, soit un écart de 5% par rapport à l'optimal au pire des cas.

Ces résultats sont très encourageants pour l'application de nos méthodes au problème de planification d'activités avec affectation de ressources.

### 5.2.3.2 Métaheuristicques basées population

Il n'existe pas à notre connaissance de méthodes pour le réglage de paramètres de l'optimisation par essaim particulière. Le meilleur paramétrage que nous avons trouvé est le suivant.

Le nombre de particules utilisé est égal à  $\frac{N}{10}$  pour chaque instance, avec  $N$  le nombre d'objets à affecter aux boîtes. Les coefficients sont choisis de la façon suivante.

- Dans 30% des cas, la particule est influencée par sa meilleure position,
- Dans moins de 10% des cas, elle est influencée par la meilleure position de l'essaim,
- Dans les autres cas, elle subit l'influence de sa propre vitesse.

Ainsi,  $c_1 = 0.6$ ,  $c_2 = 0.3$  et  $c_3 = 0.1$ .

Les perturbations sont appliquées dans le but d'éviter une convergence trop rapide vers des optimums locaux. La solution ne doit pas être trop perturbée. Les réglages les concernant sont les suivants :

- Si l'essaim est dispersée ( $D$ ) après un certain nombre d'itérations sans amélioration, ce palier est égal à  $60 \times N$ .
- Si la mutation ( $M$ ) ou l'influence du record ( $R$ ) est appliquée, la particule est influencée par cette perturbation avec une probabilité environ égale à 0.01, c'est-à-dire  $c_4 = 0.01$ .

Le déplacement de particules que nous proposons donne de meilleurs résultats que le déplacement de particules original. Les perturbations proposées permettent de résoudre un plus grand nombre d'instances. La mutation de particules avec dispersion d'essaim est la perturbation qui améliore le plus les résultats.

Sur les 1587 instances testées, 1245 ont été résolues à l'optimalité avec un couplage heuristique optimisation par essaim particulaire. Au pire des cas, les résultats sont 8% moins bons que la solution optimale.

## 5.2.4 Conclusion de la résolution du problème du bin packing

Nos méthodes qui utilisent le couplage entre une métaheuristique et une heuristique sont des méthodes simples à mettre en place qui donnent de bons résultats. Elles consistent en la combinaison de deux méthodes connues de beaucoup de scientifiques. Notre objectif principal est la mise en place d'un outil d'aide à la décision pour les systèmes hospitaliers. C'est pourquoi nous avons concentré nos travaux dans le développement de méthodes simples qui donnent de bons résultats et non pas dans celui de méthodes trop complexes qui pourraient pourtant donner de meilleurs résultats.

Les résultats montrent que le couplage résout à l'optimalité la majorité des instances de la littérature. Pour les quelques instances non résolues, la solution trouvée est au pire des cas 5% moins bonne que la solution optimale. Ainsi, nous validons nos méthodes et les appliquons à un problème réel : le problème de planification d'activités avec affectation de ressources qui est considéré comme un problème de bin packing avec incompatibilités.

## 5.3 Le modèle 1 de la CHT

### 5.3.1 Création des instances

Les instances ont été générées aléatoirement de façon à représenter des données réelles.

La CHT est composée de trois sites. Quatre à huit équipements sont répartis sur la CHT, chacun étant situé sur un site. Trois à cinq ressources humaines sont disponibles sur la CHT. Il existe trois types d'équipements. L'horizon de planification est composé de plusieurs périodes, chaque période représentant une demi-journée. L'horizon de planification dépend du nombre d'activités à planifier. Plus le nombre d'activités est important, plus l'horizon de planification est grand.

Les mêmes instances sont utilisées pour tous les modèles. Cependant, le nombre de ressources humaines ainsi que toutes les données relevant des ressources humaines ne sont

pas considérées pour le modèle 1. Pour le modèle 3, la donnée relative aux affectations sur site des ressources humaines n'est pas prise en compte.

Deux types d'instance ont été générés :

- Type A : chaque temps de traitement est compris entre 5 et 45 minutes, chacun étant un multiple de 5 minutes. Les tailles de problème testées vont de 50 activités à 500. Le temps d'ouverture de chaque équipement pendant chaque période est égal à 300 minutes. Quatre ou huit équipements sont disponibles pour le traitement des activités. Le temps de travail de chaque ressource humaine pendant chaque période est compris entre zéro et 300 minutes, avec un pas de 60 minutes. Pour les modèles 2 et 3, seul ce type d'instance est utilisé.
- Type B : 50 à 500 activités sont à planifier, quatre ou huit équipements sont considérés. Chaque équipement est disponible pendant 100 minutes durant chaque période. Les activités ont un temps de traitement entre une et quatre-vingt dix-neuf minutes. Les types d'équipement et les ressources humaines ne sont pas considérés puisque ce type d'instance n'est utilisée que dans le modèle 1.

Le Tableau 5.4 résume les données générées pour les problèmes de la CHT.

Type	$N$	$K$	$L$	$M$	$A$	$T$
A	50	3	4	3	3	8
	100	3	4	3	3	8
	200	3	8	5	3	10
	300	3	8	5	3	15
	400	3	8	5	3	20
	500	3	8	5	3	25
B	50	3	4	-	-	9
	100	3	4	-	-	17
	200	3	8	-	-	20
	500	3	8	-	-	50

TABLEAU 5.4 – Données générées pour la CHT

### 5.3.2 Résultats référence : CPLEX

La résolution par méthode exacte utilise la version 12.4.0 de CPLEX. La machine est composée d'un Intel Xeon X5687 quad-core CPU 3.6 GHz.

Les temps de calcul sont limités à une heure. Dans les tableaux présentant les résultats, si après une heure le solveur ne trouve pas de solution optimale, l'instance correspondante est notée avec une \*. Le cas échéant, les valeurs correspondant aux bornes inférieures et supérieures renseignées par le solveur sont indiquées. Plus la taille du jeu de données est grande, moins le solveur trouve la solution optimale en moins d'une heure.

Dans le Tableau 5.5, les résultats sont notés sous la forme  $(C_1; C_2)$  avec

- $C_1$  le nombre de dates limites non respectées,
- $C_2$  la somme des périodes affectées à toutes les activités.

La fonction objectif est donnée par l'Équation (5.1), avec  $\omega_1 = 5$  et  $\omega_2 = 0$ . En effet, pour toutes les instances,  $C_2$  est toujours inférieur à  $10^5$ . Ainsi, la valeur de chaque critère est facilement lisible.

$$H(X) = 10^{\omega_1} \cdot C_1(X) + 10^{\omega_2} \cdot C_2(X) \quad (5.1)$$

Instance	$(C_1; C_2)$	Temps (sec)	Borne inf	Borne sup	Gap (%)
50A	(0; 51)	0.02	-	-	-
100A	(0; 131)	0.08	-	-	-
200A	(0; 266)	0.27	-	-	-
300A*	-	-	(0; 531.31)	(0; 533)	0.32
400A	(0; 830)	241.75	-	-	-
500A*	-	-	(0; 1223.01)	(0; 1225)	0.16
50B	(1; 150)	2.27	-	-	-
100B	(0; 517)	707.82	-	-	-
200B*	-	-	(0; 1119.35)	(0; 1131)	1.03
500B*	-	-	(2; 6300.11)	(2; 6394)	0.36

TABLEAU 5.5 – Résultats pour le modèle 1 avec la méthode exacte

Dans la suite, nos méthodes sont comparées aux résultats références trouvés par le solveur. Pour chaque instance, la référence est la valeur optimale si elle a été trouvée, ou la borne supérieure.

### 5.3.3 Algorithmes de liste

Les algorithmes de liste détaillés dans le chapitre précédent sont appliquées sur ces mêmes instances. Les résultats sont présentés dans le Tableau 5.6 sous la forme  $(C_1; C_2)$ .

Les heuristiques proposées sont des extensions des heuristiques pour le bin packing de base : la compatibilité entre les activités et les équipements est vérifiée à chaque affectation d'une activité à un équipement. Les même garanties de performance que celles des heuristiques du bin packing de base sont vérifiées. L'heuristique Best Fit donne des résultats équivalents à ceux retournés par l'heuristique First Fit qui donne elle-même de meilleurs résultats que l'heuristique Next Fit. L'heuristique Worst Fit est celle qui donne les moins bons résultats.

Lorsque ces heuristiques sont appliquées sur des listes d'activités triées de façon décroissante, l'heuristique First Fit Decreasing est celle qui donne les meilleurs résultats.

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	(1; 150)	517	1131	(2; 6394)
<i>NF</i>	(1; 54)	(4; 157)	(12; 312)	(26; 672)	(30; 1053)	(45; 1667)	(27; 268)	(60; 1028)	(96; 2221)	(321; 12973)
<i>FF</i>	(1; 54)	(4; 153)	(11; 300)	(23; 633)	(27; 1009)	(43; 1587)	(19; 212)	(34; 701)	(55; 1440)	(194; 8218)
<i>BF</i>	(3; 54)	(4; 158)	(12; 313)	(22; 638)	(29; 1007)	(41; 1594)	(20; 217)	(37; 719)	(56; 1475)	(194; 8253)
<i>WF</i>	(2; 55)	(7; 163)	(8; 320)	(27; 695)	(37; 1106)	(49; 1705)	(23; 245)	(45; 793)	(72; 1710)	(211; 8677)
<i>NFD</i>	(0; 56)	(0; 156)	(0; 301)	(0; 661)	(0; 1086)	(0; 1614)	(33; 264)	(53; 859)	(3; 1834)	(201; 10468)
<i>FFD</i>	<b>(0; 54)</b>	(0; 153)	<b>(0; 297)</b>	<b>(0; 629)</b>	<b>(0; 1033)</b>	<b>(0; 1556)</b>	<b>(3; 172)</b>	<b>(0; 584)</b>	<b>(0; 1276)</b>	<b>(9; 7274)</b>
<i>BFD</i>	(0; 56)	<b>(0; 151)</b>	(0; 300)	(0; 632)	(0; 1051)	(0; 1568)	(3; 175)	(0; 606)	(0; 1318)	(9; 7364)
<i>WFD</i>	(0; 56)	(0; 159)	(0; 318)	(0; 659)	(0; 1083)	(0; 1635)	(28; 241)	(45; 769)	(0; 1804)	(289; 10312)

TABLEAU 5.6 – Résultats de l'application des algorithmes de liste sur le modèle 1

### 5.3.4 Métaheuristiques

L'application des métaheuristiques sur le problème de planification d'activités avec affectation de ressources donne des résultats intéressants. Pour les plus petites tailles d'instance, les méthodes trouvent la solution optimale. Plus les tailles d'instance sont grandes, plus les résultats sont éloignés de la solution optimale quand elle existe, ou de la borne supérieure retournée par le solveur.

Lorsque la recherche est faite dans l'ensemble  $C$  des solutions admissibles, la solution initiale est admissible mais n'est pas générée par les heuristiques présentées. En effet, si la solution initiale est de trop bonne qualité, la métaheuristique ne parvient pas à l'améliorer. Finalement, les résultats sont meilleurs lorsque l'exploration est faite dans l'ensemble  $\Omega$  de toutes les solutions et non pas dans l'ensemble  $C$  des solutions admissibles. À partir d'une solution dégradée, les métaheuristiques améliorent la solution courante mais n'atteignent pas de bonnes solutions pour les instances de grande taille.

### 5.3.5 Couplage heuristiques métaheuristiques

#### 5.3.5.1 Métaheuristiques basées individu

L'application du couplage entre les heuristiques et les métaheuristiques donne de très bons résultats. La solution optimale est atteinte pour les instances de petite taille. Pour les instances de plus grande taille, les méthodes donnent des résultats au pire à 1% de la solution optimale. Le Tableau 5.7 résume les écarts des résultats de la méthode proposée à l'optimal ou à la borne supérieure retournée par le solveur pour le critère relatif à la somme des périodes affectées aux activités. Dans le cas d'intervalle retourné par le serveur, l'écart est calculé par rapport à la borne supérieure.

Pour l'instance 500B, la somme des périodes affectées aux activités trouvée par le couplage est meilleure que la borne supérieure retournée par le solveur. Ceci est très encourageant pour l'application sur des données réelles : le couplage s'avère plus performant que les méthodes exactes avec le même temps imparti pour les instances de grande taille.

Instance	50A	100A	200A	300A	400A
Référence	51	131	266	[531.31, 533]	830
Valeur	51	131	266	534	835
Écart	0%	0%	0%	0.19%	0.60%
Instance	500A	50B	100B	200B	500B
Référence	[1223.01, 1225]	150	517	[1119.35, 1131]	[6300.11, 6394]
Valeur	1234	150	518	1135	6368
Écart	0.73%	0%	0.19%	0.35%	<b>-0.4%</b>

TABLEAU 5.7 – Résultat du couplage pour le problème de planification d'activités avec affectation de ressources

### 5.3.5.2 Métaheuristique basée population

De la même façon que pour le problème du bin packing, les résultats obtenus par le couplage avec l'optimisation par essaim particulière sont bons. Le nouveau déplacement de particules améliore les résultats. Les perturbations proposées les améliorent encore plus. Cependant, les résultats sont moins bons que ceux obtenus par le couplage avec les métaheuristicues basées individu.

Ces expérimentations permettent de valider toutes nos méthodes. En particulier, le coupage des heuristiques et des métaheuristicues est très performant. Nos propositions permettent d'obtenir une solution de planification d'activités avec affectation de ressources en moins d'une heure. La solution trouvée est une très bonne solution.

## 5.4 Les modèles avec ressources humaines

Le type d'instance A présenté pour le modèle 1 est testé par les modèles avec ressources humaines. On note  $HS$  si les heures supplémentaires des ressources humaines sont autorisées,  $\overline{HS}$  dans le cas contraire. Nous faisons l'hypothèse que les heures supplémentaires des équipements ne sont jamais autorisées.

### 5.4.1 Résolution par méthode exacte

La résolution par méthode exacte est réalisée identiquement au modèle 1. Les calculs sont limités à une heure. Les modèles sont étudiés deux à deux : les modèles  $2\overline{HS}$  et  $2HS$ , puis les modèles  $3\overline{HS}$  et  $3HS$ . La fonction objectif de base est la même que précédemment : l'objectif est de minimiser le nombre d'activités réalisées après leur date limite puis de minimiser la somme des périodes affectées aux activités. Ainsi la fonction objectif se formule tel que dans l'Équation (5.1). Cette fonction est directement utilisable pour le modèle  $2\overline{HS}$ . Le modèle  $2HS$  autorise les heures supplémentaires des ressources humaines. Puisque les heures supplémentaires ont un coût, il semble logique de les minimiser.

La fonction objectif devient donc celle donnée par l'Équation (5.2), avec

- $C_1$  le nombre de dates limites non respectées,
- $C_2$  la somme des périodes affectées aux activités,
- $C_3$  le nombre total d'heures supplémentaires réalisées par les ressources humaines pendant l'horizon temporel considéré.

$$H(X) = 10^{\omega_1} \cdot C_1(X) + 10^{\omega_2} \cdot C_2(X) + 10^{\omega_3} \cdot C_3(X) \quad (5.2)$$

Si  $\omega_1 > \omega_2 > \omega_3$ , les heures supplémentaires sont minimisées en dernier, c'est-à-dire l'objectif premier est l'augmentation de l'offre de soin malgré un coût augmenté par les heures supplémentaires. À l'inverse, si  $\omega_1 < \omega_2 < \omega_3$ , cela revient à ne pas autoriser les heures supplémentaires. Nous vérifions que dans ce cas, les résultats sont les mêmes qu'en utilisant le modèle  $2\overline{HS}$ . Ces deux configurations sont des configurations extrêmes. Une pondération plus précise permettrait d'ajuster au mieux les coûts engendrés par les heures supplémentaires. Un travail collaboratif avec le personnel de santé permettrait de régler au mieux cette pondération. Les résultats sont présentés dans le Tableau 5.8 par le duo  $(C_1, C_2)$  ou le triplet  $(C_1, C_2, C_3)$  pour chaque instance, chaque modèle et chaque



combinaison de critères précédemment présentée. Les bornes inférieures et supérieures sont renseignées. Par soucis de gain de place, si la borne inférieure n'est pas remplie, le résultat inscrit en borne supérieure est la solution optimale.

Instance	Temps (s)	Borne inf	Borne sup	Gap (%)
Modèle $2\overline{HS}$ , $\omega_1 < \omega_2$				
50A	0.21	-	(0 ; 67)	-
100A	1.69	-	(0 ; 208)	-
200A	-	(0 ; 439.83)	(0 ; 441)	0.26
300A	126.44	-	(0 ; 1036)	-
400A	-	(0 ; 1734.93)	(0 ; 1737)	0.12
500A	-	(0 ; 2700.21)	(0 ; 2704)	0.14
Modèle $2HS$ , $\omega_1 > \omega_2 > \omega_3$				
50A	0.56	-	(0 ; 59 ; 210)	-
100A	1.3	-	(0 ; 155 ; 805)	-
200A	14.21	-	(0 ; 345 ; 1295)	-
300A	2746.75	-	(0 ; 743 ; 2560)	-
400A	-	(0 ; 1205 ; 3335.35)	(0 ; 1205 ; 3355)	0.02
500A	-	(0 ; 1806 ; 3914.13)	(0 ; 1806 ; 4045)	0.12
Modèle $2HS$ , $\omega_1 < \omega_2 < \omega_3$				
50A	0.21	-	(0 ; 67 ; 0)	-
100A	1.93	-	(0 ; 208 ; 0)	-
200A	8.83	-	(0 ; 441 ; 0)	-
300A	-	(0 ; 1034.49 ; 0)	(0 ; 1036 ; 0)	0.15
400A	-	(0 ; 1735.06 ; 0)	(0 ; 1737 ; 0)	0.11
500A	-	(0 ; 2699.67 ; 0)	(0 ; 2705 ; 0)	0.20

TABLEAU 5.8 – Résolution par méthode exacte du modèle 2

Le modèle 3 considère un nouveau critère : le nombre de déplacements de la ressource humaine. En effet, l'affectation des ressources humaines sur les sites est à déterminer par l'outil. Cette affectation doit minimiser le nombre de déplacements des ressources humaines au sein de la CHT. Les modèles  $3\overline{HS}$  et  $3HS$  diffèrent par la possibilité laissée ou pas aux ressources humaines d'effectuer des heures supplémentaires. La fonction objectif est donnée par l'Équation (5.3) pour le modèle  $3\overline{HS}$  et par l'Équation (5.4) pour le modèle  $3HS$ , avec

- $C_1$  le nombre de dates limites non respectées,
- $C_2$  la somme des périodes affectées aux activités,
- $C_3$  le nombre d'heures supplémentaires réalisées par les ressources humaines,
- $C_4$  le nombre de déplacements des ressources humaines.

$$H(X) = 10^{\omega_1}.C_1(X) + 10^{\omega_2}.C_2(X) + 10^{\omega_4}.C_4(X) \quad (5.3)$$

$$H(X) = 10^{\omega_1}.C_1(X) + 10^{\omega_2}.C_2(X) + 10^{\omega_3}.C_3(X) + 10^{\omega_4}.C_4(X) \quad (5.4)$$

Le nombre de combinaisons de critères possibles est très important. Nous choisissons arbitrairement  $\omega_1 > \omega_2 > \omega_3 > \omega_4$  pour la suite des expérimentations : dans l'ordre de priorité, sont minimisées les dates limites non respectées, puis la somme des périodes affectées aux activités, puis le nombre d'heures supplémentaires le cas échéant et enfin le nombre de déplacements. Une autre configuration est testée pour le modèle  $3HS$ , celle où  $\omega_1 > \omega_2 > \omega_4 > \omega_3$ . Dans ce cas, on privilégie les déplacements aux heures supplémentaires. Les résultats obtenus par application de la méthode exacte sont donnés dans le Tableau 5.9 sous la forme  $(C_1; C_2; C_4)$  ou  $(C_1; C_2; C_3; C_4)$ .

Instance	Temps (s)	Borne inf	Borne sup	Gap (%)
Modèle $\overline{3HS}$ , $\omega_1 > \omega_2 > \omega_4$				
50A	7.65	-	(0 ; 67 ; 0)	-
100A	21.46	-	(0 ; 208 ; 0)	-
200A	-	(0 ; 439.67 ; 0)	(0 ; 441 ; 0)	0.30
300A	-	(0 ; 1033.01 ; 0)	(0 ; 1036 ; 0)	0.29
400A	-	(0 ; 1730.98 ; 0)	(0 ; 1735 ; 0)	0.23
500A	-	(0 ; 2657.99 ; 0)	(0 ; 2669 ; 0)	0.75
Modèle $3HS$ , $\omega_1 > \omega_2 > \omega_3 > \omega_4$				
50A	0.67	-	(0 ; 59 ; 210)	-
100A	10.47	-	(0 ; 155 ; 805)	-
200A	-	(0 ; 345 ; 1138.2 ; 0)	(0 ; 345 ; 1295 ; 0)	0.44
300A	-	(0 ; 743 ; 2451.3 ; 0)	(0 ; 743 ; 2560 ; 0)	0.14
400A	-	(0 ; 1205 ; 3181.90 ; 0)	(0 ; 1205 ; 3355 ; 0)	0.14
500A	-	(0 ; 1806 ; 3802.46 ; 0)	(0 ; 1806 ; 4080 ; 0)	0.33
Modèle $3HS$ , $\omega_1 > \omega_2 > \omega_4 > \omega_3$				
50A	2.18	-	(0 ; 59 ; 210 ; 0)	-
100A	5.42	-	(0 ; 155 ; 805)	-
200A	-	(0 ; 345 ; 764.44 ; 0)	(0 ; 345 ; 1295 ; 0)	0.15
300A	-	(0 ; 743 ; 1422.77 ; 0)	(0 ; 743 ; 2560 ; 0)	0.15
400A	-	(0 ; 1205 ; 1355 ; 0)	(0 ; 1205 ; 3355 ; 0)	0.17
500A	-	(0 ; 1806 ; 1304.60 ; 0)	(0 ; 1806 ; 4080 ; 0)	0.59

TABLEAU 5.9 – Résolution par méthode exacte du modèle 3

La comparaison des résultats des modèles  $\overline{3HS}$  et  $3HS$  montre que l'autorisation des heures supplémentaires permet de diminuer la somme des périodes affectées aux activités ainsi que le nombre d'activités réalisées après leur date limite. La comparaison des

résultats des modèles 2 et 3 montre que lorsque l'affectation sur site des ressources humaines est une variable et non une donnée, le système s'en trouve plus efficient : la somme des périodes affectées aux activités diminue ainsi que le nombre d'activités réalisées après leur date limite. Il faudrait cependant travailler sur d'autres instances pour pouvoir tirer une conclusion plus générale. Il serait notamment intéressant de travailler avec des instances composées d'un plus grand nombre de sites ou à l'extrême une instance où il y a moins de sites que de ressources humaines.

### 5.4.2 Limitation des méthodes exactes

Dans cette partie, l'impact du nombre de critères étudiés sur le temps de calcul est montré.

Pour chaque taille de jeu de données, pour les modèles 1,  $2\overline{HS}$ ,  $2HS$  et  $3\overline{HS}$ , toutes les combinaisons de critères ont été testées. Pour le modèle  $3HS$ , à cause du grand nombre de combinaisons de critères possibles, seulement vingt-neuf d'entre elles ont été choisies, celles qui sont les plus significatives. Le temps de calcul pour le solveur est limité à trente minutes. Une case vide ( - ) signifie que le solveur ne trouve pas la solution optimale en moins de trente minutes. Dans les autres cas, le temps de calcul peut varier entre quelques secondes et moins de trente minutes. Pour chaque cas, le Tableau 5.10 renseigne le nombre d'itérations nécessaires par le solveur pour trouver la solution optimale sur la première ligne, ainsi que le nombre de nœuds de la décomposition branch-and-cut de CPLEX sur la seconde ligne.

Plus le nombre d'activités à planifier dans la CHT est élevé, plus le nombre d'itérations pour trouver la valeur optimale de la fonction objectif est grand. De la même façon, le nombre d'itérations augmente avec le nombre de critères. Les limites des modèles sont vite atteintes. Si deux critères sont étudiés, le solveur ne trouve pas de solution en moins de trente minutes s'il y a plus de deux cents activités pour les modèles  $2\overline{HS}$  et  $2HS$ . Si trois critères ou plus sont optimisés, la limite est de cent activités pour les modèles  $3\overline{HS}$  et  $3HS$ .

## 5.5 Conclusion

Toutes les méthodes proposées dans le Chapitre 4 ont été appliquées sur le problème du bin packing puis sur le problème de la CHT : problème de planification d'activités avec affectation de ressources, défini comme étant un problème de bin packing avec incompatibilités.

Il existe de nombreuses instances de la littérature pour le problème du bin packing, ce qui nous a permis de valider nos méthodes. L'application des métaheuristiques à la résolution du problème du bin packing donne des résultats médiocres : certaines instances sont résolues à l'optimalité mais elles ne sont qu'une minorité. L'application du couplage des heuristiques et des métaheuristiques est plus prometteuse : la majorité des instances est résolue à l'optimalité par ces méthodes. Les métaheuristiques basées individu sont cependant plus performantes que la métaheuristique basée population testée. L'optimisation par essaim particulière demande un réglage précis de ses paramètres.

Nos méthodes ne donnent pas d'aussi bons résultats que les méthodes de la littérature mais sont plus simples à mettre en place. L'utilisation de nos méthodes est finalement

Instance	100A	200A	300A	400A	500A
Modèle 1					
1 critère	374	1 148	1 239	1 580	39 974
	21	22	8	0	2 103
2 critères	41 167	19 314	35 072	57 080	-
	3 684	3 553	5 280	6 856	-
Modèle $2HS$					
1 critère	236	453	821	1 570	1 948
	0	1	5	34	14
2 critères	17 857	15 771	-	-	-
	4 861	1 642	-	-	-
Modèle $2HS$					
1 critère	222	453	669	39 064	1 713
	0	0	2	2 422	2
2 critères	5 389	9 486	-	-	-
	1 244	1 273	-	-	-
3 critères	-	-	-	-	-
	-	-	-	-	-
Modèle $3HS$					
1 critère	1 397	513 507	79 036	1 008 101	1 350 326
	11	4 841	1 188	6 482	1 905
2 critères	1 213	-	-	-	-
	2	-	-	-	-
3 critères	1 302	-	-	-	-
	1	-	-	-	-
Modèle $3HS$					
1 critère	1 041	403 543	367 914	1 930 366	-
	0	4 246	4 444	4 601	-
2 critères	1 069	-	-	-	-
	0	-	-	-	-
3 critères	1 461	-	-	-	-
	26	-	-	-	-
4 critères	71 236	-	-	-	-
	2 733	-	-	-	-

TABLEAU 5.10 – Nombre d'itérations et de nœuds pour la résolution par méthode exacte des cinq modèles

applicative, c'est pourquoi nous avons privilégié le développement de méthodes simples.

Après cette première étape de validation, ces méthodes ont été appliquées au problème de planification d'activités avec affectation de ressources. Les résultats donnés par nos méthodes sont comparés avec les résultats obtenus par la résolution par méthode exacte du solveur CPLEX. Pour beaucoup d'instances, le solveur ne trouve pas de solution optimale après une heure de calcul. Nos propositions retournent une solution dans ce temps imparti. Nos propositions sont très prometteuses. Pour les instances de grande taille, nos résultats sont de meilleure qualité que la borne supérieure retournée par le solveur.

Les méthodes approchées n'ont pas été adaptées à la résolution des modèles intégrant les ressources humaines. Nous avons cependant montré que le solveur trouve très peu de solutions optimales pour ces modèles intégrant de plus en plus de contraintes. L'utilisation de méthodes approchées semble donc impérative dans un futur proche. Grâce aux méthodes couplées proposées, ces problèmes pourront être résolus grâce au développement d'heuristiques adéquates, qui seront couplées à des métaheuristiques.

# Conclusion

Le contexte hospitalier actuel nécessite de trouver des solutions pour améliorer l'efficacité actuelle du système hospitalier. Des coopérations hospitalières ont récemment vu le jour. Une mutualisation des ressources peut permettre une meilleure utilisation des différents Centres Hospitaliers au sein d'un même territoire. Mais ces coopérations hospitalières manquent parfois d'outil d'aide à la décision pour améliorer voire optimiser leur fonctionnement.

Le problème de planification d'activités avec affectation de ressources a été étudié dans ce manuscrit. La résolution de ce problème peut aider à la gestion de Communautés Hospitalières de Territoire existantes ou futures. L'analogie entre ce problème et le problème du bin packing a été réalisée : notre problème peut être considéré comme un problème de bin packing avec incompatibilités. C'est pourquoi notre manuscrit s'est concentré sur la résolution du problème du bin packing et les méthodes proposées ont pu être adaptées au problème réel de planification d'activités avec affectation de ressources.

De nombreuses méthodes de résolution existent déjà pour la résolution du problème du bin packing. Beaucoup de ces méthodes sont des méthodes complexes à mettre en place, elles allient des métaheuristiques, des méthodes de recherche locale, des méthodes de réduction de l'espace de recherche ou des heuristiques de construction. Étant donné que la résolution de notre problème doit servir de base à la conception d'un outil d'aide à la décision, nous avons concentré nos recherches sur la proposition de méthodes faciles à implémenter.

Plusieurs pistes ont été envisagées : l'utilisation de métaheuristiques basées individu ou de métaheuristiques basées population. Ces méthodes sont applicables mais les résultats ne sont pas très probants. Une nouvelle méthode a donc été proposée : le couplage hiérarchique d'heuristiques et de métaheuristiques basées individu ou population. Ce couplage utilise un codage inspiré des problèmes de permutation d'ordonnancement. Une solution est une liste d'activités à planifier. Des systèmes de voisinage classiques tel que la permutation entre activités ou l'insertion d'activité peuvent être utilisés.

Grâce à ce couplage, les caractéristiques au problème étudié sont uniquement intégrées dans l'heuristique utilisée. Ainsi, pour le passage du problème du bin packing au problème de planification d'activités avec affectation de ressources, il a suffi de prendre en compte les incompatibilités entre activités et ressources uniquement au niveau de l'heuristique. Le reste de la méthode a été utilisé sans modification.

L'application de ces méthodes a permis d'obtenir de très bons résultats. Concernant

le large benchmark existant pour le problème du bin packing, nos méthodes ont résolu une grande majorité des instances. Celles qui n'ont pas été résolues ont été approximées à 5% de la solution optimale au pire des cas. Pour la résolution du problème de planification d'activités avec affectation de ressources, nous avons créé nos propres instances représentatives de la réalité hospitalière. Ces instances ont été traitées par un solveur afin de fournir des résultats référence : la solution optimale ou une borne supérieure si la méthode exacte ne parvient pas à résoudre l'instance considérée. Nos méthodes proposées résolvent une grande partie de ces instances à l'optimalité. Pour les instances de grande taille, nos méthodes trouvent une solution meilleure que la borne supérieure retournée par le solveur. Ces résultats sont très encourageants pour la suite de nos travaux.

## Perspectives

Grâce à l'utilisation des méthodes couplées, un plus large spectre de problèmes va pouvoir être traité en intégrant les nouvelles contraintes à prendre en compte dans l'heuristique utilisée. Ainsi, les problèmes suivants pourront être étudiés :

- Prise en compte des ressources humaines,
- Intégration des relations de précédence entre activités,
- Définition d'un degré d'urgence entre les différentes activités,
- Détermination de la composition des équipes médicales,
- Considération de la spécificité des patients, etc.

Une autre perspective est l'élargissement du domaine d'étude de notre problème afin de pouvoir traiter des problèmes de dimensionnement ou d'ordonnancement. Par une augmentation de l'horizon temporel, les problèmes de dimensionnement peuvent être étudiés. Par une définition de périodes de plus petite taille, les problèmes d'ordonnancement peuvent être traités.

La perspective prioritaire à nos yeux est la prise en compte des ressources humaines. Deux modèles les concernant ont été définis : le modèle 2 où les ressources humaines ont une affectation sur site donnée, le modèle 3 où leur affectation sur site est une variable. Dans ce dernier modèle, les ressources humaines sont autorisées à se déplacer d'un site à un autre. Le cas d'étude suivant permet de se rendre compte de l'importance de leur mutualisation entre différents Centres Hospitaliers. Ce cas d'étude considère l'exemple de l'imagerie médicale : une activité est un examen.

La CHT est composée de deux sites. Quatre équipements sont localisés sur ces deux sites. Chaque équipement est d'un type. Trois types existent : IRM, radio et scanner. Trois ressources humaines travaillent sur ces sites. Sur le premier site, il y a un équipement de type radio et un du type scanner, et deux ressources humaines. Les deux ressources humaines peuvent travailler sur les deux types d'équipement présents (radio et scanner). Sur le second site, il y a deux équipements, un du type IRM et un du type scanner. Une seule ressource humaine travaille sur ce site, elle peut travailler sur les deux équipements. La Figure 1 décrit la CHT.

Une liste d'examens doit être planifiée sur cette CHT. Chaque examen a un temps de traitement, un site de référence et une date limite. En supposant que l'on relâche la contrainte d'affectation de tous les examens pendant l'horizon de planification, l'objectif

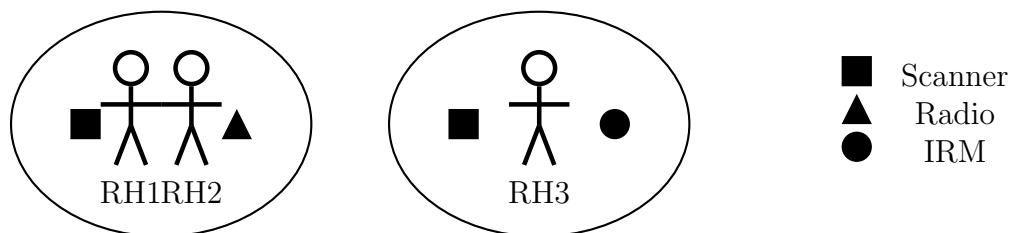


FIGURE 1 – Composition de la CHT

devient la maximisation du nombre d'examens réalisés pendant l'horizon de planification. Cent examens doivent être planifiés pendant trois périodes.

Pour l'instance créée, seuls quatre-vingt-huit examens sur cent sont réalisés si les ressources humaines ne peuvent pas se déplacer. Tous les examens sont réalisés si les ressources peuvent se déplacer. Ceci s'explique facilement. Supposons qu'il y ait moins d'examens de radio à faire que des autres types. Toutes les ressources humaines sont occupées au début. Parce qu'il y a moins de radios à réaliser, la ressource humaine qui travaillait sur l'équipement de type radio est maintenant libre. Elle ne peut pas travailler sur le scanner parce que la seconde ressource humaine travaille dessus. Sur le second site, la ressource humaine travaille soit sur l'équipement de type scanner ou celui de type IRM. Si les ressources humaines sont autorisées à se déplacer, la ressource humaine qui est libre sur le premier site peut aller travailler sur l'équipement de type scanner sur le second site, pendant que l'autre ressource humaine sur le second site travaille sur l'équipement de type IRM. Si elles ne peuvent pas se déplacer, plus de temps est nécessaire pour exécuter tous les examens.

Il est aussi intéressant d'étudier le taux d'occupation de chaque ressource dans les deux modèles considérés. Le taux d'occupation est la somme des temps de traitement des examens réalisés par la ressource divisée par son temps de disponibilité. Le taux d'occupation de l'équipement  $l_1$  est donné par l'Équation (5.5). L'objectif est de maximiser ces taux.

$$\tau_{l_1} = \frac{\sum_{i \in N} \sum_{s \in T} \sum_{m \in M} x_{i,l_1,s,m} \cdot t_i}{\sum_{s \in T} p_{l_1,s}} \quad (5.5)$$

Ces taux par ressource sont représentés par la Figure 2. Les taux d'occupation sont meilleurs dans le modèle 3 que dans le modèle 2. Quand les ressources humaines peuvent se déplacer, les ressources sont utilisées plus efficacement et plus d'examens sont réalisés.

La mutualisation de ressources n'est pas toujours utile, cela dépend de la configuration du système. S'il y a le même nombre de ressources humaines et d'équipements et si chaque ressource humaine ne peut travailler que sur un équipement, la mutualisation de ressources ne semble pas utile. La mutualisation semble être très efficace sur le système quand il y a moins de ressources humaines que d'équipements, et si les ressources humaines peuvent travailler sur plusieurs équipements, comme dans l'exemple précédent.



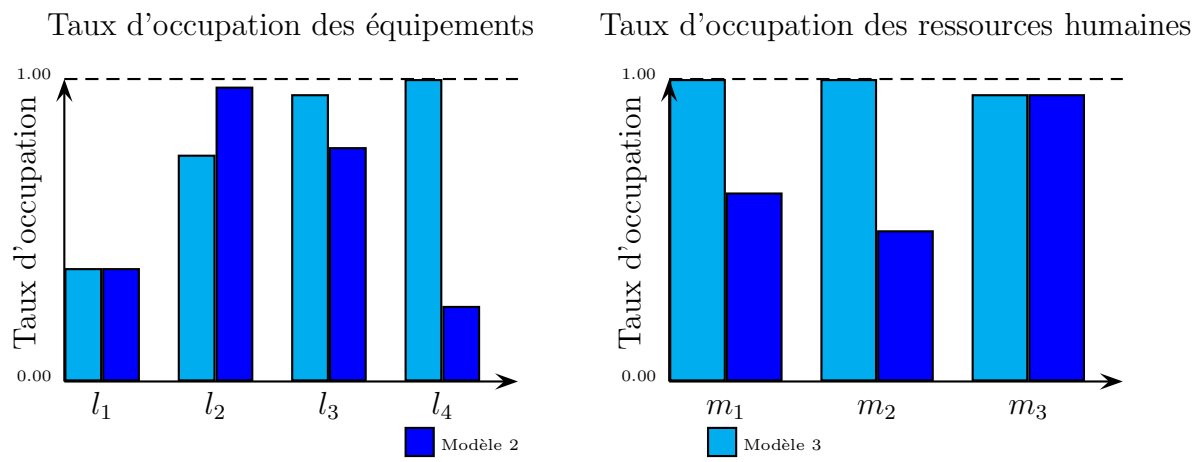


FIGURE 2 – Taux d'occupation des équipements et des ressources humaines

# Bibliographie

- Aarts, E. and van Laarhoven, P. (1987). *Simulated Annealing : Theory and Applications*. Kluwer Academic Publishers.
- Alvim, A. C., Ribeiro, C. C., Glover, F., and Aloise, D. J. (2004). A hybrid improvement heuristic for the one-dimensional bin packing problem. *Journal of Heuristics*, 10(2) :205–229.
- Ballou, R. H. (1992). *Business Logistics Management*. Prentice Hall, Englewood Cliffs, NJ.
- Balogh, J., Békési, J., and Galambos, G. (2012). New lower bounds for certain classes of bin packing algorithms. *Theoretical Computer Science*, 440 :1–13.
- Benders, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische mathematik*, 4(1) :238–252.
- Bhatia, A. and Basu, S. K. (2004). Packing bins using multi-chromosomal genetic representation and better-fit heuristic. In *Neural Information Processing*, pages 181–186. Springer.
- Bin, W., Qinke, P., Jing, Z., and Xiao, C. (2012). A binary particle swarm optimization algorithm inspired by multi-level organizational learning behavior. *European Journal of Operational Research*, 219(2) :224–233.
- Bortfeldt, A. and Wäscher, G. (2012). *Container loading problems : A state-of-the-art review*. Univ., Faculty of Economics and Management.
- Boutevin, C. (2003). *Problèmes d’ordonnancement et d’affectation avec contraintes de ressources de type RCPSP et line balancing*. PhD thesis, Université Blaise Pascal, Clermont-Ferrand II.
- Brugger, B., Doerner, K. F., Hartl, R. F., and Reimann, M. (2004). Antpacking—an ant colony optimization approach for the one-dimensional bin packing problem. In *Evolutionary Computation in Combinatorial Optimization*, pages 41–50. Springer.
- Caprara, A. and Pferschy, U. (2004). Worst-case analysis of the subset sum algorithm for bin packing. *Operations Research Letters*, 32(2) :159–166.
- Caprara, A. and Pferschy, U. (2005). Modified subset sum heuristics for bin packing. *Information processing letters*, 96(1) :18–23.
- Cardoen, B., Demeulemeester, E., and Beliën, J. (2010). Operating room planning and scheduling : A literature review. *European Journal of Operational Research*, 201(3) :921–932.

- 
- Cassagne, R., Gourgand, M., and Rodier, S. (2008). Un outil d'aide au dimensionnement, à la planification et à la visualisation d'un programme opératoire. *Gestion et Ingénierie des Systèmes Hospitaliers (GISEH)*, pages 807–814.
- Chaabane, S., Meskens, N., Guinet, A., and Laurent, M. (2008). Comparison of two methods of operating theatre planning : application in belgian hospital. *Journal of Systems Science and Systems Engineering*, 17(2) :171–186.
- Chabrol, M., Féliès, P., Gourgand, M., Tchernev, N., et al. (2005). Un environnement de modélisation pour le système d'information de la supply chain : application au nouvel hôpital d'estaing. In *INFORSID*, pages 603–618.
- Chabrol, M., Gourgand, M., and Rodier, S. (2008). A modeling methodology and its application to the design of decision - making aid tools for the hospital systems. In *IEEE International Conference on Research Challenges in Information Science (RCIS), Marrakech, Maroc, 3-6 juin 2008*, pages 161–172.
- Choi, S. and Wilhelm, W. E. (2014). On capacity allocation for operating rooms. *Computers & Operations Research*, 44 :174–184.
- Clerc, M. (1999). The swarm and the queen : towards a deterministic and adaptive particle swarm optimization. In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 3. IEEE.
- Clerc, M. (2004). Discrete particle swarm optimization, illustrated by the traveling salesman problem. In *New optimization techniques in engineering*, pages 219–239. Springer.
- Coello, C. A. (2000). An updated survey of GA-based multiobjective optimization techniques. *ACM Computing Surveys (CSUR)*, 32(2) :109–143.
- Coffman Jr, E. G., Garey, M. R., and Johnson, D. S. (1978). An application of bin-packing to multiprocessor scheduling. *SIAM Journal on Computing*, 7(1) :1–17.
- Coffman Jr, E. G., Garey, M. R., and Johnson, D. S. (1996). Approximation algorithms for bin packing : A survey. In *Approximation algorithms for NP-hard problems*, pages 46–93. PWS Publishing Co.
- Cours des comptes (2011). La sécurité sociale.
- Couty, É. (2013). Pacte de confiance pour l'hôpital.
- Csirik, J., Johnson, D. S., Kenyon, C., Orlin, J. B., Shor, P. W., and Weber, R. R. (2006). On the sum-of-squares algorithm for bin packing. *Journal of the ACM (JACM)*, 53(1) :1–65.
- Csirik, J., Johnson, D. S., Kenyon, C., Shor, P. W., and Weber, R. R. (1999). A self organizing bin packing heuristic. In *Algorithm Engineering and Experimentation*, pages 250–269. Springer.
- Dantzig, G. B. (1990). *Origins of the simplex method*. ACM.
- Denton, B., Viapiano, J., and Vogl, A. (2007). Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health care management science*, 10(1) :13–24.
- Deroussi, L., Grangeon, N., and Norre, S. (2014). Techniques d'hybridation à base de métaheuristiques pour optimiser des systèmes logistiques. In Siarry, P., editor,

- 
- Métaheuristiques : Recuits simulé, recherche avec tabous, recherche à voisinages variables, méthodes GRASP, algorithmes évolutionnaires, fourmis artificielles, essaims particuliers et autres méthodes d'optimisation.* Editions Eyrolles.
- Desaulniers, G., Desrosiers, J., and Solomon, M. M. (2005). *Column generation*, volume 5. Springer.
- Dorigo, M. (1992). Optimization, learning and natural algorithms. *Ph. D. Thesis, Politecnico di Milano, Italy.*
- Dorigo, M., Maniezzo, V., and Colomi, A. (1996). Ant system : optimization by a colony of cooperating agents. *Systems, Man, and Cybernetics, Part B : Cybernetics, IEEE Transactions on*, 26(1) :29–41.
- Eberhart, R. and Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on*, pages 39–43. IEEE.
- Eberhart, R., Simpson, P., and Dobbins, R. (1996). *Computational intelligence PC tools.* Academic Press Professional, Inc.
- Ehrgott, M. (1999). *A characterization of lexicographic max-ordering solutions.* Technische Universität Kaiserslautern, Fachbereich Mathematik.
- Everett, J. (2002). A decision support simulation model for the management of an elective surgery waiting system. *Health Care Management Science*, 5(2) :89–95.
- Falkenauer, E. (1996). A hybrid grouping genetic algorithm for bin packing. *Journal of heuristics*, 2(1) :5–30.
- Falkenauer, E. and Delchambre, A. (1992). A genetic algorithm for bin packing and line balancing. In *Robotics and Automation*, pages 1186–1192. IEEE.
- Fan, H. (2010). Discrete particle swarm optimization for TSP based on neighborhood. *Journal of Computational Information Systems*, 6(10) :3407–3414.
- Fleszar, K. and Charalambous, C. (2011). Average-weight-controlled bin-oriented heuristics for the one-dimensional bin-packing problem. *European Journal of Operational Research*, 210(2) :176–184.
- Fleszar, K. and Hindi, K. S. (2002). New heuristics for one-dimensional bin-packing. *Computers & operations research*, 29(7) :821–839.
- Fleury, G. (1993). *Méthodes stochastiques et déterministes pour les problèmes NP-difficiles.* Ph.D. thesis, Université Blaise Pascal, Clermont-Ferrand II.
- Friesen, D. K. and Langston, M. A. (1991). Analysis of a compound bin packing algorithm. *SIAM Journal on Discrete Mathematics*, 4(1) :61–79.
- Garey, M. R., Graham, R. L., Johnson, D. S., and Yao, A. C.-C. (1976). Resource constrained scheduling as generalized bin packing. *Journal of Combinatorial Theory, Series A*, 21(3) :257–298.
- Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability : A Guide to the Theory of NP-completeness.* WH Freeman and Company, New York.
- Garey, M. R. and Johnson, D. S. (1985). A 71/60 theorem for bin packing. *Journal of Complexity*, 1(1) :65–106.

- 
- Gartner, D. and Kolisch, R. (2014). Scheduling the hospital-wide flow of elective patients. *European Journal of Operational Research*, 233(3) :689–699.
- Gendreau, M., Laporte, G., and Semet, F. (2004). Heuristics and lower bounds for the bin packing problem with conflicts. *Computers & Operations Research*, 31(3) :347–358.
- Giard, V. (2003). Gestion de la production et des flux. *Economica, Paris*.
- Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5) :533–549.
- Goldberg, E. F., Goldberg, M. C., and de Souza, G. R. (2008). Particle swarm optimization algorithm for the traveling salesman problem. *Greco (2008)*, pages 202–224.
- Goldberg, D. E. et al. (1989). *Genetic algorithms in search, optimization, and machine learning*, volume 412. Addison-wesley Reading Menlo Park.
- Gourgand, M. (1984). *Outils logiciels pour l'évaluation des performances des systèmes informatiques*. Ph.D. thesis, Université Blaise Pascal, Clermont-Ferrand II.
- Gourgand, M. (2008). La modélisation, la simulation et l'optimisation des flux dans les systèmes hospitaliers. *Bulletin de la Société Française de Recherche Opérationnelle et d'Aide à la Décision*, 21 :8–12.
- Gourgand, M., Grangeon, N., and Klement, N. (2014a). Activities planning and resource assignment on multi-place hospital system : Exact and approach methods adapted from the bin packing problem. In *7th International Conference on Health Informatics, Angers, France*, pages 117–124.
- Gourgand, M., Grangeon, N., and Klement, N. (2014b). An analogy between bin packing problem and permutation problem : A new encoding scheme. In *Advances in Production Management Systems. Innovative and Knowledge-Based Production Management in a Global-Local World*, volume 438, pages 572–579. Springer Berlin Heidelberg.
- Gourgand, M., Grangeon, N., and Klement, N. (2015). Activities planning and resources assignment on distinct places : A mathematical model. *RAIRO - Operations Research*, 49, à paraître.
- Gourgand, M. and Kellert, P. (1991). Conception d'un environnement de modélisation des systèmes de production. In *Congrès International de Génie Industriel*.
- Gupta, J. N. and Ho, J. C. (1999). A new heuristic algorithm for the one-dimensional bin-packing problem. *Production planning & control*, 10(6) :598–603.
- Haimes, Y. Y., Ladson, L., and Wismer, D. A. (1971). Bicriterion formulation of problems of integrated system identification and system optimization.
- Hans, E., Wullink, G., Van Houdenhoven, M., and Kazemier, G. (2008). Robust surgery loading. *European Journal of Operational Research*, 185(3) :1038–1050.
- Hansen, P. (1986). The steepest ascent mildest descent heuristic for combinatorial programming. In *Congress on numerical methods in combinatorial optimization, Capri, Italy*, pages 70–145.
- Hansen, P. and Mladenović, N. (2001). Variable neighborhood search : Principles and applications. *European journal of operational research*, 130(3) :449–467.
- Holland, J. H. (1962). Outline for a logical theory of adaptive systems. *Journal of the ACM (JACM)*, 9(3) :297–314.

- 
- Hu, X., Eberhart, R. C., and Shi, Y. (2003). Swarm intelligence for permutation optimization : a case study of n-queens problem. In *Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE*, pages 243–246. IEEE.
- Iser, J. H., Denton, B. T., and King, R. E. (2008). Heuristics for balancing operating room and post-anesthesia resources under uncertainty. In *Winter Simulation Conference*, pages 1601–1608. IEEE.
- Izakian, H., Ladani, B. T., Abraham, A., and Snasel, V. (2010). A discrete particle swarm optimization approach for grid job scheduling. *International Journal of Innovative Computing, Information and Control*, 6(9) :4219–4233.
- Jebali, A., Hadj Alouane, A. B., and Ladet, P. (2006). Operating rooms scheduling. *International Journal of Production Economics*, 99(1) :52–62.
- Johnson, D. S. (1973). *Near-optimal bin packing algorithms*. PhD thesis, Massachusetts Institute of Technology.
- Johnson, D. S. (1974). Fast algorithms for bin packing. *Journal of Computer and System Sciences*, 8(3) :272–314.
- Johnson, D. S., Demers, A., Ullman, J. D., Garey, M. R., and Graham, R. L. (1974). Worst-case performance bounds for simple one-dimensional packing algorithms. *SIAM Journal on Computing*, 3(4) :299–325.
- Johnson, E. L. (1989). Modeling and strong linear programs for mixed integer programming. In *Algorithms and Model Formulations in Mathematical Programming*, pages 1–43. Springer.
- Kashan, A. H. and Karimi, B. (2009). A discrete particle swarm optimization algorithm for scheduling parallel machines. *Computers & Industrial Engineering*, 56(1) :216–223.
- Kelley, Jr, J. E. (1960). The cutting-plane method for solving convex programs. *Journal of the Society for Industrial & Applied Mathematics*, 8(4) :703–712.
- Kemmoé Tchomt , S. and Gourgand, M. (2009). Particle swarm optimization : a study of particle displacement for solving continuous and combinatorial optimization problems. *International Journal of Production Economics*, 121(1) :57–67.
- Kennedy, J. (1999). Small worlds and mega-minds : effects of neighborhood topology on particle swarm performance. In *Congress on Evolutionary Computation*, volume 3. IEEE.
- Kennedy, J. (2006). Handbook of nature-inspired and innovative computing. *Swarm intelligence*. Springer, pages 187–219.
- Kennedy, J. and Eberhart, R. C. (1997). A discrete binary version of the particle swarm algorithm. In *International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 5, pages 4104–4108. IEEE.
- Khanafer, A., Clautiaux, F., and Talbi, E.-G. (2010). New lower bounds for bin packing problems with conflicts. *European journal of operational research*, 206(2) :281–288.
- Kharraja, S., Albert, F., and Chaabane, S. (2006). Blocs op ratrices publics : application de la strat gie block scheduling. In *6th International Conference on Modeling, Optimization and Simulation (MOSIM), Rabat, Maroc*, pages 3–5.

- 
- Kim, B.-I. and Wy, J. (2010). Last two fit augmentation to the well-known construction heuristics for one-dimensional bin-packing problem : an empirical study. *The International Journal of Advanced Manufacturing Technology*, 50(9-12) :1145–1152.
- Kirkpatrick, S. (1984). Optimization by simulated annealing : Quantitative studies. *Journal of statistical physics*, 34(5-6) :975–986.
- Krichen, S. and Dahmani, N. (2010). A particle swarm optimization approach for the bi-objective load balancing problem. *Electronic Notes in Discrete Mathematics*, 36 :751–758.
- Labeled, S., Gherboudj, A., and Chikhi, S. (2012). A modified hybrid particle swarm optimization algorithm for solving the travelling salesmen problem. *Journal of Theoretical and Applied Information Technology*, 39(2) :132–138.
- Lamiri, M., Grimaud, F., and Xie, X. (2009). Optimization methods for a stochastic surgery planning problem. *International Journal of Production Economics*, 120(2) :400–410.
- Lawler, E. L. and Wood, D. E. (1966). Branch-and-bound methods : A survey. *Operations research*, 14(4) :699–719.
- Lemamou, E., Gagné, C., and Gravel, M. (2012). Optimisation par essais particuliers pour le problème rcpsp. In *9th International Conference on Modeling, Optimization and Simulation (MOSIM), Bordeaux, France*.
- Levine, J. and Ducatelle, F. (2004). Ant colony optimization and local search for bin packing and cutting stock problems. *Journal of the Operational Research Society*, 55(7) :705–716.
- Litvak, N., van Rijsbergen, M., Boucherie, R. J., and van Houdenhoven, M. (2008). Managing the overflow of intensive care patients. *European journal of operational research*, 185(3) :998–1010.
- Liu, D., Tan, K. C., Huang, S., Goh, C. K., and Ho, W. K. (2008). On solving multiobjective bin packing problems using evolutionary particle swarm optimization. *European Journal of Operational Research*, 190(2) :357–382.
- Loh, K.-H., Golden, B., and Wasil, E. (2008). Solving the one-dimensional bin packing problem with a weight annealing heuristic. *Computers & Operations Research*, 35(7) :2283–2291.
- Lourenço, H. R., Martin, O., Stutzle, T., Glover, F., and Kochenberger, G. (2002). Iterated local search. *Handbook of Metaheuristics*, pages 321–353.
- Marcon, E., Kharraja, S., and Simonnet, G. (2003). The operating theatre planning by the follow-up of the risk of no realization. *International Journal of Production Economics*, 85(1) :83–90.
- Martello, S. and Toth, P. (1990). Lower bounds and reduction procedures for the bin packing problem. *Discrete Applied Mathematics*, 28(1) :59–70.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21 :1087–1092.
- Muritiba, A. E. F., Iori, M., Malaguti, E., and Toth, P. (2010). Algorithms for the bin packing problem with conflicts. *Inform Journal on computing*, 22(3) :401–415.

- 
- Ninio, M. and Schneider, J. J. (2005). Weight annealing. *Physica A : Statistical Mechanics and its Applications*, 349(3) :649–666.
- Norre, S. (2005). Heuristiques et métaheuristiques pour la résolution de problèmes d’optimisation combinatoire dans les systèmes de production. *Mémoire HDR, Université Blaise Pascal, Clermont-Ferrand*.
- Padberg, M. and Rinaldi, G. (1991). A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM review*, 33(1) :60–100.
- Ponnambalam, S., Jawahar, N., and Chandrasekaran, S. (2009). Discrete particle swarm optimization algorithm for flowshop scheduling. *Particle Swarm Optimization. In-Tech*.
- Puchinger, J. and Raidl, G. R. (2005). *Combining metaheuristics and exact algorithms in combinatorial optimization : A survey and classification*. Springer.
- Raidl, G. R. (2006). A unified view on hybrid metaheuristics. In *Hybrid Metaheuristics*, pages 1–12. Springer.
- Rais, A. and Viana, A. (2011). Operations research in healthcare : a survey. *International Transactions in Operational Research*, 18(1) :1–31.
- Reeves, C. (1996). Hybrid genetic algorithms for bin-packing and related problems. *Annals of Operations Research*, 63(3) :371–396.
- Reynolds, C. W. (1987). Flocks, herds and schools : A distributed behavioral model. In *ACM SIGGRAPH Computer Graphics*, volume 21, pages 25–34. ACM.
- Rodier, S. (2010). *Une tentative d’unification et de résolution des problèmes de modélisation et d’optimisation dans les systèmes hospitaliers. Application au nouvel hôpital Estaing*. PhD thesis, Université Blaise Pascal-Clermont-Ferrand II.
- Santibáñez, P., Begen, M., and Atkins, D. (2007). Surgical block scheduling in a system of hospitals : an application to resource and wait list management in a british columbia health authority. *Health care management science*, 10(3) :269–282.
- Scholl, A. (1999). *Balancing and sequencing of assembly lines*. Physica-Verlag Heidelberg.
- Scholl, A., Klein, R., and Jürgens, C. (1997). Bison : A fast hybrid procedure for exactly solving the one-dimensional bin packing problem. *Computers & Operations Research*, 24(7) :627–645.
- Schwerin, P. and Wäscher, G. (1997). The bin-packing problem : A problem generator and some numerical experiments with FFD packing and MTP. *International Transactions in Operational Research*, 4(5-6) :377–389.
- Shahzad, F., Baig, A. R., Masood, S., Kamran, M., and Naveed, N. (2009). Opposition-based particle swarm optimization with velocity clamping (ovcpso). In *Advances in Computational Intelligence*, pages 339–348. Springer.
- Shi, X. H., Liang, Y. C., Lee, H. P., Lu, C., and Wang, Q. (2007). Particle swarm optimization-based algorithms for tsp and generalized tsp. *Information Processing Letters*, 103(5) :169–176.
- Singh, A. and Gupta, A. K. (2007). Two heuristics for the one-dimensional bin-packing problem. *OR Spectrum*, 29(4) :765–781.
- Stawowy, A. (2008). Evolutionary based heuristic for bin packing problem. *Computers & Industrial Engineering*, 55(2) :465–474.



- 
- Talbi, E.-G. (2002). A taxonomy of hybrid metaheuristics. *Journal of heuristics*, 8(5) :541–564.
- Trilling, L. (2006). *Aide à la décision pour le dimensionnement et le pilotage de ressources humaines mutualisées en milieu hospitalier*. PhD thesis, INSA de Lyon.
- Valério de Carvalho, J. (1999). Exact solution of bin-packing problems using column generation and branch-and-bound. *Annals of Operations Research*, 86 :629–659.
- Valério de Carvalho, J. (2002). LP models for bin packing and cutting stock problems. *European Journal of Operational Research*, 141(2) :253–273.
- Van den Bergh, J., Beliën, J., De Bruecker, P., Demeulemeester, E., and De Boeck, L. (2013). Personnel scheduling : A literature review. *European Journal of Operational Research*, 226(3) :367–385.
- Van Oostrum, J. M., Van Houdenhoven, M., Hurink, J. L., Hans, E. W., Wullink, G., and Kazemier, G. (2008). A master surgical scheduling approach for cyclic scheduling in operating room departments. *OR spectrum*, 30(2) :355–374.
- VanBerkel, P. T. and Blake, J. T. (2007). A comprehensive simulation for wait time reduction and capacity planning applied in general surgery. *Health care management Science*, 10(4) :373–385.
- Vanderbeck, F. (1999). Computational study of a column generation algorithm for bin packing and cutting stock problems. *Mathematical Programming*, 86(3) :565–594.
- Wäscher, G. and Gau, T. (1996). Heuristics for the integer one-dimensional cutting stock problem : A computational study. *Operations-Research-Spektrum*, 18(3) :131–144.
- Wee, T. and Magazine, M. J. (1982). Assembly line balancing as generalized bin packing. *Operations Research Letters*, 1(2) :56–58.
- Yan, X., Zhang, C., Luo, W., Li, W., Chen, W., and Liu, H. (2012). Solve traveling salesman problem using particle swarm optimization algorithm. *International Journal of Computer Science*, 9(2012) :264–271.

## Détail des résultats

Les résultats détaillés de l'application de toutes les méthodes proposées pour le problème du bin packing et le problème de planification d'activités avec affectation de ressources sont présentés.

Pour chaque problème étudié et chaque méthode utilisée, deux tableaux sont donnés.

Dans un premier tableau, le résultat est composé de quatre valeurs pour chaque combinaison instance/méthode :

- la moyenne sur les quatre réplifications,
- la valeur minimale,
- la valeur maximale,
- l'écart-type.

Pour le problème du bin packing, les résultats correspondent au nombre de boîtes supplémentaires par rapport à la valeur optimale. Si ce nombre est nul, la valeur optimale a été trouvée. Pour le problème de planification d'activités avec affectation de ressources, les résultats correspondent à la fonction objectif  $H = 10^5 \cdot C_1 + C_2$  avec  $C_1$  le nombre de périodes limites non respectées et  $C_2$  la somme des périodes affectées.

Un second tableau résume le nombre de valeurs optimales trouvées par chacune des méthodes pour le problème du bin packing. Pour le second problème, le second tableau résume le meilleur couple  $(C_1; C_2)$  trouvé par chacune des méthodes pour chacune des instances.

Les tableaux qui ne sont pas remplis en intégralité le seront très prochainement.

Les notations utilisées sont les suivantes :

- Métaheuristique : RS pour recuit simulé, AK pour algorithme du kangourou,
- Ensemble de recherche :
  - $\Omega$  : la recherche est faite dans l'ensemble  $\Omega$  de toutes les solutions,
  - $C$  : la recherche est faite dans l'ensemble  $C$  des solutions admissibles.
- PSO : optimisation par essaim particulaire :
  - Déplacement : *PSO1* pour le déplacement 1, *PSO2* pour le déplacement 2.
  - Perturbation :
    - R* : dispersion avec influence du record,
    - M* : mutation,
    - MD* : mutation et dispersion.

- Heuristique : NF pour Next Fit, FF pour First Fit et BF pour Best Fit.
- Système de voisinage : P pour permutation  $P_{i,j}$ , I pour insertion.

## A.1 Problème du bin packing

### Métaheuristiques

Le Tableau A.1 donne les résultats de l'application des métaheuristiques basées individu au problème du bin packing sur toutes les instances de la littérature. Les méthodes proposées sont robustes : les écarts-types sont faibles.

Le Tableau A.2 résume le nombre de valeurs optimales trouvées. Les méthodes proposées sont globalement justes. Même si certaines valeurs optimales sont atteintes, ces méthodes en trouvent relativement peu. Si le critère d'arrêt de nos méthodes avaient été un temps de calcul plus long, les résultats s'en seraient trouvés améliorés. Le choix d'arrêter la recherche après une heure est un choix délibéré. L'objectif est de proposer des méthodes simples qui s'exécutent rapidement. Les résultats de l'application des métaheuristiques justifient la proposition du couplage entre les heuristiques et les métaheuristiques.

### Couplage heuristiques métaheuristiques

#### Métaheuristiques basée individu

Le Tableau A.3 donne les résultats du couplage du recuit simulé et des heuristiques. Le Tableau A.4 donne ceux du couplage de l'algorithme du kangourou et des heuristiques. Une fois encore, les méthodes proposées sont robustes : les écarts-types sont faibles.

Le Tableau A.5 résume le nombre de valeurs optimales trouvées par chacune des méthodes. Les résultats sont très bons. La majorité des instances traitées est résolue à l'optimalité en moins d'une heure de recherche. Un bémol peut être noté sur les instances t\_60 à t\_501. Peu de ces instances ont été résolues à l'optimalité. Ces instances sont particulièrement difficiles : la solution optimale est composée de boîtes contenant exactement trois objets.

Globalement, le recuit simulé est meilleur que l'algorithme du kangourou, l'heuristique Best Fit et First Fit sont équivalentes et meilleures que l'heuristique Next Fit, le système de voisinage permutation  $P_{i,j}$  est meilleur que le système de voisinage insertion.

#### Métaheuristique basée population

Le Tableau A.6 donne ces mêmes résultats pour l'heuristique First Fit. Le Tableau A.7 donne les résultats de l'application du couplage de l'optimisation par essaim particulaire et de l'heuristique Best Fit. Les méthodes proposées sont robustes.

Le Tableau A.8 résume le nombre de valeurs optimales trouvées par toutes ces méthodes. Les méthodes proposées donnent de bons résultats. Les méthodes utilisant l'heuristique First Fit sont plus performantes que celles utilisant l'heuristique Best Fit. First Fit est plus rapide que Best Fit et les temps de calcul sont limités à une heure.

Il est cependant regrettable que les résultats utilisant une métaheuristique basée population soient finalement moins bons que ceux utilisant les métaheuristiques basées individu.

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17
RS_C	1.26	2.73	6.39	22.34	1.85	3.35	6.14	12.59	1.35	1.2	2.95	1.19	1.16	0.82
	0	1	4	16	1	2	4	9	0	0	1	0	0	0
	2	5	8	29	3	4	9	15	12	11	4	2	2	1
	0.47	0.81	0.92	2.87	0.45	0.61	0.96	1.41	1.93	1.71	0.71	0.42	0.42	0.38
AK_C	0.96	2.64	6.63	23.3	1	1.91	5.23	12.79	1.33	1.11	2.05	0.92	0.81	0.82
	0	1	5	19	1	1	3	10	0	0	1	0	0	0
	2	4	9	28	1	3	7	15	12	11	4	2	2	1
	0.4	0.68	0.84	1.85	0	0.48	0.91	1.22	2.1	1.73	0.71	0.29	0.45	0.38
RS_Ω	1.4	3.94	12.04	36.19	1.91	3.25	7.29	16.61	3.27	2.84	3.8	1.21	1.25	1.21
	1	2	9	30	1	2	5	13	0	0	2	1	1	0
	3	6	17	42	3	5	9	20	19	16	6	2	2	5
	0.54	0.91	1.67	2.49	0.48	0.7	1.07	1.59	3.94	3.16	0.84	0.41	0.43	0.68
AK_Ω	1.46	4.11	12.39	34.18	1.06	2.43	6.96	16.86	3.11	2.96	3.58	1.61	1.63	1.34
	1	2	10	25	1	2	5	13	0	0	2	1	0	0
	3	6	16	41	2	4	10	20	18	14	5	3	3	4
	0.52	0.96	1.37	3.68	0.24	0.52	1.13	1.62	3.69	3.12	0.86	0.52	0.57	0.74

TABLEAU A.1 – Résultats de l'application des métaheuristiques basées individu

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1	TOTAL
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17	1587
RS_C	1	0	0	0	0	0	0	0	396	258	0	4	8	3	670
AK_C	4	0	0	0	0	0	0	0	423	281	0	19	41	3	771
RS_Ω	0	0	0	0	0	0	0	0	3	77	0	0	0	1	81
AK_Ω	0	0	0	0	0	0	0	0	2	63	0	0	2	1	68

TABLEAU A.2 – Nombre de solutions optimales trouvées par les métaheuristiques basées individu

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17
RS_NF_P	0.23	0.46	0.99	5.66	1.21	2.01	3.86	8.48	0.26	0.83	3.45	0.76	0.59	0.25
	0	0	0	3	1	1	2	5	0	0	2	0	0	0
	1	2	3	10	2	4	6	12	4	12	5	1	2	1
	0.42	0.52	0.78	1.43	0.41	0.66	0.98	1.38	0.59	1.56	0.8	0.43	0.5	0.43
RS_NF_I	0.21	1.66	3.54	9.96	1.01	1.53	2.96	6.43	0.64	0.32	0.8	0.28	0.11	0.49
	0	0	2	8	1	1	2	4	0	0	0	0	0	0
	1	3	5	12	2	2	4	8	7	5	2	1	1	1
	0.41	0.67	0.61	0.93	0.11	0.5	0.62	0.83	1.20	0.74	0.56	0.45	0.31	0.5
RS_FF_P	0.01	0.24	0.35	2.1	0.95	0.99	1.15	2.74	0.09	0.02	0.28	0	0	0.19
	0	0	0	1	0	0	1	1	0	0	0	0	0	0
	1	1	2	4	1	1	3	8	2	2	1	0	0	1
	0.11	0.43	0.5	0.62	0.22	0.11	0.5	2.02	0.29	0.14	0.45	0	0	0.39
RS_FF_I	0.24	1.18	3.13	9.9	1	1.01	2.08	6.24	0.47	0.17	0.35	0.04	0.02	0.24
	0	0	2	7	1	1	1	4	0	0	0	0	0	0
	1	2	5	15	1	2	3	8	6	4	1	1	1	1
	0.43	0.47	0.7	1.32	0	0.11	0.52	0.91	1.03	0.50	0.48	0.2	0.15	0.42
RS_BF_P	0.03	0.28	0.64	2.96	1	1	1.31	5.43	0.07	0.05	0.28	0	0.01	0.18
	0	0	0	1	1	1	1	2	0	0	0	0	0	0
	1	1	2	7	1	1	4	9	2	2	1	0	1	1
	0.16	0.45	0.51	0.81	0	0	0.68	2.13	0.28	0.22	0.45	0	0.05	0.38
RS_BF_I	0.16	0.98	2.81	8.99	1	1	2.8	7.91	0.27	0.2	0.5	0.06	0.04	0.25
	0	0	2	7	1	1	2	7	0	0	0	0	0	0
	1	2	4	11	1	1	4	10	7	4	1	1	1	1
	0.37	0.57	0.61	0.9	0	0	0.56	0.78	0.88	0.57	0.5	0.24	0.2	0.43

TABLEAU A.3 – Résultats de l'application du couplage du recuit simulé avec les heuristiques

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17
AK_NF_P	0.23	1	1.06	5.56	0.4	1	3.08	8.36	0.22	0.71	2.28	0.08	0.08	0.25
	0	1	0	3	0	1	1	5	0	0	1	0	0	0
	1	1	3	10	1	1	5	12	5	12	3	1	1	1
	0.42	0	0.7	1.36	0.49	0	0.82	1.35	0.56	1.57	0.6	0.27	0.26	0.43
AK_NF_I	0.78	0.89	3.41	9.63	1	1	1.96	6.36	0.44	0.26	0.35	0.03	0.03	0.49
	0	0	2	8	1	1	1	5	0	0	0	0	0	0
	2	2	6	12	1	1	3	8	7	6	1	1	1	1
	0.47	0.5	0.74	0.94	0	0	0.25	0.73	1.07	0.7	0.48	0.16	0.16	0.5
AK_FF_P	0	0.3	0.4	1.89	0.89	1	1	2.88	0.11	0.02	0.2	0	0	0.24
	0	0	0	1	0	1	1	1	0	0	0	0	0	0
	0	1	2	4	1	1	1	7	3	1	1	0	0	1
	0	0.46	0.51	0.69	0.32	0	0	2.18	0.35	0.15	0.4	0	0	0.42
AK_FF_I	0.18	1.19	3.25	9.96	1	1	2.09	6.34	0.54	0.17	0.35	0.02	0.02	0.24
	0	0	2	7	1	1	1	5	0	0	0	0	0	0
	1	2	6	16	1	1	3	8	9	4	1	1	1	1
	0.38	0.53	0.7	1.5	0	0	0.5	0.72	1.2	0.50	0.48	0.15	0.15	0.42
AK_BF_P	0.03	0.34	0.63	3.08	0.99	1	1.28	5.73	0.07	0.05	0.28	0	0.01	0.22
	0	0	0	2	0	1	1	2	0	0	0	0	0	0
	1	1	2	5	1	1	3	11	3	2	1	0	1	1
	0.16	0.47	0.53	0.61	0.11	0	0.59	2.37	0.29	0.22	0.45	0	0.07	0.41
AK_BF_I	0.16	0.99	2.74	9.21	1	1	2.94	8.29	0.27	0.21	0.5	0.07	0.04	0.29
	0	0	2	7	1	1	2	6	0	0	0	0	0	0
	1	2	4	12	1	1	5	11	7	6	1	1	1	1
	0.37	0.46	0.59	1.06	0	0	0.51	0.87	0.88	0.59	0.5	0.25	0.2	0.46

TABLEAU A.4 – Résultats de l'application du couplage de l'algorithme du kangourou et des heuristiques

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1	TOTAL
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17	1587
RS_NF_P	<b>20</b>	13	12	0	0	0	0	0	621	347	0	49	75	13	1150
RS_NF_I	16	2	0	0	0	0	0	0	518	397	4	86	94	12	1129
RS_FF_P	<b>20</b>	<b>16</b>	<b>15</b>	0	3	<b>1</b>	0	0	682	<b>474</b>	8	<b>100</b>	<b>100</b>	15	<b>1434</b>
RS_FF_I	17	1	0	0	0	0	0	0	580	431	7	98	98	13	1245
RS_BF_P	<b>20</b>	<b>16</b>	12	0	0	0	0	0	<b>690</b>	469	8	<b>100</b>	<b>100</b>	<b>16</b>	1431
RS_BF_I	17	6	0	0	0	0	0	0	642	428	7	<b>100</b>	99	13	1312
AK_NF_P	18	14	10	0	0	0	0	0	640	374	0	99	96	13	1264
AK_NF_I	7	6	0	0	0	0	0	0	589	408	8	98	98	9	1223
AK_FF_P	<b>20</b>	14	14	0	<b>7</b>	0	0	0	673	473	<b>9</b>	<b>100</b>	<b>100</b>	13	1423
AK_FF_I	18	2	0	0	0	0	0	0	572	431	8	99	98	13	1241
AK_BF_P	<b>20</b>	15	11	0	1	0	0	0	686	467	8	<b>100</b>	<b>100</b>	14	1422
AK_BF_I	17	4	0	0	0	0	0	0	641	426	6	98	98	12	1302

TABLEAU A.5 – Nombre de solutions optimales trouvées par le couplage entre les métaheuristiques basées individu et les heuristiques



Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17
FF_PSO1	1.45	3.33	6.58	13.64	1.64	2.88	6.09	12.71	1.29	0.81	2.73	0.96	0.82	0.78
	1	2	5	11	1	2	4	11	0	0	2	0	0	0
	2	4	8	15	2	4	7	15	8	9	4	1	1	1
	0.5	0.59	0.63	0.88	0.48	0.4	0.66	0.81	1.48	1.36	0.63	0.21	0.39	0.41
FF_PSO1R	0.94	3.23	6.44	13.58	1.08	2.1	5.75	12.79	1.24	0.68	2.33	0.47	0.2	0.76
	0	2	5	12	1	2	4	11	0	0	1	0	0	0
	2	4	8	16	2	3	7	15	12	8	3	1	1	1
	0.46	0.52	0.63	0.89	0.26	0.3	0.73	0.9	2.15	1.34	0.52	0.5	0.4	0.42
FF_PSO2	0.98	3.49	8.59	18.64	1.04	1.81	4.78	15.83	1.16	0.94	1.95	0.44	0.15	0.73
	0	1	5	16	1	1	2	8	0	0	0	0	0	0
	3	6	12	23	2	4	9	20	16	13	4	2	2	1
	0.92	1.17	1.16	1.26	0.19	0.96	2.08	2.71	2.63	1.97	1.4	0.51	0.38	0.44
FF_PSO2R	0.89	4.06	8.3	18.37	1.71	3.21	5.75	12.06	1.01	0.9	2.43	0.018	0.03	0.68
	0	2	6	16	1	1	2	7	0	0	0	0	0	0
	3	6	11	21	3	5	9	19	18	13	6	2	2	1
	1.14	0.9	1.12	1.23	0.57	0.82	2.21	4.09	2.45	1.92	1.91	0.15	0.21	0.47
FF_PSO2M	0.1125	1.89	6.04	14.38	0.99	1	2.94	11.33	0.76	0.49	0.35	0.01	0.02	0.41
	0	1	5	13	0	1	1	10	0	0	0	0	0	0
	1	3	8	16	1	1	5	13	11	7	1	1	1	1
	0.32	0.52	0.6	0.69	0.11	0	0.71	0.67	1.82	1.16	0.48	0.07	0.12	0.49
FF_PSO2MD	0.1	1.1	5.58	15.14	0.99	1.05	2.15	10.25	0.66	0.41	0.25	0.01	0.01	0.4
	0	0	4	13	0	1	1	8	0	0	0	0	0	0
	1	2	7	18	2	3	4	12	10	6	1	1	1	1
	0.3	0.54	0.65	1.3	0.25	0.27	0.69	0.78	1.68	1.03	0.43	0.07	0.09	0.49

TABLEAU A.6 – Résultats de l'application du couplage de la PSO et de l'heuristique First Fit

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17
BF_PSO1	1.25	2.79	5.43	11.91	1.58	2.85	6.03	12.76	0.81	0.79	2.68	0.94	0.81	0.79
	1	2	4	10	1	2	5	11	0	0	2	0	0	0
	2	4	7	15	2	4	7	14	11	8	4	1	2	1
	0.43	0.61	0.63	1.02	0.49	0.42	0.65	0.9	1.76	1.3	0.52	0.23	0.4	0.4
BF_PSO1R	0.81	2.6	5.48	11.8	1.05	2.26	6	13.01	0.71	0.66	2.72	0.69	0.5	0.76
	0	2	4	10	1	2	5	11	0	0	2	0	0	0
	2	4	7	14	2	3	7	15	11	8	4	1	1	1
	0.42	0.56	0.69	0.9	0.22	0.44	0.55	0.84	1.77	1.26	0.55	0.46	0.5	0.42
BF_PSO2	0.45	2.74	7.75	17.38	1.11	2.09	6.43	17.48	0.82	0.95	2.6	0.07	0.14	0.72
	0	0	4	14	1	1	2	12	0	0	0	0	0	0
	2	6	10	20	3	5	10	22	16	13	6	2	2	1
	0.69	1.12	0.96	1.25	0.35	1.09	2.11	1.44	2.46	2.08	1.53	0.29	0.37	0.45
BF_PSO2R	0.4	2.51	7.06	17	1.61	3.18	5.24	16.24	0.73	0.9	3.15	0.07	0.09	0.65
	0	0	4	13	1	1	2	8	0	0	0	0	0	0
	2	6	11	20	3	5	10	21	15	12	5	2	2	1
	0.77	1.32	1.42	1.31	0.51	1.08	2.25	2.65	2.26	1.99	1.41	0.29	0.38	0.48
BF_PSO2M	0.14	1.48	4.9	13.12	1	1.01	4.03	12.19	0.48	0.5	0.93	0.02	0.03	0.54
	0	0	4	12	1	1	3	11	0	0	0	0	0	0
	1	2	6	15	1	2	5	14	10	7	2	1	1	1
	0.34	0.52	0.54	0.65	0	0.11	0.47	0.69	1.56	1.18	0.52	0.14	0.16	0.5
BF_PSO2MD	0.11	1.19	4.81	13.78	1	1.05	3.38	11.23	0.43	0.44	0.48	0.01	0.02	0.5
	0	0	3	12	1	1	2	9	0	0	0	0	0	0
	1	2	7	17	1	2	5	13	10	7	2	1	1	1
	0.32	0.45	0.92	1.3	0	0.22	0.75	0.84	1.45	1.05	0.55	0.1	0.15	0.5

TABLEAU A.7 – Résultats de l'application du couplage de la PSO et de l'heuristique Best Fit

Instance	u_120	u_250	u_500	u_1000	t_60	t_120	t_249	t_501	set_1	set_2	set_3	was_1	was_2	gau_1	TOTAL
Référence	20	20	20	20	20	20	20	20	720	480	10	100	100	17	1587
FF_PSO1	0	0	0	0	0	0	0	0	313	291	0	8	34	4	650
FF_PSO1R	4	0	0	0	0	0	0	0	429	324	0	63	84	4	908
FF_PSO2	12	0	0	0	0	0	0	0	536	341	4	71	91	5	1060
FF_PSO2R	13	0	0	0	0	0	0	0	559	354	4	99	99	8	1136
FF_PSO2M	18	0	0	0	1	0	0	0	537	375	<b>8</b>	<b>100</b>	<b>100</b>	<b>13</b>	1152
FF_PSO2MD	18	<b>4</b>	0	0	<b>3</b>	0	0	0	567	389	<b>8</b>	<b>100</b>	<b>100</b>	<b>13</b>	1202
BF_PSO1	0	0	0	0	0	0	0	0	499	288	0	12	34	4	837
BF_PSO1R	7	0	0	0	0	0	0	0	552	321	0	43	68	4	995
BF_PSO2	15	2	0	0	0	0	0	0	601	348	2	98	91	6	1163
BF_PSO2R	16	2	0	0	0	0	0	0	613	356	2	98	96	7	1190
BF_PSO2M	<b>19</b>	1	0	0	0	0	0	0	615	<b>393</b>	3	<b>100</b>	99	9	1239
BF_PSO2MD	18	1	0	0	0	0	0	0	<b>629</b>	381	<b>8</b>	<b>100</b>	99	9	<b>1245</b>

TABLEAU A.8 – Nombre de solutions optimales trouvées par l'application du couplage de la PSO et des heuristiques

## A.2 Problème de planification d'activités avec affectation de ressources

### Métaheuristique

Le Tableau A.9 présente les résultats de l'application du recuit simulé et de l'algorithme du kangourou à la résolution du modèle 1. Les méthodes proposées sont normalement robustes. Cela n'est pas toujours visible à cause de la pondération des critères. En effet, un écart d'une unité sur le critère relatif au nombre de dates limites non respectées se traduit par un écart de 100 000 sur la valeur de la fonction objectif, puisque ce critère a une pondération de  $10^5$ . C'est pourquoi, la valeur de l'écart-type peut paraître impressionnante.

Le Tableau A.10 résume les meilleurs résultats trouvés. Pour les instances de petite taille, toutes les métaheuristicques testées trouvent la valeur optimale. Plus la taille de l'instance est importante, plus la solution retournée par nos méthodes est éloignée de la solution référence. L'application directe des métaheuristicques ne semblent pas adaptée à notre problème en un temps restreint. Lorsque la recherche est faite dans l'ensemble  $C$  des solutions admissibles, la métaheuristique n'arrive pas à améliorer la solution initiale sans pénaliser les contraintes, en particulier pour l'instance 500B.

### Couplage heuristiques métaheuristicques

#### Métaheuristicques basée individu

Le Tableau A.11 présente les résultats du couplage du recuit simulé et des heuristiques. Le Tableau A.12 présente ceux du couplage de l'algorithme du kangourou et des heuristiques. Les méthodes proposées sont robustes.

Le Tableau A.13 résume les meilleures valeurs trouvées par chacune des méthodes. Il montre les bons résultats du couplage des heuristiques et des métaheuristicques basées individu. Les résultats du couplage des heuristiques et des métaheuristicques sont meilleurs que les résultats des métaheuristicques. En fonction du critère étudié, le couplage heuristique métaheuristique peut donner de meilleurs résultats que la résolution par méthode exacte avec le solveur CPLEX.

#### Métaheuristique basée population

Le Tableau A.14 présente les résultats de l'application du couplage de l'optimisation par essaim particulaire et de l'heuristique First Fit. Le Tableau A.15 présente ceux de l'application du couplage de l'optimisation par essaim particulaire et de l'heuristique Best Fit. Les méthodes proposées sont robustes.

Le Tableau A.16 résume les meilleurs résultats trouvés par chacune des méthodes. Il montre les bons résultats du couplage des heuristiques et de l'optimisation par essaim particulaire. De la même façon que pour le problème du bin packing, on peut cependant regretter que les résultats utilisant une métaheuristique basée population soient finalement moins bons que ceux utilisant les métaheuristicques basées individu.

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	100150	517	1131	206394
RS_C	51	131	266.5	551.25	893.75	1365.5	3550153.5	9575529	22201201	19408218
	<b>51</b>	<b>131</b>	<b>266</b>	549	891	1356	<b>1000147</b>	<b>2000535</b>	<b>3001197</b>	<b>19408218</b>
	51	131	267	554	896	1375	6600153	16500518	47001212	19408218
	0	0	0.5	2.28	1.92	6.95	2227668.43	5933530.09	15808864.15	0
AK_C	51	132	268	556.75	5300881	12601357.25	11550148	29850534.5	52101238	19408218
	<b>51</b>	132	267	554	200880	801352	1200148	2400535	5601225	<b>19408218</b>
	51	132	269	559	9000865	26401357	22000146	57200548	96801265	19408218
	0	0	0.71	1.92	3753657.78	9110434.95	7759348.75	20415376.78	33540885.94	0
RS_Ω	51	131	266.25	550.75	898.5	1374	10900148.75	23425518	54751197.25	119656644.75
	<b>51</b>	<b>131</b>	<b>266</b>	<b>548</b>	<b>890</b>	<b>1350</b>	2600152	10400525	22101194	47006733
	51	131	267	554	908	1385	16500144	33000518	93501207	186306507
	0	0	0.43	2.17	6.73	14.02	5880897.87	8551129.84	27260826.88	52893333.33
AK_Ω	51	132.75	272.5	569.25	5350901.5	21001348.75	18600161.25	54000572.25	181201110.5	124356758.25
	<b>51</b>	132	272	566	916	8001366	6000159	17000570	85001049	23206706
	51	133	273	575	15600891	31201344	31200157	93600561	285001034	232006784
	0	0.43	0.5	3.49	6068559.51	10061798.79	9391483.38	28389076.97	73268114.74	78412062.71

TABLEAU A.9 – Résultats de l'application des métaheuristiques basées individu

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	(1 ; 150)	517	1131	(2 ; 6394)
RS_C	51	<b>131</b>	<b>266</b>	549	891	1356	<b>(10 ; 147)</b>	<b>(20 ; 535)</b>	<b>(30 ; 1197)</b>	<b>(194 ; 8218)</b>
AK_C	51	132	267	554	(2 ; 880)	(8 ; 1352)	(12 ; 148)	(24 ; 535)	(56 ; 1225)	<b>(194 ; 8218)</b>
RS_Ω	51	<b>131</b>	<b>266</b>	<b>548</b>	<b>890</b>	<b>1350</b>	(26 ; 152)	(104 ; 525)	(221 ; 1194)	(470 ; 6733)
AK_Ω	51	132	272	566	916	(80 ; 1366)	(60 ; 159)	(170 ; 570)	(850 ; 1049)	(232 ; 6706)

TABLEAU A.10 – Meilleurs résultats de l'application des métaheuristiques basées individu

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	100150	517	1131	206394
RS_NF_P	51	131	266	536	838	1241.25	175147.25	300502.75	401113	1906588
	51	131	<b>266</b>	536	837	1241	100153	300501	301117	<b>1806659</b>
	51	131	266	536	839	1242	200146	300506	501108	2106393
	0	0	0	0	0.71	0.43	43297.95	2.05	70707.5	122359.16
RS_NF_I	51	131	266	536.75	840	1250.75	250147.75	575503.5	1201145.5	7281329
	51	131	<b>266</b>	536	839	1248	200150	500501	1001147	6606297
	51	131	266	537	841	1254	300147	600511	1301146	8306431
	0	0	0	0.43	1	2.17	49997.75	43302.71	122471.02	626054.47
RS_FF_P	51	131	266	535	835.75	1235.75	100150	25517.25	1136	2256151.25
	51	131	<b>266</b>	535	<b>835</b>	1235	<b>100150</b>	<b>518</b>	<b>1135</b>	2206200
	51	131	266	535	836	1237	100150	100513	1138	2306102
	0	0	0	0	0.43	0.83	0	43298.82	1.22	49938.75
RS_FF_I	51	131	266	534.5	837	1235	100150	75513.25	26135.75	5431384.5
	51	131	<b>266</b>	<b>534</b>	836	<b>1234</b>	<b>100150</b>	519	1137	5006368
	51	131	266	535	838	1236	100150	100512	101131	5706405
	0	0	0	0.5	0.71	0.71	0	43297.95	43298.53	294763.76
RS_BF_P	51	131	266.25	537.5	847	1326.25	100150	25519.5	176135.25	3606461.75
	51	131	<b>266</b>	536	843	1306	<b>100150</b>	521	1143	3406479
	51	131	267	539	852	1342	100150	100514	301126	3706499
	0	0	0.43	1.12	3.24	14.57	0	43298.09	108966.8	122468.77
RS_BF_I	51	131	266	542	862.25	51314.25	100150	100512.5	451138.5	8831792
	51	131	<b>266</b>	538	857	1304	<b>100150</b>	522	301147	8406731
	51	131	266	548	866	201318	100150	200504	601129	9306981
	0	0	0	3.67	3.27	86604.71	0	70704.31	111796.69	426532.13

TABLEAU A.11 – Résultats de l'application du couplage du recuit simulé et des heuristiques

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	100150	517	1131	206394
AK_NF_P	51	131	266	541.25	850.5	1258.75	150154.25	225529	376180.75	2006472.25
	51	131	<b>266</b>	541	850	1256	100157	200526	301176	<b>1906382</b>
	51	131	266	542	852	1262	200149	300525	501187	2106608
	0	0	0	0.43	0.87	2.38	49994.75	43298.96	82919.77	70790.59
AK_NF_I	51	131	266	541.25	850.5	1261.75	200148.75	575522.25	1551151.25	9881368.75
	51	131	<b>266</b>	540	850	1259	200148	500528	1501149	9206280
	51	131	266	542	851	1266	200150	600523	1701138	10306427
	0	0	0	0.83	0.5	2.95	0.83	43297.95	86594.89	414635.55
AK_FF_P	51	131	266	537.5	841.75	1244	100151	100518.25	76146	2306173.75
	51	131	<b>266</b>	<b>537</b>	<b>841</b>	<b>1241</b>	<b>100151</b>	<b>100516</b>	<b>1154</b>	2006216
	51	131	266	538	842	1246	100151	100521	101144	2706239
	0	0	0	0.5	0.43	1.87	0	1.79	43296.65	273878.45
AK_FF_I	51	131	266	537.75	842.5	1244.75	100151	200513	101153	5306411.25
	51	131	<b>266</b>	<b>537</b>	842	1244	<b>100151</b>	200511	1162	5006397
	51	131	266	538	843	1245	100151	200514	201148	5606420
	0	0	0	0.43	0.5	0.43	0	1.22	70705.73	300008.75
AK_BF_P	51	131	267	542.5	852	1346	100151.75	175519.25	201160.25	3881469
	51	131	<b>266</b>	542	850	1334	<b>100151</b>	100532	201155	3606477
	51	131	268	543	853	1369	100152	200516	201167	4106495
	0	0	0.71	0.5	1.22	14.3	0.43	43293.91	4.44	178539.63
AK_BF_I	51	131	267	542.5	873.25	101323.75	100151.75	250519.25	576148.5	8781706.75
	51	131	267	542	857	1302	<b>100151</b>	200524	501152	7906667
	51	131	267	544	883	201359	100152	300512	601152	9906817
	0	0	0	0.87	9.81	100017.75	0.43	49989.75	43299.25	759595.68

TABLEAU A.12 – Résultats de l'application du couplage de l'algorithme du kangourou et des heuristiques



Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	(1 ; 150)	517	1131	(2 ; 6394)
RS_NF_P	51	131	<b>266</b>	536	837	1241	(1 ; 153)	(3 ; 501)	(3 ; 1117)	<b>(18 ; 6659)</b>
RS_NF_I	51	131	<b>266</b>	536	839	1248	(2 ; 150)	(5 ; 501)	(10 ; 1147)	(66 ; 6297)
RS_FF_P	51	131	<b>266</b>	535	<b>835</b>	1235	<b>(1 ; 150)</b>	<b>518</b>	<b>1135</b>	(22 ; 6200)
RS_FF_I	51	131	<b>266</b>	<b>534</b>	836	<b>1234</b>	<b>(1 ; 150)</b>	519	1137	(50 ; 6368)
RS_BF_P	51	131	<b>266</b>	536	843	1306	<b>(1 ; 150)</b>	521	1143	(34 ; 6479)
RS_BF_I	51	131	<b>266</b>	538	857	1304	<b>(1 ; 150)</b>	522	(3 ; 1147)	(84 ; 6731)
AK_NF_P	51	131	<b>266</b>	541	850	1256	(1 ; 157)	(2 ; 526)	(3 ; 1176)	(19 ; 6382)
AK_NF_I	51	131	<b>266</b>	540	850	1259	(2 ; 148)	(5 ; 528)	(15 ; 1149)	(92 ; 6280)
AK_FF_P	51	131	<b>266</b>	537	841	1241	(1 ; 151)	(1 ; 516)	1154	(20 ; 6216)
AK_FF_I	51	131	<b>266</b>	537	842	1244	(1 ; 151)	(2 ; 511)	1162	(50 ; 6397)
AK_BF_P	51	131	<b>266</b>	542	850	1334	(1 ; 151)	(1 ; 532)	(2 ; 1155)	(36 ; 6477)
AK_BF_I	51	131	267	542	857	1302	(1 ; 151)	(2 ; 524)	(5 ; 1152)	(78 ; 6667)

TABLEAU A.13 – Meilleurs résultats de l'application du couplage des métaheuristiques basées individu et des heuristiques

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	100150	517	1131	206394
FF_PSO1	51.75	141.25	75287.5	300600.5	700979	1101484.75	800158.75	1775561.75	2751230.5	10106789.5
	51	140	280	595	500982	901478	600159	1600539	2601237	10006789
	53	143	100291	400605	900985	1301484	1100150	1900573	2901256	10206740
	0.83	1.09	43305.6	173208.26	141422.42	141423.48	187077.93	108985.49	111804.97	70693.38
FF_PSO1R	51.25	25138.75	25282	25587.5	475969.25	1226479.75	925158.25	1475565.5	2026227.5	10431815.25
	51	136	281	586	300964	1101483	800154	1200553	1801236	9006761
	52	100139	100279	100583	900983	1301519	1200165	1600561	2201240	11106722
	0.43	43301.41	43299.54	43298.67	248753.92	82924.74	163939.89	163940.16	147901.4	834811.38
FF_PSO2	51	137	274.5	557.75	876	51313	175150.75	1175548	901215.25	4031521.5
	51	137	272	556	863	1303	100156	900551	801212	3106465
	51	137	278	561	887	101323	200150	1500540	1101202	4906575
	0	0	2.18	2.05	8.66	50005.5	43298.24	216502.31	122467.96	637905.93
FF_PSO2R	51	131	266	537.5	857.25	1283	100151.25	527.25	1154.75	3481520.75
	51	<b>131</b>	<b>266</b>	<b>537</b>	853	1279	<b>100151</b>	<b>526</b>	<b>1152</b>	3006523
	51	131	266	538	863	1286	100152	530	1160	4806506
	0	0	0	0.5	3.77	2.55	0.43	1.64	3.27	766067.39
FF_PSO2M	51	131.25	266	539.25	848	1260.75	125150.25	100522	26158.75	2706587.25
	51	<b>131</b>	<b>266</b>	538	<b>844</b>	<b>1256</b>	<b>100151</b>	527	<b>1152</b>	<b>2606530</b>
	51	132	266	540	855	1266	200145	200513	101159	2806635
	0	0.43	0	0.83	4.18	3.96	43298.24	70705.73	43301.41	100023.25
FF_PSO2MD	51	131	266	540.25	848.75	1262	200148.25	125526	100844.25	2706539.5
	51	<b>131</b>	<b>266</b>	539	847	1260	100154	531	1164	2606575
	51	131	266	542	852	1263	300145	200523	200523	2806526
	0	0	0	1.09	1.92	1.22	70707.5	82910.19	70484.39	70693.36

TABLEAU A.14 – Résultats de l'application du couplage de la PSO et de l'heuristique First Fit

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	100150	517	1131	206394
BF_PSO1	51.5	142.25	287.75	200605.25	650981.5	1151500.5	900162.75	1725577.5	2776253.5	10981894
	51	139	281	100609	200976	901501	800155	1700562	2501251	10506974
	52	146	291	300600	1200979	1501490	1000171	1800583	2901254	11706881
	0.5	2.86	4.09	99993.25	357071.91	229120.49	100004.75	43304.45	163936.34	476304.11
BF_PSO1R	51	139	283.25	100597.75	675984	1151482.75	1000163.25	1850573.25	2226253.75	10656896.25
	51	138	279	596	400983	701457	900159	1700563	2001252	10306877
	51	140	285	200608	1000989	1701472	1200169	2100579	2601239	11006923
	0	0.71	2.49	70714.92	238489.41	384053.28	122478.16	165834.03	248742.56	304175.82
BF_PSO2	51.25	138.75	276.75	557.75	891	1326.75	175157.5	1000556	901223.5	3906644.5
	51	138	272	556	885	1315	100160	800560	501238	3606693
	52	140	282	559	896	1346	200163	1200559	1101220	4106633
	0.43	0.83	3.56	1.09	4.06	12.11	43299.83	141421	244938.97	187064.83
BF_PSO2R	51	131	266.25	546.25	876.5	1351.25	100151	532.75	26178.25	4881709.25
	51	<b>131</b>	<b>266</b>	543	871	1319	<b>100151</b>	<b>531</b>	1174	4106721
	51	131	267	549	884	1373	100151	536	101168	5306669
	0	0	0.43	2.17	5.02	19.83	0	2.05	43295.35	471015.36
BF_PSO2M	51	132.25	267.25	542	862.25	1314.75	175149.75	150515.5	76182.5	4356718.5
	51	132	267	<b>540</b>	<b>856</b>	<b>1302</b>	<b>100151</b>	100520	<b>1171</b>	<b>3606643</b>
	51	133	268	544	869	1335	200153	200513	101191	4906647
	0	0.43	0.43	1.58	4.6	13.5	43300.55	49994.5	43307.91	471716.23
BF_PSO2MD	51	131	268	543	865.25	1326.5	175151.75	150520.75	201181	4481650.25
	51	<b>131</b>	268	542	863	1314	<b>100151</b>	100516	101182	3806601
	51	131	268	545	867	1336	200154	200520	301191	6006778
	0	0	0	1.22	1.48	8.99	43301.7	49998.75	70713.86	887134.19

TABLEAU A.15 – Résultats de l'application du couplage de la PSO et de l'heuristique Best Fit

Instance	50A	100A	200A	300A	400A	500A	50B	100B	200B	500B
Référence	51	131	266	533	830	1225	(1 ; 150)	517	1131	(2 ; 6394)
FF_PSO1	51	140	280	595	(5 ; 982)	(9 ; 1478)	(6 ; 159)	(16 ; 539)	(26 ; 1237)	(100 ; 6789)
FF_PSO1R	51	136	281	586	(3 ; 964)	(11 ; 1483)	(8 ; 154)	(12 ; 553)	(18 ; 1236)	(90 ; 6761)
FF_PSO2	51	137	272	556	863	1303	(1 ; 156)	(9 ; 551)	(8 ; 1212)	(31 ; 6465)
FF_PSO2R	51	<b>131</b>	<b>266</b>	<b>537</b>	853	1279	<b>(1 ; 151)</b>	<b>526</b>	<b>1152</b>	(30 ; 6523)
FF_PSO2M	51	<b>131</b>	<b>266</b>	538	<b>844</b>	<b>1256</b>	<b>(1 ; 151)</b>	527	<b>1152</b>	<b>(26 ; 6530)</b>
FF_PSO2MD	51	<b>131</b>	<b>266</b>	539	847	1260	(1 ; 154)	531	1164	(26 ; 6575)
BF_PSO1	51	139	281	(1 ; 609)	(2 ; 976)	(9 ; 1501)	(8 ; 155)	(17 ; 562)	(25 ; 1251)	(105 ; 6974)
BF_PSO1R	51	138	279	596	(4 ; 983)	(7 ; 1457)	(9 ; 159)	(17 ; 563)	(20 ; 1252)	(103 ; 6877)
BF_PSO2	51	138	272	556	885	1315	(1 ; 160)	(8 ; 560)	(5 ; 1238)	(36 ; 6693)
BF_PSO2R	51	<b>131</b>	<b>266</b>	543	871	1319	<b>(1 ; 151)</b>	531	1174	(41 ; 6721)
BF_PSO2M	51	132	267	540	856	1302	<b>(1 ; 151)</b>	(1 ; 520)	1171	(36 ; 6643)
BF_PSO2MD	51	<b>131</b>	268	542	863	1314	<b>(1 ; 151)</b>	(1 ; 516)	(1 ; 1182)	(38 ; 6601)

TABLEAU A.16 – Meilleurs résultats de l'application du couplage de la PSO avec une heuristique

