



HAL
open science

**Contribution à l'optimisation de la chaîne logistique
pharmaceutique dans un contexte multi-sites :
application au groupe hospitalier de la région Mulhouse
et sud Alsace**

Benoit Beroule

► **To cite this version:**

Benoit Beroule. Contribution à l'optimisation de la chaîne logistique pharmaceutique dans un contexte multi-sites : application au groupe hospitalier de la région Mulhouse et sud Alsace. Autre. Université Bourgogne Franche-Comté, 2018. Français. NNT : 2018UBFCA013 . tel-01876744

HAL Id: tel-01876744

<https://theses.hal.science/tel-01876744>

Submitted on 18 Sep 2018

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.

THÈSE DE DOCTORAT DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ

PRÉPARÉE À L'UNIVERSITÉ DE TECHNOLOGIE DE BELFORT-MONTBÉLIARD

École doctorale n°37

Sciences Pour l'Ingénieur et Microtechniques

Doctorat d'Informatique

par

BENOÎT BEROULE

Contribution à l'optimisation de la chaîne logistique pharmaceutique dans un contexte multi-sites : application au Groupe Hospitalier de la Région de Mulhouse et Sud Alsace

Thèse présentée et soutenue à Belfort, le 10 juillet 2018

Composition du Jury :

ABED MOURAD	Professeur à l'Université de Valenciennes	Président du jury
ARTIBA ABDELHAKIM	Professeur à l'Université de Valenciennes	Rapporteur
AUJOULAT OLIVIER	Chef de pôle de la pharmacie de Mulhouse	Invité
BARAKAT OUSSAMA	Maitre de Conférences, HDR, Université de Franche compté	Codirecteur de thèse
GRUNDER OLIVIER	Maitre de Conférences, HDR, Université de Technologie de Belfort-Montbéliard	Directeur de thèse
HAMMADI SLIM	Professeur à l'école centrale de Lille	Rapporteur

Titre : Contribution à l'optimisation de la chaîne logistique pharmaceutique dans un contexte multi-sites : application au Groupe Hospitalier de la Région de Mulhouse et Sud Alsace

Mots-clés : hôpital,logistique,informatique,multi-site,groupement hospitalier, metaheuristique

Résumé :

Ce travail de thèse a pour but d'étudier et d'améliorer la logistique du groupement hospitalier de Mulhouse sud-Alsace. Il est donc divisé en deux parties distinctes ayant pour dénominateur commun la gestion des dispositifs médicaux.

En effet, dans un premier temps nous envisageons l'optimisation des méthodes d'implantation pour le stockage de ces dispositifs pour minimiser la distance moyenne parcourue par les agents lors de la phase de collecte. Afin d'affiner ce travail, et

de se projeter dans l'avenir, nous proposons aussi une méthode d'optimisation des tournées de collecte d'une liste de dotations au sein du stock.

Dans un second temps, nous proposons des méthodes de planification des interventions chirurgicales se focalisant d'avantage sur l'assignation des boites de dispositifs médicaux pour lisser la charge de travail du service de stérilisation. De plus, pour compléter ce travail, nous présentons une méthode d'ordonnancement qui tient compte également de la livraison de ces boites.

Title: Contribution à l'optimisation de la chaîne logistique pharmaceutique dans un contexte multi-sites : application au Groupe Hospitalier de la Région de Mulhouse et Sud Alsace

Keywords: hospital, logistic, computer science, multi-sites, hospital grouping, metaheuristic

Abstract:

The purpose of this thesis is to study and improve the logistical aspects of the hospital group named "Groupement Hospitalier de Mulhouse Sud-Alsace" (GHRMSA). This work is divided into two different parts on the management of the surgical devices

Indeed, as a first step we study the optimization of the layout to store surgical devices to minimize the mean distance traveled by the employees during the picking step. In order to improve this work and plan the future, we also propose methods to optimize the

routing of the employees to completes endowment lists using a transport vehicle.

As a second step, we propose surgical operations scheduling methods. This work focuses on surgical boxes assignment in order to smooth the sterilization workload. Moreover, we extend this problem with the transport between hospital locations, and present a method to create plannings for surgical operations and for the delivery of the surgical boxes to minimize the global cost considering the parameters of the transport system.

Cette thèse est dédiée aux artistes morts pendant sa rédaction qui m'ont marqués au cours de ma vie et fait de moi l'homme que je suis aujourd'hui.

-Jean Rochefort

-Carrie Fisher

-Debbie Reynolds

-Alan Rickman

-Christopher Lee

-George Romero

REMERCIEMENTS

Vous entrez dans la section de ce mémoire de thèse dédiée aux remerciements. Armés uniquement de votre courage et de votre pugnacité légendaire vous décidez de chercher votre chemin à travers les méandres de cette partie (allez en **1**).

(1)

Votre périple débute au paragraphe où je remercie les membres du jury, à savoir monsieur Abed Mourad et monsieur Artiba Abdelhakim tous deux professeurs à l'université de Valenciennes, ainsi que monsieur Hamadi Slim professeur à l'école Centrale de Lille et monsieur Aujoulat Olivier chef de pôle de la pharmacie de l'hôpital de Mulhouse. Merci à eux d'avoir donné de leur temps et de leur expertise dans l'évaluation de mon travail.

A présent deux chemins s'offrent à vous : choisissez vous le chemin familial (allez en **2**) ou le chemin professionnel (allez en **3**).

(2)

Vous pénétrez à présent dans les remerciements familiaux. C'est ici que je remercie mes parents pour m'avoir mis au monde et m'avoir supporté chez eux jusqu'à un âge "avancé". Je remercie bien sûr ma compagne Karol pour avoir été présente lorsque j'en avais besoin et pour avoir corrigé en partie mon travail. Je n'oublie pas les autres membres de ma famille qui m'ont permis de répondre souvent aux questions : "mais qu'est ce que c'est qu'une thèse ?" et "mais tu veux pas faire un vrai travail?". Enfin je remercie ma chatte Ragnout d'avoir toujours su m'offrir du réconfort généralement non-consenti et non-réciproque.

Vous faites maintenant face à 3 chemins : choisissez vous le chemin des corrections du manuscrit (allez en **7**), le chemin de l'hôpital (allez en **4**) ou l'étrange sentier qui ne semble mener nul part (allez en **5**) ?

(3)

Vous accédez alors aux remerciements adressés à mon directeur et co-directeur de thèse. A ce titre je remercie vivement Olivier Grunder et Oussama Barakat qui m'ont

énormément aidé et aiguillé et sans qui rien de tout cela n'aurait été possible.

Vous faites maintenant face à 3 chemins : choisissez vous le chemin des corrections du manuscrit (allez en **7**), le chemin de l'hôpital (allez en **4**) ou l'étrange sentier qui ne semble mener nul part (allez en **5**) ?

(4)

Vous parvenez aux remerciements des cadres de la pharmacie de l'hôpital de Mulhouse. Je remercie tout particulièrement Sophie Ligner (responsable de la CAMPS) et Hélène Lustig (responsable du service de stérilisation) qui m'ont offert un accueil chaleureux et m'ont aidé à mieux comprendre le fonctionnement de l'hôpital.

Vous pouvez continuer votre route (Allez en **6**) ou prendre un étrange passage sans indication (Allez en **5**).

(5)

Vous découvrez un lieu visiblement oublié de tous. Ici, pas de remerciement, la seule chose que vous trouvez est une clef rouillée que vous vous empressez de mettre dans votre poche.

Vous faites maintenant face à 2 chemins : choisissez vous le chemin des amis (allez en **8**) ou des collègues de l'hôpital (allez en **7**).

(6)

Vos pas vous mènent aux remerciement des collègues de l'hôpital. Ici je remercie chaleureusement Stéphanie, Antoine et Valérie grâce à qui j'ai pu mener de longues discussions sur des sujets futiles pendant les heures de travail (et avec qui je ne reparlerai sans doute plus jamais politique).

Vous faites maintenant face à 2 chemins : choisissez vous le chemin des amis (allez en **9**) ou celui des séries animées (allez en **11**).

(7)

Vous entrez à présent dans la partie liée aux remerciements de mes amis qui ont eu le courage de relire l'intégralité de ce présent mémoire afin d'en vérifier l'orthographe et les coquilles en tout genre. C'est pourquoi je remercie Clémence et Morgane sans qui ma dysorthographe auto-diagnostiquée ne m'aurait pas permis d'écrire ces lignes.

Vous pouvez continuer votre route (allez en **8**) ou prendre un étrange passage sans indication (allez en **5**).

(8)

Vous vous trouvez à présent dans la section vouée à remercier les amis qui ont voulu m'aider mais dont la raison fut plus forte que le cœur. Je remercie donc Pierre G. et Pierre B. dont j'attends encore les corrections avec impatience.

Vous faites maintenant face à 2 chemins : choisirez vous le chemin des amis (allez en **9**) ou celui des collègues laborantins (allez en **10**).

(9)

Vous accédez à présent aux remerciements des amis encore non-cités. Ici je remercie Renaud et Pauline (ainsi que mes autres amis précédemment cités) pour les nombreuses heures de bonheur qu'ils m'ont apporté ainsi que les nombreuses histoires complexes de relations humaines qu'ils m'ont fait vivre.

Vous faites maintenant face à 3 chemins : choisirez vous le chemin des collègues laborantins (allez en **10**), le chemin des séries animées (allez en **11**) ou prenez vous directement la sortie (allez en **12**) ?

(10)

Vous pénétrez dans la section des remerciements adressés aux collègues du laboratoire et ses environs. Je remercie Jeremy, Esmaeil, Yong Shi, Citlalli, Justine et tous ceux dont je ne peux écrire les noms par manque de place. Grâce à eux, la vie de doctorant s'est révélée plaisante et pleine de surprises.

Il ne vous reste plus qu'à rejoindre la fin de la route (Allez en **12**).

(11)

Vous arrivez à une section liée à des remerciements un peu spéciaux. En effet, j'adresse un grand merci à Matt Groening, Trey Parker et Matt Stone dont les immenses talents pour l'écriture de séries animées m'ont permis de garder le moral et la force de continuer quand j'en avais le plus besoin.

Il ne vous reste plus qu'à rejoindre la fin de la route (Allez en **12**).

(12)

Félicitation vous êtes parvenus à la fin des remerciements. Devant vous se tient une large porte menant à la suite de ce mémoire, si vous possédez la clef rouillée vous pouvez l'utiliser pour poursuivre votre lecture, dans le cas contraire il vous faut recommencer les remerciements depuis le début et tenter de prendre un chemin différent.

SOMMAIRE

1	Introduction générale	1
1.1	Contexte de la thèse	4
1.1.1	L'Hôpital Emile Muller	4
1.1.2	De la pharmacie de l'hôpital à la pharmacie de territoire	5
1.2	État de l'art	6
1.3	La logistique pharmaceutique dans un contexte hospitalier	9
1.3.1	Le service de stérilisation	10
1.3.2	Analyse du fonctionnement du service de stérilisation	16
1.3.2.1	Le processus de pré-lavage	16
1.3.2.2	Le processus de reconditionnement	18
1.3.3	La centrale d'achat de matériel et pansements	20
1.3.4	Le service de dispensation des médicaments	22
1.3.5	Problématique et méthodes	24
2	Stockage multiniveau avec transporteur	27
2.1	État de l'art	28
2.2	Système étudié et possibilité d'optimisation	30
2.3	Représentation graphique et modèle mathématique	32
2.3.1	Les carrefours	33
2.3.2	Les nœuds orientés	35
2.3.3	Modèle mathématique	41
2.4	Algorithme génétique	45
2.4.1	Principe et algorithme	45
2.4.2	Croisements et mutations	47

2.4.2.1	Opérateur de croisement	48
2.4.2.2	Opérateur de mutation	49
2.5	Résultats expérimentaux	50
2.6	Conclusion	53
3	Agencement du stock de dispositifs médicaux	57
3.1	Système étudié et possibilités d'amélioration	58
3.2	Modélisation	61
3.2.1	Analyse du problème	64
3.3	Méthodes de résolution	66
3.3.1	Approche heuristique	66
3.3.1.1	Algorithme	66
3.3.1.2	Amélioration du MFIF avec un TSP	68
3.3.2	Méthode basée sur le branch and bound	71
3.3.2.1	Algorithme	71
3.3.2.2	Borne inférieure	74
3.3.3	Algorithme génétique	77
3.3.3.1	Représentation des solutions	78
3.3.3.2	Croisements et mutations	78
3.3.4	Les méthodes mixtes	79
3.4	Résultats expérimentaux	81
3.5	Conclusion	84
4	Affectation des dispositifs médicaux aux interventions	87
4.1	État de l'art	88
4.2	Le système étudié	91
4.3	Modélisation	92
4.3.1	Définition du problème	92
4.3.2	Caractérisation du problème	95
4.4	modèle mathématique	97

4.4.1	La contrainte de "non-chevauchement"	100
4.5	Méthodes de résolution	101
4.5.1	Choix de la méthode	101
4.5.2	L'optimisation par essaim particulaire	103
4.5.2.1	Modélisation	104
4.5.2.2	Choix des paramètres	109
4.6	Résultats expérimentaux	112
4.7	Conclusion	114
5	Affectation des dispositifs médicaux dans un contexte intersite	115
5.1	État de l'art	116
5.2	Système étudié	119
5.3	Modélisation	120
5.4	Méthode de résolution	123
5.4.1	Modification du modèle	124
5.4.2	Gestion des retours	125
5.4.3	Heuristique dédiée	130
5.4.4	Résultats expérimentaux	130
5.5	Conclusion	132
6	Conclusion générale et perspectives	135
I	Annexes	167
A	Application de l'algorithme (1)	169
B	Modèle MILP d'optimisation de stock	171
C	Modèle MILP d'optimisation des plannings d'interventions	175

INTRODUCTION GÉNÉRALE

Au fil du temps et grâce aux progrès de la médecine, l'espérance de vie humaine n'a cessé de croître. À titre d'exemple, lors de la dernière décennie elle a augmenté de 1.7 an aux USA, de 1.8 an en France ou encore de 2.3 ans au Japon [Organization, 2016]. Cela implique une hausse de la fréquentation des hôpitaux, de nombreux patients venant y recevoir des soins médicaux de natures variées (interventions chirurgicales, analyses sanguines, radiographies...). C'est donc tout naturellement que les domaines de recherches liés à la gestion des systèmes de santé ont gagné un intérêt grandissant d'année en année aux yeux du public comme des chercheurs.

À cette sollicitation toujours plus importante des centres de soins s'ajoute l'impact des différentes crises financières récentes sur l'équilibre économique des pays occidentaux. Cette situation économique épineuse impliquant une réduction des budgets de fonctionnement octroyés aux établissements de santé, on comprend que ces derniers, particulièrement en Europe, soient confrontés au besoin impérieux de maîtriser leurs finances. Les centres hospitaliers doivent ainsi réduire leurs dépenses tout en assurant une qualité de service et une sécurité optimale des soins pour les patients.

Dans ce but, les chercheurs se penchent de plus en plus sur des questions liées à l'optimisation des nombreux aspects de la logistique hospitalière. Le présent mémoire de thèse résume trois ans de travail de recherche sur un sujet directement lié à l'une de ces problématiques. Dans le cas présent, il s'agit de l'optimisation de la chaîne logistique pharmaceutique au sein d'un hôpital dans un contexte multi site. L'origine de ce travail fut la création d'un nouveau groupement hospitalier, le **Groupe Hospitalier de la Région Mulhouse et Sud-Alsace (GHRMSA)**, un groupement alors naissant qui réunit aujourd'hui certains des principaux acteurs Alsaciens du domaine de la santé publique. Le travail de thèse en lui-même s'est déroulé au sein de la pharmacie de l'hôpital Emile Muller à Mulhouse afin de pouvoir étudier les problématiques et les données d'un cas réel.

Un tel rassemblement d'établissements sous une direction commune implique de repenser les processus logistiques internes à chacun d'eux. En effet, dans un tel groupement, les différents centres de soin doivent fonctionner conjointement en partageant des ressources matérielles, humaines et décisionnelles. Pour ce faire, une importante partie de ces ressources est généralement centralisée pour assurer une meilleure répartition et une minimisation des coûts. C'est dans ce but que la pharmacie de l'hôpital de Mulhouse est amenée à centraliser les activités de stérilisation du matériel médical et d'approvisionnement des autres sites en médicaments et dispositifs médicaux au sein du GHRMSA. Ces nouveaux enjeux nous ont amené à reconsidérer certains aspects de la chaîne logistique globale de la pharmacie afin qu'elle soit en adéquation avec la nouvelle politique du groupement. Pour ce faire, nous avons axé ces 3 ans de travail sur deux aspects logistiques.

I- L'utilisation des particularités intrinsèques au milieu hospitalier pour établir un agencement optimal des stocks sur palettes de médicaments et dispositifs médicaux.

II- La création d'une méthode de planification des interventions chirurgicales cherchant à lisser la charge de travail du service de stérilisation et la mise en place d'une logistique de transport associée.

Afin de répondre au mieux à ces problématiques, le présent mémoire est divisé en 5 parties.

Dans une première partie, nous présentons le contexte de cette étude en proposant un état des lieux des différents services impliqués. La seconde partie porte sur la possibilité de mise en place d'un stock sur étagères impliquant l'utilisation de véhicules de levage et de transport dans une pharmacie. Le but est d'identifier les contraintes qui en découlent ainsi que d'optimiser au mieux le chemin du véhicule à travers le stock. Toujours dans l'optique de minimiser les durées des opérations liées au stockage, la troisième partie détaille une méthode de placement des produits (médicaments et matériels) dans le stock en utilisant le fait que les éléments à fournir à chaque service sont connus à l'avance. Cette particularité propre au domaine médical offrant, en effet, de nouvelles possibilités d'optimisation des flux matériels. La quatrième partie traite de l'ordonnancement des interventions chirurgicales en proposant une méthode de création de plannings cherchant à minimiser la charge de travail du service de stérilisation. Cette méthode est complétée par la cinquième partie dans laquelle le problème est considéré à l'échelle d'un groupement hospitalier forçant à mettre en place des plannings prenant en compte la livraison et le retour du matériels médical via des transporteurs dédiés.

Chaque étape de ce travail a donc pour but d'optimiser une partie des processus logistiques gérés par la pharmacie et d'adapter ces derniers aux exigences d'un groupement

hospitalier.

1.1/ CONTEXTE DE LA THÈSE

Ce chapitre présente le contexte dans lequel la thèse s'est déroulée. Le fonctionnement de l'hôpital Emile Muller y est détaillé ainsi que son implication dans le récemment formé groupement hospitalier de la région Mulhouse est sud-Alsace. Dans la suite de ce mémoire, différents services de la pharmacie de l'hôpital font l'objet d'études approfondies afin d'améliorer certains aspects logistiques. Ces services sont donc présentés en détails et des modélisations sont utilisées pour mettre en évidence les différentes étapes des processus liés à ces derniers. Ce travail préliminaire d'analyse est capital pour être en capacité de comprendre les enjeux logistiques du présent travail.

1.1.1/ L'HÔPITAL EMILE MULLER

La ville de Mulhouse, située dans le Haut-Rhin en région Grand Est (anciennement Alsace), est pourvue de deux hôpitaux. L'hôpital du Hasenrain se trouve au centre de la ville et accueille des services de gérontologie et de psychiatrie ainsi que l'ensemble des services de maternité, gynécologie et obstétrique qui à l'heure de la rédaction de ce mémoire se prépare à être déplacés à l'hôpital Emile Muller. Ce dernier, quant à lui, se trouve en périphérie de la ville et fut créé en 1972 à partir de ce qui n'était encore qu'une maison médicale pour personnes âgées. Le reste des bâtiments de l'hôpital a été construit jusqu'en 1994. Longtemps désigné sous le nom du site qui l'accueille, le "Moenschberg" a pris le nom d'Emile Muller en 1989. Il s'agit d'un hommage à l'ancien président du Conseil d'administration de l'hôpital et maire de Mulhouse, décédé un an plus tôt. L'hôpital héberge la plupart des services de médecine et de chirurgie, et les hautes spécialités du GHRMSA. Il accueille par ailleurs le service des urgences ainsi que la base hélicoptère du SAMU [JDS, 2017].

De nombreux échanges ont lieu chaque jour entre les deux hôpitaux (informations, patients, documents, médicaments, dispositifs médicaux, personnels...), ils sont généralement effectués par des navettes à horaires fixes, des transporteurs dédiés ou des livreurs dépêchés pour l'occasion en cas d'urgence.

Dans le but de minimiser les coûts et de simplifier les processus, ces deux hôpitaux ont fusionné avec d'autres centres de soins géographiquement proches formant ainsi le GHRMSA.

1.1.2/ DE LA PHARMACIE DE L'HÔPITAL À LA PHARMACIE DE TERRITOIRE

Le **Groupe Hospitalier de la Région Mulhouse et Sud-Alsace** (ou GHRMSA) est un groupement hospitalier en perpétuelle évolution. À sa création il était constitué de divers centres de soins se trouvant dans les villes de Thann, Bitschwiller-lès-Thann, Cernay, Saint-Louis ainsi que les deux hôpitaux de Mulhouse. Depuis sa création en 2015 se sont rajoutés à cette liste Rixheim, Sierentz et Altkirch pour un total de neuf établissements comme on peut le voir sur la figure (1.1). Le groupement est né de la volonté de rassembler des établissements publics de santé dans le sud de l'Alsace afin d'y maintenir et d'y développer l'offre de soin, pour l'ensemble des bassins de population. L'un des résultats de cette fusion est que la pharmacie de l'hôpital Emile Muller est devenue la pharmacie centrale du groupement ; c'est donc à elle de gérer les livraisons de médicaments, des dispositifs médicaux ainsi que la stérilisation des dispositifs médicaux réutilisables. Le but global du travail de doctorat est donc d'identifier et d'optimiser certains aspects logistiques des processus de la pharmacie pour mettre en place de nouveaux modes de fonctionnement plus adaptés à un contexte multisite.

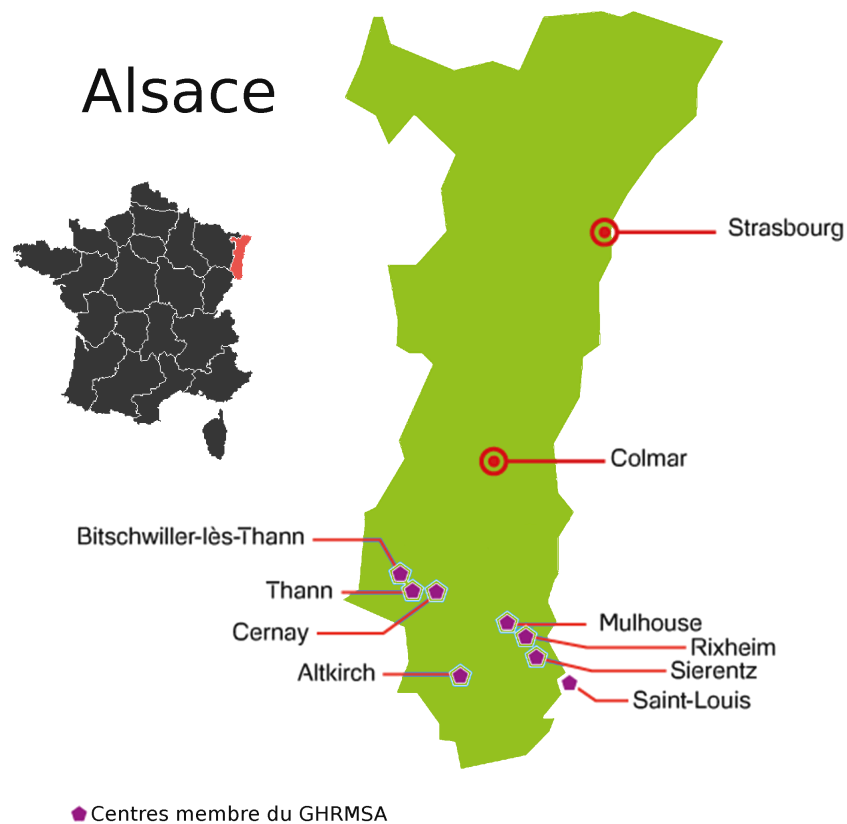


FIGURE 1.1 – Centres membre du GHRMSA

1.2/ ÉTAT DE L'ART

Le contexte décrit dans l'introduction explique l'augmentation significative des études liées à l'optimisation des processus hospitaliers depuis ces quarante dernières années. Dans la littérature il est possible de trouver plusieurs définitions de la logistique hospitalière. Dans le cadre de cette thèse, nous utilisons la définition proposée par [Landry et al., 2001] qui la décrit comme " l'ensemble d'activités de conception, de planification et d'exécution permettant l'achat, la gestion des stocks et le réapprovisionnement des biens, et services entourant la prestation de services médicaux aux patients ". D'autres propositions ont été faites [Aslog, 2002, Sampieri et al., 2000] mais toutes tournent autour de l'idée de maîtrise des flux de matériels et d'informations circulant dans l'hôpital.

Il est important de noter que cette prise de conscience de l'importance des aspects logistiques fut un processus lent et graduel. En effet, [Canter, 1991, Landry et al., 2001] ont montré que le travail de logisticien dans un centre de soins est souvent mal compris et peu considéré. De nombreux chercheurs ont tenté d'adapter les méthodes appliquées dans l'industrie ainsi que les concepts de la recherche opérationnelle dans ce nouveau cadre. L'un des premiers états de l'art sur le sujet est celui de [Fries, 1976], suivis par d'autres tel que [Rosenhead, 1978]. Ces derniers ont mis l'accent sur les problèmes de planification des services de santé et notamment les différences qui existent avec ceux de l'industrie. Plus tard, [Boldy et al., 1982] ont dressé un état des lieux de la recherche opérationnelle appliquée aux systèmes de santé en couvrant un large champ de domaines (contrôle des stocks, gestion du personnel, système d'affectation des tâches. . .). De plus, [DeJohn, 2009] estimaient en 2009 que compte tenu des prévisions de croissance des dépenses du secteur de la santé, il serait probable que les centres hospitaliers dépenseraient bientôt d'avantage dans leurs chaînes logistiques que dans leurs activités principales de santé. La chaîne logistique d'un hôpital peut être représentée, selon [Beaulieu et al., 2010], par une chaîne logistique externe composée elle-même de plusieurs chaînes logistiques internes. La chaîne externe regroupe un ensemble d'intervenants allant des fournisseurs (généralement de grands groupes industriels pharmaceutiques) jusqu'au consommateur final qu'est le patient, en passant par différents intermédiaires (transformateurs, grossistes, transporteurs, distributeurs...). Les chaînes internes, quant à elles, ont pour objectif l'achat, la réception, le stockage et la livraison des différents produits nécessaires au bon fonctionnement des services de soins. Cette représentation est synthétisée par la figure (1.2). Face à ce constat, de nouvelles méthodes d'optimisation de certains aspects de la logistique hospitalière ont alors fait leur apparition dans la recherche scientifique. [Groene et al., 2008]

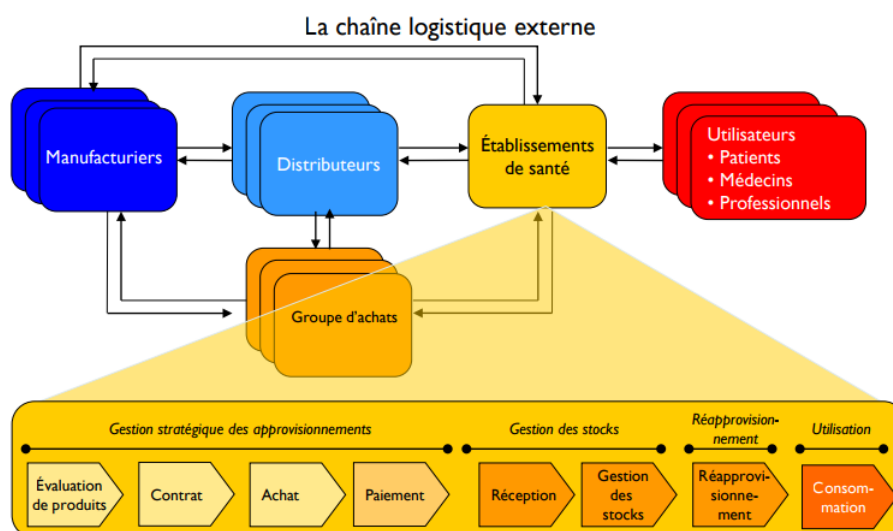


FIGURE 1.2 – La chaîne logistique interne [Beaulieu et al., 2010]

et [Jobin et al., 2004] ont proposé un outil d'évaluation des performances pour hôpitaux en identifiant un ensemble d'indicateurs clefs liés au milieu hospitalier alors que selon [Morgan, 2000], l'identification en elle-même de ces indicateurs est déjà en soi une tâche complexe qui mérite d'être étudiée. [Hassan, 2006] ont démontré dans leurs travaux que certaines méthodes d'analyse et outils de résolutions habituellement réservés aux domaines industriels pouvaient être appliqués, parfois avec quelques modifications, au domaine hospitalier car leurs enjeux partageaient des similarités.

Les enjeux du secteur hospitalier étant proches de ceux du secteur industriel, un certain nombre de chercheurs ont tenté d'y appliquer des méthodes inspirées du "lean management". Le lean management (ou simplement "lean") est un concept inventé par [Krafcik, 1988] en 1988 et popularisé par [Womack et al., 1990] au MIT deux ans plus tard ainsi que par le groupe Toyota [Womack et al., 1996] dont les origines japonaises sont en parfaite adéquation avec les principes du lean. Le lean est une méthode managériale basée sur la gestion de production qui se concentre sur la "gestion sans gaspillage", ou "gestion allégée" ou encore gestion "au plus juste". La définition (1) donne une bonne représentation du lean.

Définition 1 : lean management

Système visant à générer la valeur ajoutée maximale au moindre coût et au plus vite, ceci en employant les ressources juste nécessaires pour fournir aux clients ce qui fait de la valeur à leurs yeux. [Hohmann, 2012]

[Hasle et al., 2016] ont identifié les aspects du travail en hôpital qui pourraient freiner le lean (à savoir la jeunesse du concept, sa complexité et le concept de "valeur" cher au lean [Jacquier, 2013] afin de les améliorer.

On trouve dans la littérature de nombreuses publications détaillant les méthodes pour mettre en place le lean dans les établissements de santé [King et al., 2006, Womack et al., 2005, Ben-Tovim et al., 2007, Schuur et al., 2007]. Malheureusement cette pratique est aussi beaucoup controversée, autant par les chercheurs que par les travailleurs des secteurs de la santé. En effet, l'Agence nationale pour l'amélioration des conditions de travail (Anact) alerte l'opinion publique sur les risques de conditions de travail dues au lean [Anact, 2012]. [Conti et al., 2006] ont mis en évidence le lien entre la pratique du lean et le stress des travailleurs selon une relation non linéaire. D'un autre côté, [Dombrowski et al., 2017] ont montré que les nouveaux managers ont tendance à ne pas comprendre la relation entre l'utilisation de méthodes lean et le stress des employés.

Parmi les différents secteurs d'un centre hospitalier, le service de pharmacie est un point central de la logistique globale de l'établissement de santé. En effet, la pharmacie est la branche de l'hôpital qui se charge des flux de matériels de santé (médicaments, dispositifs médicaux, préparations de chimiothérapies...). Or, ces flux ne peuvent souffrir de perturbations trop importantes, en raison de leur caractère vital [Engel et al., 2000]. En effet, [Landry et al., 2001] ont mené des études sur différents hôpitaux français, néerlandais, québécois et américains pour finalement mettre en évidence que les coûts engendrés par le fonctionnement de la pharmacie peuvent représenter de 30% à 40% des coûts annuels de fonctionnement d'un hôpital. Conséquemment, de nouvelles études sur les améliorations possibles du fonctionnement de ce secteur ont vu le jour. [Romero, 2013] ont mené une étude sur une pharmacie d'un hôpital nord-américain pour identifier toutes les imperfections logistiques impliquées dans le traitement des flux de médicaments. Ils identifient notamment une mauvaise gestion du stock, un travail manuel intensif, de nombreux rappels de produits périssables ou encore une mauvaise utilisation de certaines technologies. En effet, les nouvelles technologies ouvrent de nouvelles perspectives en terme d'organisation et semblent se répandre de plus en plus au sein des centres hospitaliers [Meinert et al., 2016]. Par exemple, [Romero et al., 2015] ont proposé d'utiliser à la fois des codes-barres à scanner et des systèmes RFID afin de simplifier la traçabilité des flux et de permettre une plus forte informatisation et donc une fiabilité plus importante. Pour ce qui est des technologies de la communication [Rangel-Mayoral et al., 2012] ont étudié l'impact de l'utilisation du "web 2.0" (applications de communication web, groupes de discussion, réseaux sociaux...) dans les milieux hospitaliers et en déduisent qu'il s'agit de solutions efficaces pour favoriser et clarifier les échanges entre les différents services. Il a été ainsi montré qu'au fil du temps, les pharmacies tendent vers une part d'informatisation et d'automatisation de plus en plus grande simplifiant ainsi les processus tout en augmentant la sécurisation des différents flux [Zhiqiang et al., 2012]. Dans un autre

domaine, [Ritchie et al., 2000] ont étudié la "logistique inverse" de la pharmacie, c'est-à-dire le recyclage, la destruction et éventuellement la réutilisation des produits pharmaceutiques. L'étude en question s'est déroulée au sein de 28 hôpitaux et a montré que l'amélioration de cet aspect organisationnel, souvent négligé, permettrait de faire d'importantes économies budgétaires. [Böhme et al., 2016] ont quant à eux analysé les moyens grâce auxquels les pharmacies pouvaient améliorer leur fiabilité. Ils ont pu mettre en évidence que les managers pouvaient utiliser des méthodes de test de performance non dédiées au milieu hospitalier pour identifier efficacement les modifications possibles de leurs pratiques.

Une autre méthode de rassemblement des flux logistiques est la création d'un groupement hospitalier partageant une pharmacie commune. L'apparition de plus en plus systématique de ce genre de groupements en France est un phénomène récent qui intéresse de plus en plus les chercheurs. [Hargadon, 1999] ont noté que ce genre de groupements apparaissait souvent aux USA afin de créer une instance avec un poids politique et économique plus important et donc de modifier le rapport de force pendant les négociations avec des MCO (**M**anaged **C**are **O**rganization) [Miller et al., 1994]. Dans le même ordre d'idée, [Beaulieu et al., 2004] ont décrit la démarche ainsi que le fonctionnement du réseau de gestion des approvisionnements entre trois établissements de santé québécois qui ont décidé de partager leurs ressources dans le but de rehausser les activités de logistique hospitalière. De nouvelles méthodes plus adaptées à ce mode de fonctionnement ont alors été imaginées. [Danas et al., 2002] ont, par exemple, proposé de mettre en place une pharmacie centrale virtuelle en se basant sur la méthode "zéro stock" de [Cunningham et al., 1992] et de la popularisation des méthodes "juste-à-temps" dans un contexte hospitalier [Whitson, 1997]. Le juste-à-temps (aussi appelé "flux tendu") est une méthode d'organisation et de gestion de la production, propre au secteur de l'industrie, qui consiste à minimiser les stocks et les en-cours de fabrication. Le principe est donc de ne gérer aucun stock général en le répartissant directement entre les différents membres du groupement et en gérant les commandes en "juste à temps".

1.3/ LA LOGISTIQUE PHARMACEUTIQUE DANS UN CONTEXTE HOSPITALIER

Au sein d'un hôpital, la **Pharmacie à Usage Interne** (aussi appelée PUI) est une branche de l'hôpital nécessitant généralement un bâtiment dédié. Son rôle principal est de gérer le stock de médicaments et de **Dispositifs Médicaux** (aussi appelé DM) afin de pouvoir les distribuer dans les différents services de soins de l'hôpital ou dans les blocs opératoires.

Définition 2 : Dispositif médical

Un dispositif médical (souvent raccourci en DM) est un instrument, appareil, équipement ou encore un logiciel destiné, par son fabricant, à être utilisé chez l'homme à des fins, notamment, de diagnostic, de prévention, de contrôle, de traitement, d'atténuation d'une maladie ou d'une blessure.

De plus, on trouve souvent (mais pas systématiquement) au sein de la pharmacie le service de stérilisation dont le rôle est de laver, stériliser et reconditionner les DM réutilisables après qu'ils aient été utilisés lors d'une intervention chirurgicale. Bien que rien n'oblige ce service à se trouver dans les murs de la pharmacie, cette disposition permet de facilement gérer de façon conjointe les DM "réutilisables" et "à usage unique". On peut donc généralement décomposer la pharmacie en trois principaux sous-services.

- Le **service de stérilisation** qui stérilise et gère des DM réutilisable.
- La **centrale d'achat de matériel et pansements** qui gère les stocks de DM à usage unique.
- Le **service de dispensation des médicaments** qui gère les stocks de médicaments sous toutes leurs formes

NB : Il existe d'autres services qui peuvent être délégués à la pharmacie, mais ils n'entrent pas dans le contexte de la présente étude.

Chaque sous-service est sujet à des normes spécifiques résumées dans un texte de loi édicté conjointement par le ministère de l'Emploi et de la Solidarité et par le ministère de la santé en juin 2001 nommé "les **Bonnes Pratiques de Pharmacie Hospitalière**" (abrégé en BPHP) [Jean-Claude, 2001]. Ces bonnes pratiques sont un ensemble de recommandations provenant de l'expérience des établissements de santé pour qu'ils puissent chacun se remettre en cause et s'organiser pour maîtriser la qualité de leur production.

1.3.1/ LE SERVICE DE STÉRILISATION

Le service de stérilisation a pour rôle de gérer les DM réutilisables et notamment de les stériliser après chaque intervention chirurgicale. La stérilisation des DM sert à éviter le plus possible le risque d'infection d'un patient dans un établissement de santé (appelée aussi infection nosocomiale). Elle est définie dans les bonnes pratiques hospitalières comme suit :

Définition 3 : Stérilisation - Extrait des BPPH

Pour qu'un dispositif ayant subi une stérilisation puisse être étiqueté "stérile", la probabilité théorique qu'un micro-organisme viable soit présent doit être inférieure ou égale à 1 pour 10^6 .

La figure (1.3) détaille les différentes étapes classiques du cycle de vie d'un DM réutilisable. Les étapes sont les suivantes :

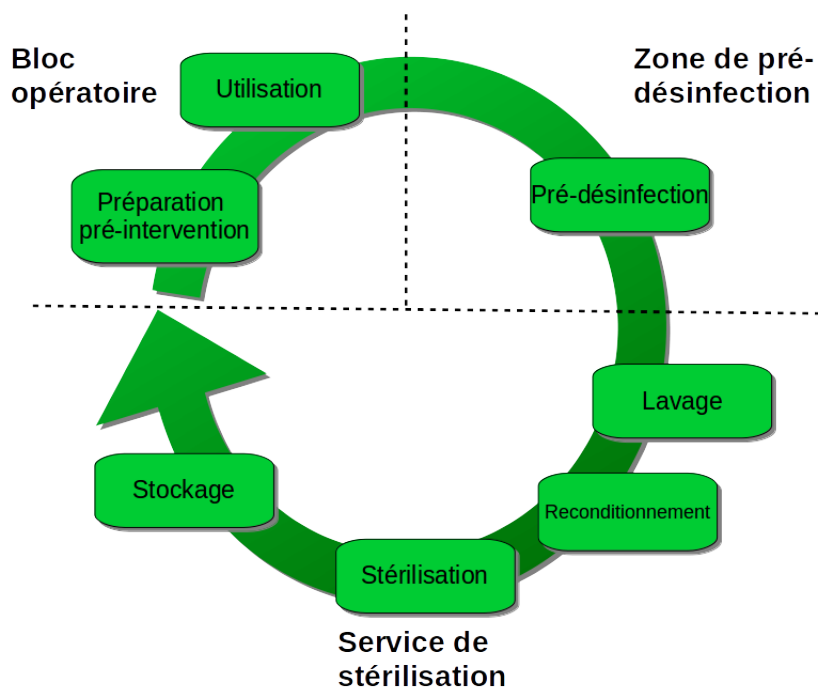


FIGURE 1.3 – Cycle de vie d'un DM

Préparation pré-intervention : Le DM est contenu dans une boîte dédiée. Cette boîte est hermétiquement scellée et contient tout ou une partie des DM stériles nécessaires à l'intervention chirurgicale concernée (une intervention peut nécessiter plusieurs boîtes). Avant le début de l'intervention, un infirmier ouvre la boîte et prépare les instruments sur une table. Il est important de noter qu'une fois la boîte ouverte la stérilité n'est plus garantie, c'est pourquoi une boîte ouverte ne peut pas être utilisée sans avoir été stérilisée au préalable.

Utilisation : le DM est utilisé lors de l'intervention. Il est techniquement possible que le DM ne soit pas utilisé, car les boîtes contiennent souvent plus d'instruments que nécessaire afin de pouvoir utiliser le même type de boîte pour des interventions similaires ou pour correspondre aux pratiques de tous les chirurgiens qui peuvent légèrement varier d'une intervention à l'autre. Que le DM soit utilisé ou pas il sera

considéré comme sale et suivra les mêmes étapes.

Pré-désinfection : une fois l'intervention terminée, le DM est alors déplacé vers une zone dédiée pour être plongé dans un bac de liquide désinfectant. La pré-désinfection est le premier traitement à effectuer sur les objets et matériels souillés dans le but de diminuer la population de micro-organismes et de faciliter le nettoyage ultérieur. Elle a également pour but de protéger le personnel lors de la manipulation des instruments et de protéger l'environnement. Les bacs sont ensuite chargés dans des chariots de transport puis acheminés vers le service de stérilisation. Ce trajet ne doit pas être trop long, car le DM doit idéalement rester immergé 15 à 20 min. En cas de dépassement trop important de cette durée, il y a d'importants risques de corrosion du DM.

Lavage : une fois le DM réceptionné par le service de stérilisation (toujours dans le bac de liquide de pré-désinfection) il est pris en charge par les agents du service dans la zone de pré-lavage (dite zone "sale"). Dans cette zone les DM sont sortis des bacs de pré-désinfection pour être placés sur des embases (c'est à dire des grilles conçues pour entrer dans les laveurs) avant d'être mis dans des laveurs. Les laveurs sont des machines qui lavent les instruments, certains modèles sont munis de deux portes (avant et arrière) et peuvent être encastrés dans un mur afin d'être chargés en zone "sale" puis déchargés en zone stérile (dite zone "propre") comme on peut l'observer sur le schéma (1.4). Ce type de laveur permet de respecter la séparation des activités décrites dans les BPPH (voir définition (4))

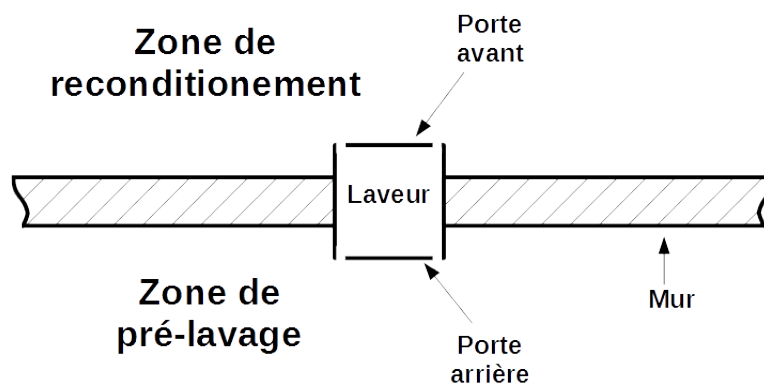


FIGURE 1.4 – Positionnement d'un laveur

Définition 4 : Cloisonnement des tâches - Extrait des BPPH

L'organisation des locaux permet de séparer physiquement les opérations de réception et de nettoyage des opérations de conditionnement. Le tri et le pliage du linge sont réalisés dans un local séparé.

Reconditionnement : Une fois en zone stérile le DM est pris en charge par de nouveaux agents. En effet les agents des différentes zones ne doivent pas passer d'une zone à l'autre afin de respecter "la marche en avant" (la circulation se fait toujours de la zone la plus contaminée vers la moins contaminée sans jamais de retour en arrière). Les DM sont alors reconditionnés, c'est-à-dire que les boîtes sont refermées et scellées avant d'être mises en attente sur de nouvelles embases pour être stérilisées.

Stérilisation : Une fois les DM reconditionnés dans leurs boîtes d'origine (ou une boîte équivalente), cette dernière est mise dans un autoclave. Un autoclave est une machine qui stérilise les boîtes de DM scellées grâce à de la vapeur d'eau saturée sous pression. À l'instar des laveurs, les autoclaves peuvent être munis de deux portes (avant et arrière) afin que les boîtes soient chargées en zone stérile et récupérées en zone de stockage. Il est important de noter qu'il existe d'autres modes de stérilisation (comme la stérilisation dite "à froid" par exemple), mais cette méthode est la plus répandue.

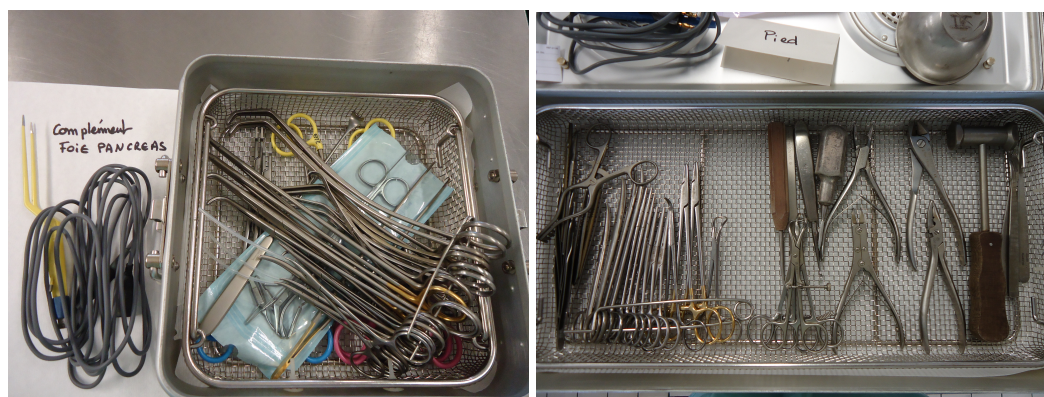


FIGURE 1.5 – Exemple de boîte de DM reconstituées

Stockage : une fois stérilisée, la boîte est stockée dans une zone dédiée. Elle y restera jusqu'à être envoyée au bloc opératoire pour une nouvelle intervention chirurgicale et ainsi recommencer le cycle.

Le service de stérilisation est donc sollicité pendant une bonne partie du cycle de vie



FIGURE 1.6 – boîtes reconstituées en attente devant un autoclave

des DM réutilisables (lavage, reconditionnement, stérilisation, stockage). Ces opérations doivent être effectuées le plus rapidement possible pour rendre les DM disponibles en temps et en heure pour de nouvelles interventions. Parallèlement, un soin extrême doit être apporté à chaque opération, en effet, une erreur dans le processus global de stérilisation pourrait conduire à de graves conséquences sur la vie des patients. Il est donc primordial d'optimiser au mieux les méthodes de travail du service de stérilisation pour parvenir à augmenter son efficacité sans diminuer la qualité des opérations menées. Pour ce faire, il semble nécessaire de passer par une première étape d'analyse du fonctionnement du service.

Afin d'affiner le niveau de détails des processus mis en œuvre au sein du service, nous proposons une représentation des flux de DM grâce à un diagramme UML. L'UML (pour **Unified Modeling Language** ou langage de modélisation unifié en français) est un langage de modélisation graphique développé en 1994 par Grady Booch, Ivar Jacobsin et James Rumbaugh [Booch et al., 1996, Fowler, 2004]. Il est conçu pour fournir une méthode normalisée pour visualiser la conception d'un système en fournissant un ensemble de diagrammes représentant des scénarios d'utilisation, des possibilités d'action, des flux et bien d'autres. Dans ce chapitre, nous présentons les diagrammes d'activités



FIGURE 1.7 – Logo de l'UML

représentant les principaux flux circulant dans la pharmacie. Le diagramme d'activité a pour fonction de représenter des flux de toutes natures en mettant en évidence leurs parcours et les actions des différents agents sur ces flux. Pour gérer au mieux les flux physiques, la pharmacie est pourvue d'un système de transport autonome. En effet, de nombreux produits à destination (ou en provenance) du bloc opératoire sont rangés dans des armoires roulantes. Quand une armoire est prête à être envoyée, elle est acheminée jusqu'à une gare et mise en attente avec les autres. Dans la gare sont disposés des capteurs qui permettent de savoir le nombre d'armoires en attente ainsi que leurs emplacements. Périodiquement, des véhicules autonomes appelés AGV (pour **A**utomatic **G**uided **V**ehicles soit en français des véhicules à guidage automatique) prennent en charge les armoires pour les transporter vers leur destination. Les AGV sont complètement autonomes, ils sont capables de prendre des ascenseurs qui leur sont réservés, d'éviter des obstacles fixes ou mouvants et de se brancher à une borne de recharge quand leur batterie est faible. Ce genre de véhicules est de plus en plus utilisé dans les centres de soins de taille importante pour simplifier les transports de matériels [Ozkil et al., 2009]. La figure (1.9) présente un diagramme d'activité des flux de DM au sein du service de



FIGURE 1.8 – Des AGV sur leurs stations de recharge

stérilisation de la pharmacie de l'hôpital Emile Muller. Le diagramme proposé ici présente aussi le flux de consommable (sachets, gants...) qui sont entreposés dans un stock à part. On y retrouve certaines des grandes étapes citées précédemment (lavage, reconditionnement, stérilisation et stockage) ainsi que d'autres sous-étapes permettant d'avoir une vision plus précise des processus en jeu. Les DM peuvent donc passer par cinq zones durant leur séjour dans le service de stérilisation : la gare dans laquelle sont amenées les armoires roulantes en provenance (généralement) du bloc opératoire, la

zone de prélavage dans laquelle les armoires sont amenées pour être déchargées avant que les instruments ne soient vérifiés puis mis sur les embases des laveurs, la zone de reconditionnement dans laquelle les DM sont réceptionnés après avoir été lavés pour être remis dans leurs boîtes d'origine qui sont alors mises sur les embases des autoclaves et enfin le quai qui permet de charger ou décharger les armoires roulantes dans des transporteurs afin d'alimenter les autres sites distants en DM.

1.3.2/ ANALYSE DU FONCTIONNEMENT DU SERVICE DE STÉRILISATION

Il est important de garder à l'esprit que l'analyse qui va suivre a pour sujet le service de stérilisation de l'hôpital Emile Muller. Les normes et les pratiques en terme de pharmacie hospitalière étant, en grande partie, commune en France d'un établissement de santé à l'autre elle reste vraie dans la majorité des cas.

Nous avons vu précédemment que le processus global de stérilisation était en réalité composé de plusieurs étapes. Nous allons ici surtout nous intéresser à celles qui demandent l'intervention de plusieurs agents c'est-à-dire le prélavage et le reconditionnement. En effet, ces deux tâches requièrent un nombre plus ou moins important d'agents qui doivent agir de façon synchrone pour effectuer une tâche commune.

1.3.2.1/ LE PROCESSUS DE PRÉ-LAVAGE

La figure (1.10) représente un logigramme du processus de prélavage, c'est-à-dire la suite des tâches qui doivent être exécutées par les agents (et les conditions pour les réaliser) lorsqu'une armoire roulante chargée de DM à stériliser est réceptionnée.

L'agent doit alors vérifier la traçabilité des boîtes réceptionnées pour savoir si elle est correcte ou éventuellement si certaines informations importantes doivent être prises en compte (comme un patient avec une maladie rare par exemple). Si la traçabilité est correcte, les DM peuvent alors être déchargés pour subir un contrôle de l'agent. Il est en effet nécessaire de vérifier que chaque DM a été correctement nettoyé à la sortie du bloc opératoire, dans le cas contraire les DM concernés doivent être relavés à la main. De plus, certains DM possédant des cavités doivent être nettoyés à part grâce à un système de nettoyage pneumatique.

Une fois ces contrôles terminés, les DM doivent être chargés sur une embase qui est elle-même chargée dans un laveur. L'armoire roulante doit alors être nettoyée avant d'être amenée dans une nouvelle zone de garage. Le chargement des laveurs est déjà en soi une opération qui peut être optimisée pour minimiser le temps de traitement global de chaque DM. En effet, les agents prennent constamment des décisions pour savoir quand lancer les cycles de lavage en fonction du nombre de DM à nettoyer, du nombre

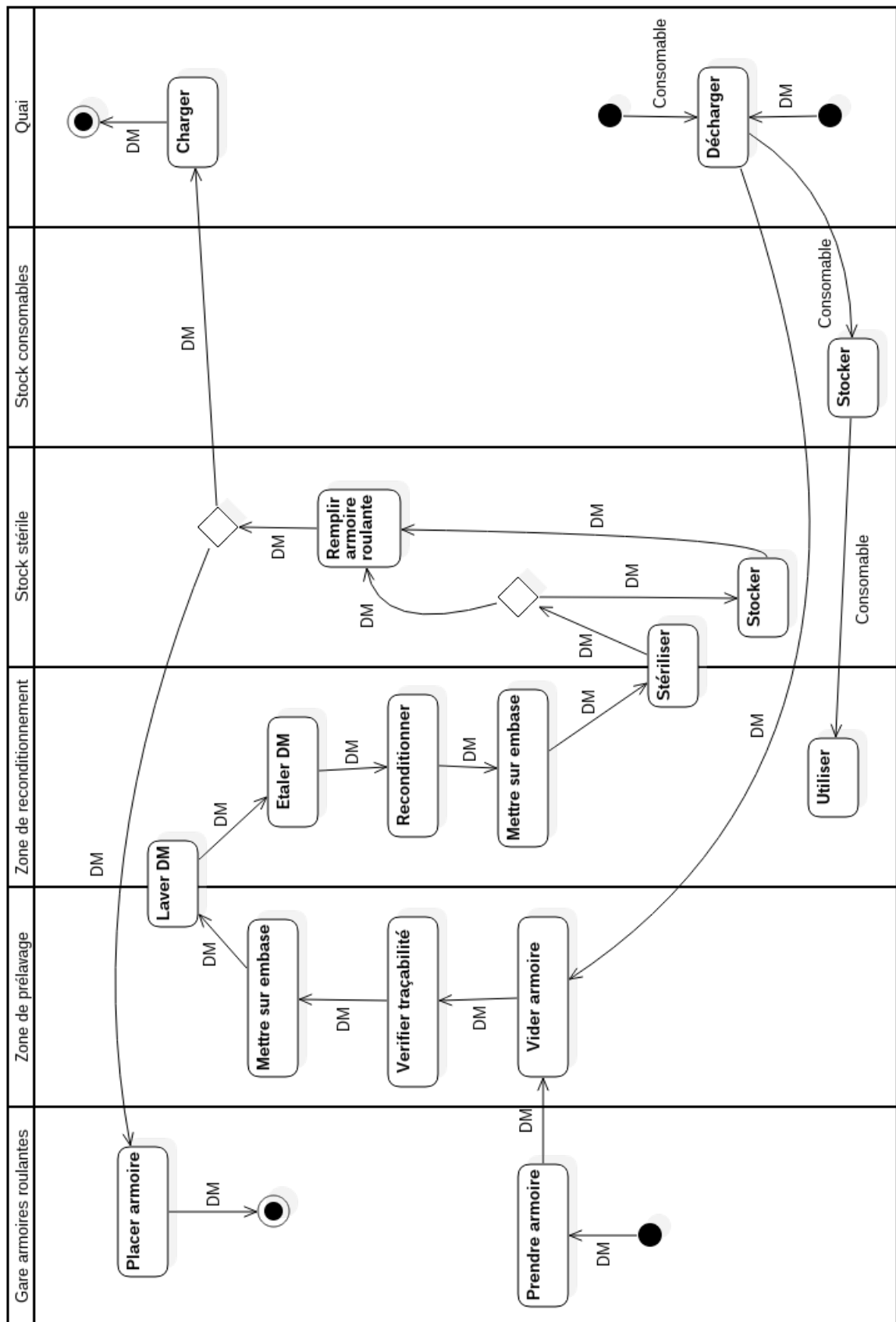


FIGURE 1.9 – Diagramme d'activité du service de stérilisation

de laveurs disponible et leurs capacités ou encore du nombre de DM qu'ils pensent réceptionner prochainement. [Ozturk, 2012]. Il est important de noter que ces tâches sont généralement effectuées par des agents dans l'ordre décrit par le logigramme, mais il est bien sûr possible de scinder le processus en plusieurs étapes et de répartir ces dernières entre les agents.

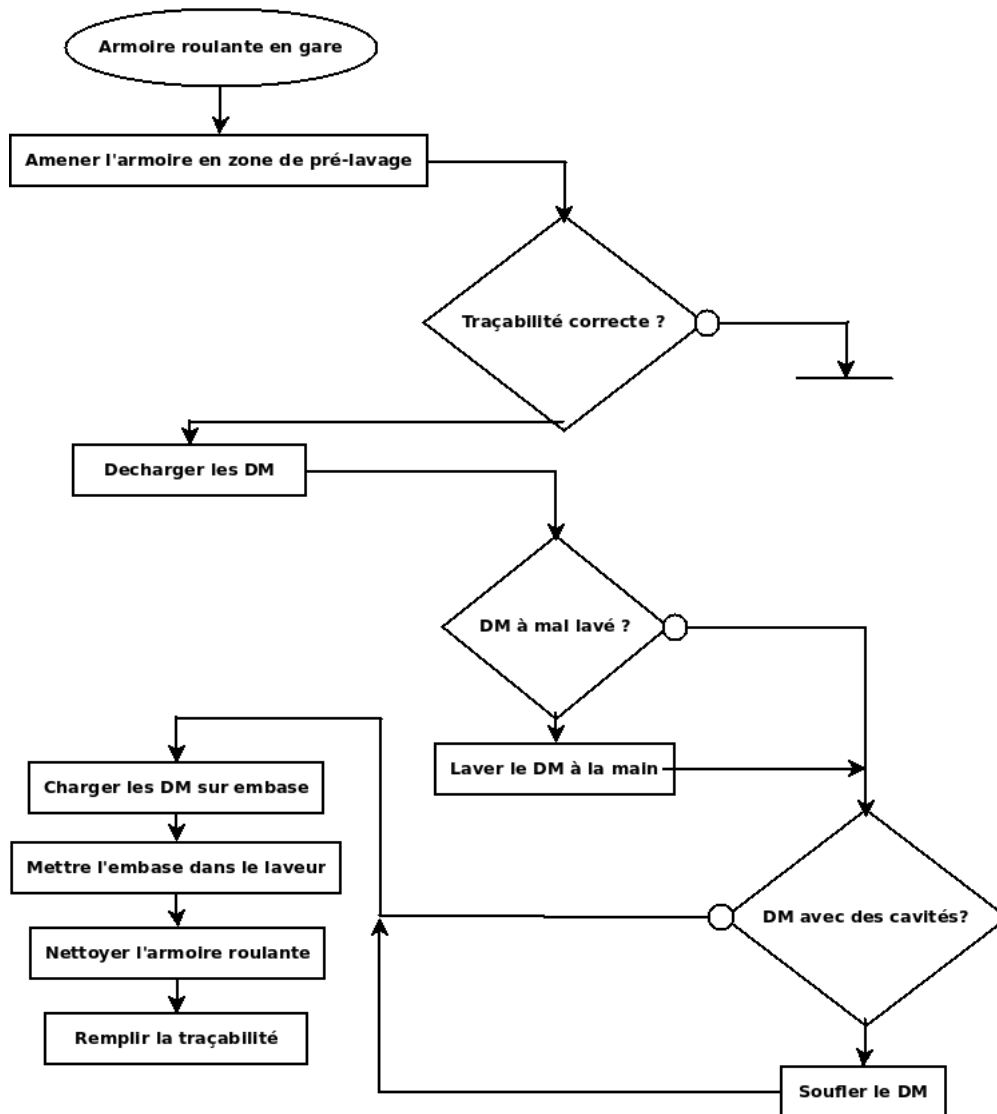


FIGURE 1.10 – Logigramme du processus de pré-lavage des DM

1.3.2.2/ LE PROCESSUS DE RECONDITIONNEMENT

La figure (1.11) présente, quant à elle, un logigramme du processus global de reconditionnement (c'est-à-dire les tâches à effectuer entre le moment où un DM vient d'être lavé et le moment où il est prêt à être stérilisé).

Lorsqu'un cycle de lavage est terminé, les embases sont déchargées et les DM sont

séchés par un système pneumatique avant d'être disposés sur des tables et triés en fonctions des interventions chirurgicales auxquelles ils sont liés, chaque groupe de DM constitue alors une boîte à reconstituer. Chaque DM est alors vérifié : si un DM est cassé ou manquant il doit être remplacé, si ce n'est pas immédiatement possible l'agent peut alors reconditionner une autre boîte. Une fois la boîte recomposée, elle est scellée et posée sur les embases d'un autoclave pour pouvoir être chargée en temps voulu.

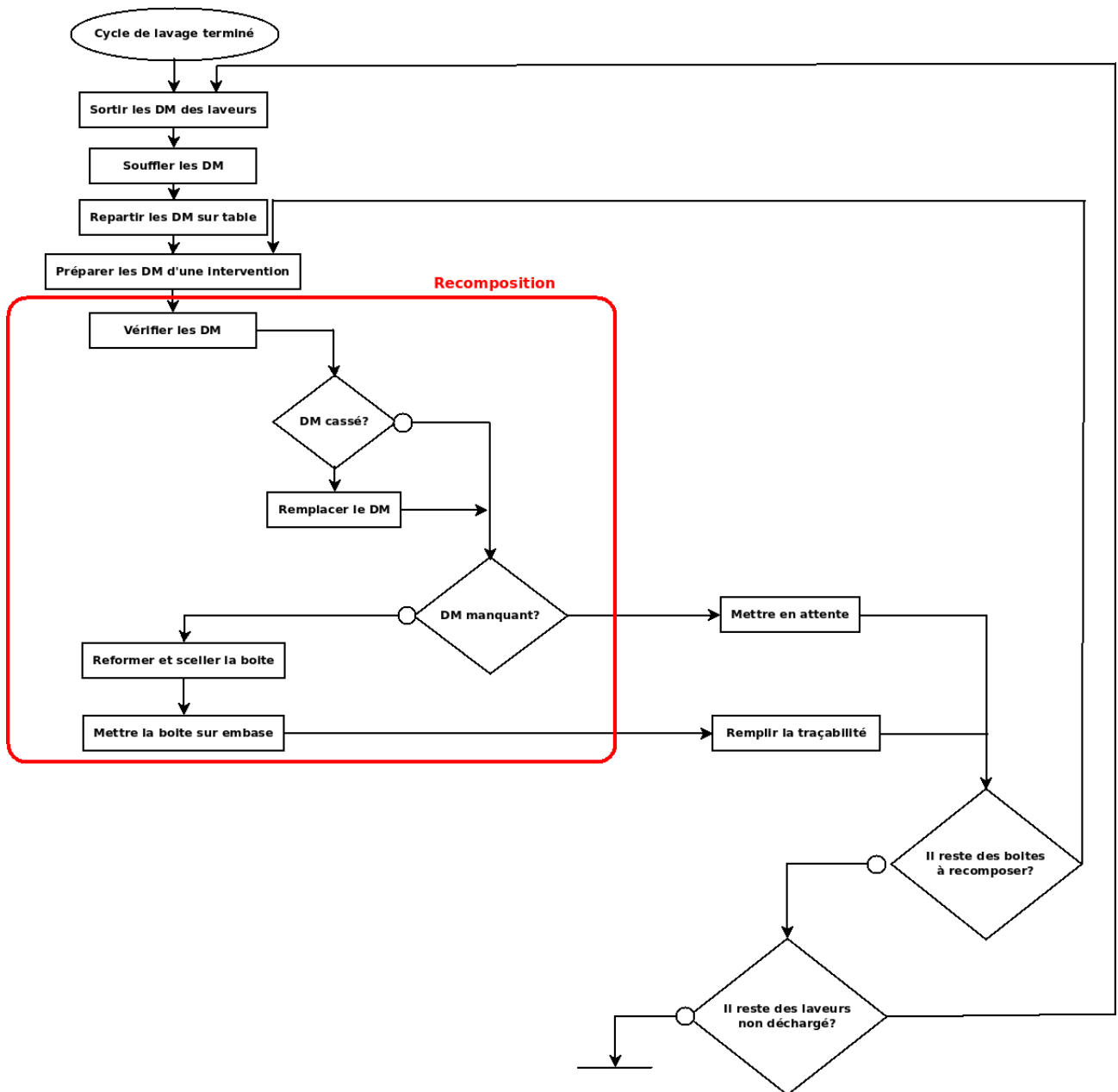


FIGURE 1.11 – Logigramme du processus de reconditionnement des DM

1.3.3/ LA CENTRALE D'ACHAT DE MATÉRIEL ET PANSEMENTS

La CAMSP (**C**entrale d'**A**chat de **M**atériel **S**térile et **P**ansements) est un service de la pharmacie dont le rôle est d'assurer l'approvisionnement, la gestion et la dispensation des dispositifs stériles au sein des services de soins de l'hôpital.

En d'autres termes ce service se charge d'acheter le matériel stérile (instruments, seringues, perfuseurs, gants...) et de le stocker pour pouvoir, le moment venu, le distribuer dans les différents services de soins.

La figure (1.12) présente un diagramme d'activité représentant le flux de matériel au sein du service depuis la livraison par le fournisseur jusqu'à la distribution dans l'hôpital Emile Muller. Ces flux peuvent passer dans différentes zones pour y subir certains traitements. Tout d'abord, la livraison est faite par un transporteur qui apporte les commandes au niveau du quai extérieur pour qu'elles puissent être déchargées (souvent sous forme de palettes). Une fois déchargées, les commandes sont immédiatement placées en zone tampon sur un emplacement dédié pour éviter de surcharger le quai en période d'affluence. Lorsqu'un agent est disponible, il vérifie que la commande concerne bien la CAMSP et le cas échéant la transporte vers une nouvelle zone tampon (proche du magasin) pour qu'elle puisse être identifiée. Les produits sont alors séparés en deux groupes : ceux qui concernent des références présentes dans le magasin sont directement stockés à un emplacement dédié fixe et les autres (aussi appelé "hors stock") sont directement posés sur des chariots roulants pour pouvoir rapidement être distribués dans le service concerné. Ils peuvent être envoyés directement (en cas d'urgence) ou en étant transférés dans des armoires roulantes pour être amenés à la gare en attendant le prochain départ. Si le produit est destiné à un site distant, il sera alors amené sur le quai pour être pris en charge par un transporteur interne.

La CAMSP de l'hôpital Emile Muller comptabilise environ 700 références différentes dont 150 sont stockées sur palettes. Pour pouvoir effectuer les livraisons, chaque service de soins doit fournir une liste de dotations. Il s'agit d'une liste de produits que le service souhaite se faire livrer de façon périodique (hebdomadaire, semi-mensuelle, mensuelle...), ces listes changent généralement peu d'une année sur l'autre et servent de base aux agents de la CAMSP pour effectuer la préparation des commandes.

En français on parle de "préparation de commandes" ou encore de "cueillette", mais dans la littérature on trouve souvent en remplacement de ces termes le mot anglais "picking" (voir définition (5)) c'est donc celui-ci que nous utiliserons dans la suite de ce mémoire.

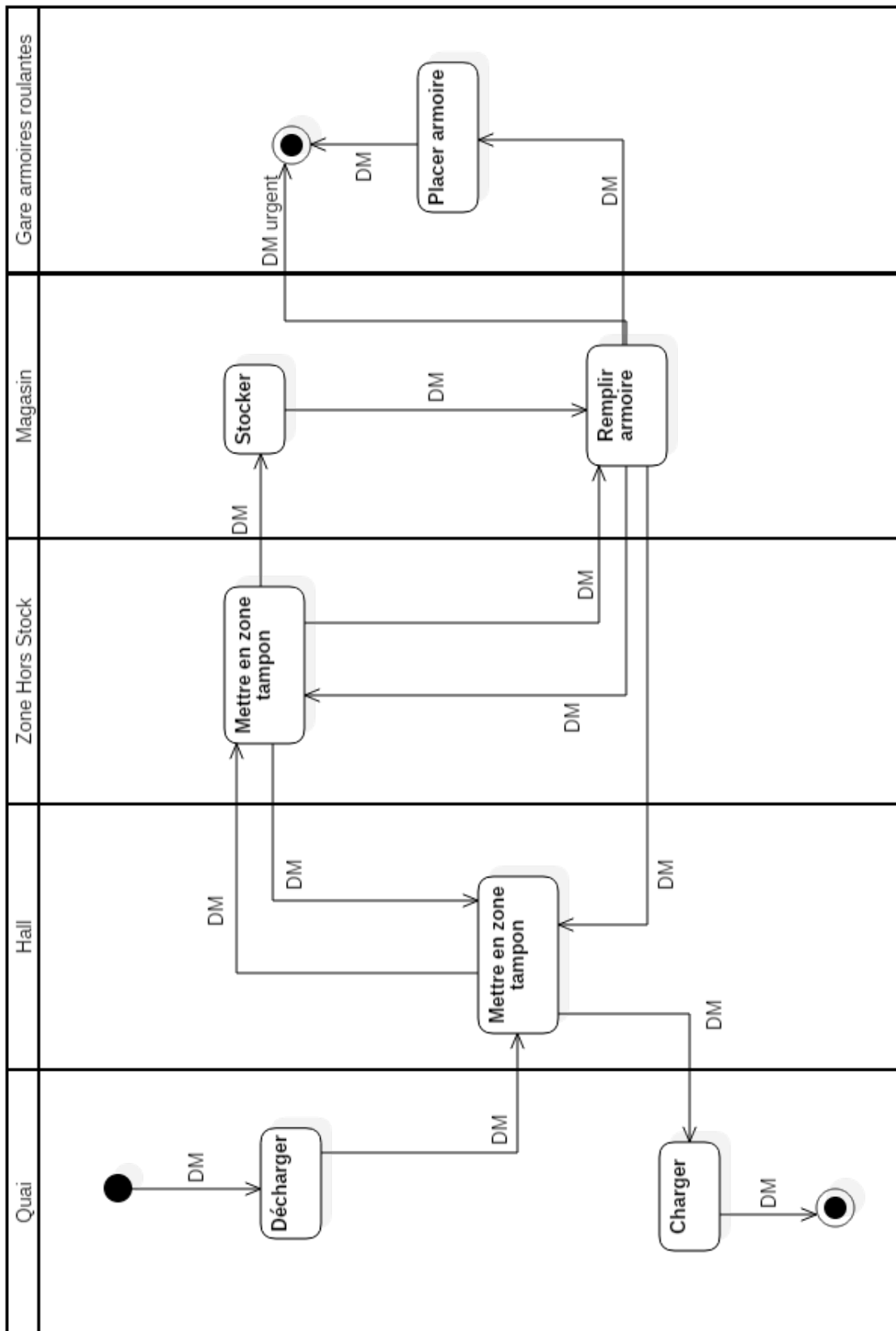


FIGURE 1.12 – Diagramme d'activité de la CAMSP

Définition 5 : Picking

Le picking (ou préparation de commandes) est l'opération qui consiste à prélever et rassembler les articles dans la quantité spécifiée par la commande avant l'expédition de celle-ci.

Le picking est un processus crucial de la CAMSP, en effet ce dernier peut être fait selon de nombreuses façons (comme nous le verrons par la suite) et laisse assez de marge de manœuvre pour être optimisé.

1.3.4/ LE SERVICE DE DISPENSATION DES MÉDICAMENTS

La dispensation des médicaments est l'une des activités principales d'une PUI. En effet, ce sont les agents de la pharmacie qui se chargent de commander, stocker et délivrer les médicaments, mais dans certains cas c'est aussi eux qui les fabriquent ou les transforment. C'est le cas, par exemple, des perfusions de chimiothérapies qui sont généralement préparées dans une unité spéciale (l'unité de pharmacotechnie). De plus, certains médicaments nécessitent des conditions de conservation particulières comme les stupéfiants et les ATU (voir définitions (6) et (7)) qui d'après les bonnes pratiques hospitalières doivent être stockés dans un lieu sécurisé.

Définition 6 : Stockage des stupéfiants – Extrait des BPPH

Les matières premières, préparations et spécialités pharmaceutiques classées comme stupéfiants sont détenues dans des locaux fermant à clef et munies d'un système d'alerte ou de sécurité renforcé contre toute tentative d'effraction, relié à un poste de sécurité.

Définition 7 : Autorisation temporaire d'utilisation

Une autorisation temporaire d'utilisation (ATU) est une procédure française permettant à certaines catégories de malades d'utiliser des médicaments n'ayant pas encore été mis sur le marché

D'autres médicaments doivent être conservés dans une fourchette de température précise ce qui impose l'utilisation d'une chambre froide et de précautions particulières pour ne pas briser la chaîne du froid lors du transport.

Le service de dispensation distribue les médicaments aux différents services de soins, mais aussi à des patients extérieurs (on parle alors de rétrocession). La rétrocession concerne certains médicaments dit en "double circuit" (disponible aussi en officine (voir définition (9))) et d'autres disponibles uniquement en PUI. Les bonnes pratiques hospi-

talières imposent de nouvelles contraintes sur ce sujet pour garantir le secret médical et empêcher le personnel non autorisé à entrer dans les zones non dédiées de la PUI (voir définition (8)).

Définition 8 : Rétrocession – Extrait des BPPH

La dispensation au public des médicaments ou produits en application des articles L. 5126-2, L. 5126-4, L. 6112-6 est organisée dans des locaux spécifiques convenablement équipés, disposant de sanitaires à proximité, comportant une zone d'attente, concourant à la qualité de l'acte de dispensation et garantissant le respect du secret professionnel et de la confidentialité ainsi que la sécurité du personnel concerné et du public. Celui-ci n'a pas accès aux autres locaux de la pharmacie à usage intérieur.

Définition 9 : Pharmacie d'officine

La pharmacie d'officine est la spécialité regroupant les pharmaciens qui travaillent dans les pharmacies de ville, appelées aussi officines, où les médicaments sont délivrés au public.

Les médicaments peuvent se trouver sous plusieurs formes (gélules, soluté, gaz...) on parle de "forme galénique" du médicament. Certaines formes galéniques peuvent être stockées sans difficulté majeure tels que les pastilles, les comprimés les granules et bien d'autres. Celles-là peuvent être rangées sur des étagères ou via des systèmes d'armoires rotatives. D'autres peuvent être particulièrement lourdes (comme les solutés massifs) et doivent alors être déplacées avec un véhicule ou un transporteur manuel et être stockées sur des palettes.

La figure (1.13) présente un diagramme d'activité représentant le flux de médicaments au sein du service de dispensation depuis la livraison par le fournisseur jusqu'à la distribution. À l'instar des DM, les médicaments sont livrés par des transporteurs et réceptionnés sur le quai avant d'être mis en zone tampon dans un hall. Ensuite, les médicaments sont apportés en salle de commandes pour être triés, à l'exception de certains solutés massifs qui sont directement amenés dans le stock dédié. Les ATU, stupéfiants, médicaments de rétrocession et médicaments à température contrôlée sont directement stockés dans les zones appropriées ; les autres sont stockés temporairement dans la salle de commande avant d'être rangés dans des armoires rotatives. Lorsque ces médicaments doivent être envoyés dans les services, ils sont rangés dans des caissettes (une par service) puis sont rangés dans des armoires roulantes avant d'être amenés à la gare. Avant cette opération, les médicaments de la chambre froide sont bien sûr reconditionnés toujours dans l'optique de ne jamais briser la chaîne du froid.

1.3.5/ PROBLÉMATIQUE ET MÉTHODES

Comme nous l'avons vu tout au long de cette introduction, une PUI abrite un système complexe de processus et sous-processus qu'il est difficile de maîtriser complètement. Cette thèse a pour but d'optimiser certains aspects de la logistique hospitalière et pharmaceutique en prévision d'une transition vers la mise en place d'une PUI centrale au sein d'un groupement hospitalier. Afin d'y parvenir nous avons choisi d'améliorer au mieux la circulation des flux de DM et de médicaments dans leur globalité, c'est à dire autant intrasite qu'intersites. Les études préliminaires du fonctionnement de l'hôpital Emile Muller nous ont permis d'aboutir à plusieurs constats. Tout d'abord, les différents stocks sur palettes (solutés massifs et DM) sont organisés selon des règles arbitraires plus liées au "bon sens" qu'à des considérations logistiques. Ensuite, l'activité journalière du service de stérilisation est très dépendante de celle du bloc opératoire. En effet, on observe souvent des périodes de creux et des périodes de forte hausse de la charge de travail. Enfin, il est nécessaire de mettre en place un modèle de logistique de transport intersites qui n'existait pas jusqu'à maintenant dans la PUI de l'hôpital Emile Muller. C'est fort de ces constatations que nous proposons les deux grands axes de travail décrit dans l'introduction général de cette thèse à savoir le réagencement des stocks de DM et médicaments (chapitre 2 et 3) et la création d'une méthode de planification des interventions lissant la charge de travail du service de stérilisation (chapitre 4 et 5).

Pour répondre à ces différents problématiques, nous proposons une approche analytique basées sur l'utilisation des outils de recherche opérationnelle. La recherche opérationnelle est l'ensemble des méthodes et techniques rationnelles orientées vers la recherche du meilleur choix lors d'une prise de décision en vue d'aboutir au résultat visé ou au meilleur résultat possible. Pour obtenir ces solutions, nous cherchons à établir des sous-problèmes et à les traduire en modèle mathématique linéaire. Ces séries équations peuvent théoriquement être résolu de façon exacte mais la complexité des système étudié dans cette thèse rend cette opération souvent difficile. Pour pallier à ce problème nous cherchons systématiquement à obtenir des solutions aussi proches que possible de l'optimal en se basant sur des méthodes utilisant certains principes connus du domaine de la recherche opérationnelle (comme la programmation dynamique, les heuristiques ou encore les métaheuristiques).

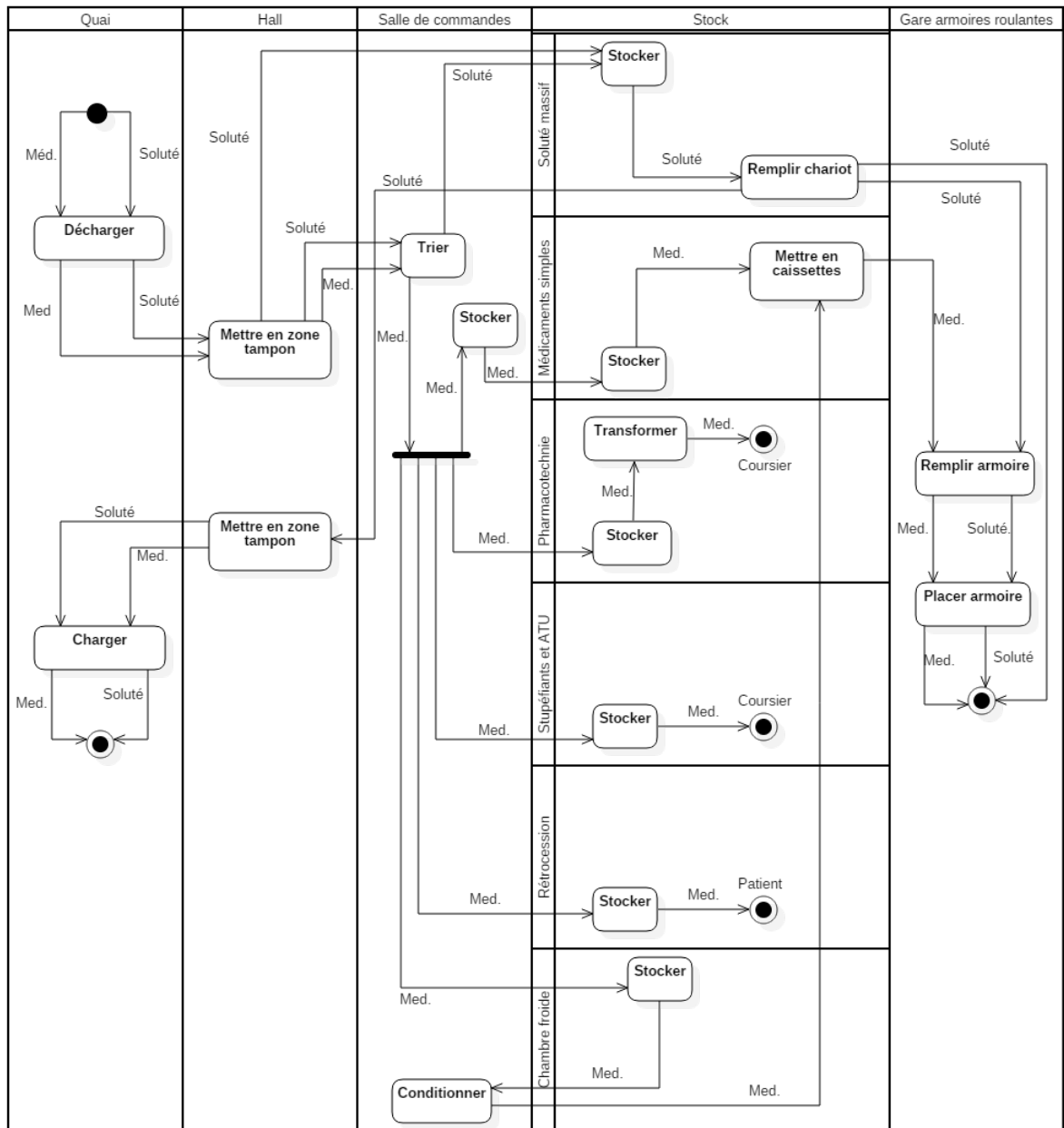


FIGURE 1.13 – Diagramme d'activité du service de dispensation

STOCKAGE MULTINIVEAU AVEC TRANSPORTEUR

Ce chapitre se concentre sur une problématique liée à l'utilisation de véhicules de transport lors des opérations de picking au sein du stock de matériel (médicaments et dispositifs médicaux) de la PUI. En effet, investir dans un véhicule peut s'avérer bénéfique pour simplifier le transport et le stockage (par exemple en cas de rangement en hauteur ou de déplacement d'objets lourds). En effet, de nombreux entrepôts industriels de grande taille (et même de taille intermédiaire) ont recours à des flottes de transporteurs motorisé pour réduire les temps des différentes opérations de stockages quotidiennes. Malheureusement, l'utilisation de tels véhicules entraîne aussi l'apparition de nouvelles contraintes liées, notamment, à la sécurité des conducteurs ou des manutentionnaires durant les phases de stockage ou de cueillette ainsi qu'à la difficulté d'effectuer certaines manœuvres (comme les demi-tours) dut à la taille réduite des allés par rapport aux gabarits des véhicules. La prise en compte de ces nouvelles contraintes impose de reconsidérer la logistique globale des agents de manutention au sein du stock. Pour pallier à cette difficulté, ce chapitre propose une modélisation (à la fois graphique et mathématique) du problème de picking adaptée aux particularités des véhicules de transport. Cette modélisation permet de faciliter l'utilisation de méthodes de résolution pour déterminer la route la plus rapide qu'un agent motorisé puisse emprunter pour effectuer les opérations de picking. Deux de ces méthodes sont mises en place. D'un côté un algorithme génétique dont le but est de s'approcher le plus possible de la solution optimale en peu de temps, de l'autre la résolution du modèle mathématique linéaire par un solveur exacte pour obtenir la solution optimale en un temps plus ou moins important. Ces deux méthodes sont appliquées à des exemples tirés de cas réels pour pouvoir être évaluées et comparées autant en terme de qualité des solutions que de temps de calcul.

2.1/ ÉTAT DE L'ART

Comme nous l'avons vu dans l'introduction, le picking (ou cueillette en français) est l'opération qui consiste à rassembler un ensemble défini d'articles dans un stock. La liste des articles à collecter est généralement fixée par un bon de commande ou, dans le cas d'une PUI, une liste de dotation (on parle aussi plus généralement de "liste de picking"). D'après [Coyle et al., 1996], les opérations de picking représentent entre 50% et 70% du coût de fonctionnement total d'un entrepôt du domaine industriel. La gestion même de l'entrepôt, quant à elle, représente généralement environ 20% des coûts totaux d'une entreprise d'après [Kearney et al., 2004]. L'importance de ces coûts explique logiquement que de nombreuses études aient été menées sur l'optimisation de la gestion des stocks de toutes natures.

[De Koster et al., 2007] ont montré que l'opération de picking est composée de plusieurs phases qui peuvent être indépendamment améliorées telles que l'agencement du stock, l'affectation des emplacements de stockage, la création de zones de picking, l'établissement de parcours de picking et bien d'autres. Ce chapitre se concentre sur l'amélioration des parcours de picking (généralement désigné par le terme anglais "routing"). Dans un contexte industriel classique, les problèmes de routing sont fréquemment reliés à des résolutions de problèmes du voyageur de commerce (voir définition (10)), aussi appelé TSP (pour **T**raveling **S**alesman **P**roblem). Appliqué à une problématique de picking, le TSP consiste à trouver le plus court chemin qu'un agent doit suivre à travers un stock pour compléter un bon de commande.

[Ratliff et al., 1983] ont utilisé les algorithmes de résolutions du TSP pour minimiser la durée des opérations de picking dans un entrepôt organisé en allées formées par des étagères et créant des carrefours. Les résultats ont pu être obtenus en un temps raisonnable sur des exemples contenant moins d'une cinquantaine d'allées. Afin de minimiser les temps de calcul, [Miliotis, 1978] ont déterminé une méthode exacte de résolution du TSP en modifiant la contrainte d'élimination des sous-tours (que nous verrons en détail dans la suite de ce chapitre). Le principe est de relâcher cette contrainte pour déterminer une solution plus rapidement. Tant que la solution est irréalisable, le problème est à nouveau résolu en ajoutant des contraintes pour interdire les solutions déjà trouvées. Cette approche implique de faire de nombreux calculs intermédiaires pour obtenir le résultat final, mais, permet pourtant de réduire drastiquement le temps de résolution global du TSP et donc, de le résoudre pour des stocks contenant plus d'emplacements.

Définition 10 : Problème du voyageur de commerce

Le problème du voyageur de commerce est un problème d'optimisation défini comme suit.

Soit n villes et, pour toute paire (i, j) de villes, la distance (ou le coût) correspondant au trajet direct de la ville i à la ville j est connu. Le problème consiste alors à déterminer une permutation cyclique des n villes qui soit de coût minimum. ([Teghem, 2012]). Voici une formulation mathématique possible du TSP.

Paramètres :

- n : nombre de villes.
- $d_{i,j}$: distance à parcourir pour aller de la ville i à la ville j .

Variables :

- $x_{i,j}$: variable binaire égale à 1 si et seulement si lors le chemin solution impose d'aller de la ville i à la ville j directement.
- u_i : l'ordre dans lequel la ville i est visité.

La fonction objectif est donnée par :

$$\text{Minimiser } \sum_{i=1}^n \sum_{j=1}^n x_{i,j} d_{i,j} \quad (2.1)$$

sujet à un certains nombre de contraintes :

$$\sum_{i=1}^n x_{ij} = 1 \quad \forall 1 \leq j \leq n \quad (2.2)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall 1 \leq i \leq n \quad (2.3)$$

$$u_1 = 1 \quad (2.4)$$

$$u_i - u_j + n \times x_{ij} \leq n - 1 \quad \forall 1 \leq i, j \leq n \quad (2.5)$$

L'équation (2.1) est la fonction objectif, elle permet de classer les solutions pour en déterminer la meilleure. Ici, on cherche une solution qui minimise le plus possible la longueur du chemin passant par toutes les villes. Les autres équations correspondent à des contraintes que la solution devra respecter pour être acceptable. Les contraintes (2.2) et (2.3) assurent respectivement que pour chaque ville, le chemin trouvé ne peut venir que d'une unique ville et ne peut repartir que vers une unique ville. La contrainte (2.4) quant à elle, est optionnelle et précise l'index de la première ville visitée. Enfin, la contrainte (2.5) définit la notion d'ordre des villes visitées.

Les enjeux logistiques des entreprises étant de plus en plus importants au fil du temps, on trouve de nombreux articles scientifiques traitant de méthodes, exactes ou approchées, s'inspirant du TSP pour résoudre le problème de routing au sein d'un stock. [Mohsen et al., 2002] ont développé des outils d'aide à la décision pour déterminer certains aspects de la gestion d'un entrepôt tels que le nombre et la localisation des quais de chargement/déchargement, le nombre et l'orientation des allées dans le stock ou encore l'emplacement des points de dépôt. Ces outils prennent en compte un certain nombre de spécificités des problématiques de stockage pour s'assurer de produire les meilleures solutions possibles en fonction d'un objectif défini par l'utilisateur. [Moeller, 2011] ont axé leurs études sur les séquences de déplacement des agents effectuant le picking obtenu par résolution du TSP en les séparant en plusieurs catégories. Une séquence peut, en effet, être obtenue en considérant le stock de façon globale, en considérant simplement les agents ou alors en résolvant mathématiquement le TSP. L'étude met en évidence le fait que parfois, une séquence de picking peut sembler "étrange" à un agent. Dans ce cas, les risques que ce dernier choisisse volontairement de modifier sa route sont importants.

2.2/ SYSTÈME ÉTUDIÉ ET POSSIBILITÉ D'OPTIMISATION

Comme nous l'avons vu dans l'introduction, la PUI de l'hôpital Emile Muller abrite le stock de DM et de médicaments. À l'heure de la rédaction de ce mémoire, les deux stocks sont séparés pour des raisons plus historiques que logistiques. Le rassemblement des stocks permettrait de grouper les flux logistiques matériels et donc de simplifier la gestion des opérations de picking, de livraison de répartitions des commandes et bien d'autres. Bien sûr, un stock plus grand implique de nouveaux enjeux logistiques à commencer par la place nécessaire pour entreposer chaque référence. Pour résoudre ce problème, certaines PUI de taille importante ont opté pour un système plus classique d'étagères utilisées dans l'industrie (voir figure (2.1)).

L'utilisation de ce genre de solutions entraîne l'apparition de nouvelles problématiques. En effet, prélever un article en hauteur sur une étagère implique de mettre en place un système adapté comme un chariot élévateur (voir figure (2.2)). Dans le cadre du présent travail de thèse, nous imaginons un système de convoi de chariots transporteurs menés par un véhicule capable de prélever les articles sur les étagères. Un convoi (voir figure (2.3)) permet de transporter un grand nombre d'articles en une seule fois limitant ainsi les aller-retour à travers le stock. Dans le but de prendre en compte le plus de contraintes possible pour une future implantation, nous supposons que les contraintes suivantes doivent être respectées.

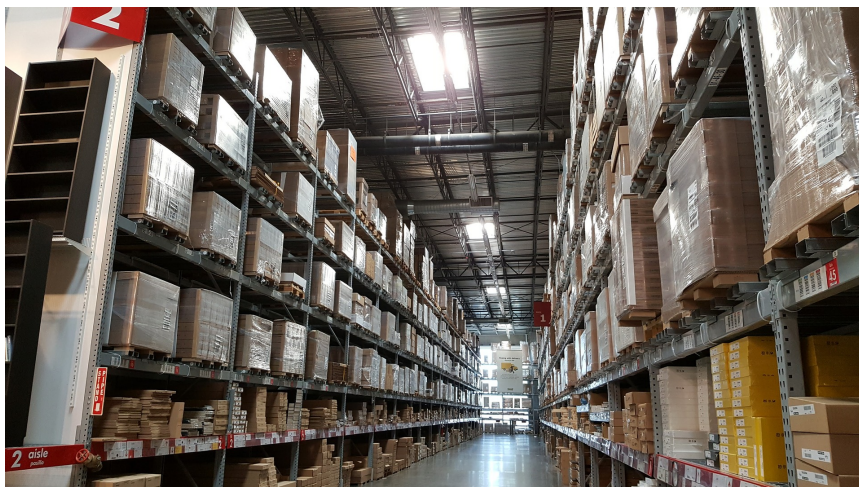


FIGURE 2.1 – Exemple de stock industriel avec des étagères



FIGURE 2.2 – Un chariot élévateur dans un entrepôt industriel

- 1 Le convoi ne peut pas faire demi-tour, car les allées ne sont pas assez larges pour que le conducteur puisse facilement manœuvrer.
- 2 Les véhicules utilisés dans les entrepôts peuvent être munis de barrières de sécurité pour protéger le conducteur de chutes d'objets éventuels. La barrière est positionnée d'un certain côté et le picking se fait de l'autre ce qui implique que le véhicule roule toujours du même côté des allées. Il existe aussi certains véhicules qui ne peuvent prendre en charge que les articles se trouvant d'un certain côté pour éviter d'avoir à manœuvrer (voir exemple figure 2.4).
- 3 Le convoi permet de transporter un certain nombre d'articles, mais sa capacité reste limitée. Si la quantité d'articles à collecter dépasse la capacité du véhicule,



FIGURE 2.3 – un convoi d'articles

un système de tournée doit être mis en place pour effectuer le picking en plusieurs fois (ou avec plusieurs véhicules).

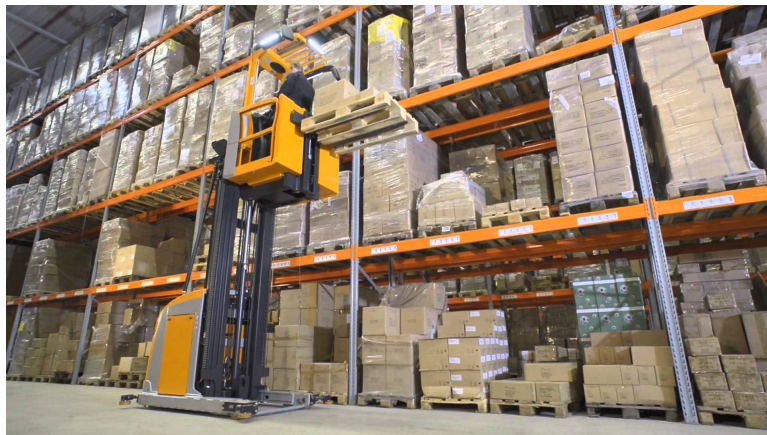


FIGURE 2.4 – Un chariot élévateur collectant uniquement sur le coté

2.3/ REPRÉSENTATION GRAPHIQUE ET MODÈLE MATHÉMATIQUE

Le problème que nous cherchons à résoudre ici consiste à minimiser la durée de l'opération de picking avec un véhicule. Concrètement, pour une liste de dotation donnée, on cherche à établir la route des différentes tournées du véhicule à travers le stock afin de collecter tous les articles de la liste en parcourant la plus petite distance possible et en respectant les contraintes énoncées précédemment. Ce problème s'apparente à un

problème de tournées de véhicules (aussi appelé VRP pour **V**ehicle **R**outing **P**roblem) avec des contraintes supplémentaires. Résoudre le VRP consiste à résoudre un TSP en considérant la possibilité de faire plusieurs tournées si la capacité du transporteur l'impose (voir définition (11)).

Définition 11 : Problème de tournées de véhicules

La classe des problèmes de tournées de véhicules consiste à gérer de manière optimale une flotte de véhicules (ou un véhicule effectuant plusieurs tournées), stationnés en un ou plusieurs dépôts, de façon à satisfaire les demandes d'un ensemble de clients ; il convient de déterminer le circuit visitant un sous-ensemble de clients en débutant et finissant aux dépôts, que doit effectuer chaque véhicule de manière à minimiser une fonction de coût. ([Teghem, 2012])

NB : Il est important de noter que la formulation basique du VRP ne fait pas référence à la capacité des véhicules. La prise en compte de cette dernière constitue une variante du problème appelé CVRP (pour **C**apacity **V**ehicle **R**outing **P**roblem).

Lors de la résolution d'un VRP, il est possible (et même utile) de représenter la situation sous forme de graphe. Pour un problème de picking, on utilisera un graphe dont les nœuds représentent les emplacements des articles à collecter et les arcs les chemins possibles entre ces nœuds. La figure (2.5) donne un exemple de représentation graphique des éléments d'une liste de picking au sein d'un stock. Notons qu'un graphe simple comme celui de la figure (2.5) ne permet pas de rendre compte des contraintes liées à l'utilisation de véhicules. Pour palier à ce problème, nous introduisons de nouveaux éléments de modélisation graphique.

2.3.1/ LES CARREFOURS

Comme nous l'avons vu précédemment, la première contrainte interdit au convoi de faire demi-tour à cause de la difficulté d'effectuer une manœuvre dans un espace exigu comme un entrepôt. Il est alors nécessaire de proposer des outils de modélisation graphiques pour représenter cette particularité. Le premier d'entre eux est le carrefour, c'est à dire un croisement entre deux allées qui permet au convoi d'aller dans trois directions (gauche, avant, droite) mais qui interdit de retourner en arrière (et donc de pouvoir faire de demi-tours). Pour pouvoir représenter cet élément, il est nécessaire de définir un point d'entrée et un point de sortie pour chacune des 4 directions (donc 8 en tout) et de les relier avec des arcs orientés. La modélisation complète est détaillée par la figure (2.6). Il est important de noter que sur cette représentation graphique, les croix cerclées

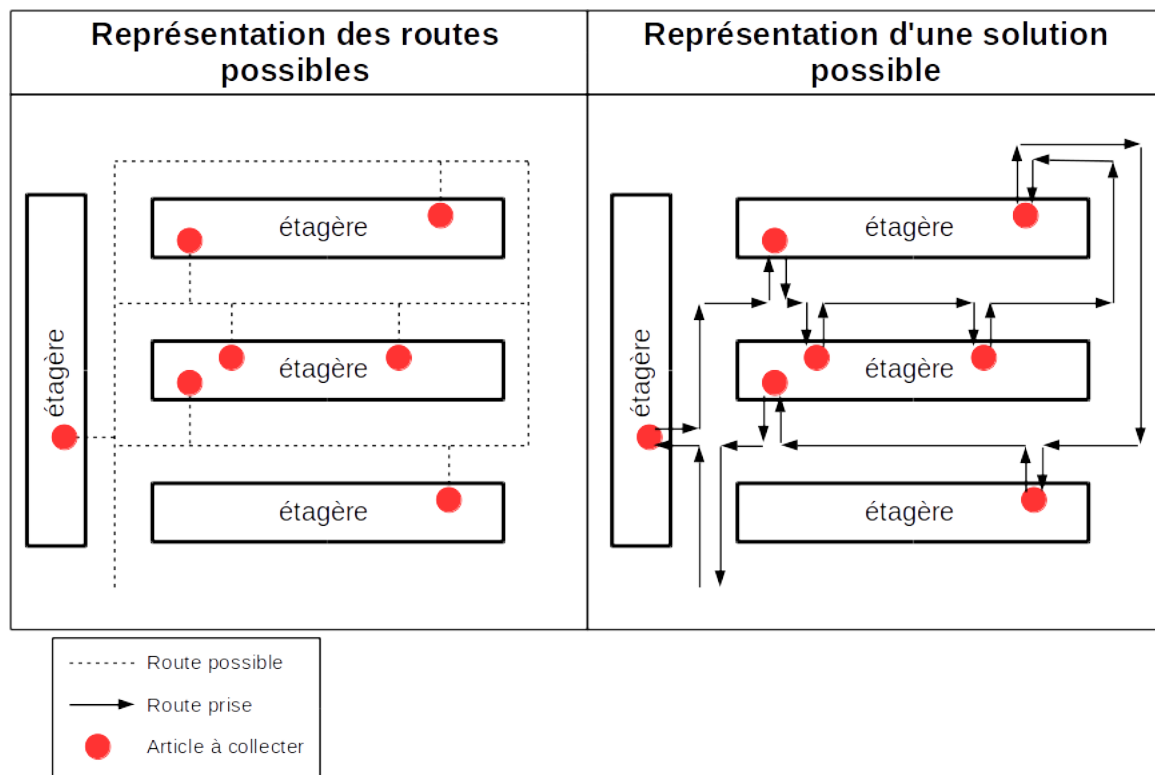


FIGURE 2.5 – exemple de représentation graphique d'un problème de routing

représentent des nœuds secondaires. Il s'agit de nœuds du graphe uniquement utilisés pour symboliser un chemin plus complexe qu'une simple ligne droite, contrairement aux nœuds principaux (les points rouges sur la figure (2.5)) il n'est pas nécessaire qu'ils soient atteints par un agent collecteur et n'augmentent pas la complexité du problème de picking (qui est détaillé plus bas).

On remarque que les arcs qui arrivent et partent du carrefour sont orientés. En effet,

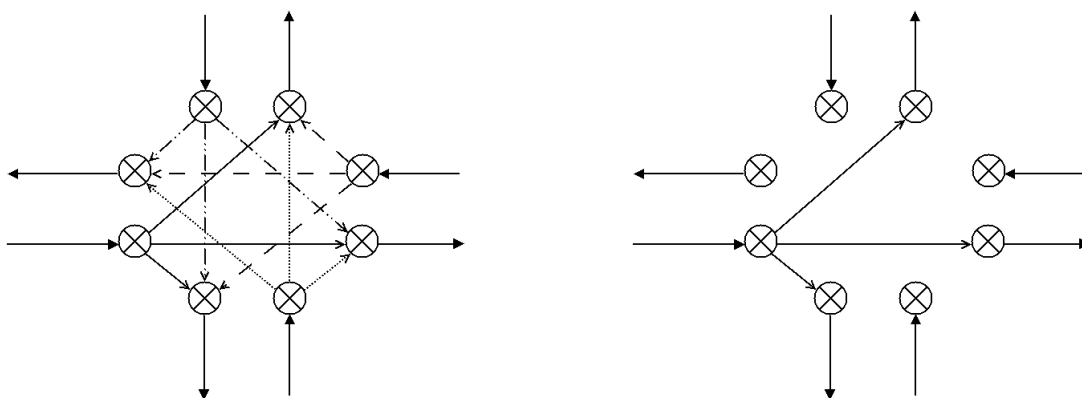


FIGURE 2.6 – Carrefour complet (gauche) et carrefour avec une direction détaillée (droite)

l'utilisation de ce genre d'arcs est primordiale pour être capable de gérer correctement la

direction des véhicules arrivant sur le carrefour. En conséquence, utiliser ces carrefours dans un graphe implique d'utiliser uniquement des arcs orientés. Si une allée est à double sens de circulation (ce qui est souvent le cas), il faut alors définir un arc pour chaque sens. La figure (2.7) permet de visualiser les différences entre une représentation graphique d'un entrepôt avec et sans carrefours, on notera que le premier cas est composé de deux fois plus d'arcs, mais en contrepartie permet de gérer la contrainte d'interdiction des demi-tours.

Les carrefours permettent donc de créer des graphes qui prennent en compte la

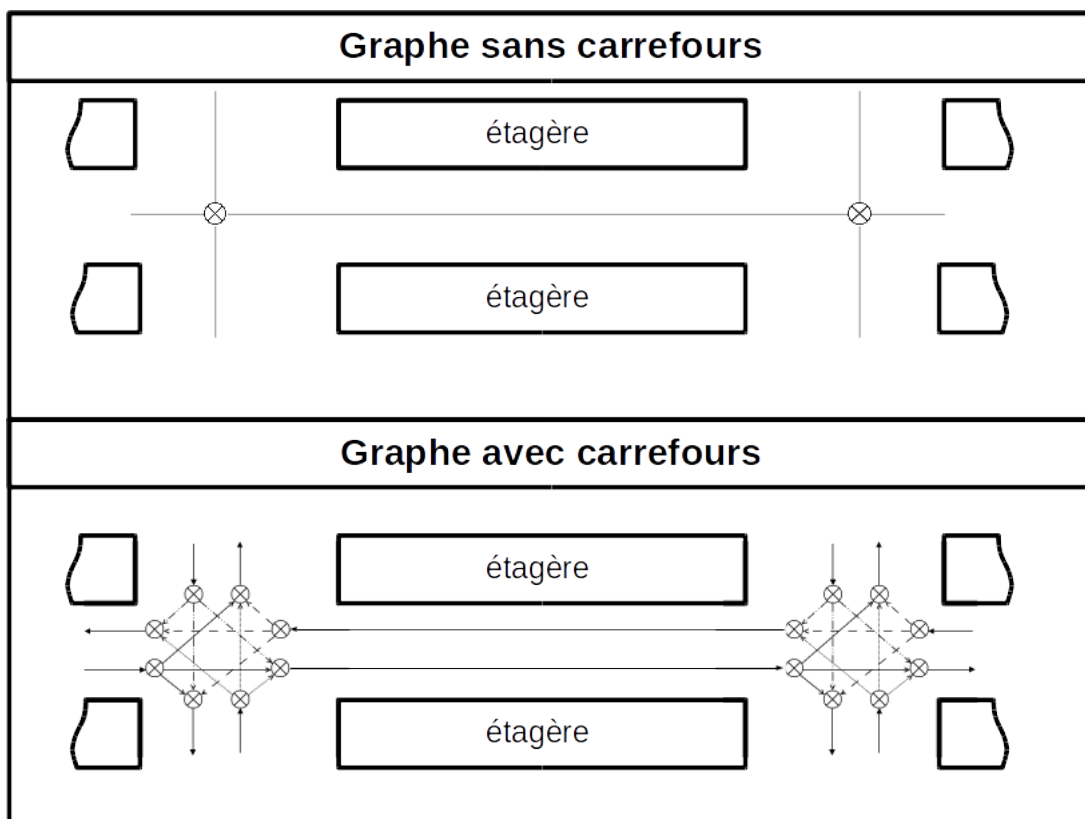


FIGURE 2.7 – représentation graphique avec et sans carrefours

contrainte des demi-tours. Pour régler le problème posé par la seconde contrainte (présence d'une barrière de sécurité) nous proposons une autre solution de modélisation graphique.

2.3.2/ LES NŒUDS ORIENTÉS

Toujours dans l'optique de proposer une modélisation du problème de picking qui respectent les contraintes proposées au début de ce chapitre, nous introduisons une nouvelle norme liée cette fois à la seconde contrainte. Pour rappel, la seconde contrainte im-

pose que lorsqu'un article est collecté, le véhicule qui pratique la collecte doit se trouver dans un certain sens pour des raisons de sécurité. Comme la première contrainte interdit les demi-tours, le véhicule doit forcément arriver dans le bon sens vers l'article. Cela implique, par exemple, qu'il est impossible de collecter deux articles se trouvant sur des étagères se faisant face en un seul passage.

Pour respecter cette contrainte d'orientation, nous introduisons donc la notion de nœuds orientés. Un nœud orienté est un nœud simple avec une contrainte d'orientation représentée par une flèche (voir figure 2.8). La flèche permet de savoir d'où doit venir l'agent ou le véhicule pour pouvoir collecter l'article qui se trouve sur le nœud. Il est im-



FIGURE 2.8 – nœud simple – nœud orienté vers la droite – nœud orienté vers la gauche

portant de noter que tel quel un nœud orienté est inutile puisqu'il est considéré comme un nœud simple lors du calcul de la solution (détaillé dans la section suivante). En réalité l'avantage de ce type de nœuds et de pouvoir créer un graphe en précisant sur chaque nœud l'orientation que doit avoir le véhicule, mais en utilisant des arcs non orientés. Puis, à partir de ce graphe, on utilise un algorithme pour remplacer les arcs non orientés par des arcs orientés correctement reliés aux nœuds et aux carrefours (comme représenté par la figure (2.9)). La méthode complète est détaillée par l'algorithme (2.9). Ce dernier doit être appliqué sur chaque allée, c'est-à-dire chaque série de nœuds orientés reliée à des extrémités de carrefours (même s'ils ne forment pas de lignes droites). L'algorithme est prévu pour créer une allée en créant les arcs orientés de la gauche vers la droite, il est bien sûr possible d'inverser le sens de construction ou de traiter les allées verticales, coudées ou de n'importe quelle forme. Le détail de ces algorithmes n'est pas donné ici, car ils sont analogues à celui présenté.

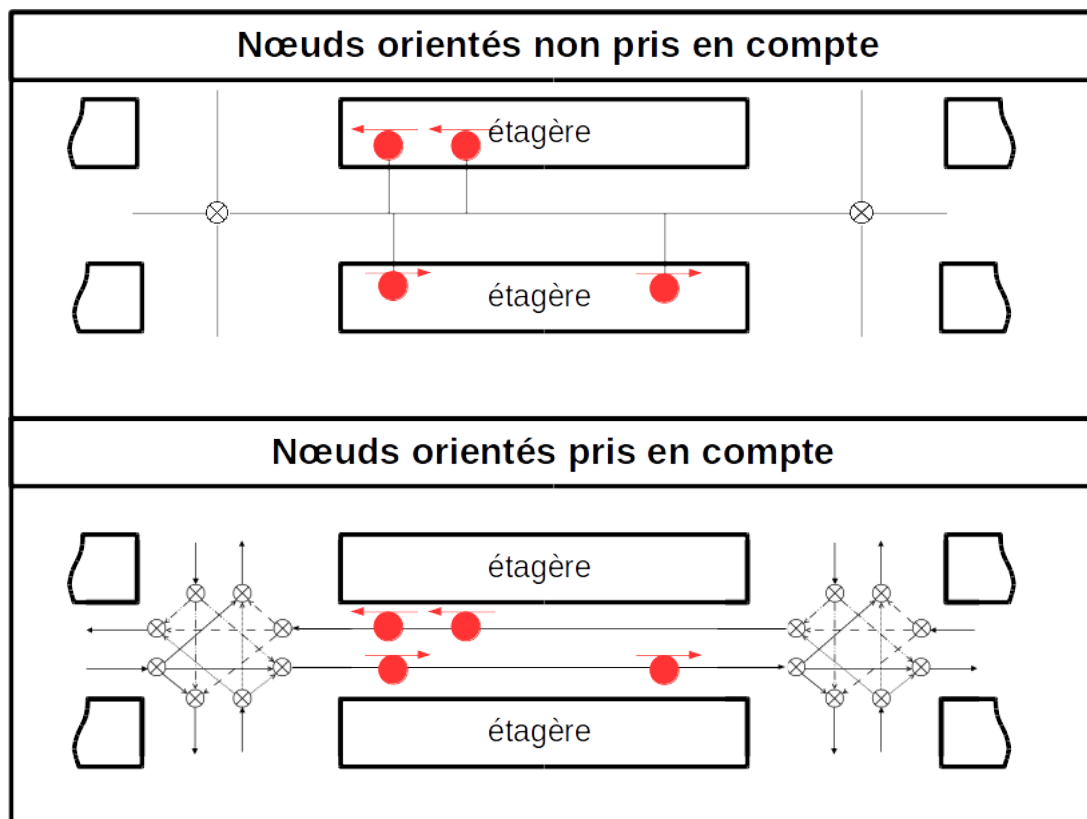


FIGURE 2.9 – Représentation graphique avec et sans nœuds orientés

la section suivante est dédié à l'explication des différentes étapes de l'algorithme. Initialement, cette allée est constituée de plusieurs nœuds orientés et d'un nœud secondaire, le tout relié sans logique particulière à des carrefours. L'algorithme étant récursif, il est itéré en tout 5 fois pour construire l'allée de cet exemple.

Initialisation : pour l'initialisation de l'algorithme, la variable "nœudGauche" correspond au nœud haut du côté droit du carrefour de gauche alors que la variable "nœudDroite" correspond au nœud bas du côté droit du carrefour de gauche. La variable "nœudCourant" quant à elle correspond au premier nœud de l'allée (voir figure (2.10)).

Itération 1 : le nœud courant est considéré. Il s'agit d'un nœud orienté vers la droite, il est donc relié au nœud d'entrée du carrefour par un arc orienté. Le nœud courant devient alors le prochain nœud de l'allée et le nœud droite devient l'ancien nœud courant, car il s'agit du dernier nœud orienté à droite traité. Pour la même raison, le nœud gauche n'est pas modifié (voir figure (2.11)).

creerArcSuivant

Inputs :

graph : ensemble d'arcs (non orientés)
nœudDroit : nœud droit du carrefour de départ
nœudGauche : nœud gauche du carrefour de départ
nœudCourant : premier nœud de l'allée
nœudTraites : nœuds secondaires déjà traités

tableDesArcs $\leftarrow \emptyset$

trouve \leftarrow **faux**

si *nœudCourant.direction()* = "droite" **alors**

a ajouter *arc*(*nœudDroit*, *nœudCourant*) à *tableDesArcs*

nœudDroit \leftarrow *nœudCourant*

fin

sinon si *nœudCourant.direction()* = "gauche" **alors**

a ajouter *arc*(*nœudCourant*, *nœudGauche*) à *tableDesArcs*

nœudGauche \leftarrow *nœudCourant*

fin

sinon si *nœudCourant.AppartientAUnCarrefour()* = "faux" **alors**

nœudtemp \leftarrow *nœudCourant.dupliquer()*

a ajouter *arc*(*nœudDroit*, *nœudCourant*) à *tableDesArcs*

nœudDroit \leftarrow *nœudCourant*

a ajouter *arc*(*nœudCourant*, *nœudGauche*) à *tableDesArcs*

nœudGauche \leftarrow *nœudTemp*

fin

sinon

si *nœudCourant.emplacementCarrefour* = "GAUCHE_BAS" **alors**

a ajouter *arc*(*nœudDroit*, *nœudCourant*) à *tableDesArcs*

a ajouter *arc*(*nœudCourant.Carrefour.nœud*("GAUCHE_HAUT"), *nœudGauche*) à *tableDesArcs*

 retourner *tableDesArcs*

fin

fin

pour *arc* de *graphe.premierElem()* à *graphe.dernierElem()* **faire**

si *arc.nœud1()* = *nœudCourant* **et** *nœudTraites.NeContientPas*(*arc.nœud2()*) **alors**

nœudCourant \leftarrow *arc.nœud2()*

trouve \leftarrow **vrai**

fin

sinon si *arc.nœud2()* = *nœudCourant* **et** *nœudTraites.NeContientPas*(*arc.nœud1()*)

alors

nœudCourant \leftarrow *arc.nœud1()*

trouve \leftarrow **vrai**

fin

fin

si *trouve* = **vrai** **alors**

creerArcSuivant(*graphe*, *nœudDroit*, *nœudGauche*, *nœudCourant*, *nœudtraites*)

fin

retourner *tableDesArcs*

Algorithme 1 : Algorithme de création d'allées avec nœuds orientés

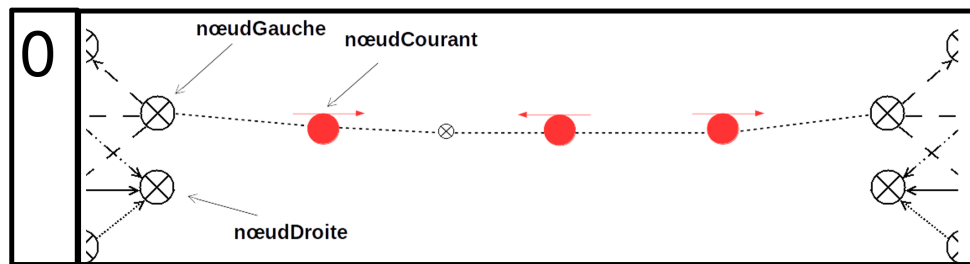


FIGURE 2.10 – Exemple d'application de l'algorithme de création d'allées - Étape 0

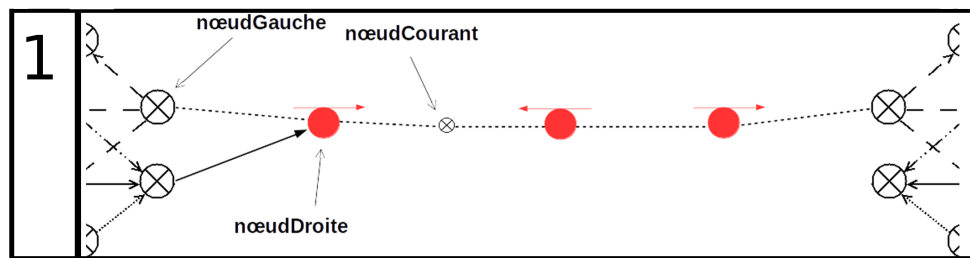


FIGURE 2.11 – Exemple d'application de l'algorithme de création d'allées - Étape 1

Itération 2 : le nœud courant est un nœud secondaire, il est donc non orienté. Pour résoudre ce problème, le nœud est alors dupliqué et deux arcs orientés sont créés pour relier le premier nœud au nœud gauche et sa duplication au nœud droite. Les deux nœuds secondaires deviennent alors respectivement les nouveaux nœud gauche et nœud droite et le nœud suivant de l'allée devient le nœud courant (voir figure (2.12)).

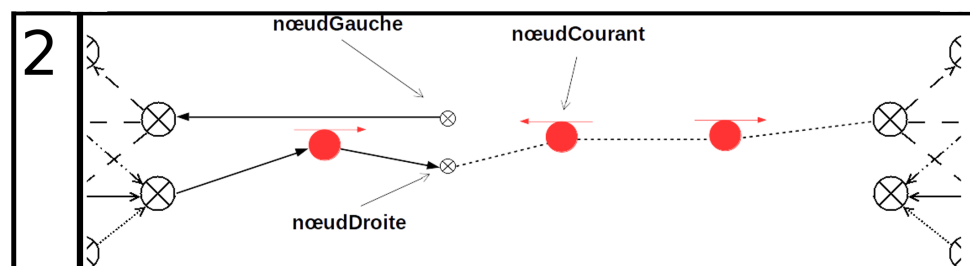


FIGURE 2.12 – Exemple d'application de l'algorithme de création d'allées - Étape 2

Itération 3 : le nœud courant est à présent un nœud orienté vers la gauche, il est donc relié par un arc orienté au nœud gauche. Le nœud courant devient alors le prochain nœud sur l'allée et le nœud gauche devient l'ancien nœud courant. Le nœud droite, quant à lui, n'est pas modifié (voir figure (2.13)).

Itération 4 : le nouveau nœud courant est un nœud orienté vers la droite il est donc relié au nœud droite. Le nœud droite devient alors le nœud courant et le nœud

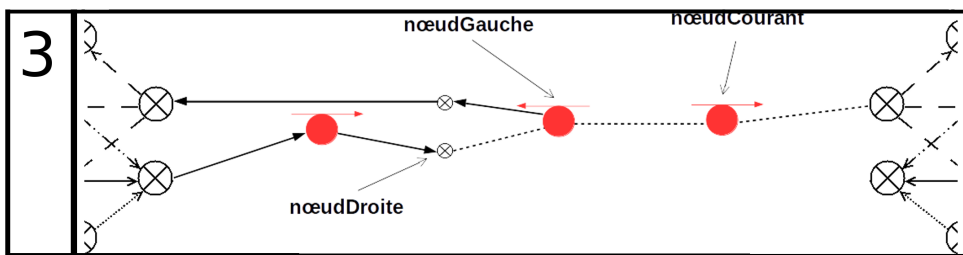


FIGURE 2.13 – Exemple d'application de l'algorithme de création d'allées - Étape 3

gauche n'est pas modifié. Le nœud courant, quant à lui, devient le prochain nœud de l'allée, c'est-à-dire un nœud du carrefour de droite (voir figure (2.14)).

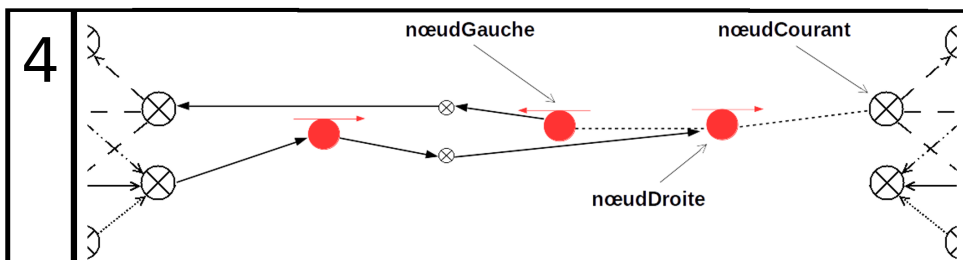


FIGURE 2.14 – Exemple d'application de l'algorithme de création d'allées - Étape 4

Itération 5 : le nœud courant est donc, à présent, celui d'un carrefour. Il s'agit donc de la dernière itération. Le nœud gauche est donc relié au nœud haut du côté gauche du carrefour de droite tandis que le nœud droite est relié au nœud bas du côté gauche du carrefour de droite (voir figure (2.15)).

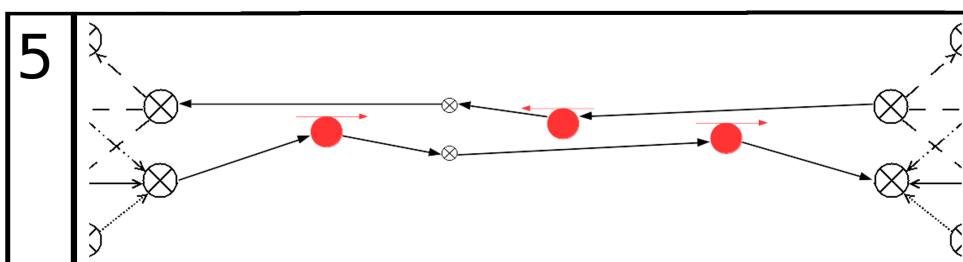


FIGURE 2.15 – Exemple d'application de l'algorithme de création d'allées - Étape 5

Les arcs non orientés peuvent alors être supprimés pour laisser place à une allée composée uniquement d'arcs orientés respectant la contrainte d'orientation du véhicule lors du picking.

Il est important de noter que sur l'exemple certains nœuds ont été légèrement décalés pour des raisons de visibilité. En réalité, la position des nœuds reste fixe afin de pouvoir, par la suite, correctement calculer les distances. En particulier, lors de la duplication des nœuds secondaires, l'original et sa copie sont en réalité superposés.

Grâce aux méthodes de construction graphique mises en place dans ce chapitre, il est possible de représenter le problème de picking par un graphe qui respecte les contraintes d'impossibilité des demi-tours et d'orientation du véhicule de transport. La troisième contrainte (la capacité du véhicule) n'a pas d'impact d'un point de vue graphique. Il s'agit alors de mettre en place un modèle mathématique qui prenne en compte cette spécificité.

2.3.3/ MODÈLE MATHÉMATIQUE

Comme nous l'avons vu précédemment, le VRP est un problème connu de recherche opérationnelle dont le but est de gérer des tournées de transporteurs pour livrer des clients. Ce problème, proposé par Dantzig et Ramser en 1959 (voir [Dantzig et al., 1959]), peut tout à fait être adapté au problème de picking dans un entrepôt en considérant un agent qui doit relier les emplacements d'un ensemble d'articles à collecter en une ou plusieurs tournées (ou plusieurs agents se partageant les tournées). Un algorithme de résolution du VRP permettrait alors d'obtenir l'ordre dans lequel l'agent devra collecter les articles et les moments où il devra les décharger pour pouvoir continuer sans dépasser la capacité maximum des chariots de transports.

Afin de parvenir à résoudre ce problème, la première étape consiste alors à établir un modèle mathématique fiable.

Cette section détaille le modèle d'optimisation linéaire en nombre entier (que par la suite nous appellerons par son acronyme anglais MILP pour **Mixed Integer Linear Programming**) du problème (voir définition (13)).

Définition 12 : modèle LP (Linear Programming)

Problème mathématique consistant à rechercher l'optimum d'une fonction de variables, étant donné que celles-ci doivent vérifier un certain nombre d'équations et/ou d'inéquations (appelées contraintes). Autant la fonction à optimiser que les contraintes doivent être linéaires, c'est à dire du premier degré en les variables. ([Teghem, 2012])

Définition 13 : modèle MILP (Mixed Integer Linear Programming)

Modèle mathématique LP (voir définition (12)) dont tout ou partie des variables peuvent être à valeurs entières (on parle aussi de contraintes d'intégrité). ([Teghem, 2012])

Pour écrire le modèle MILP du VRP il est nécessaire de poser certains paramètres et variables.

- n : nombre de références (et donc d'emplacements).
- $V = v_1, \dots, v_n$: ensemble des articles (nœuds) à collecter.
- Q : volume maximum qu'un convoi peut transporter.
- q_i : volume de l'article v_i .
- $d_{i,j}$: distance à parcourir pour aller de l'emplacement de l'article v_i à celui de l'article v_j .
- $x_{i,j,k}$: variable binaire égale à 1 si et seulement si lors de la tournée k l'agent doit aller directement de v_i vers v_j .
- $u_{i,k}$: l'ordre dans lequel l'article v_i sera collecté lors de la tournée k .
- n_k : nombre d'articles collectés lors de la tournée k .

Il est alors possible d'écrire le modèle MILP.

La fonction objectif est donnée par :

$$\text{Minimiser } \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^m x_{ijk} d_{ij} \quad (2.6)$$

sujet à un certains nombre de contraintes :

$$\sum_{i=1}^n \sum_{k=1}^m x_{ijk} = 1 \quad \forall 1 \leq j \leq n \quad (2.7)$$

$$\sum_{j=1}^n \sum_{k=1}^m x_{ijk} = 1 \quad \forall 1 \leq i \leq n \quad (2.8)$$

$$\sum_{j=1}^n x_{jik} = \sum_{j=1}^n x_{ijk} \quad \forall 1 \leq i \leq n \quad \forall 1 \leq k \leq m \quad (2.9)$$

$$\sum_{j=1}^n x_{0jk} = 1 \quad \forall 1 \leq k \leq m \quad (2.10)$$

$$\sum_{i=1}^n x_{i0k} = 1 \quad \forall 1 \leq k \leq m \quad (2.11)$$

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} \times q_i \leq Q \quad \forall 1 \leq k \leq m \quad (2.12)$$

$$u_{1k} = 1 \quad \forall 1 \leq k \leq m \quad (2.13)$$

$$u_{ik} - u_{jk} + n_k \times x_{ijk} \leq n_k - 1 \quad \forall 1 \leq i, j \leq n \quad \forall 1 \leq k \leq m \quad (2.14)$$

$$\sum_{i=1}^n \sum_{j=1}^n x_{ijk} = n_k \quad \forall 1 \leq k \leq m \quad (2.15)$$

L'équation (2.6) représente la fonction objectif, c'est-à-dire la façon d'évaluer une solution. Ici, on cherche une solution qui minimise le plus possible la distance parcourue par le ou les agents pour collecter l'intégralité des articles demandés. En plus de cela, la solution doit respecter un certain nombre de contraintes. Les contraintes (2.7) et (2.8) assurent respectivement que pour chaque nœud, l'agent ne peut venir que depuis un unique nœud et ne peut repartir que vers un unique nœud. Ces deux contraintes assurent donc la continuité du chemin de l'agent collecteur lors de l'opération de picking. La contrainte (2.9) est une contrainte classique de problèmes de détermination de meilleur chemin. Son but est d'assurer la conservation des flux, en d'autres termes, pour chaque nœud il y a autant de chemins qui y mènent que de chemins qui en partent. La contrainte (2.10) force chaque tournée à démarrer au même point. Ce point est le nœud d'indice 0 qui sert de point de départ (aussi appelé dépôt). Conjointement, la contrainte (2.11) force chaque tournée à se terminer à ce même dépôt pour que les articles soient déchargés. Ainsi, chaque tournée forme un cycle à travers l'entrepôt. La contrainte (2.12) permet de s'assurer que la capacité totale du véhicule transporteur n'est jamais dépassée, c'est-à-dire qu'il est capable de transporter tous les articles de chaque tournée.

La résolution de ce modèle par un logiciel peut donner une solution contenant des sous-tours, c'est à dire des solutions où certains nœuds ne sont pas reliés à certains autres même indirectement (voir exemple figure (2.16)). Pour pallier à ce problème, on définit la variable $u_{i,k}$ qui représente l'ordre dans lequel l'article i sera collecté lors de la tournée k . En effet, si la solution déterminée possède des sous-tours, il est impossible de définir un ordre de collecte alors que ce dernier est évident avec un unique tour. Pour correctement définir cet ordre, la contrainte (2.13) précise tout d'abord que le dépôt doit être le premier nœud visité. Ensuite, la contrainte (2.14) définit la notion d'ordre des articles. Notons que la contrainte (2.13) n'est en réalité pas nécessaire, car théoriquement n'importe quel article d'une tournée peut être considéré comme le premier, mais dans ce cas la variable u n'a plus qu'un intérêt calculatoire et ne représente plus la réalité. Ces deux dernières équations font appel à la variable n_k qui représente le nombre d'articles collectés lors de la tournée k . La valeur de cette variable est déterminée par l'équation (2.15). Nous verrons dans le chapitre suivant qu'il existe d'autres méthodes pour éviter la formation de sous-tours.

L'élaboration du modèle MILP permet de mettre en évidence les variables et les contraintes qui entrent en jeu dans la résolution du problème. Un tel modèle peut être résolu de façon exacte grâce à des programmes spécifiques appelés solveurs (voir définition (14)).

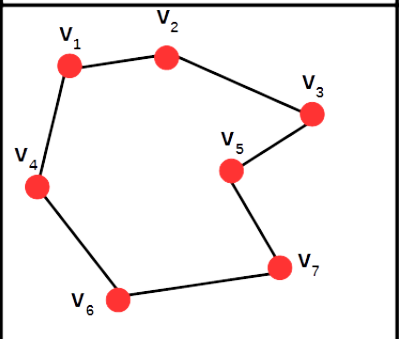
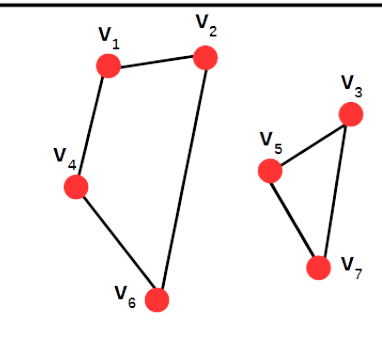
Sans sous-tours	Avec sous-tours
$X_{1,2,1} = 1$ $X_{7,6,1} = 1$ $X_{2,3,1} = 1$ $X_{6,4,1} = 1$ $X_{3,5,1} = 1$ $X_{4,1,1} = 1$ $X_{5,7,1} = 1$	$X_{1,2,1} = 1$ $X_{5,3,1} = 1$ $X_{2,6,1} = 1$ $X_{3,7,1} = 1$ $X_{6,4,1} = 1$ $X_{7,5,1} = 1$ $X_{4,1,1} = 1$
	

FIGURE 2.16 – Exemple d'une solution comprenant des sous-tours

Définition 14 : Solveur

Un solveur est un programme logiciel permettant, sous certaines contraintes, de calculer et fournir la solution d'un problème après sa transcription en équations mathématique ou langage informatique.

Ils en existe plusieurs sortes sur le marché, certains sont des modules de programmes plus importants tels que Matlab ([Guide, 1998]) ou Scilab ([Enterprises, 2012]). D'autre, sont des programmes dédiés comme Gurobi ([Optimization, 2014]) ou Cplex ([CPLEX, 2009]).

Malheureusement des test préliminaires ont montré que la résolution par à un solveur du problème étudié ici s'avère être une tâche nécessitant un temps de calcul important. En conséquence, cette solution ne peut pas être utilisée pour obtenir une liste de picking en temps réel (c'est à dire dès sa réception par les gestionnaires du stock). Pour résoudre ce problème, nous proposons de mettre en place un algorithme génétique. Il s'agit d'une méthode qui a déjà fait ses preuves pour résoudre des problèmes dérivés du TSP et capable de déterminer une bonne solution (à priori non optimale) en un temps respectable. De plus, les résultats obtenus avec le MILP permettront de mesurer l'efficacité de notre méthode sur des exemples types.

2.4/ ALGORITHME GÉNÉTIQUE

Un algorithme génétique (ou AG) est une métaheuristique (voir définition (15)) basée sur le principe de l'évolution d'une population de solutions à l'instar de "l'optimisation par essaim particulaire" ou des algorithmes de "colonies de fourmis" ([Talbi, 2009]). La méthode a été inventée en 1960 par John Holland et popularisée par David Goldberg ([Golberg, 1989]). Le principe est d'utiliser les mécanismes évolutifs d'une population d'individus (au sens darwinien du terme). Pour ce faire, une population composée de solutions initiales est tout d'abord créée. Cette population est considérée comme la première génération. Puis, génération après génération, de nouvelles solutions sont créées à partir des précédentes en effectuant des croisements et des mutations sur les solutions.

Les algorithmes basés sur l'évolution de populations ont déjà prouvé, par le passé, leur efficacité lorsqu'il s'agit de résoudre des problèmes complexes dérivés du TSP et particulièrement lorsqu'il existe potentiellement plusieurs solutions optimales ([Snyder et al., 2006, Grefenstette et al., 1985, Dorigo et al., 1997]).

Définition 15 : Métaheuristique

Une métaheuristique est une heuristique (voir définition (16)) qui n'est pas dédiée à un problème particulier.

Définition 16 : Heuristique

Une heuristique est une méthode de résolution dédiée à un problème précis et dont le but est de fournir une solution aussi proche que possible de l'optimal sans certitude de l'atteindre. Dans le langage courant on parle généralement de "bonnes" solutions. ([Teghem, 2012])

2.4.1/ PRINCIPE ET ALGORITHME

Pour implémenter un algorithme génétique, il est nécessaire de commencer par proposer une représentation des solutions. Pour le problème étudié, une solution consiste en un ensemble d'articles ordonnés et repartis en plusieurs (ou une unique) tournées. Supposons pour l'exemple que l'on cherche à collecter 9 articles numérotés de 1 à 9, alors la figure (2.17) donne une représentation d'une solution possible. Cette solution est composée de 4 tournées contenant respectivement les articles [1,6,2], [4,9], [3,8,5] et [7]. Lors de chaque tournée, les articles doivent être collectés dans l'ordre inscrit.

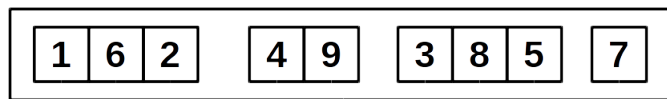


FIGURE 2.17 – Représentation d'une solution

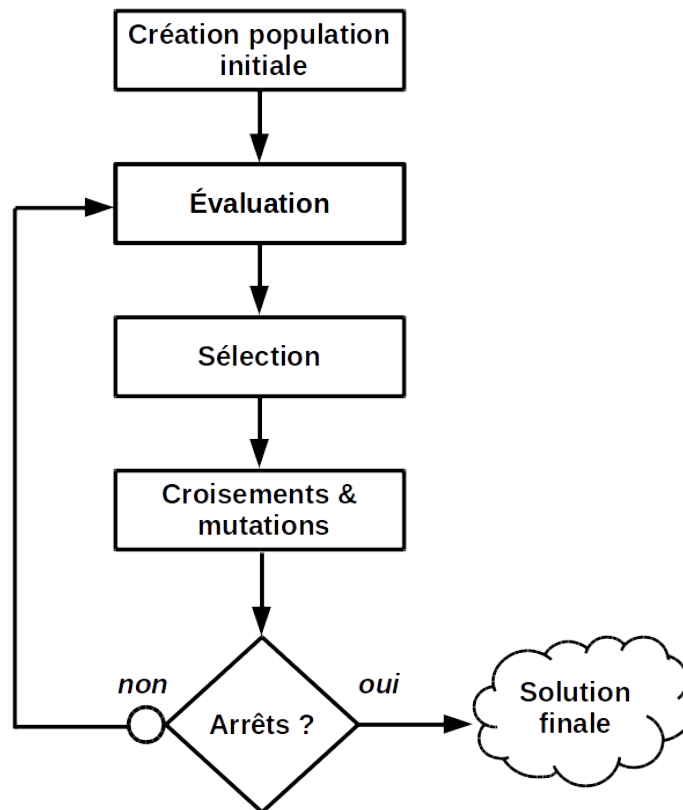


FIGURE 2.18 – Les étapes de l'algorithme génétique

Lors de l'exécution d'un algorithme génétique, les solutions sont appelées des individus et les composantes d'une solution sont des gènes. Comme dit précédemment, l'idée est alors de simuler une population de solutions qui génération après génération se reproduisent et mutent pour s'adapter à leur environnement.

L'algorithme génétique détaillé dans ce manuscrit se décompose en plusieurs étapes (détaillées par la figure (2.18)).

- 1 – Initialisation :** une population initiale est tout d'abord créée, elle est constituée de solutions (ou individus) créées aléatoirement. La taille de cette population est définie par un paramètre modifiable.
- 2 - Évaluation :** chaque individu de la population courante est évalué selon la distance totale de picking qu'il représente (voir équation (2.6)). La population est alors triée du meilleur au pire individu (c'est à dire de la plus petite à la plus

grande distance à parcourir).

3 - Sélection : une nouvelle population est alors créée en sélectionnant les meilleurs individus de la population courante (ce qui est simplifié par le tri effectué à l'étape précédente). Le nombre d'individus sélectionnés est généralement identique à la taille de la population initiale bien que ça ne soit pas une obligation.

4 – Croisements et mutations : les meilleurs individus de la population courante sont croisés pair à pair grâce à un opérateur de croisement pour créer de nouveaux individus. Ensuite, une autre partie des meilleurs individus est mutée pour en créer encore de nouveau. Les nouveaux individus ainsi créés sont rajoutés à la population courante. Il est important de noter que le taux de la population sélectionnée pour les croisements et les mutations est un paramètre modifiable.

5 – Vérification de la condition d'arrêt : les conditions d'arrêt peuvent être de natures diverses. Dans le cas de la présente étude, il s'agira d'avoir atteint un certain nombre de générations ou d'avoir dépassé une limite de temps fixée. Si l'une de ces conditions est remplie, l'algorithme s'arrête et fournit la meilleure solution de la population courante comme solution finale. Dans le cas contraire, l'algorithme repasse à l'étape 2. Ici encore, le nombre limite de générations et la limite de temps sont des paramètres modifiables.

L'algorithme génétique complet basé sur ces étapes est détaillé par l'algorithme (2) Comme nous venons de le voir, pour mettre en place un algorithme génétique il est donc nécessaire de détailler les mécanismes de croisement et de mutation.

2.4.2/ CROISEMENTS ET MUTATIONS

Les méthodes de croisement et de mutations des individus sont au cœur du fonctionnement d'un algorithme génétique. En effet, c'est grâce à ces mécanismes, hérités des observations sur la théorie de l'évolution de Charles Darwin, que les individus peuvent s'améliorer en produisant des solutions de plus en plus performantes (à l'image d'une population qui devrait s'adapter à un changement génération après génération).

Les fonctions produisant ces nouvelles solutions sont appelées des opérateurs (de croisement ou de mutation).

Inputs :

taillePop : taille de la population initiale
maxIteration : nombre maximal d'itérations (générations)
tauxCrois : taux de croisement (en pourcentage)
tauxMut : taux de mutation (en pourcentage)

```

pop ← ∅
pour i ← 0 à taillePop faire
  | ajouter solutionAleatoire() a pop
fin
pop ← trier(pop)
nouvellePop ← pop
pour j ← 0 à maxIteration faire
  | pour k ← 0 à taillePop * tauxCrois/100 faire
  |   | r ← entierAleatoire() mod 2
  |   | si r = 0 alors
  |   |   | ajouter croisement1(pop.get(k), pop.get(k + 1)) a nouvellePop
  |   |   | sinon si r = 1 alors
  |   |   |   | ajouter croisement2(pop.get(k), pop.get(k + 1)) a nouvellePop
  |   |   |   | fin
  |   |   | fin
  |   | pour l ← 0 à taillePop * tauxMut/100 faire
  |   |   | r ← entierAleatoire() mod 3
  |   |   | si r = 0 alors
  |   |   |   | ajouter mutation1(pop.get(k)) a nouvellePop
  |   |   |   | sinon si r = 1 alors
  |   |   |   |   | ajouter mutation2(pop.get(k)) a nouvellePop
  |   |   |   |   | sinon si r = 2 alors
  |   |   |   |   |   | ajouter mutation3(pop.get(k)) a nouvellePop
  |   |   |   |   |   | fin
  |   |   |   |   | fin
  |   |   |   | fin
  |   |   | fin
  |   | nouvellePop ← trier(nouvellePop)
  |   | pop ← ∅
  |   | pour i ← 0 à taillePop faire
  |   |   | ajouter nouvellePop.get(i) a pop
  |   |   | fin
  |   | fin
fin

```

Algorithme 2 : Algorithme génétique

2.4.2.1/ OPÉRATEUR DE CROISEMENT

L'opérateur de croisement sert à simuler la "reproduction" de deux individus. L'idée est donc, à partir de deux solutions parentes, de créer une solution fille partageant des gènes communs avec ces deux parents. On fait alors se "reproduire" les meilleurs individus en espérant que les enfants ainsi créés garderont les gènes qui ont permis d'obtenir des solutions de bonne qualité. La figure (2.19) présente un exemple de croisement entre deux individus. L'opérateur de croisement utilisé dans cette étude identifie les parties robustes de chaque parent, c'est-à-dire les articles positionnés à la même place dans

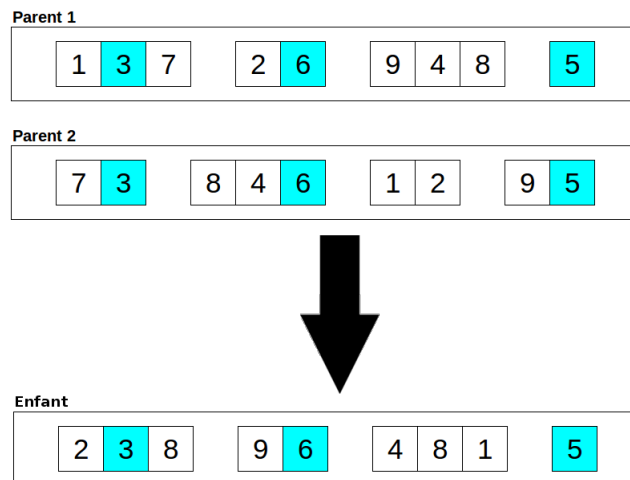


FIGURE 2.19 – Exemple de croisement de deux individus

chaque solution (en bleu sur la figure). Ces parties sont les gènes communs aux deux parents et ils seront donc transmis à leur enfant. Les autres parties seront aléatoirement choisies pour créer la nouvelle solution.

Effectué des croisements au sein d'une population permet, théoriquement, d'obtenir des solutions de plus en plus proches de l'optimal, mais il reste possible que ces dernières finissent par rester conditionnée à un minimum local dont il sera impossible de sortir. Pour régler ce problème, nous faisons appel à un opérateur de mutation.

2.4.2.2/ OPÉRATEUR DE MUTATION

Un opérateur de mutation, comme son nom l'indique, a pour but de simuler la "mutation génétique" d'un individu. Les mutations sont des modifications des gènes qui peuvent intervenir de façon sporadique au sein d'une population. Faire muter un individu permet d'explorer une nouvelle partie de l'espace de solutions et donc d'éviter de faire converger une population vers un optimum local. Si un individu muté possède une bonne évaluation, il pourra être sélectionné pour être croisé avec d'autres et sa particularité génétique pourra ainsi se transmettre au reste de la population. La figure (2.20) présente un exemple de l'utilisation de l'opérateur de croisement créé pour notre étude. Cet opérateur nécessite en effet une seule opérande, il choisit aléatoirement deux gènes de la solution et les intervertit pour en former une nouvelle. La solution ainsi créée peut être ajoutée à la population courante. Il est important de noter que l'ancienne solution n'est pas détruite pour s'assurer de ne pas perdre de bons individus.

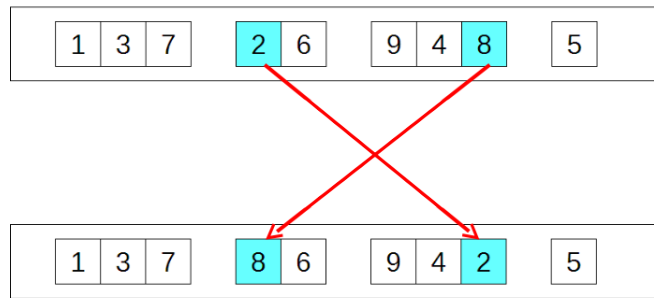


FIGURE 2.20 – Exemple de mutation d'un individu

2.5/ RÉSULTATS EXPÉRIMENTAUX

Pour être en mesure de valider l'efficacité de l'algorithme génétique développé précédemment, il est nécessaire de tester ce dernier dans plusieurs cas réalistes en comparant les résultats obtenus avec ceux d'un solveur exact. Pour ce faire, on imagine un entrepôt composé de plusieurs allées formées par des étagères remplies d'articles sur plusieurs niveaux et d'un dépôt où les articles sont déchargés entre chaque tournée (voir figure (2.21)).

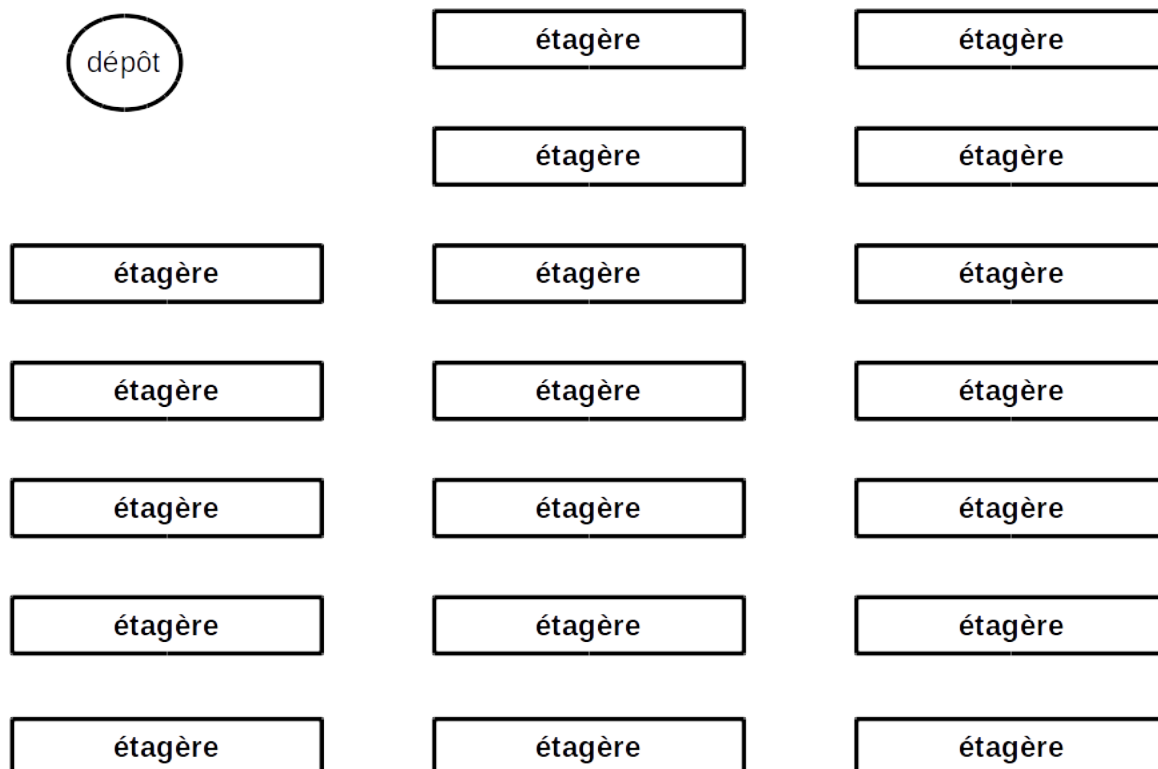


FIGURE 2.21 – Entrepôt organisé en allées

Pour effectuer les opérations de picking, un agent conduit un véhicule type "chariot élévateur". On suppose que ce chariot est pourvu d'une barrière de sécurité qui impose de toujours conduire à droite dans les allées (notons que les résultats présentés dans cette section sont équivalents pour un chariot élévateur fonctionnant avec une logique inverse). Des chariots transporteurs sont accrochés à ce véhicule pour facilement déplacer les articles collectés à travers le stock formant ainsi un convoi.

On suppose pour les besoins de l'expérience que la capacité de transport totale du convoi est de 7 unités de volume ($Q = 7$) et que chaque article représente 1 unité de volume ($q_i = 1 \forall 1 \leq i \leq n$), ce qui correspond grossièrement au nombre d'articles moyens transportable par les chariots de la PUI étudiée ici.

Les expérimentations sont menées sur plusieurs listes de picking possédants de 15 à 150 éléments. Pour chaque liste, une représentation graphique est faite, les articles à collecter sont représentés par des nœuds orientés dont l'orientation dépend de son positionnement dans les allées (c'est-à-dire de façon à s'assurer que le véhicule roule toujours à droite). Un carrefour est placé à chaque intersection pour gérer la contrainte des demi-tours et les liaisons entre les nœuds et les carrefours sont construites à l'aide de l'algorithme (1). Enfin, un nœud est placé au niveau du dépôt pour que l'agent puisse déposer les articles collectés à la fin de chaque tournée.

Afin d'illustrer la situation, les figures (2.22) et (2.23) présentent un exemple de résolution du VRP dans le cadre d'une liste de picking contenant 17 articles (donc $n = 18$ en comptant le dépôt) ce qui implique, au vu des contraintes énoncées précédemment, un minimum de $m = 3$ tournées. Sur la figure (2.22) on peut observer le placement des nœuds orientés en fonction du placement des articles correspondants sur les étagères ainsi que la liaison avec les carrefours. Attention toute fois, ici la représentation graphique des carrefours a été modifiée pour des raisons de lisibilité. La figure (2.22), quant à elle, représente la solution optimale du VRP avec 3 tournées. Chaque chemin est représenté d'une couleur différente et peut être effectué par l'agent concerné dans n'importe quel ordre. De plus, les chemins commencent et finissent tous au dépôt. Les expérimentations se déroulent de la façon suivante. Pour chaque scénario (comprendre chaque liste de picking), le VRP est résolu grâce à un solveur informatique via son modèle MILP. La solution obtenue étant optimale elle permet de connaître la plus petite distance possible que l'agent collecteur peut effectuer lors de l'opération de picking. Ensuite, l'algorithme génétique est utilisé pour obtenir une nouvelle solution. Les solutions sont comparées en termes de temps de calcul et de distance à parcourir, sur ce dernier point un ratio d'erreur est calculé. Les calculs sont effectués sur un ordinateur Intel® Core™ i5-3550 CPU @ 3.30 GHz * 4 processeurs et 4 GB de RAM. Le modèle MILP, quant à lui, a été résolu avec le solveur GUROBI™.

Les résultats de ces expérimentations sont résumés par la table (2.1). Les performances

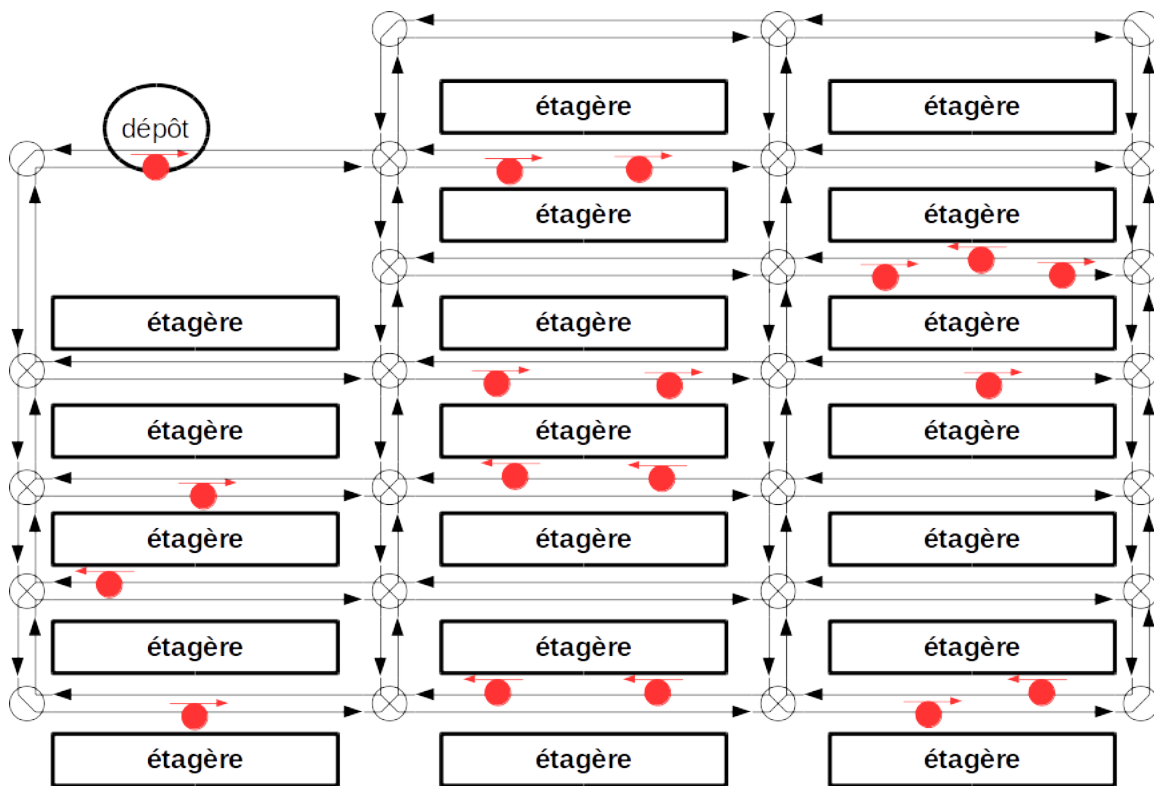


FIGURE 2.22 – Exemple de représentation graphique d'une liste de picking

de l'algorithme génétique données par cette table (autant en termes de temps que de distance) sont des moyennes obtenues sur 10 exécutions. Les résultats montrent que

TABLE 2.1 – Comparaison de performance entre le MILP et l'algorithme génétique

	Temps de calcul – algorithme génétique	Distance	Temps de calcul - Gurobi	Distance	Ratio d'erreur
15 nœuds	6 min. 20 sec.	220 m.	0 min. 0.80 sec.	220 m.	0 %
20 nœuds	7 min. 50 sec.	225 m.	0 min. 0.42 sec.	221 m.	1.8 %
25 nœuds	12 min. 25 sec.	280 m.	0 min. 2.46 sec.	275 m.	1.8%
30 nœuds	15 min.	302 m.	5 min. 6.32 sec.	296 m.	5,3%
35 nœuds	16 min. 15 sec.	342 m.	5 min. 7.63 sec.	329 m.	3.8%
40 nœuds	21 min.	362 m.	103 min. 4.52 sec.	-	-
45 nœuds	27 min. 40 sec.	397 m.	> 1 hour	-	-
50 nœuds	28 min. 55 sec.	424 m.	> 1 hour	-	-
70 nœuds	37 min. 25 sec.	453 m.	> 1 hour	-	-
90 nœuds	54 min. 45 sec.	515 m.	> 1 hour	-	-
110 nœuds	82 min. 20 sec.	562 m.	> 1 hour	-	-
130 nœuds	112 min. 40 sec.	624 m.	> 1 hour	-	-
150 nœuds	137 min. 25 sec.	703 m.	> 1 hour	-	-

le temps nécessaire pour déterminer une solution avec l'algorithme génétique évolue de

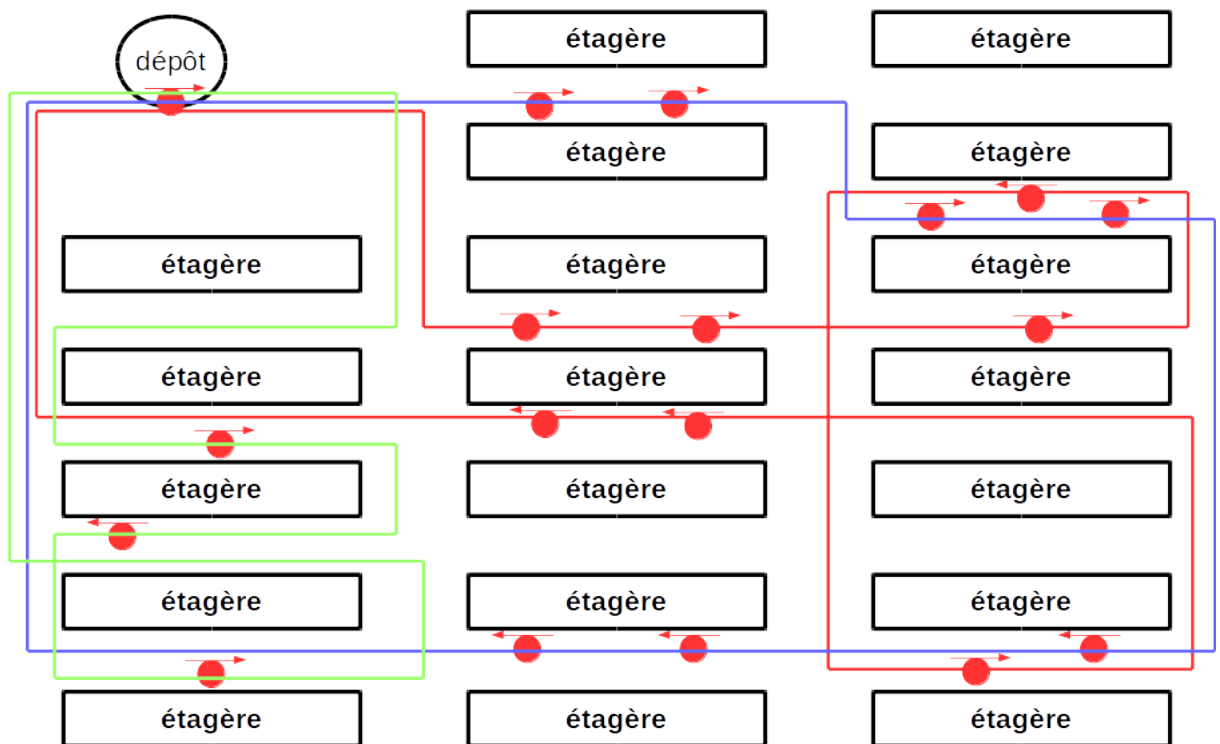


FIGURE 2.23 – Solution optimale du VRP appliqué à la figure (2.22)

façon linéaire avec le nombre de nœuds contenus dans le graphe. Par contre, le solveur exact est capable de fournir une solution optimale de façon très rapide pour de petites instances, mais à partir de 40 nœuds on constate qu'il est nécessaire d'attendre plus d'une heure pour l'obtenir. Pour l'exemple présenté ici, les performances, en termes de temps de calcul de l'algorithme génétique, ne sont effectives que pour les listes de picking contenant 40 éléments ou plus.

Pour ce qui est de la distance parcourue par l'agent collecteur, l'écart entre l'algorithme génétique et la solution optimale reste inférieur à 6%. Il n'est malheureusement pas possible de connaître cet écart pour les grandes instances puisque pour ces dernières la solution optimale n'est pas connue.

2.6/ CONCLUSION

Comme nous l'avons vu tout au long de ce chapitre, de nombreux entrepôts sont pensés comme une série d'allées formées par des étagères que ce soit dans le domaine industriel ou hospitalier. Il est plutôt logique de souvent voir apparaître ce type d'implantation, car il possède un avantage majeur. En effet, cela permet de stocker un grand nombre d'articles avec une faible surface au sol en favorisant le stockage sur plusieurs

niveaux. Malheureusement, la présence d'articles en hauteur impose l'utilisation d'un ou plusieurs véhicules qui apportent leurs lots de contraintes. Afin d'être exhaustif nous avons considéré trois contraintes majeures : l'impossibilité pour le véhicule d'effectuer des demi-tours, l'obligation de respecter une certaine orientation du véhicule lors de la collecte des articles en fonction de leurs places dans le stock et enfin le respect de la capacité totale du véhicule transporteur en terme de volume. Ces trois contraintes peuvent être considérées ensemble ou séparément pour s'adapter à différents choix d'implantation de stock.

Pour respecter les deux premières contraintes, nous avons mis en place de nouveaux éléments graphiques. Il est en effet possible de représenter un scénario de collecte d'articles par un graphe en commençant par ajouter une nouvelle modélisation des carrefours interdisant le retour en arrière et placé à chaque intersection d'allées. Ensuite, l'introduction de la notion de nœud orienté a permis de s'assurer de l'orientation du véhicule lors de la collecte. Ces nouvelles considérations graphiques ne sont pas sans conséquence sur la création même du graphe. Un graphe utilisant les carrefours et les nœuds orientés devra en effet utiliser des arcs orientés et les raccords entre ces arcs et les différents éléments devront être faits minutieusement pour représenter fidèlement la réalité et ne pas bloquer certaines voies. Pour régler ce problème, nous avons mis en place un algorithme automatisant la création graphique d'allées avec des arcs orientés.

Une fois le graphe correctement créé, nous avons défini un modèle MILP de VRP pour s'adapter à la contrainte de capacité des véhicules de transport. En effet, ce type de problèmes prend en compte la possibilité de retourner au dépôt pour décharger les articles collectés pendant l'opération de picking. Pour le résoudre, nous avons proposé un algorithme génétique utilisant des opérateurs de croisement et de mutation afin obtenir une solution impliquant de parcourir la plus petite distance possible. En comparant les performances de cet algorithme avec celui d'un solveur exact (autant en terme de qualité de la solution que de temps de calcul), on observe qu'il est un bon compromis pour les instances de problèmes concernant des listes de picking de plus d'une trentaine d'articles. En effet, il permet d'obtenir une solution correcte en moins de 20 minutes alors que la solution optimale ne peut plus être obtenue en un temps raisonnable.

Dans le cas d'un entrepôt industriel, il est alors possible d'utiliser soit l'une soit l'autre des deux méthodes en fonction de la taille de la liste de picking concernée, mais ceci implique d'avoir un système informatisé capable d'effectuer les calculs en temps réel. Pour le cas d'une PUI, la même méthode peut être employée, mais dans la mesure ou généralement les listes de dotation sont connues à l'avance (point qui sera discuté dans le chapitre suivant) il est possible de déterminer les chemins optimaux en amont de l'opération de picking. L'avantage de cette solution étant qu'elle ne nécessite pas l'utilisation d'un système informatisé pour gérer le stock, mais simplement de réorganiser l'ordre

de picking des articles sur les listes de dotations en précisant les éventuels retours au dépôts.

AGENCEMENT DU STOCK DE DISPOSITIFS MÉDICAUX

Le chapitre précédent présentait des méthodes pour optimiser les opérations de picking en améliorant le chemin parcouru par les agents collecteurs au sein du stock. Le présent chapitre, quant à lui, propose des méthodes d'optimisation en amont des opérations de picking. Le principe est de profiter de certaines des caractéristiques intrinsèques au domaine pharmaceutique (tel que l'existence de listes de dotations préétablies pour chaque service) pour optimiser le rangement des articles dans le stock et l'ordonnancement des listes de picking. En procédant ainsi, il est possible de raccourcir encore le temps nécessaire pour compléter les commandes des différents services. Pour plus de simplicité, le problème est divisé en deux sous-problèmes : d'un côté les choix du placement des références dans le stock et de l'autre l'ordonnancement des listes de picking. Ce chapitre présente donc plusieurs approches pour résoudre ces sous-problèmes. Tout d'abord, un modèle MILP inspiré par le LRP (voir ci-après) est présenté et détaillé, ce dernier permet d'identifier les différentes variables qui entrent en jeu dans ce problème. Ce modèle ne permet par contre pas de résoudre le problème de façon exacte en temps réel via un solveur notamment à cause de la trop grande complexité induite par la contrainte d'élimination des sous-tours. Pour pallier à ce problème, nous proposons une méthode de construction d'une solution valide à partir d'une solution contenant des sous-tours. Cette méthode permet alors de se soustraire de la contrainte d'élimination des sous-tours en créant une solution valide d'un des deux sous-problèmes en deux étapes.

Afin de construire une solution complète, trois approches sont proposées. Une approche basée sur le "bon sens" qui consiste à trier les références suivant leurs taux d'utilisation et de placer les plus utilisés le plus proche possible du dépôt. On peut compléter la solution obtenue grâce à un algorithme de TSP appliqué aux listes de picking couplé à la méthode d'élimination des sous-tours précédemment citée. Ensuite deux métaheuristiques sont proposées : un algorithme génétique et un branch & bound. Afin d'utiliser au mieux les

avantages des différentes approches, des méthodes mixtes les combinant sont proposés. Le principe est de déterminer une solution basée sur le "bon sens" et de l'utiliser pour initialiser l'une des deux des métaheuristiques précédemment citées.

Les différentes méthodes sont alors comparées sur différents scénarii en termes de qualité de solution. Ces expérimentations tentent à montrer que l'algorithme génétique mixte produit les meilleurs résultats.

Les méthodes mises en place dans ce chapitre ont conduit à la publications de [Beroule et al., 2017].

3.1/ SYSTÈME ÉTUDIÉ ET POSSIBILITÉS D'AMÉLIORATION

Comme dit dans l'introduction, le stock de la PUI de l'hôpital Emile Muller est de type "picker to part low level policy". Cette appellation signifie que lors de l'opération de picking, un agent doit se déplacer physiquement vers les emplacements des différents éléments à collecter un à un ("picker to part") et que les éléments en question se trouvent sur des palettes au niveau du sol ("low level"). Il est important de garder à l'esprit que les méthodes développées dans ce chapitre peuvent aisément être appliquées à d'autres politiques de gestion de stock et donc à d'autres PUI sous réserve d'effectuer de légers changements. D'un point de vue purement formel, le stock de la PUI peut être considéré comme un ensemble d'emplacements contenant des références à collecter (médicaments, seringues, gants...). Chaque service de l'hôpital fournit alors à la direction de la PUI une liste contenant l'équipement médical dont ils ont besoin pour une période donnée (une semaine, deux semaines, un mois. . .). Ces listes sont appelées des "listes de dotations" et sont l'équivalent dans l'industrie des bons de commande. Néanmoins, contrairement au secteur industriel, dans un contexte hospitalier les listes de dotations sont connues à l'avance et ne sont que peu modifiées au fil du temps. En effet, un service de soins donné aura toujours besoin peu ou prou du même matériel pour pouvoir pratiquer son activité correctement. Chaque jour ouvré, les agents gérant le stock de la PUI ont pour objectif (entre autres) de collecter les références d'un certain nombre de listes de dotations. Lors du traitement d'une liste, l'agent concerné va se déplacer vers les emplacements de chaque élément de ladite liste dans l'ordre imposé par cette dernière afin de collecter chaque référence demandée dans la quantité adéquate. Une fois terminé, l'ensemble des objets collectés forme la "dotation" du service et peut être envoyé à ce dernier pour qu'il soit approvisionné.

En considérant le contexte décrit plus haut, le but de cette section est de proposer une modélisation du problème de picking ainsi que des solutions efficaces. Le problème de picking consiste à déterminer les paramètres qui permettent de minimiser la dis-

tance moyenne qu'un agent doit parcourir pour compléter une liste de dotation (et conséquemment le temps nécessaire). Pour parvenir à ce but, il nous faut considérer deux aspects logistiques de l'opération de picking.

- La configuration du stock (c'est-à-dire les emplacements de chaque référence).
- L'ordre dans lequel l'agent va collecter les objets dans le stock.

L'originalité de ce travail réside dans le fait de considérer ces deux aspects non pas séparément, mais ensemble. En effet, dans le cadre d'un problème industriel classique, les bons de commande (équivalent des listes de dotations) ne sont généralement pas connus à l'avance, mais sont directement établis par un client lors de sa commande (voir peuvent regrouper les commandes de plusieurs clients). Dans ce cas, l'emplacement des références peut être déterminé en se basant sur des statistiques de demandes. Il est par exemple possible de physiquement grouper les références qui sont souvent commandées ensemble ou de placer les références les plus demandées le plus proche possible du point de départ de l'opération de picking. Le chemin à parcourir dans l'entrepôt peut alors être déterminé à l'aide d'un algorithme de "Problème de Voyageur de Commerce" appelé en anglais "Traveling Salesman Problem" (TSP) ou dans le cas d'agents effectuant des tournés pour collecter les objets en plusieurs fois, d'un algorithme de "Problème de Routage de Véhicule" ou en anglais "Vehicle Routing Problem" (VRP).

Le problème que nous présentons ici peut, quant à lui, être comparé à un "Problème de routage avec emplacement" ou "Location Routing Problem" (LRP). En effet, les deux problèmes partagent un certain nombre de similarités malgré des méthodes de résolution différentes.

Définition 17 : Location Routing Problem

Le LRP se définit comme suit.

Soit un ensemble d'emplacements de clients N , un ensemble d'emplacements potentiels de dépôts $R \subset N$ et une liste de livraisons à effectuer par un ou plusieurs transporteurs. Le but est de déterminer où placer les dépôts parmi les emplacements potentiels proposés afin de minimiser le temps global de livraison tout en satisfaisant les demandes des clients. La solution dépend en partie des transporteurs choisis et de leurs caractéristiques intrinsèques (vitesse, capacité de transport, durée du chargement/déchargement). La définition (18) présente une possibilité de formulation mathématique du problème.

Définition 18 : Location Routing Probem - MILP**Paramètres :**

- S : un sous-ensemble de $N - R$.
- $c_{i,j}$: le coût de transport du site i vers le site j (distance, temps, essence...).
- d_k : la demande du client k (en nombre d'articles)
- \underline{P} et \bar{P} : respectivement le nombre minimal et maximal de sites qui peuvent être choisis pour abriter un dépôt

Variables :

- $x_{i,j}$: nombre de fois que l'arc (i,j) est emprunté par un transporteur ($i \geq j$)
- y_r : variable binaire égale à 1 si et seulement si le site r est choisi pour abriter un dépôt.

La fonction objectif est donnée par :

$$\text{Minimiser } \sum_{i \in N} \sum_{j \in N} x_{i,j} c_{i,j} \quad (3.1)$$

sujet à un certain nombre de contraintes :

$$\sum_{i < k} x_{i,k} + \sum_{k < j} x_{k,j} = 2 \quad \forall k \in N - R \quad (3.2)$$

$$\sum_{i \in S} \sum_{j \in S} x_{i,j} \leq |S| - \lceil \frac{\sum_{k \in S} d_k}{D} \rceil \quad \forall S \subset N - R, |S| \geq 3 \quad (3.3)$$

$$x_{i_1, i_2} + 3 \times x_{i_2, i_3} + x_{i_3, i_4} \leq 4 \quad \forall i_1, i_4 \in R; i_2, i_3 \in N - R \quad (3.4)$$

$$x_{i_1, i_2} + x_{i_{h-1}, i_h} + 2 \times \sum_{i, j \in \{i_2, \dots, i_{h-1}\}} x_{i,j} \leq 2h - 5 \quad \forall h \geq 5; i_1, i_h \in R; i_2, \dots, i_{h-1} \in N - R \quad (3.5)$$

$$\underline{P} \leq \sum_{r \in R} y_r \leq \bar{P} \quad (3.6)$$

La fonction objectif (3.1) stipule que la solution trouvée devra minimiser le coût total de transport. Cette solution est soumise à diverses contraintes. La contrainte (3.2) spécifie que chaque site non choisi pour abriter un dépôt doit être visité une fois par chaque véhicule. La contrainte (3.3), quant à elle, s'assure que la solution trouvée ne contient pas de sous-tours. Les contraintes (3.4) et (3.5) s'assurent que chaque tourné commence et finisse au même point. Enfin, la contrainte (3.6) définit les bornes du nombre de sites choisis pour abriter un dépôt.

Par analogie avec le problème étudié dans ce manuscrit, les clients du LRP peuvent être comparés aux objets à collecter et l'emplacement de l'unique dépôt est connu à l'avance et peut être comparé au point de départ de l'opération de picking. En conséquence, le problème traité dans ce chapitre devient alors de déterminer les emplacements des clients (ici les produits) afin de minimiser le temps global nécessaire pour compléter chaque liste de dotations. La fonction objectif de ce problème est donc la même que celle du LRP, mais les contraintes sont différentes.

Maintenant que le concept de base du problème est posé, nous proposons d'en définir une modélisation mathématique.

3.2/ MODÉLISATION

Afin de parvenir à résoudre le problème décrit précédemment, la première étape consiste à établir un modèle MILP (voir définition (13)). Les cas d'études réels présentent souvent des particularités bien précises qui complexifient les calculs inutilement, c'est pourquoi pour nous assurer de nous concentrer sur l'essentiel nous choisissons de faire les simplifications suivantes :

- les collisions entre les agents de collecte ne sont pas considérées
- les agents de collecte (ou leurs véhicules) ne sont jamais surchargés
- le chemin de picking commence et termine au même point (que nous appellerons le dépôt)
- une même référence ne peut pas se trouver sur plusieurs emplacements à la fois.

Comme nous l'avons vu précédemment, le problème étudié partage un certain nombre de points communs avec le TSP ou le VRP. Il est donc légitime dans ce cas de réutiliser certaines variables de ces deux problèmes connus. L'ensemble des variables utilisées est le suivant.

- n : nombre de références (et donc d'emplacements)
- m : nombre de listes de dotation fournies par l'hôpital
- C_t : ensemble des objets contenu dans la liste t
- $d_{k,l}$: distance entre l'emplacement k et l'emplacement l
- $x_{i,k}$: variable binaire égale à 1 seulement si l'objet i se trouve à l'emplacement k
- $l_{i,j,t}$: variable binaire égale à 1 seulement si dans la liste t l'objet i est directement suivi par l'objet j
- $u_{i,t}$: l'ordre de l'objet i dans la liste t

NB : On suppose que chaque liste commence et finit par une référence "1" qui correspond à l'emplacement du dépôt.

Le but est de minimiser la distance moyenne parcourue par les agents lors de l'opération

de picking d'une liste de dotation. Sachant que minimiser la distance moyenne revient à minimiser la distance totale, on obtient la fonction objectif suivante.

$$\text{Minimize } z = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{l=1}^n \sum_{t=1}^m x_{i,k} \cdot x_{j,l} \cdot l_{i,j,t} \cdot d_{k,l} \quad (3.7)$$

Dans cette équation, la variable z représente la somme des distances parcourues par un agent qui collecterait les objets de chaque liste de picking une à une en respectant l'ordre des listes. Pour s'en convaincre, il faut décortiquer l'équation. L'expression " $x_{i,k} \cdot x_{j,l} \cdot l_{i,j,t}$ " est égale à 1 seulement si les conditions suivantes sont réunies :

- $x_{i,k} = 1$: l'objet i se trouve à l'emplacement k .
- $x_{j,l} = 1$: l'objet j se trouve à l'emplacement l .
- $l_{i,j,t} = 1$: la liste t contient l'objet i directement suivi par l'objet j .

Si toutes ces contraintes sont respectées pour un 5-uplet (i, j, k, l, t) alors l'agent va effectivement se déplacer de l'emplacement k à l'emplacement l pendant l'opération de picking et donc la distance $d_{k,l}$ doit être ajoutée à z . Une illustration de ce cas est donnée par la figure (3.1). Il est important de noter que la fonction objectif proposée ici n'est pas

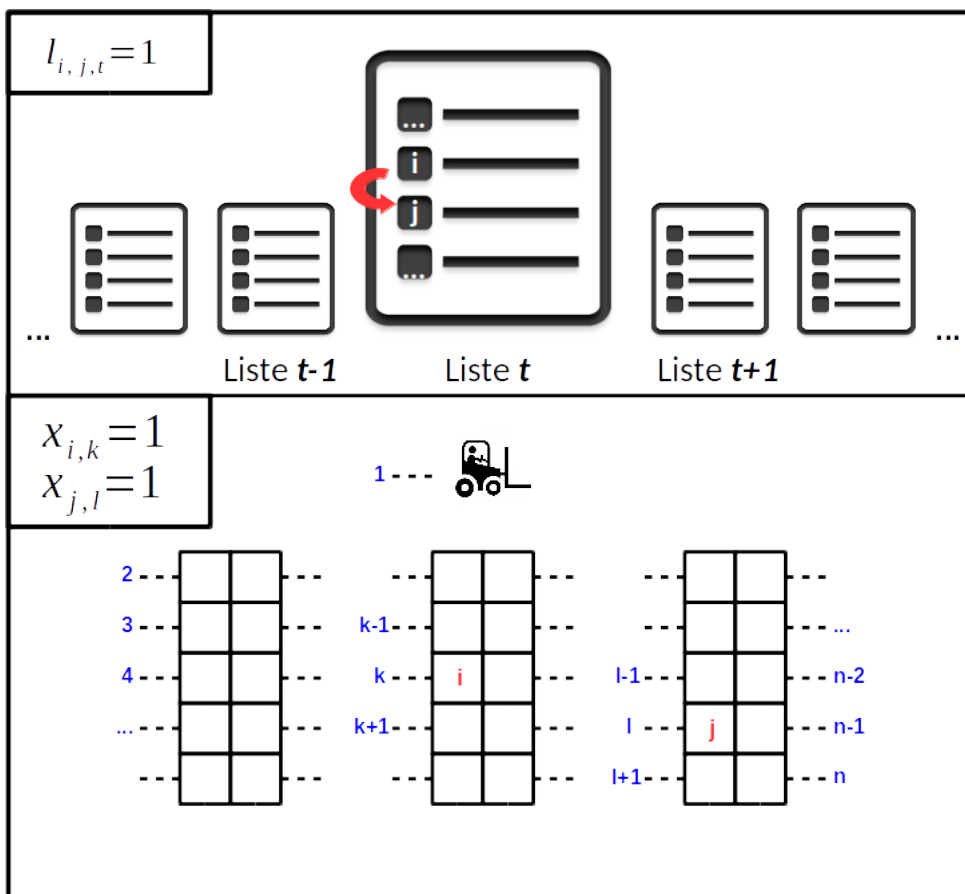


FIGURE 3.1 – Explication de la fonction objectif

linéaire puisqu'elle demande de multiplier des variables entre elles. Nous avons choisi de présenter cette version pour des raisons de facilité de compréhension. La version linéarisée du modèle peut être trouvée en annexe (B).

Une fois la fonction objectif déterminée, il faut considérer un certain nombre de contraintes.

$$\sum_{j=1}^n x_{i,j} = 1 \quad 1 \leq i \leq n \quad (3.8)$$

$$\sum_{i=1}^n x_{i,j} = 1 \quad 1 \leq j \leq n \quad (3.9)$$

$$x_{1,1} = 1 \quad (3.10)$$

$$\sum_{j \in C_t} l_{i,j,t} = 1 \quad i \in C_t \quad 1 \leq t \leq m \quad (3.11)$$

$$\sum_{i \in C_t} l_{i,j,t} = 1 \quad j \in C_t \quad 1 \leq t \leq m \quad (3.12)$$

$$l_{i,i,t} = 1 \quad 1 \leq t \leq m \quad (3.13)$$

$$u_{1,t} = 1 \quad 1 \leq t \leq m \quad (3.14)$$

$$u_{i,t} - u_{j,t} + n \times l_{i,j,t} \leq n - 1 \quad \begin{array}{l} 1 \leq i \leq n \\ 2 \leq j \leq n \\ 1 \leq t \leq m \\ i \neq j \end{array} \quad (3.15)$$

Les équations (3.8) et (3.9) contraignent respectivement chaque objet à être placé sur un emplacement unique et chaque emplacement à être occupé par un unique objet. Comme nous l'avons vu précédemment, l'emplacement 1 correspond au dépôt, c'est à dire au point de départ et d'arrivée de chaque opération de picking. C'est pourquoi l'équation (3.10) empêche le dépôt d'être placé ailleurs qu'à l'emplacement 1. Les équations (3.11) et (3.12) représentent le fait que dans chaque liste, chaque référence est suivie et précédée d'une unique autre référence (ou du dépôt). L'équation (3.13) sert à empêcher une liste de contenir une même référence plusieurs fois consécutivement (car cela ne modifie pas la route empruntée). Enfin, les équations (3.14) et (3.15) empêchent les listes de contenir des sous-tours. En effet, le modèle proposé étant basé sur celui du TSP il est possible qu'une liste contienne des sous-tours comme montré par la figure (3.2).

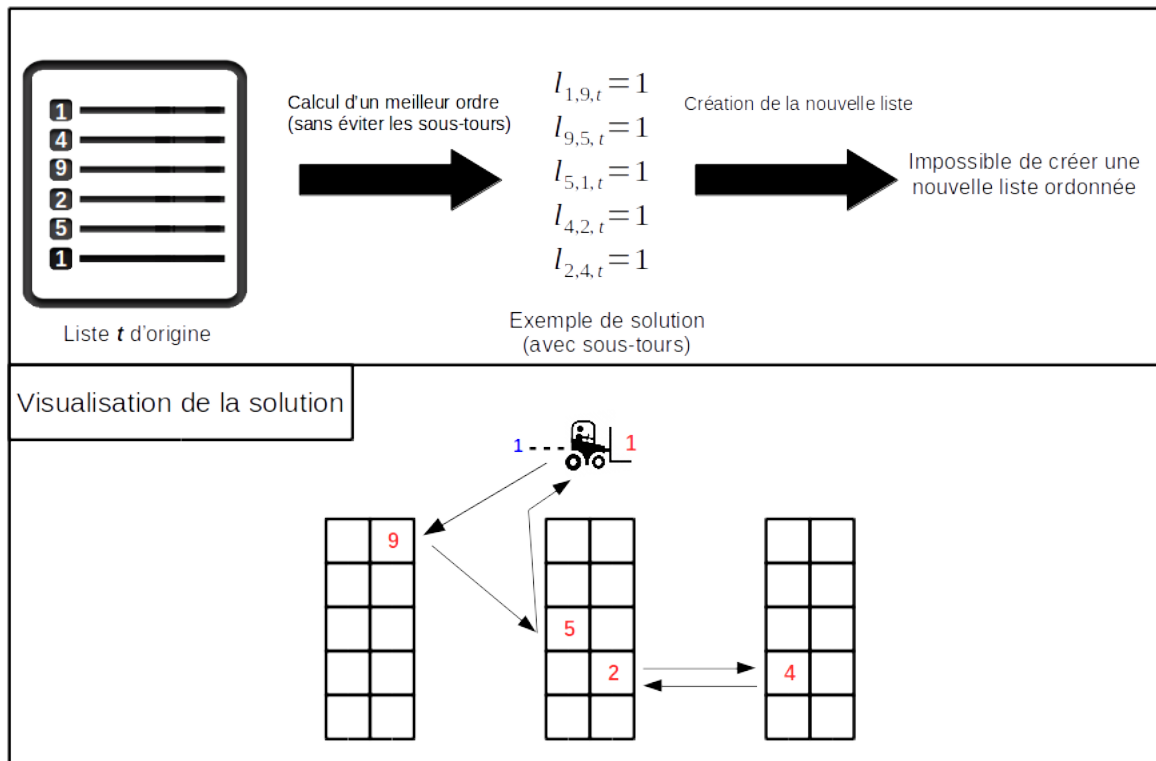


FIGURE 3.2 – Exemple de solution avec sous-tours

3.2.1/ ANALYSE DU PROBLÈME

La solution optimale du problème présenté ici peut être déterminée par un solveur exact. Malheureusement, le temps de calcul nécessaire pour obtenir une telle solution peut facilement augmenter de façon importante pour des instances de plus en plus grandes de problèmes (comprendre avec de plus en plus d'emplacements ou de listes). Comme nous l'avons dit précédemment, ce problème est basé sur le TSP. En effet, si l'on fixe les emplacements de chaque référence dans le stock alors le problème se résume à trouver comment atteindre le plus rapidement possible les références de chaque liste de picking, il s'agit donc bien d'un TSP. La conséquence de cette constatation est que le problème étudié est NP-complet (voir définition (19)) ce qui implique un temps de calcul qui peut être important.

Définition 19 : Problème NP-complet

Un problème est NP-complet s'il vérifie les deux propriétés suivantes :

- il est possible de vérifier une solution efficacement en un temps polynomial (problème NP)
- tous les problèmes de la classe NP se ramènent à celui-ci via une réduction polynomiale.

Ces considérations prises en compte, il peut s'avérer intéressant de définir la dimension du problème. Dans le cadre de cette étude, nous appellerons "dimension du problème" le nombre de solutions possibles de ce dernier. Dans ce contexte, une instance du problème est composée d'un ensemble de n références ainsi que m listes de picking. Chaque liste t ($1 \leq t \leq m$) contient $|C_t|$ éléments. Comme dit précédemment, une solution consiste en un appariement des références et des emplacements ainsi qu'un ensemble de listes de picking ordonnées. Tout d'abord, trouver un appariement implique de choisir parmi $n!$ possibilités. Ensuite, il y a $|C_t|!$ ordre possible pour chaque liste t , soit un total de $\sum_{t=1}^m |C_t|!$ m -uplet de listes possibles. Enfin, le nombre total de possibilités (noté δ) correspond à la totalité des couples "appariement emplacement-référence" et "ensemble de listes" qu'il est possible de former. La proposition (1) donne la valeur de δ pour n'importe qu'elle instance.

Proposition 1. *Le nombre de solutions possibles du problème (dimension) considéré ici est défini comme suit.*

$$\delta = n! \times \sum_{t=1}^m |C_t|!$$

Cette valeur peut facilement devenir très grande et ce même pour de petites instances. Prenons un exemple en considérant une petite PUI et donc un petit entrepôt avec $n = 50$ références et $m = 10$ listes de dotations à traiter composées chacune de $|C_t| = 10$ ($\forall t \in \llbracket 1, m \rrbracket$) éléments. Dans ce cas (qui est clairement plus petit qu'un cas réel), on obtient un nombre total de solutions possibles de $\delta = 1.1 \times 10^{72}$. De plus d'après la proposition (1) le nombre de solutions augmente en suivant un comportement factoriel avec le nombre de références n . Des résultats expérimentaux préliminaires ont montré que cette évolution empêchait d'utiliser des solveurs exacts pour résoudre le problème sans que le temps de calcul soit significativement trop important. Cette constatation nous oblige à envisager des méthodes de résolution alternatives capables de fournir un résultat aussi proche que possible de l'optimal mais, en un temps respectable.

3.3/ MÉTHODES DE RÉOLUTION

Le but de cette section est de fournir des méthodes approximatives pour résoudre le problème étudié et d'évaluer ces méthodes en fonction de la qualité des solutions obtenues. Nous étudierons aussi la possibilité de combiner certaines méthodes pour en tirer de meilleurs résultats.

3.3.1/ APPROCHE HEURISTIQUE

La première étape de cette recherche de méthodes alternatives consiste à proposer un moyen simple et intuitif de trouver une bonne solution. Nous proposons pour ce faire de nous baser sur les principes de l'analyse ABC ([Flores et al., 1986]) afin d'identifier le taux d'utilisation de chaque référence de la PUI et de réagencer le stock en minimisant la distance de picking.

L'analyse ABC est une méthode qui consiste à trier les produits à stocker en trois catégories A, B et C en fonction de leur valeur (A étant les produits avec les plus fortes valeurs et C les plus faibles). Pour définir une méthode intuitive, nous nous basons donc sur le concept général de tri des références par valeur, comme la PUI n'a pas pour but de vendre ses produits le concept de valeur est ici remplacé par celui de taux d'utilisation des produits.

La méthode consiste donc à établir une liste des références disponibles dans le stock en les triant par leur nombre d'apparitions dans les listes de dotations (de la plus fréquente à la moins demandée). Ensuite, le stock de la pharmacie est organisé de façon à ce que les références les plus récurrentes soient assignées aux emplacements les plus proches du dépôt et inversement. Dans la suite de ce manuscrit cette méthode sera notée MFIF (pour **M**ost **F**requent **I**tem **F**irst). Puisque le principe de la méthode MFIF est déterminé, il est nécessaire d'en décrire l'algorithme afin de pouvoir la mettre en pratique.

3.3.1.1/ ALGORITHME

Comme nous l'avons vu précédemment, le principe de la méthode MFIF implique de trier les références selon leurs taux utilisations (les demandes des services de soins) puis, de trier les emplacements selon la distance qui les sépare du dépôt. On obtient ainsi un vecteur de références et un vecteur d'emplacements ordonnés. Il faut alors faire un appariement de ces deux vecteurs (c'est à dire coupler le premier emplacement avec la première référence, le deuxième emplacement avec la deuxième référence et ainsi de suite).

La procédure complète de la méthode MFIF est décrite par l'algorithme (3). Dans cet

algorithme on cherche à construire le vecteur P qui représente les emplacements de chaque référence. Il se lit de la façon suivante : si $P(i) = j$ alors la référence i se trouve à l'emplacement j . Pour bien comprendre l'algorithme il est important de préciser que le but de la fonction $trie(V_1, V_2)$ est de trier le vecteur V_1 dans l'ordre croissant de l'évaluation de ses composants donnés par V_2 . L'algorithme se découpe en trois étapes.

Inputs :

C_t : vecteur contenant les objets de la liste t

d_0 : vecteur de taille n contenant les distances entre les différents emplacements et le dépôt

V : vecteur de taille n contenant l'ensemble des objets contenus dans le stock

V' : vecteur de taille n contenant l'ensemble des emplacements de stock

$occur \leftarrow$ nouveau vecteur nul de taille n

$P \leftarrow$ nouveau vecteur nul de taille n

pour $t \leftarrow 1$ à m **faire**

pour $i \leftarrow 1$ à $|C_t|$ **faire**

$occur[C_t[i]] \leftarrow occur[C_t[i]] + 1$

fin

fin

$trie(V, occur)$

$trie(V', d_0)$

pour $i \leftarrow 1$ à n **faire**

$P[V[i]] \leftarrow V'[i]$

fin

$solution \leftarrow creerSolution(P, C)$

$renvoyer(solution)$

Algorithme 3 : MFIF

Étape 1 : chaque liste de picking est lue et le nombre d'occurrences de chaque référence est compté (le résultat de ce comptage est représenté par le vecteur $occur$). Les références sont alors triées (de la plus fréquente à la moins fréquente) et le résultat est représenté par le vecteur V ($V(1)$ est la référence la plus fréquente, puis $V(2)$, etc.).

Étape 2 : les emplacements potentiels des références dans l'entrepôt sont ensuite triés en fonction de leur distance par rapport au dépôt (du plus proche au plus éloigné). Le résultat est représenté par le vecteur V' ($V'(1)$ est l'emplacement le plus proche du dépôt, puis $V'(2)$, etc.)

Étape 3 : enfin, chaque référence est associée à un emplacement s'ils ont la même place dans les vecteurs V et V' . Autrement dit : la référence $V(i)$ est as-

signée à l'emplacement $V'(i)$ ($\forall i \in \llbracket 1, n \rrbracket$).

3.3.1.2/ AMÉLIORATION DU MFIF AVEC UN TSP

Il est important de noter que l'algorithme MFIF modifie uniquement les emplacements des références, mais pas l'ordre des listes de picking. Il est donc possible de l'améliorer en utilisant un algorithme de TSP pour modifier l'ordre de ces dernières (et donc les routes des agents de collecte) une fois l'algorithme MFIF exécuté. Cette nouvelle méthode est notée MFIF+TSP. Le problème est que la méthode de résolution exacte du TSP peut facilement demander un temps de calcul très important pour de grandes instances de problèmes. Or, le but de cette section reste de proposer une méthode de résolution rapide. Pour pallier à ce problème, nous utilisons le modèle MILP classique du TSP (voir définition (10) en modifiant la contrainte d'élimination des sous-tours (équation (3.15)) pour utiliser la méthode dite "des plans sécants" (voir [Miliotis, 1978]). Le principe de cette méthode est de résoudre le TSP avec un solveur exact sans considérer la formation éventuelle de sous-tours. Ensuite, tant que la solution obtenue n'est pas faisable (car elle contient des sous-tours), la résolution est relancée en ajoutant une nouvelle contrainte pour interdire cette solution particulière (on parle alors de coupe). La nouvelle contrainte, ainsi mise en évidence, s'ajoute aux autres coupes et empêche, lors d'une prochaine itération, d'obtenir à nouveau cette solution. On procède ainsi jusqu'à obtenir une solution faisable.

Ce processus peut être plus ou moins long, c'est pourquoi nous définissons une limite de temps. Si le calcul d'une solution atteint cette limite avant d'avoir déterminé une solution faisable, alors la meilleure solution trouvée jusque-là est considérée comme résultat de l'algorithme (donc avec des sous-tours). Il faut alors mettre en place un moyen d'obtenir une solution faisable à partir de cette solution contenant des sous-tours.

Pour ce faire, nous proposons une méthode dont le principe de base est d'identifier, pour chaque liste de picking, les deux références les plus proches qui n'appartiennent pas au même sous-tour. Ensuite, ces deux références sont reliées par un arc pour fusionner les deux sous-tours en un seul (bien sûr, d'autres arcs doivent être modifiés pour parvenir au résultat final). Une fois les deux sous-tours fusionnés en un, on réitère le processus jusqu'à ce que la solution courante ne contienne plus de sous-tours.

Le processus complet est divisé en 3 étapes illustrées par la figure (3.3) :

Etape 1 : on détermine le couple de nœuds les plus proches (n_1, n_2) tel que n_1

appartient au sous-tour identifié et pas n_2 . Ce couple va permettre de créer une jonction entre les sous-tours.

Étape 2 : on identifie alors les nœuds voisins de n_1 et n_2 dans leurs sous-tours respectifs. Ils sont nommés $n_{1voisin}$ et $n_{2voisin}$. $n_{1voisin}$ est défini comme étant le nœud précédant n_1 et $n_{2voisin}$ comme le nœud succédant n_2 .

NB : Dans l'algorithme (4) (voir ci-après), la valeur de $n_{2voisin}$ est déterminé lors de l'étape 1 ce qui est certes moins lisible, mais permet de gagner du temps de calcul.

Étape 3 : enfin, les arcs $(n_{1voisin}, n_1)$ et $(n_2, n_{2voisin})$ sont supprimés de l'ensemble des arcs valides et remplacés par (n_1, n_2) et $(n_{1voisin}, n_{2voisin})$. Les sous-tours contenant initialement n_1 et n_2 sont alors fusionnés en un seul.

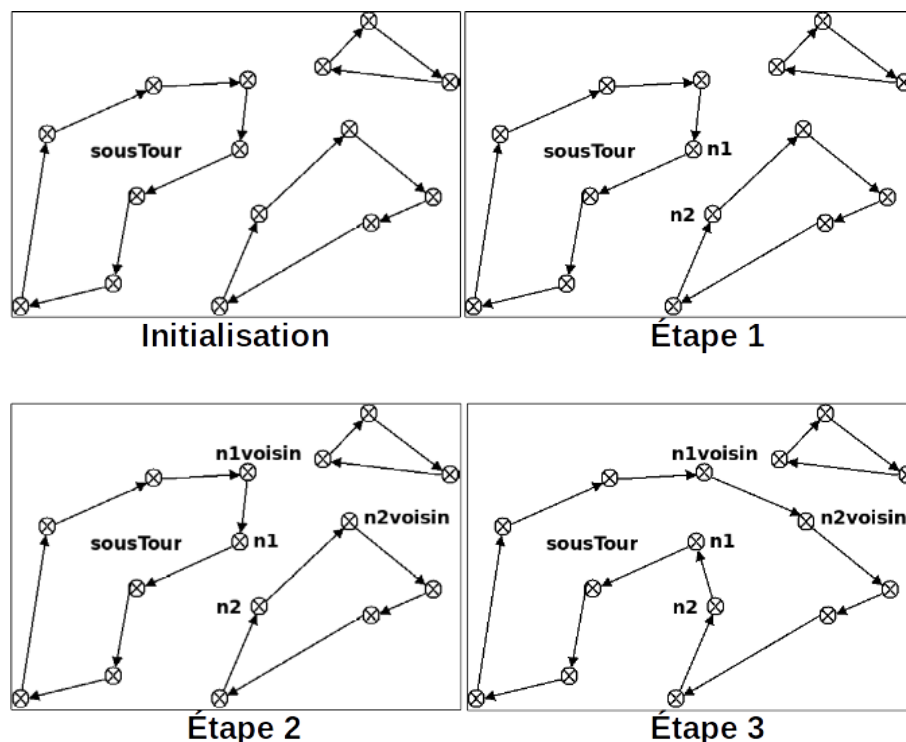


FIGURE 3.3 – Les étapes d'élimination des sous-tours

Le détail complet de la méthode est résumé par l'algorithme (4)

NB : Pour lancer l'algorithme, il faut au préalable identifier un sous-tour (dont les arcs constitueront l'ensemble $arcsSousTour$).

En conclusion, la méthode MFIF permet de déterminer une solution en un temps de calcul raisonnable. Cette solution peut être améliorée en utilisant un solveur exact pour

Entrées :

arcsSousTour : ensemble des arcs contenus dans le sous-tour courant

arcsAutres : ensemble des arcs ne faisant pas partie de *arcsSousTour*

allEd : ensemble de tout les arcs

d : matrice des distances

```

n1 ← ∅
n2 ← ∅
n1voisin ← ∅
n2voisin ← ∅
meilleur ← d[1][2]

; // Étape 1
pour i ← 1 à |arcsSousTour| faire
  pour j ← 1 à |arcsAutres| faire
    si d[arcsSousTour[i].noeudDebut][arcsAutres[j].noeudDebut] < meilleur alors
      n1 ← arcsSousTour[i].noeudDebut
      n2 ← arcsAutres[j].noeudDebut
      n2voisin ← arcsAutres[j].noeudFin
      meilleur ← d[n1][n2]
    fin
  fin
fin
; // Étape 2
pour i ← 1 à |arcsSousTour| faire
  si arcsSousTour[i].noeudFin = n1 alors
    n1voisin ← arcsSousTour[i].noeudDebut
  fin
fin
; // Étape 3
pour i ← 1 à |arcsSousTour| faire
  si arcsSousTour[i].noeudFin = n1 alors
    supprimer arcsSousTour[i] de arcsGlobal
  fin
fin
pour j ← 1 à |arcsAutres| faire
  si arcsAutres[j].noeudDebut = n2 alors
    supprimer arcsAutres[j] de arcsGlobal
  fin
fin
ajouter nouvelArc(n2, n1) a arcsGlobal
ajouter nouvelArc(n1voisin, n2voisin) a arcsGlobal

```

Algorithme 4 : methode d'élimination des sous-tours

résoudre le TSP sur le parcours des agents dans le stock. Mais, dans ce cas, une limite de temps doit être imposée pour assurer l'aboutissement du calcul. Si la solution trouvée est incomplète, il faut alors utiliser la méthode d'élimination des sous-tours autant que

nécessaire pour obtenir une solution faisable.

Afin d'identifier la meilleure façon de résoudre le problème courant, il semble intéressant de comparer les performances de l'heuristique dédiée MFIF avec celles de méthodes plus classiques.

3.3.2/ MÉTHODE BASÉE SUR LE BRANCH AND BOUND

Le branch and bound (en français "séparation et évaluation"), souvent abrégé en B&B est une méthode générique de résolution de problèmes en recherche opérationnelle. Elle fut mise au point en 1960 par A. H. Land and A. G. Doig [Land et al., 1960]. Le principe global est de construire un arbre de solutions et de l'explorer de façon intelligente pour éviter de considérer des branches peu prometteuses.

Comme nous l'avons vu plus tôt, le problème étudié ici consiste en réalité à résoudre deux sous-problèmes, à savoir, l'ordre des listes de dotations et l'emplacement de chaque référence dans le stock. Des études préliminaires ont montré que le nombre de solutions potentielles au problème (voir section (3.2.1)) impliquerait de créer un arbre de solutions de taille importante (et donc long à parcourir). Des tests empiriques ont confirmé que l'utilisation classique d'un algorithme de branch and bound implique un important temps de calcul même pour de petites instances. C'est riche de ces observations que nous proposons une nouvelle méthode de résolution basée sur le branch and bound et dont le principe est le suivant :

- tout d'abord, un algorithme classique de branch and bound (que nous verrons en détail par la suite) est utilisé pour déterminer les meilleurs emplacements possibles de chaque référence en prenant en compte les listes de picking avec leurs ordres d'origines. Ensuite, on utilise un autre algorithme de branch and bound qui, cette fois, a pour but de déterminer le meilleur ordre possible de chaque liste de picking en prenant en compte les emplacements des références déterminées précédemment. Une fois la solution obtenue elle est prise comme référence pour recommencer le processus, et ce jusqu'à ce que la solution ne soit plus modifiée ou qu'une limite de temps soit atteinte. Ce fonctionnement est résumé par la figure (3.4).

3.3.2.1/ ALGORITHME

L'algorithme (5) décrit la méthode pour déterminer la meilleure disposition possible des références (c'est-à-dire le meilleur appariement références – emplacements) en partant d'un ordre donné pour chacune des listes de picking. Cette disposition est représentée par le vecteur P dans l'algorithme (voir section (3.3.1.1) pour la définition précise du vec-

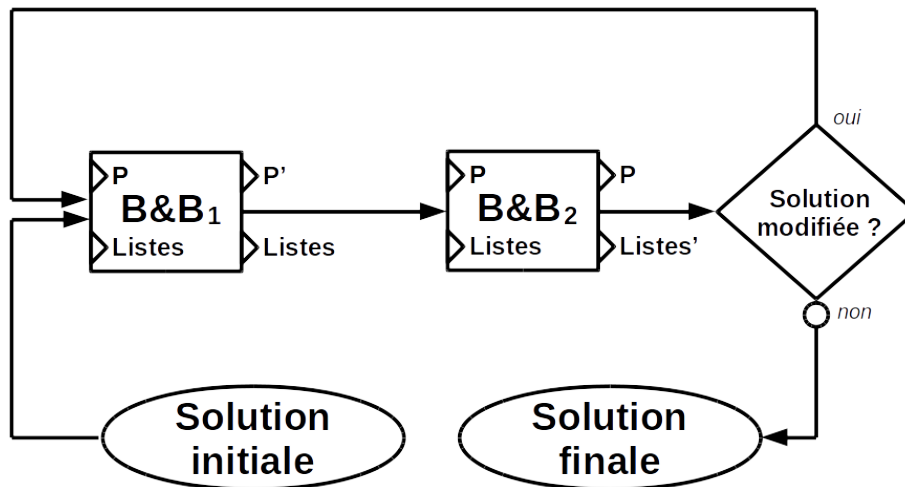


FIGURE 3.4 – Les étapes de la méthode basée sur le B&B

teur P).

Il est important de noter que l'algorithme du second branch and bound décrit précédemment (celui qui détermine l'ordre des listes) n'est pas détaillé ici pour des raisons de lisibilité et de simplicité. En effet, son fonctionnement est analogue au premier.

Pour décrire au mieux un B&B, il faut avant tout définir deux aspects :

- la méthode de construction de l'arbre de solutions (la "séparation", le "branch")
- la méthode d'évaluation des solutions incomplètes aussi appelée "borne inférieure" ("l'évaluation", le "bound")

Construire l'arbre des solutions revient à construire le vecteur P . Pour ce faire, le vecteur est tout d'abord initialisé avec uniquement l'emplacement du dépôt ($P = [1]$) ce qui correspond au nœud racine de l'arbre des solutions. Ensuite, une composante est ajoutée à P pour déterminer l'emplacement de la première référence. Le nœud racine possède donc autant de branches que le nombre de possibilités de placement de cette première référence (à savoir $n - 1$). Un exemple de construction d'arbre de solution est présenté par la figure (3.5) sur laquelle, pour des raisons de lisibilité, une seule branche est totalement développée. Les nœuds fils sont alors explorés selon un ordre défini par l'utilisateur et représenté dans l'algorithme (5) par le vecteur "*ordreExploration*". L'arbre est ensuite exploré en profondeur et les branches sont systématiquement créées en ajoutant de nouvelles composantes à P .

Définir un ordre d'exploration des nœuds permet d'explorer des groupes de solutions plus prometteuses en priorité en espérant trouver de bonnes solutions pour faire, par la suite, des coupes plus importantes lors de l'exploration de l'arbre. Pour définir l'ordre d'exploration, nous procédons de façon analogue à la méthode MFIF (voir section (5)), c'est-à-dire que les références sont prétriées en fonction de leur récurrence dans les

Entrées :

P : vecteur de localisation des produits (complet ou non)
 $listeVec$: vecteur contenant chaque liste de dotation
 $meilleureEval$: évaluation de la meilleure solution rencontrée
 $ordreExploration$: ordre d'exploration de l'arbre de solution

determinerMeilleurSolution si $borneInf(P, listeVec) \leq meilleureEval$ alors

```

si  $|P| = n$  alors
   $P_{meilleur} \leftarrow P$ 
  renvoyer( $borneInf(P, listeVec)$ )
sinon
  pour  $i \leftarrow 2$  à  $n$  faire
    si  $ordreExploration(i) \notin P$  alors
      ajouter  $ordreExploration(i)$  à  $P$ 
      si  $determinerMeilleurSolution(P, listeVec, meilleureEval) \leq best$  alors
         $meilleureEval \leftarrow determinerMeilleurSolution(P, listeVec, meilleurEval)$ 
      fin
      supprimer  $ordreExploration(i)$  de  $P$ 
    fin
  fin
  renvoyer( $meilleurEval$ )
fin
sinon
  renvoyer( $meilleurEval$ )
fin

```

Algorithme 5 : algorithme B&B d'allocation des emplacements

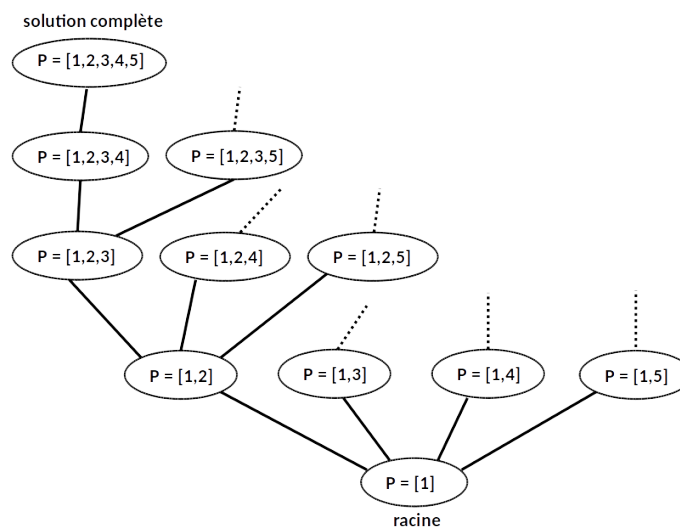


FIGURE 3.5 – Représentation partielle d'un arbre de solution (ordre standard)

listes de picking. Ensuite le vecteur d'ordre d'exploration est construit de façon à ce que la première solution construite corresponde à celle donnée par le MFIF, pour qu'ensuite des solutions proches soit explorées.

Un exemple de construction d'arbre avec prise en compte d'un ordre prédéfini est donné par la figure (3.6)

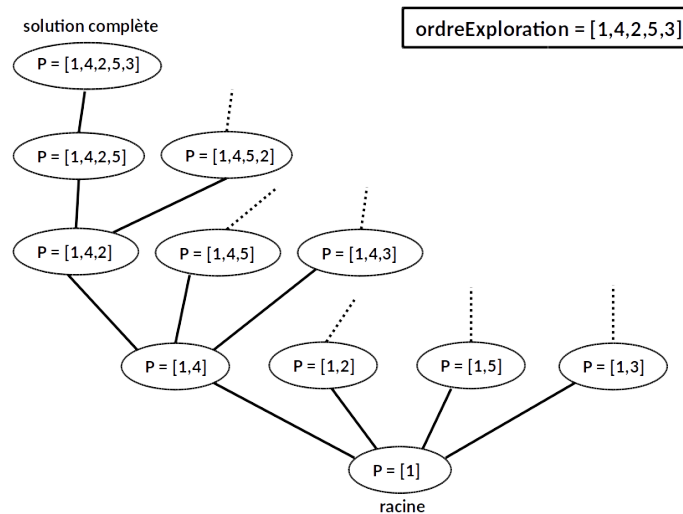


FIGURE 3.6 – Représentation partielle d'un arbre de solution (ordre prédéfinis)

Une fois l'exploration de l'arbre des solutions définit, il est nécessaire de se pencher sur la méthode d'évaluation des nœuds via la borne inférieure.

3.3.2.2/ BORNE INFÉRIEURE

Définir correctement la borne inférieure est nécessaire pour pouvoir réaliser un algorithme de B&B. La borne inférieure est une fonction qui évalue la distance de picking minimum que l'on peut espérer d'une solution incomplète. En d'autres termes, elle permet de connaître l'évaluation minimum de toutes les solutions complètes construites à partir de cette solution. Comme nous l'avons vu dans l'algorithme (5), la borne inférieure permet d'éliminer des branches de l'arbre des solutions en ne les explorant pas. Dans les faits, si la borne inférieure d'une solution incomplète est supérieure à l'évaluation de la meilleure solution trouvée jusqu'à présent alors cette solution et toutes ses branches sont ignorées (on dit qu'on fait une coupe) puisque leur exploration ne pourra en aucun cas donner une meilleure solution.

Le principe de la borne inférieure proposé dans ce chapitre est de sommer les plus petite distances possible que pourrais parcourir un agent lors de la collecte des différentes listes de picking. Concrètement, on considère chaque paire de références successives de chaque liste et on détermine une distance théorique minimale entre les deux emplacements correspondants, on distingue alors plusieurs cas de figure. Si les deux emplacements sont connus la distance entre les deux est elle aussi connue. Si l'un des deux emplacements est connu et pas l'autre alors la distance minimale théorique correspond

à la distance entre l'emplacement connu et son plus proche voisin (qui n'est pas déjà assigné à une référence). Enfin, si aucun des deux emplacements n'est connu, alors la distance minimale correspond à la plus petite distance possible entre deux emplacements non-assignés.

$$\begin{aligned}
borneInf(P, L) = & \sum_{t=1}^m \left[\sum_{i=1}^{|C_t|-1} [d[P(L_{i,t}), P(L_{i+1,t})].\omega_{L_{i,t}}(P).\omega_{L_{i+1,t}}(P)] \right. \\
& + \left(\min_{\substack{b \in [2,n] \\ b \notin P}} d[P(L_{i,t}), b]).\omega_{L_{i,t}}(P).(1 - \omega_{L_{i+1,t}}(P)) \right. \\
& + \left(\min_{\substack{a \in [2,n] \\ a \notin P}} d[a, P(L_{i+1,t})]).(1 - \omega_{L_{i,t}}(P)).\omega_{L_{i+1,t}}(P) \right. \\
& \left. \left. + \left(\min_{\substack{a,b \in [2,n] \\ a \neq b \\ b \notin P \\ a \notin P}} d[a, b]).(1 - \omega_{L_{i,t}}(P)).(1 - \omega_{L_{i+1,t}}(P)) \right] \right] \quad (3.16)
\end{aligned}$$

$$\omega_i(P) = \begin{cases} 1 & \text{si } i \leq \text{tailleDe}(P) \\ 0 & \text{sinon} \end{cases} \quad (3.17)$$

Pour pouvoir calculer la borne inférieure, on introduit deux paramètres. Tout d'abord, on définit $L_{i,t}$ comme étant l'objet numéro i de la liste t . Ensuite, on introduit le paramètre $\omega(P)$ défini par l'équation (3.17). $\omega(P)$ est un vecteur binaire de dimension n tel que $\omega_i(P) = 1$ si et seulement si l'emplacement de la référence i est déterminé par P . Ce paramètre permet de distinguer les différents cas qui peuvent se présenter lors du parcours de l'arbre de solutions.

La borne inférieure en elle-même est définie par l'équation (3.16). Elle est en effet la somme de quatre termes, correspondant chacun à un cas précis.

- $\omega_{L_{i,t}}(P).\omega_{L_{i+1,t}}(P) = 1$ si et seulement si l'emplacement de la référence de $L_{i,t}$ et de $L_{i+1,t}$ sont définis par P . Dans ce cas la distance entre ces deux emplacements ($d[P(L_{i,t}), P(L_{i+1,t})]$) est ajoutée à la valeur de la borne inférieure.
- $\omega_{L_{i,t}}(P).(1 - \omega_{L_{i+1,t}}(P)) = 1$ si est seulement si l'emplacement de la référence de $L_{i,t}$ est défini par P mais pas celui de $L_{i+1,t}$. Dans ce cas on ne peut pas connaître la distance entre les deux emplacements puisque l'un d'eux n'est pas connu. On calcul donc la distance entre l'emplacement de $L_{i,t}$ et son plus proche voisin non défini par P que l'on ajoute à la valeur de la borne inférieure.
- Le cas ou $(1 - \omega_{L_{i,t}}(P)).\omega_{L_{i+1,t}}(P) = 1$ est analogue au précédent. Dans ce cas, on ajoute la distance entre l'emplacement de la référence de $L_{i+1,t}$ et son plus proche voisin non défini par P à la valeur de la borne inférieure.
- $(1 - \omega_{L_{i,t}}(P)).(1 - \omega_{L_{i+1,t}}(P)) = 1$ si et seulement si ni l'emplacement de la référence

TABLE 3.1 – Temps de calcul du B&B avec la borne inférieure standard et améliorée

nombre de noeuds	borne inf. standard	borne inf. amélioré	amélioration
5	1.7 s	1.3 s	31%
10	4.3 s	3.1 s	28%
20	203 s	115 s	43%
30	1030 s	537 s	43%
40	53.3 min	27.3 min	49%
50	103 min	55.3 min	46%
100	8.63 h	4.52 h	47%

de $L_{i,t}$ ni celui de $L_{i+1,t}$ n'est défini par P . Dans ce cas, ne sachant rien sur ces emplacements on ajoute à la valeur de la borne inférieure la plus petite distance possible entre deux emplacements encore non défini par P .

On obtient donc la borne par sommation de ces quatre termes dont un seul est systématiquement non-nul pour un élément d'une liste de picking donnée. Ce qui correspond à la somme des plus petites distances possibles que pourrait parcourir l'agent effectuant le picking pour chaque liste.

Il est intéressant de comparer cette borne inférieure à sa version plus classique. En effet, la borne inférieure classique consisterait ici à sommer uniquement les distances entre les références dont les emplacements sont défini par P . Après quelques expérimentations sur des instances de tailles variées (résumé dans la table 3.1), il apparaît que plus le nombre de noeuds est élevé, plus l'amélioration induite par l'utilisation de la borne inférieure proposée s'approche des 50% par rapport à sa version classique sa version classique (sommation uniquement des distances connues).

Afin de bien comprendre la définition de la borne inférieure proposée ici, nous nous proposons de traiter un exemple simplifié représenté par la figure (3.7). Dans cet exemple on considère une unique liste de picking qui contient les références 2, 5, 6 et 3 (dans cet ordre). Le stock, quant à lui, est constitué en tout de 9 emplacements (donc 8 références et le dépôt). Enfin, le vecteur P déterminant l'emplacement des références n'est que partiellement défini, seul l'emplacement des références 1 à 4 sont connues. Comme vu précédemment, la borne inférieure se calcule en plusieurs étapes.

Tout d'abord, on considère la distance entre le dépôt (le nœud 1) et la première référence de la liste, à savoir la référence 2 qui se trouve à l'emplacement 3 (car $P(2) = 3$). On note donc la première distance à prendre en compte $d_1 = d(1, 3)$.

Ensuite, on cherche à calculer la distance entre l'emplacement du premier élément de la liste ($P(2)$) et le second ($P(5)$). Malheureusement, $P(5)$ n'est pas défini, il va donc falloir considérer la plus petite distance que l'agent pourrait parcourir en partant de l'empla-

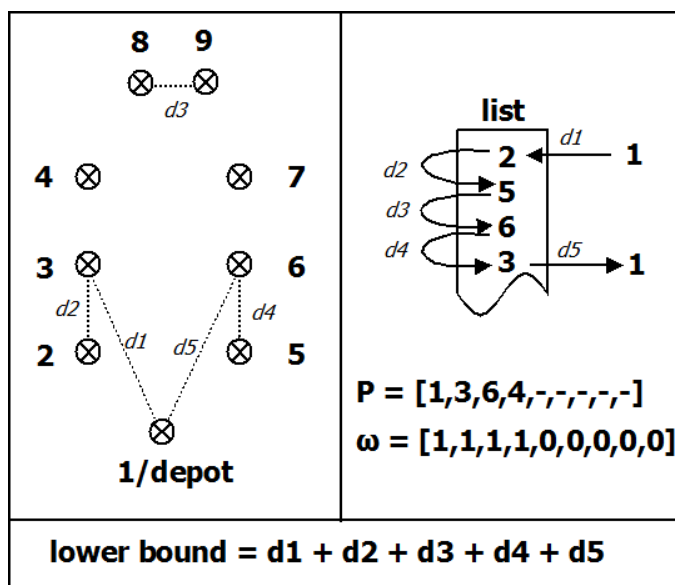


FIGURE 3.7 – Exemple de calcul de la borne inférieure

ement 3 c'est-à-dire la distance entre cet emplacement et son plus proche voisin non défini par P . Dans ce cas, il s'agit de l'emplacement 2. On note alors la seconde distance $d_2 = d(3, 2)$.

Le problème pour calculer la prochaine distance, c'est que ni l'emplacement de la référence 5 ni celui de la référence 6 ne sont connus. Dans ce cas, on considère que la plus petite distance que l'agent puisse faire pour rejoindre ces deux emplacements correspond à la distance entre les deux emplacements les plus proches qui ne sont pas déjà définis par P , à savoir le 8 et le 9. On note donc $d_3 = d(8, 9)$.

Les distances suivantes se calculent de façon analogue et on obtient donc $d_4 = d(5, 6)$ et $d_5 = d(6, 1)$.

La borne inférieure se calcule alors en faisant la somme de toutes les distances précédemment calculées ($d_1 + d_2 + d_3 + d_4 + d_5$).

La méthode de résolution du problème principal basé sur le concept du B&B est complètement défini. Afin de pouvoir proposer plusieurs approches et de décider de la meilleure, nous proposons dans la suite de ce manuscrit de tester l'utilisation d'une autre méthode, les algorithmes génétiques.

3.3.3/ ALGORITHME GÉNÉTIQUE

Comme nous l'avons vu dans le chapitre précédent, les algorithmes génétiques donnent de bons résultats lorsqu'il s'agit de résoudre des problèmes dérivés du TSP. C'est pourquoi ce type d'algorithme semble à nouveau tout indiqué dans ce cas.

3.3.3.1/ REPRÉSENTATION DES SOLUTIONS

Pour implémenter un algorithme génétique, la première étape consiste à proposer une représentation de la solution. Cette représentation est donnée par l'équation (3.18). Pour rappel, dans ce genre d'algorithme, les solutions sont appelées des individus et leurs composants des gènes.

$$S = \begin{cases} L = \{L_1, L_2, \dots, L_m\} \\ P = [p_1, p_2, \dots, p_n] \end{cases} \quad (3.18)$$

Chaque solution S est composée d'un ensemble de listes de picking ordonnées L et d'un vecteur d'emplacement P . Chaque liste L_i est elle-même un vecteur contenant des références du stock et respectant un ordre précis qui sera l'ordre dans lequel l'agent devra effectuer l'opération de picking. Pour rappel, le vecteur P correspond à l'appariement entre les références stockées et leurs emplacements (la référence i se trouve à l'emplacement $P(i)$).

3.3.3.2/ CROISEMENTS ET MUTATIONS

Comme vu précédemment, pour mettre en place un algorithme génétique, il est nécessaire de définir des opérateurs de croisement et de mutation. Les opérateurs sont les fonctions qui vont permettre de créer de nouvelles solutions (enfants ou filles) à partir de solutions déjà existantes (parents).

Les opérateurs de croisements ont pour rôle de simuler la reproduction de deux individus. Le principe est de trouver une façon astucieuse de fusionner les composantes de deux solutions pour en créer une troisième qui va garder des aspects communs avec ses parents. Dans le cadre de la thèse, nous proposons deux opérateurs de croisement dont les équations (3.19) et (3.20) donnent des exemples d'application. Le premier opérateur identifie les éléments robustes de chaque liste de picking des deux solutions parentes. Les éléments robustes sont les références qui se trouvent à la même place dans les listes comparées deux à deux de chaque solution, ces éléments sont conservés dans la solution fille et les autres éléments sont choisis aléatoirement. Pour ce qui est du vecteur P celui d'un des deux parents est aléatoirement sélectionné pour être transmis à la solution fille. Ainsi, la solution fille contient toutes les parties communes des listes de picking de ses deux parents.

Le second opérateur de croisement fonctionne selon un principe similaire, à la différence que ce dernier s'effectue non pas sur les listes mais sur le vecteur P . Dans ce cas, c'est l'ensemble des listes d'une des solutions parentes qui est aléatoirement choisi pour être transmis à la solution fille.

Lorsque deux individus doivent être croisés, l'un des deux opérateurs est aléatoirement choisi.

$$\left. \begin{array}{l} L_i = [2, \underline{5}, 8, 6, \underline{7}, \underline{4}] \\ L'_i = [8, \underline{5}, 6, 2, \underline{7}, \underline{4}] \end{array} \right\} \rightarrow [6, \underline{5}, 2, 8, \underline{7}, \underline{4}] \quad (3.19)$$

$$\left. \begin{array}{l} P = [\underline{1}, 4, 6, \underline{2}, 3, 7, \underline{5}, 8] \\ P' = [\underline{1}, 6, 7, \underline{2}, 8, 4, \underline{5}, 3] \end{array} \right\} \rightarrow [\underline{1}, 7, 3, \underline{2}, 4, 6, \underline{5}, 8] \quad (3.20)$$

Une fois les opérateurs de croisement définis, il est nécessaire de créer les opérateurs de mutation. Les opérateurs de mutation ont pour rôle de simuler les mécanismes de mutation génétique spontanée d'un individu en modifiant un ou plusieurs de ses gènes. Du point de vue de l'algorithme génétique, il s'agit concrètement de modifier légèrement une composante d'une solution. Pour l'algorithme génétique détaillé ici, on définit trois opérateurs de mutation.

Le premier crée une nouvelle solution à partir d'une solution mère en sélectionnant aléatoirement une liste de picking L_i , puis deux éléments de cette liste. Les deux éléments sont alors intervertis pour créer la solution fille, ce qui revient à modifier l'ordre dans lequel l'agent récoltera les objets de la liste. Un exemple de cet opérateur est donné par l'équation (3.21).

Le second opérateur effectue la même opération que le premier, mais sur toutes les listes de picking au lieu d'une seule choisit aléatoirement.

Enfin, le troisième opérateur de mutation va modifier le vecteur P . Pour ce faire, deux éléments de P sont aléatoirement sélectionnés puis intervertis. Concrètement, cette manipulation intervertit l'emplacement de deux références au sein du stock. Un exemple de cet opérateur est donné par l'équation (3.22).

$$L_i = [2, \underline{5}, 8, \underline{6}, 7, 4] \longrightarrow [2, \underline{6}, 8, \underline{5}, 7, 4] \quad (3.21)$$

$$P = [1, \underline{4}, 6, 2, \underline{3}, 7, 5, 8] \longrightarrow [1, \underline{3}, 6, 2, \underline{4}, 7, 5, 8] \quad (3.22)$$

3.3.4/ LES MÉTHODES MIXTES

Les sections précédentes de ce chapitre ont permis de détailler les différentes méthodes de résolution possédant chacune leurs avantages et leurs inconvénients. Afin d'obtenir la meilleure solution possible, il est possible de combiner ces méthodes pour combiner les avantages de chacune. Dans cette section, nous présentons deux méthodes construites sur ce principe que nous appellerons des méthodes mixtes. On définit ainsi "l'algorithme génétique mixte" et la "méthode B&B mixte", le principe de ces méthodes est détaillé dans la figure (3.8).

- **Etape 1** : l'ensemble des listes de dotations (L) est donné et utilisé pour exécuter

un MFIF (voir section (3.3.1)). L'algorithme du MFIF permet d'obtenir un vecteur d'emplacement P à partir des listes de dotations.

- **Etape 2** : l'ensemble des listes L et le vecteur d'emplacement P sont utilisés pour construire une solution. Un algorithme de TSP est utilisé avec une limite de temps définie pour optimiser les chemins de cette solution. Une fois le problème résolu (ou la limite de temps atteinte) et si la solution n'est pas faisable, la méthode d'utilisation des sous-tours est utilisée pour obtenir une solution réaliste (voir section (3.3.1.2)).
- **Etape 3** : la solution obtenue est utilisée pour initialiser les deux autres méthodes. Dans le cas de l'algorithme génétique (voir section (3.3.3)) cette solution initiale est ajoutée à la population initiale. Pour ce qui est de la méthode basée sur le B&B (voir section (3.3.2)), l'évaluation de la meilleure solution trouvée est initialisée en évaluant la solution initiale.

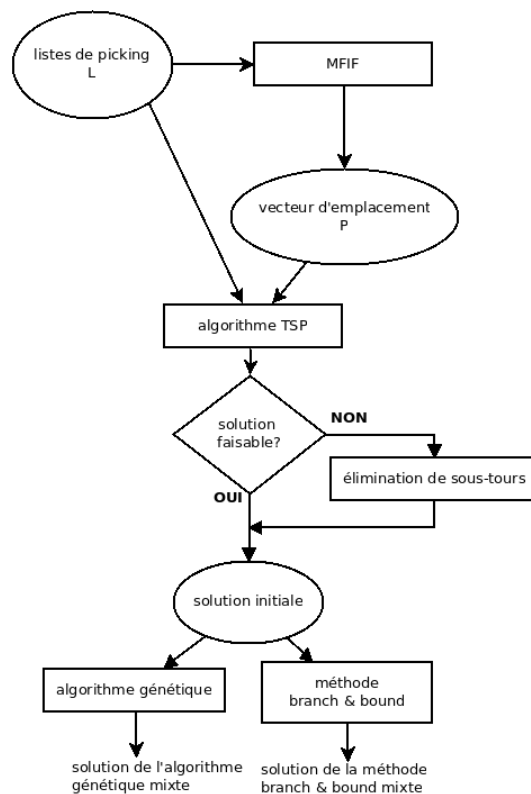


FIGURE 3.8 – construction des méthodes mixtes

La philosophie dernière de la création de ces méthodes est de combiner la rapidité d'obtention d'une solution intuitive avec la puissance des méthodes d'optimisations.

L'algorithme génétique mixte est donc basé sur l'idée d'insérer, dans la population de départ, un individus globalement de meilleure qualité de que autres (comprendre avec

une meilleure évaluation). Cet individu devrait naturellement être conservé pendant les premiers cycles et ainsi transmettre ces meilleurs gènes aux autres individus accélérant ainsi de processus de sélection génétique. Il est ainsi possible d'identifier les meilleurs aspects de la solution initiale et de les compléter avec d'autres gènes obtenus grâce aux mécanismes d'évolution propre aux algorithmes génétiques. Précisons que pour que cette méthode mixte soit efficace il faut s'assurer que les solutions ne convergent pas vers un optimal local formé par la solution initiale, auquel cas l'algorithme génétique n'apporterait aucune amélioration supplémentaire.

Pour ce qui est du B&B mixte, l'idée est d'optimiser les coupes lors de l'exploration de l'arbre de solutions. En effet, partir d'une solution de référence avec une bonne évaluation permet de faire des coupes dès l'instant où la borne inférieure d'une solution partielle dépasse l'évaluation de la référence. De cette manière, il est possible de ne pas explorer certaines parties de l'arbre peu prometteuses. Notons que pour que cette méthode mixte soit efficace, il faut s'assurer qu'une solution de meilleure qualité que celle de référence soit trouvée avant la fin de la limite de temps, ce qui n'est pas toujours évident pour des arbres de solutions de tailles importantes.

3.4/ RÉSULTATS EXPÉRIMENTAUX

Afin de déterminer la meilleure méthode de résolution parmi celles détaillées précédemment, chacune d'elle doit être testée et évaluée sur un large spectre d'instances du problème. Une instance du problème est définie par une disposition d'emplacement dans un stock de pharmacie et un ensemble de listes de dotations pré-ordonné. Dans le cadre de ce travail de thèse, les expérimentations sont menées sur un stock de DM contenant 400 emplacements (environ le nombre total de palettes présentes dans la pharmacie de l'hôpital Emile Muller), la disposition du stock en question est représentée sur la figure (3.9) Plusieurs méthodes de résolution ont donc été testées et des notations leurs sont attribuées.

- [MFIF + TSP] La première méthode consiste à exécuter un MFIF suivi d'un algorithme de résolution du TSP avec une limite de temps comme décrit dans la section (3.3.1.2). La limite de temps est paramétrée à 10 minutes.
- [AG] Ensuite, l'algorithme génétique est testé (voir section (3.3.3)) avec une population initiale aléatoire. Les paramètres de l'algorithme sont les suivants :

taillePopulationInitiale = 200

nombreDeGeneration = 1000

tauxDeCroisement = 50%

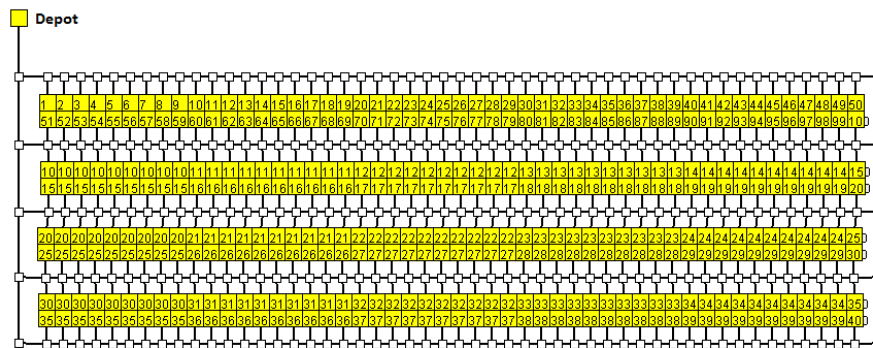


FIGURE 3.9 – stock de 400 emplacements

$$\text{tauxDeMutation} = 30\%$$

$$\text{limiteDeTemps} = 10\text{min}$$

- $[AG_{mix}]$ L'algorithme génétique mixte est alors testé (voir section (3.3.4)) avec les mêmes paramètres que l'algorithme génétique classique.
- $[BB]$ La méthode basée sur le B&B est ensuite évaluée avec une limite de temps de 10 minutes et aucune initialisation particulière (voir section (3.3.2))
- $[BB_{mix}]$ Enfin, la dernière méthode testée est le B&B mixtes (voir section (3.3.4)) avec les mêmes paramètres que le B&B classique.

Il est important de noter que le solveur utilisé pour les expérimentations (dans le cas du TSP) est GUROBI™ . GUROBI™ est un solveur exacte créé par Zonghao Gu, Edward Rothberg and Robert Bixby. Il est, entre autres, utilisé pour résoudre des problèmes avec une formulation de type MILP.

Toutes les méthodes détaillées précédemment sont testées sur un ensemble de 27 scénarii. Cet ensemble représente un large panel de possibilités faisant varier le nombre de listes de dotation de 5 à 20 et le nombre d'éléments dans chaque liste de 10 à 90. Les tables (3.2) et (3.3) présentent les résultats obtenus pour chaque scénario avec respectivement les méthodes basées sur l'algorithme génétique et celles basées sur le B&B. Dans chaque cas, les résultats donnés sont des moyennes calculées sur 10 tests. Pour chaque scénario, deux indicateurs sont présentés. Tout d'abord, l'évaluation de la meilleure solution, c'est à dire, la distance parcourue (en unité de distance) par l'agent lors de la collecte des listes de dotation. Puis, les écarts comparatifs entre les méthodes mixtes et les autres. Ces écarts permettent de mesurer l'amélioration induite par l'utilisation conjointe des méthodes qui composent les méthodes mixtes, ils sont donnés en pourcentage et détaillés par les équations (3.23), (3.24), (3.25), (3.26) et (3.27). Pour être plus précis, les équations (3.23) et (3.24) présentent les méthodes de calcul des écarts entre, respectivement, les méthodes AG et $MFIF + TSP$ et la méthode AG_{mix} . Les équations (3.25) et (3.26), quant à elles, présentent les méthodes de calcul des

écarts entre, respectivement, les méthodes BB et $MFIF + TSP$ et la méthode BB_{mix} . Enfin, l'équation (3.27) présente la méthode de calcul de l'écart entre les méthodes AG_{mix} et BB_{mix} .

$$\Delta_{AG_{mix}/AG} = \frac{Fit_{AG_{mix}} - Fit_{AG}}{Fit_{AG}} \times 100 \quad (3.23)$$

$$\Delta_{AG_{mix}/MFIF+TSP} = \frac{Fit_{AG_{mix}} - Fit_{MFIF+TSP}}{Fit_{MFIF+TSP}} \times 100 \quad (3.24)$$

$$\Delta_{BB_{mix}/BB} = \frac{Fit_{BB_{mix}} - Fit_{BB}}{Fit_{BB}} \times 100 \quad (3.25)$$

$$\Delta_{BB_{mix}/MFIF+TSP} = \frac{Fit_{BB_{mix}} - Fit_{MFIF+TSP}}{Fit_{MFIF+TSP}} \times 100 \quad (3.26)$$

$$\Delta_{GA_{mix}/BB_{mix}} = \frac{Fit_{GA_{mix}} - Fit_{BB_{mix}}}{Fit_{BB_{mix}}} \times 100 \quad (3.27)$$

Les valeurs de chaque écart sont calculées afin d'être comparées. La table (3.4) permet de faire cette comparaison en détaillant pour chaque écart sa valeur minimum, sa valeur maximale et sa moyenne sur les 27 scénarii proposés. Cette table met en évidence la supériorité de l'algorithme génétique mixte dont les résultats sont bien meilleurs que ceux de ses constituants. La méthode mixte a pu tirer parti de la solution intuitive produite par le MFIF en l'améliorant grâce à l'algorithme génétique classique. Par contre, on observe que la méthode du B&B mixte n'apporte presque aucune amélioration par rapport à la méthode MFIF. Concrètement, on observe que le B&B mixte donne de meilleurs résultats que sa version classique, mais en réalité cette comparaison améliorative est uniquement due à la supériorité de la solution obtenue grâce à la méthode MFIF par rapport au B&B. Il semble que la dimension trop importante du problème détaille précédemment (voir section (3.2.1)) entraîne une taille trop importante de l'arbre de solutions et donc un faible efficacité des mécanismes de B&B. C'est pourquoi on observe une nette supériorité de l'algorithme génétique mixte qui est capable d'obtenir de très bon résultats pour l'ensemble des scénarii proposée.

De plus, on remarque que les écarts de performance entre les méthodes standards et les méthodes mixtes augmentent avec la taille des problèmes traités. En effet, pour des problèmes de petites tailles (faible nombre de liste et de références dans les listes) les deux types de méthodes produisent des résultats similaires (il arrive même que les méthodes standards donnent de meilleurs résultats). Cela s'explique par l'efficacité de l'algorithme génétique et du B&B pour trouver la solution optimale, si le problème est d'assez petite taille il est en effet possible de déterminer la solution optimale avec ces méthodes alors que les méthodes mixtes peuvent se bloquer dans un optimum local formé par la solution du MFIF.

Dans le cas des problèmes de grandes tailles, les écarts se stabilisent et on note une nette différence entre les deux types de méthodes.

TABLE 3.2 – optimization methods efficiency (GA)

lists	items	MFIF		AG		+AG _{mix}
		fit.	$\Delta_{AG_{mix}/MFIF+TSP}$	fit.	$\Delta_{AG_{mix}/AG}$	fit.
5	10	8370	13.88	6979	-3.28	7208
5	20	14265	7.00	13842	4.16	13266
5	30	18646	5.86	20714	15.26	17552
5	40	32332	29.41	28777	20.70	22820
5	50	39307	30.44	35509	23.00	27339
5	60	45793	31.84	43978	29.02	31212
5	70	53241	33.31	54070	34.33	35504
5	80	34483	0.67	65632	47.81	34249
5	90	128217	53.85	74733	20.77	59204
10	10	19730	13.57	15884	-7.34	17051
10	20	30961	6.61	31106	7.05	28912
10	30	39599	4.68	47122	19.90	37744
10	40	69363	28.79	64122	22.97	49387
10	50	106389	39.90	82720	22.70	63937
10	60	130788	42.61	104219	27.99	75048
10	70	64099	0.72	126895	49.85	63632
10	80	223895	48.01	148636	21.69	116386
10	90	123661	27.45	174740	48.66	89710
15	10	32489	13.72	26806	-4.56	28030
15	20	60472	22.16	51740	9.03	47067
15	30	60873	3.02	74740	21.01	59031
15	40	71115	2.20	105079	33.81	69546
15	50	123543	27.57	133226	32.83	89481
15	60	142665	27.27	171390	39.46	103747
15	70	163630	27.42	207755	42.83	118758
15	80	324778	43.44	247260	25.71	183671
15	90	112575	0.21	289054	61.13	112332
20	10	45196	11.79	38753	-2.87	39866
20	20	83278	21.61	72885	10.43	65280
20	30	82195	2.41	106004	24.33	80206
20	40	174657	35.10	149826	24.35	113339
20	50	225604	38.17	193137	27.78	139481
20	60	190407	25.79	244661	42.25	141285
20	70	218442	25.56	302044	46.17	162588
20	80	139193	0.35	353486	60.76	138700
20	90	368919	34.02	416561	41.56	243410

3.5/ CONCLUSION

Ce chapitre a présenté des méthodes pour résoudre le problème de répartition des emplacements du stock de la pharmacie avec modification de l'ordre de picking des listes de dotation. Le but de ces méthodes est de produire (en un temps raisonnable) une so-

TABLE 3.3 – optimization methods efficiency (BB)

lists	items	MFIF		BB		BB_{mix}
		fit.	$\Delta_{BB_{mix}/MFIF+TSP}$	fit.	$\Delta_{BB_{mix}/BB}$	fit.
5	10	8370	0.08	13674	38.84	8363
5	20	14265	0.35	30219	52.96	14214
5	30	18646	0.17	49027	62.03	18613
5	40	32332	1.39	69569	54.17	31881
5	50	39307	1.71	92147	58.07	38633
5	60	45793	1.80	112426	60.05	44968
5	70	53241	1.42	133541	60.70	52480
5	80	34483	0.03	157683	78.13	34470
5	90	128217	2.58	184263	32.21	124906
10	10	19730	0.83	30243	35.30	19565
10	20	30961	0.08	69149	55.26	30936
10	30	39599	0.30	112184	64.81	39477
10	40	69363	0.90	155180	55.70	68735
10	50	106389	1.16	196162	46.3	105154
10	60	130788	1.99	242227	47.08	128179
10	70	64099	0	287278	77.68	64099
10	80	223895	0.90	332058	33.18	221859
10	90	123661	0	377194	67.21	123661
15	10	32489	0.22	52988	38.82	32416
15	20	60472	0.27	111860	46.08	60304
15	30	60873	0.06	172949	64.82	60835
15	40	71115	0.007	236134	69.88	71109
15	50	123543	1.26	298333	59.11	121986
15	60	142665	0.55	371471	61.80	141878
15	70	163630	1.24	436824	63.00	161589
15	80	324778	0.70	500769	35.59	322502
15	90	112575	0	568117	80.18	112575
20	10	45196	0.27	70452	36.02	45070
20	20	83278	1.31	150948	45.55	82186
20	30	82195	0.01	236374	65.23	82182
20	40	174657	0.13	322005	45.83	174419
20	50	225604	0.36	410927	45.30	224772
20	60	190407	0.49	496585	61.84	189458
20	70	218442	0.75	589363	63.21	216782
20	80	139193	0	672640	79.30	139193
20	90	368919	0.33	767810	52.11	367678

lution logistique sous la forme d'un vecteur d'emplacement P permettant de définir la position de chaque référence dans le stock ainsi que d'un ensemble L de listes de picking ordonnées. Les solutions produites doivent être réalisables et impliquer le moins de déplacement possible pour l'agent qui devra effectuer les opérations de picking. L'application de ces méthodes suppose que l'agencement de la pharmacie est connu ainsi

TABLE 3.4 – gaps analysis

	Min.	Max.	Moy.
$\Delta_{AG_{mix}/AG}$	-7.3	61.1	26.1
$\Delta_{AG_{mix}/MFIF+TSP}$	0.22	53.8	21.7
$\Delta_{BB_{mix}/BB}$	32.2	80.1	55.4
$\Delta_{BB_{mix}/MFIF+TSP}$	0	2.58	0.66
$\Delta_{GA_{mix}/BB_{mix}}$	0.04	35.6	10.6

que l'ensemble des listes de dotation (dans un ordre initial) qui est supposé ne pas (ou peu) être modifié au fil du temps.

Comme nous l'avons vu, le problème est trop complexe pour être résolu en un temps raisonnable par un solveur exact et c'est pour cette raison que cinq méthodes produisant de bonnes solutions non optimales ont été proposées. Les tests menés sur un ensemble représentatif de scénarii ont pu montrer que la méthode MFIF est capable de produire une bonne solution basée sur des améliorations intuitives en un temps très court, de plus, elle peut être améliorée par l'utilisation d'un algorithme de résolution du TSP. L'algorithme génétique et le branch and bound sont des méta heuristiques aux performances reconnues qui permettent d'obtenir des solutions sans prendre en compte le contexte du problème, ces solutions peuvent même s'améliorer tant que les algorithmes ne sont pas stoppés par une condition d'arrêt. C'est pour combiner la force de ces deux approches que nous avons mis en place les méthodes mixtes dont le but est d'initialiser les deux métaheuristiques précédemment citées avec une solution intuitive obtenue par la méthode MFIF. Bien qu'il semble que l'utilisation des principes du branch and bound ne soit (dans ce cas) pas plus efficace que la méthode intuitive, l'algorithme génétique mixte, lui, permet d'obtenir de bonne qualité parfaitement exploitable et surtout supérieurs aux performances de ses deux composants (la méthode MFIF et l'algorithme génétique classiques).

AFFECTATION DES DISPOSITIFS MÉDICAUX AUX INTERVENTIONS

Dans l'introduction générale de ce mémoire, nous avons vu que le deuxième point abordé lors du travail de thèse est l'ordonnancement des interventions chirurgicales. Concrètement, il s'agit de mettre en place des méthodes de création de plannings d'interventions qui prennent en compte les problématiques liées au service de stérilisation. Les résultats de l'étude préliminaire du fonctionnement de la PUI ont permis de mettre en évidence la répartition aléatoire de la charge de travail du service de stérilisation. En effet on observe au fil d'une journée de travail des périodes de faible activité (peu de DM à stériliser) suivit de périodes de forte activité (plusieurs boîtes de DM réceptionnées en même temps). Ce phénomène est dû à la politique de planification des interventions chirurgicales d'avantage basé sur des considérations liées au bloc opératoire.

En conséquence, ce chapitre présente des méthodes de planification dont l'objectif est de lisser la charge de travail du service de stérilisation. Pour ce faire, nous proposons de minimiser le nombre total de boîtes de DM nécessaire pour pratiquer un nombre d'interventions donné pour une période de temps.

Un modèle MILP du problème considéré est proposé. Ce dernier est basé sur le modèle du problème de bin packing avec de nouvelles contraintes liées à l'aspect temporel. Afin de résoudre le problème en un temps raisonnable plusieurs méthodes sont envisagées : la recherche tabou, les algorithmes génétiques et l'optimisation par essaim particulaire. Le choix des méthodes est basé sur les recommandations de la littérature. Afin de faire une sélection, des versions simplifiées des algorithmes de chaque méthode sont évaluées et comparées. L'optimisation par essaim particulaire semblant être la plus prometteuse, une version plus complexe est détaillée dans la suite du chapitre.

Pour s'assurer d'obtenir la meilleure solution possible, les paramètres de l'optimisation par essaim particulaire sont étudiés. Un algorithme de comparaison des résultats obtenus avec différentes combinaisons de paramètres est alors présenté et permet de statuer

sur les meilleurs valeurs possible de ces derniers dans le cadre de cette étude. De plus, on remarque que l'espace des solutions comprend une importante proportions de solutions infaisables complexifiant le calcul. Pour palier à ce problème, on définit un espace de calcul abstrait sans "zone morte" ainsi qu'une application bijective permettant de faire de lien entre les deux espaces.

Enfin, dans la section des résultats expérimentaux, la comparaison est faite entre l'algorithme d'optimisation par essaim particulière paramétré avec les valeurs précédemment définit et celui avec les paramètre communément admis dans la littérature. Les résultats confirment que les particularités du problème traité ici rendent légitimes le changement de paramètres proposé.

Les méthodes mises en place dans ce chapitre ont conduit aux publications de [Beroule et al.,], [Beroule et al., 2016a] et [Beroule et al., 2016b].

4.1/ ÉTAT DE L'ART

Au sein de la littérature scientifique, il existe de très nombreuses études qui traitent des problèmes liés à l'ordonnancement des interventions chirurgicales au sein d'un hôpital. La plupart du temps dans ce genre de travaux, l'accent est mis sur les méthodes employées ou sur les objectifs proposés. La disponibilité des salles d'opération et souvent un facteur déterminant. Par exemple, [Dexter et al., 1999] ont proposé d'utiliser une approche basée sur un problème de bin packing (voir définition (20)) inspiré du domaine industriel en y ajoutant des contraintes floues dans le but de maximiser le taux d'occupation des salles d'opération dans le cadre d'un problème d'ordonnancement au sein d'un bloc opératoire multidisciplinaire. En effet, comme nous le verrons par la suite le problème du bin packing, simple en apparence, peut être adapté à de nombreux domaines et notamment à l'ordonnancement.

[Hanset et al., 2007], quant à eux, ont fait le choix d'utiliser une métaheuristique, la recherche tabou, en l'adaptant au contexte hospitalier pour créer une méthode d'ordonnancement d'interventions qui prend en compte la disponibilité des chambres de réveil. Ce facteur est déterminant puisque sans ces chambres de nombreuses interventions ne peuvent simplement pas avoir lieu. Bien sûr, de nombreux autres aspects entrent en jeu lors du choix du planning opératoire. [Guinet et al., 2003] ont mis en place un outil d'aide à la décision basé sur un modèle MILP (voir définition (13)) pour faire de l'ordonnancement tout en minimisant le plus possible les coûts induits pour l'hôpital. À l'inverse, [Dexter et al., 2002] ont aussi proposé un modèle MILP pour cette fois maximiser les coûts et ainsi déterminer les scénarii extrêmes et les intégrer comme bornes extremums à d'autres méthodes.

Définition 20 : Bin packing [Teghem, 2012]

Le bin packing est un problème de recherche opérationnelle qui consiste à ranger des objets plus ou moins volumineux dans un nombre minimal de boîtes possédant une certaine capacité. Voici une formulation mathématique possible du problème de bin packing.

Paramètres :

- n : nombre d'objets (et aussi de boîtes).
- c_i : volume de l'objet i .
- V : capacité des boîtes.

Variables :

- $x_{i,j}$: variable binaire égale à 1 si et seulement si l'objet i est rangé dans la boîte j
- y_j : variable binaire égale à 1 si et seulement si la boîte j est utilisée.

La fonction objectif est donnée par :

$$\text{Minimiser } \sum_{j=1}^n y_j \quad (4.1)$$

sujette à un certain nombre de contraintes :

$$\sum_{i=1}^n c_i \times x_{ij} \leq V \times y_j \quad \forall 1 \leq j \leq n \quad (4.2)$$

$$\sum_{j=1}^n x_{ij} = 1 \quad \forall 1 \leq i \leq n \quad (4.3)$$

La fonction objectif (4.1) implique de trouver la solution qui minimise le plus possible le nombre de boîtes utilisées. La contrainte (4.2) s'assure que la somme des volumes des objets contenus dans une boîte ne dépasse pas la capacité de ladite boîte. On notera que l'ajout de la variable y_j au second membre de cette équation (qui peut sembler maladroit au premier abord) est nécessaire pour lier la fonction objectif et les contraintes, de cette manière on s'assure qu'une boîte non utilisée ne peut qu'être vide. Enfin l'équation (4.3) s'assure qu'un objet est rangé dans une unique boîte.

De plus, au-delà des méthodes d'ordonnement, des chercheurs se sont penchés sur l'analyse de certains aspects logistiques du bloc opératoire. Dans cet ordre d'idée, [Wullink et al., 2007] ont mené une étude pour déterminer l'importance de créer une ou plusieurs salles d'opération réservées aux interventions urgentes (non planifiées) afin d'améliorer la réactivité du service face aux imprévus et quantifier l'impact sur la prise en

charge globale de tous les patients (urgents ou non).

On peut déjà observer à la vue de ces quelques exemples que le choix des critères de performances est un aspect crucial de l'ordonnement des interventions chirurgicales. Les critères de performances sont les points que l'on cherche à améliorer par l'utilisation d'une méthode ou d'un outil et leurs choix influencent grandement la direction d'un travail de recherche. [Cardoen et al., 2010] ont identifié les huit principaux critères de performances les plus utilisés dans la littérature.

- Le temps d'attente.
- La cadence des interventions.
- Le taux d'utilisation des chambres d'opération.
- La durée totale d'une série d'interventions (ou "makespan" en anglais).
- Les annulations d'interventions.
- Le coût financier.
- Les préférences des agents concernés.

Par exemple, on trouve de nombreuses études cherchant à éviter la sur-utilisation ou la sous-utilisation des chambres d'intervention du bloc opératoire en utilisant des méthodes héritées de la recherche opérationnelle ([Dexter et al., 2003, Dexter et al., 1999, Dexter et al., 2004]). On trouve aussi [Marcon et al., 2006] qui proposent des solutions pour ordonner les interventions en minimisant la durée totale desdites interventions (la durée entre le début de la première et la fin de la dernière) en considérant différentes fenêtres de temps. [Cardoen et al., 2009a] et [Cardoen et al., 2009b] ont, pour leur part, étudié une version multi objectifs du problème et proposé plusieurs méthodes pour le résoudre dans le cadre d'un service ambulatoire en déterminant le meilleur ordre possible pour traiter les patients.

On remarque, en étudiant la littérature scientifique, que bien que de nombreux critères soient pris en compte dans les différentes études traitant du problème d'ordonnement des interventions chirurgicales, la disponibilité des DM nécessaires pour pratiquer les interventions n'est que rarement prise en compte. C'est pourquoi, dans ce chapitre, le problème est vu sous un angle nouveau qui est celui des enjeux logistiques du service de stérilisation (dont dépend grandement la disponibilité des DM). Les premières observations à l'hôpital Emile Muller nous ont mené à la conclusion que les décisions logistiques liées au bloc opératoire se prenaient généralement sans prendre en compte celles du service de stérilisation. En effet, les décisions liées aux interventions chirurgicales sont prises en amont par des décisionnaires de l'hôpital et l'activité de stérilisation doit s'adapter au mieux pour répondre à la demande.

4.2/ LE SYSTÈME ÉTUDIÉ

L'hôpital Emile Muller n'utilise pas de méthodes d'optimisation ou d'outils d'aide à la décision pour établir ces plannings d'interventions. Les rendez-vous sont directement pris par les chirurgiens et les salles d'opération sont réservées en conséquence si elles sont disponibles au moment de la prise de rendez-vous. Afin de s'assurer que le planning ainsi créé est réalisable, un comité dédié se réunit chaque semaine. Ce comité (constitué de cadres de l'hôpital) statue sur la validité du planning de la semaine suivante et si besoin effectue des modifications. Il est important de noter qu'aucun agent de la pharmacie n'est convié à cette réunion. Bien sûr, cela ne signifie pas que les différentes entités ne communiquent pas entre elles. En cas de problèmes (par exemple la non-disponibilité d'un DM pour un horaire donné) les cadres du service de stérilisation peuvent directement entrer en contact avec ceux du bloc pour modifier à nouveau le planning.

Le fait de ne pas inclure les dirigeants de la pharmacie dans le processus de planification pose certains problèmes. Une première analyse du fonctionnement du service de stérilisation au quotidien a permis, en effet, de mettre en évidence que ce service était soumis au cours de la journée à des pics d'activité. Il apparaît que pendant ces pics, le travail des agents est ralenti à cause, notamment, du manque de place pour les DM en attente et de la quantité importante d'informations circulant (DM prioritaires, mauvaise traçabilité, machine défectueuse...). Ces pics d'activités sont dus à de multiples arrivages simultanés de DM sales en provenance du bloc opératoire. Ces arrivages sont la conséquence d'interventions chirurgicales se terminant simultanément sur un même site. Cette situation reflète la non-prise en compte de la disponibilité des DM lors de la planification des interventions. Cette négligence pose deux problèmes majeurs. Tout d'abord, la présence de pics d'activité ralentit le processus de stérilisation et augmente le risque d'erreurs. Ensuite, chaque boîte de DM est disponible en quantité limitée, il faut donc s'assurer que la quantité disponible est suffisante pour assurer toutes les interventions. Le lissage de la charge de travail au sein du service de stérilisation permettrait donc de rendre plus sûr le circuit de stérilisation des DM.

Le but sous-jacent de ce travail est de mettre en place des outils concrets pour que les décisions logistiques concernant à la fois le bloc opératoire et la PUI se prennent en ayant conscience des enjeux logistiques des deux entités. Ainsi nous espérons poser les jalons d'une nouvelle logistique transversale liant chaque entité de l'hôpital.

4.3/ MODÉLISATION

Afin de correctement appréhender l'étendue du problème et ses enjeux, il est nécessaire de correctement l'identifier et d'en proposer une modélisation. Le problème présenté dans ce chapitre est un problème d'ordonnement avec certaines particularités. Il est donc important de commencer par mettre en évidence ses spécificités liées à la prise en compte du service de stérilisation.

4.3.1/ DÉFINITION DU PROBLÈME

L'ordonnement d'interventions chirurgicales est un problème complexe traitant avec de nombreuses variables (ressources humaines et matérielles, salles d'opération, créneaux horaires, compétences...). Afin de se concentrer sur le cœur du problème, nous posons les hypothèses suivantes.

- Chaque intervention est assignée à une équipe de personnel hospitalier.
- Les ressources humaines sont disponibles en nombre illimité.
- Les horaires d'ouverture des salles d'opération sont identiques d'une salle à l'autre.
- Les patients sont tous prêts pour subir l'intervention.
- Les patients admis en urgence ne sont pas considérés.
- Les problèmes relatifs à la disponibilité des chambres de réveils ne sont pas pris en compte.
- Aucun problème de transport de patients ou de matériel ne peut se produire.
- Une intervention nécessite l'utilisation d'une unique boîte de DM.
- La boîte de DM nécessaire pour une intervention ne dépend que de la nature de l'intervention.

En considérant ces hypothèses, le problème consiste alors en la création d'un planning d'interventions chirurgicales qui minimise le nombre de boîtes de DM nécessaire pour être réalisé. En effet, une boîte de DM étant stérilisée après utilisation, il est possible de la réutiliser. Ainsi une même boîte peut être utilisée plusieurs fois dans la même semaine pour des interventions de même type. Minimiser le nombre global de boîtes utilisées apporte plusieurs avantages, cela permet de garder des boîtes de DM en réserve en cas d'urgence ou de problèmes inattendus, mais aussi de naturellement lisser la charge de travail du service de stérilisation qui aura alors moins de DM différents à gérer et moins de chance que de nombreuses interventions se terminent en même temps.

Pour s'assurer de bien comprendre le problème ainsi posé, nous allons commencer par en donner une représentation graphique. La figure (4.1) présente un exemple de planning résultant de l'ordonnement de six interventions de même type (et donc, utili-

sant les mêmes boîtes). Chaque intervention I est notée de 1 à 6 (I_1, I_2, \dots, I_6). Pour

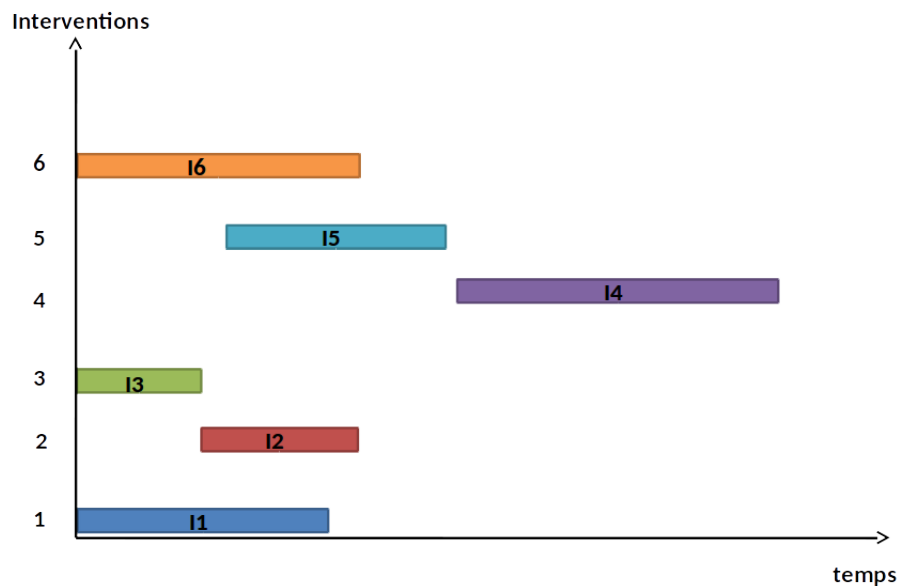


FIGURE 4.1 – Exemple de planning de six interventions

rappel, notre étude se focalise sur les boîtes de DM, c'est pourquoi les bandes de couleurs représentées sur la figure (aussi appelés "tâches") représentent un cycle de vie complet (voir figure (1.3)) de la boîte de DM utilisée lors de l'intervention (utilisation, transport, stérilisation, stockage). Par commodité ces tâches sont nommées comme les interventions qui les ont générées, elles sont représentées sur l'axe des ordonnées et sont placées sur l'axe temporel (à savoir les abscisses). La longueur des tâches permet de connaître la durée du cycle depuis l'utilisation des DM en salle d'opération jusqu'au stockage dans le service de stérilisation, en conséquence, la boîte concernée n'est pas disponible pour une nouvelle intervention pendant toute la durée de la tâche. Il découle de cette propriété que deux tâches qui se chevauchent ne peuvent pas utiliser la même boîte de DM. Il est donc possible, à partir d'une telle représentation, d'identifier graphiquement le nombre de boîtes minimum requis comme le montre la figure (4.2). Sur cette figure, on observe qu'il existe une fenêtre de temps (rectangle rouge) pendant laquelle les tâches I_1, I_2, I_5 et I_6 se chevauchent. On remarque qu'il n'existe pas de chevauchement impliquant plus d'interventions, c'est pourquoi le nombre minimum de boîtes nécessaire pour respecter ce planning est 4. On peut, en effet, observer que la boîte de DM nécessaire pour l'intervention I_3 peut être réutilisée pour l'intervention I_2 ou I_5 . Quant à l'intervention I_4 elle peut s'effectuer en utilisant la boîte de n'importe quelle autre intervention, car elle n'en chevauche aucune.

Pour plus de clarté sur l'utilisation des boîtes, la figure (4.3) présente le même planning que celui de la figure (4.1) mais cette fois ci en fonction des boîtes de DM. Cette

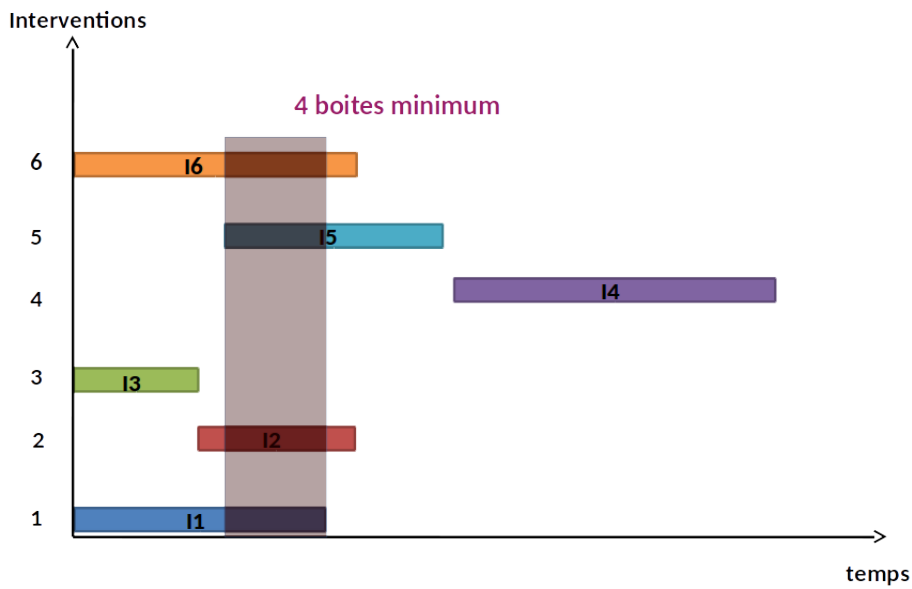


FIGURE 4.2 – Identification du nombre minimum de boîtes de DM

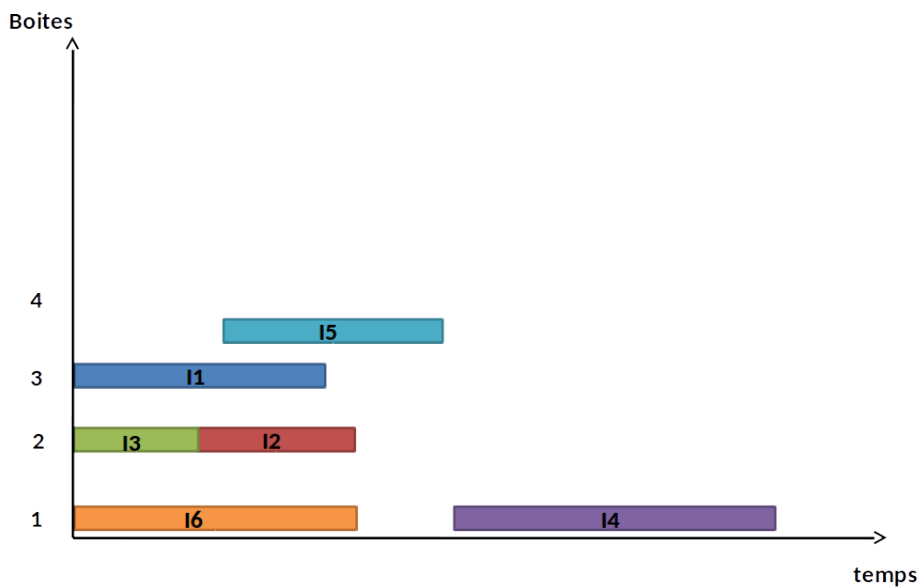


FIGURE 4.3 – Exemple de planning avec identification des boîtes

représentation présente les boîtes en ordonnée ce qui permet de voir rapidement la répartition desdites boîtes entre les différentes interventions. Dans cet exemple, la boîte 1 est utilisée pour les interventions I_6 et I_4 et la boîte 2 pour les interventions I_3 et I_2 . On remarque que cette dernière ne sera stockée qu'un très court instant, car directement réutilisée après sa stérilisation. Les boîtes 3 et 4 sont, quant à elles, utilisées chacune pour une unique intervention, respectivement I_1 et I_5 . Cette nouvelle représentation graphique permet aussi de facilement visualiser les modifications possibles pour diminuer

le nombre de boîtes minimum nécessaires. C'est dans cette optique que la figure (4.4) propose un nouvel ordonnancement des six interventions pour créer un planning qui ne demande qu'un minimum de 2 boîtes de DM pour être respecté. Cet exemple permet de mettre en évidence l'intérêt d'une optimisation de l'ordonnancement des interventions chirurgicales.

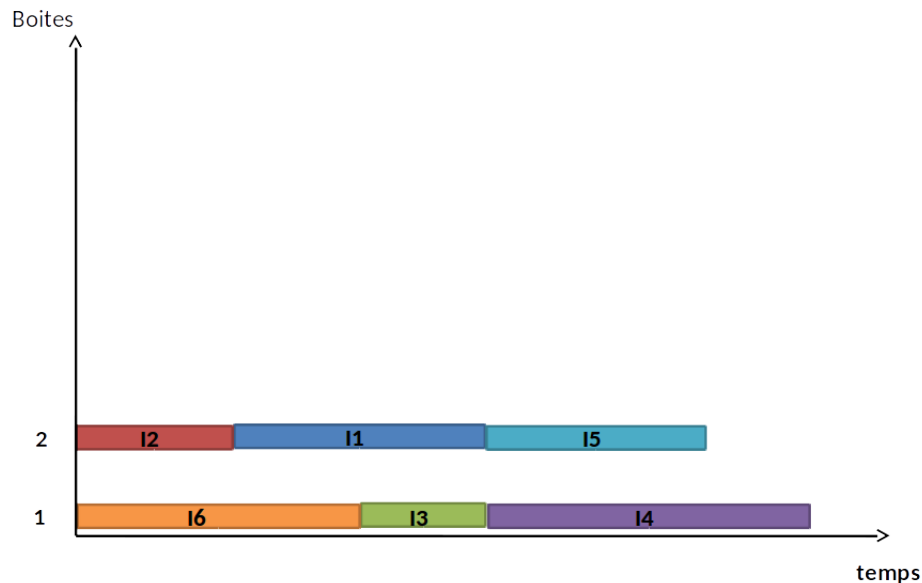


FIGURE 4.4 – Nouvel exemple de planning

4.3.2/ CARACTÉRISATION DU PROBLÈME

Une fois la possibilité d'amélioration des plannings d'interventions mise en évidence, il est important de se pencher sur les caractéristiques du problème. On remarque que ce dernier partage un certain nombre de similarités avec son équivalent dans le domaine industriel. Dans une industrie, il n'est en effet pas rare de devoir organiser des plannings de personnels ou de créations de produits en fonction de disponibilité de machines-outils ou de matières premières. En conséquence, il semble intéressant de considérer les méthodes d'optimisation utilisées dans ce domaine pour tenter de les adapter au problème considéré ici. Par exemple, une méthode de caractérisation de problèmes fréquemment utilisée dans l'industrie est une nomenclature proposée par [Graham et al., 1979]. Cette nomenclature permet d'identifier un problème d'ordonnement de produits. On suppose qu'un ensemble de produits doit être traité par un ensemble de machines qui doit répondre à certaines contraintes. Cette méthode de classification met l'accent sur certains aspects tels que le fonctionnement global de l'atelier, les

contraintes mises en jeu ou encore les critères de performances. L'idée est de classer chaque problème d'ordonnement suivant trois paramètres α , β et γ .

- Le paramètre α permet de caractériser l'atelier, il est lui-même composé de deux paramètres : $\alpha = \alpha_1\alpha_2$
 - α_1 représente l'agencement des machines de l'atelier. Ce dernier peut, par exemple, être un atelier à machine unique, à machines parallèles, à cheminement unique (flow shop), à cheminement multiples (job shop) ou autre configuration particulière. Chaque agencement est représenté par une lettre.
 - α_2 indique le nombre de machines dans l'atelier et si ce nombre est fixe ou variable.
- Le paramètre β liste les contraintes qui restreignent le problème. Ces contraintes peuvent être de différentes natures.
 - Les contraintes de temps alloué, issues généralement d'impératifs de gestion et relatives aux dates limites des tâches ou à la durée totale d'un projet.
 - Les contraintes d'antériorité et plus généralement les contraintes de cohérence technologique, qui décrivent le positionnement relatif de certaines tâches par rapport à d'autres.
 - Les contraintes de calendrier liées au respect d'horaires de travail, etc.
- Le paramètre γ , quant à lui, représente les critères de performances considérés. Le choix de ce paramètre est crucial, car il détermine l'aspect qui sera optimisé.

Le principe ici est de comparer l'ordonnement de produits et de machines à celui d'interventions chirurgicales. Il existe de nombreuses études qui identifient le bloc opératoire à un "atelier hybride à cheminement unique" ([Fei et al., 2010, Saadani et al., 2006, Jebali et al., 2006, Hanset et al., 2007]). Concrètement, les chercheurs derrière ces études substituent les patients aux produits et les différentes phases d'une intervention aux machines. On considère alors que chaque patient doit subir chaque phase dans l'ordre pour que l'intervention soit terminée.

Le travail de cette thèse se focalise, comme nous l'avons vu, sur la disponibilité des DM. En conséquence, nous proposons d'assimiler les interventions aux produits et les machines aux boîtes de DM. Ainsi, le problème considéré devient, selon la nomenclature, un *RI/prec, pmtn/"minNbBoites"*.

Commençons par identifier ces paramètres.

- $\alpha_1 = R$: la lettre R signifie que les interventions se font en parallèle et sont indépendantes l'une de l'autre. Malgré le fait que les boîtes de DM (donc les machines) puissent être identiques
- $\alpha_2 = I$: la lettre I signifie que le nombre de machines est fixe. Ce choix de pa-

ramètre découle directement du fait que le nombre de boîtes de DM est lui-même fixe.

· $\beta = prec, pmtn$:

prec : ce terme est la contraction de "precedence" (soit "précédence" en français).

Cette contrainte est utilisée lorsque certaines tâches ne peuvent débuter tant qu'une ou plusieurs autres tâches précises ne sont pas terminées. Dans notre cas, chaque tâche (chaque intervention) sera, comme nous le verrons par la suite, séparée en deux sous-tâches. La première représente l'utilisation de la boîte concernée lors de l'intervention et son transport vers le service de stérilisation tandis que la seconde concerne la phase de stérilisation et de stockage de la ladite boîte.

pmtn : ce terme est la contraction de "preemption" (soit "priorité" en français).

Cette contrainte signifie qu'une tâche peut être stoppée pendant son exécution pour permettre, par exemple, à une autre tâche plus prioritaire de s'effectuer. Dans le cas de la présente étude, il est en effet possible de stopper le processus de stérilisation à certains moments clés en faisant "attendre" une boîte. Il est important de noter que ceci est impossible lors d'une intervention puisqu'une fois cette dernière démarrée elle doit être terminée. Cette particularité justifie le découpage de chaque tâche en deux sous-tâches.

· $\gamma = minNbBoites$: le critère choisi consiste à créer un planning qui utilise le moins de boîtes de DM possible. En effet, puisqu'une boîte peut être stérilisée pour être réutilisée, il est possible d'agencer les interventions de façon à utiliser le plus possible chaque boîte afin que globalement le nombre de boîtes utilisées soit minimal. Il n'existe pas de paramètres préétablis dans la nomenclature de Graham pour représenter ce critère de performance. Cette constatation s'explique par le fait que dans notre analogie consistant à comparer un bloc opératoire et un atelier, le critère de performance considéré ici se traduirait par la minimisation du nombre de machines utilisées. Ce critère de performance inhabituelle nous force donc à considérer le problème d'ordonnement suivant un point de vue nouveau.

Maintenant que les caractéristiques du problème sont définies, il devient possible d'en proposer une modélisation mathématique.

4.4/ MODÈLE MATHÉMATIQUE

Comme nous l'avons vu précédemment, le problème considéré ici peut difficilement être considéré comme un problème d'ordonnement usuel à cause du critère de perfor-

mance choisit. C'est pourquoi, pour nous assurer d'en avoir une modélisation fidèle, nous proposons d'en définir un modèle MILP. Au vu des travaux menés que l'on peut trouver dans la littérature, il semble qu'une modélisation basée sur le problème de bin-packing soit la plus adaptée (voir définition 20). Dans notre cas, les boîtes du bin packing sont remplacées par les boîtes de DM et les objets par les interventions chirurgicales. De plus, dans le problème du bin packing chaque objet possède un certain volume qui, ici, sera remplacé par la durée de l'intervention correspondante, ou pour être plus précis, de la tâche qui en découle (intervention, transport, stérilisation, stockage). Le problème devient alors d'allouer chaque intervention à une boîte de DM en s'assurant de ne pas dépasser le temps disponible (généralement une semaine). Il sera nécessaire d'ajouter des contraintes temporelles pour respecter les contraintes de précédence et de priorité vue précédemment, mais aussi pour s'assurer que deux interventions utilisant la même boîte de DM ne se chevauchent pas (temporellement parlant).

Pour rappel, on considère que chaque intervention à ordonnancer induit la création de deux sous tâches qui correspondent pour la première au processus lié au bloc opératoire et pour la seconde à ceux liés à la stérilisation.

Nous définissons alors le modèle MILP de l'ordonnancement des interventions chirurgicales sur une semaine avec comme objectif d'utiliser le moins de boîtes de DM possible. Les paramètres du modèle peuvent alors être définis.

- n : nombre d'interventions.
- P_i : durée de l'intervention i et du transport de la boîte correspondante.
- S_i : durée du processus de stérilisation complet de la boîte de l'intervention i .
- S^+ : date d'ouverture du service de stérilisation.
- S^- : date de fermeture du service de stérilisation.
- P^+ : date d'ouverture du bloc opératoire.
- P^- : date de fermeture du bloc opératoire.
- T : durée d'ouverture du service de stérilisation sur une journée ($T = S^- - S^+$).

On considère ensuite les indices d'indexation suivants :

- i : index des interventions ($1 \leq i \leq n$).
- j : index des boîtes de DM ($1 \leq j \leq n$).
- k : index les jours de la semaine ($1 \leq k \leq 5$).

Ce qui nous permet d'introduire les variables du problème :

variables principales :

- $x_{i,j,k}$: variable binaire égale à 1 si et seulement si l'intervention i utilise la boîte de DM j pendant le jour k .

- y_j : variable binaire égale à 1 si et seulement si la boîte de DM j est utilisée au moins une fois dans la semaine.

variables secondaires :

- t_i^+, t_i^- : respectivement date de début et de fin de la tâche i .
- a_i, b_i, c_i : variables binaires sans signification particulière, utilisées pour créer les contraintes temporelles.

Une fois les différents paramètres définis, on pose pour fonction objectif la minimisation du nombre de boîte de DM utilisée.

$$\text{Minimiser } \sum_{j=1}^n y_j \quad (4.4)$$

Cette fonction est sujette à un certain nombre de contraintes.

$$\sum_{i=1}^n x_{i,j,k} \times (P_i^+ + S_i^+) \leq T \times y_j \quad \forall j \in \llbracket 1, n \rrbracket \quad (4.5)$$

$$\forall k \in \llbracket 1, 5 \rrbracket$$

$$\sum_{j=1}^n \sum_{k=1}^5 x_{i,j,k} = 1 \quad \forall i \in \llbracket 1, n \rrbracket \quad (4.6)$$

$$x_{i_1,j,k} \times x_{i_2,j,k} \times (t_{i_1}^+ - t_{i_2}^-) \times (t_{i_1}^- - t_{i_2}^+) \geq 0 \quad \forall i_1 \in \llbracket 1, n \rrbracket$$

$$\forall i_2 \in \llbracket 1, n \rrbracket \quad (4.7)$$

$$\forall j \in \llbracket 1, n \rrbracket$$

$$\forall k \in \llbracket 1, 5 \rrbracket$$

$$P^+ + a_i \times T \leq t_i^+ \leq P^- + a_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \quad (4.8)$$

$$P^+ + a_i \times T \leq t_i + P_i^+ \leq P^- + a_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \quad (4.9)$$

$$S^+ + b_i \times T \leq t_i^+ + P_i^+ \leq S^- + b_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \quad (4.10)$$

$$S^+ + c_i \times T \leq t_i^- \leq S^- + c_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \quad (4.11)$$

$$(k-1) \times T \times x_{i,j,k} \leq t_i^+ \leq 5 \times T \times (1-k) \times x_{i,j,k} \quad \forall i \in \llbracket 1, n \rrbracket$$

$$\forall j \in \llbracket 1, n \rrbracket \quad (4.12)$$

$$\forall k \in \llbracket 1, 5 \rrbracket$$

Ce modèle cherche donc à minimiser le nombre total de boîtes de DM utilisées pour respecter un planning établi sur une semaine (équation 4.4). Les contraintes (4.5) et (4.6) représentent des contraintes classiques du problème de bin packing. Dans les faits,

elles assurent que chaque tâche soit bien assignée à un jour et une boîte de DM précise. La contrainte (4.7), quant à elle, empêche deux tâches utilisant une même boîte de se chevaucher temporellement parlant. Ensuite, les contraintes (4.8) à (4.11) forcent les sous-tâches à être placées de façon à respecter les horaires d'ouverture du bloc et du service de stérilisation. En effet, le début et la fin d'une intervention doivent se situer le même jour et pendant les horaires dédiés (idem pour la stérilisation des DM). Enfin, la contrainte (4.12) permet de définir le jour de chaque intervention. On notera que la l'équation (4.7) n'est pas linéaire, elle est notée ainsi pour plus de clarté. La version linéaire de ce modèle peut être trouvée en annexe (C). De plus, cette contrainte est non-triviale, c'est pourquoi, la section suivante sera consacrée à l'explication de cette dernière.

4.4.1/ LA CONTRAINTE DE "NON-CHEVAUCHEMENT"

Comme nous l'avons vu précédemment, la contrainte de non-chevauchement (équation 4.7) du modèle MILP permet de s'assurer que deux tâches nécessitant une même boîte de DM ne se chevauchent pas temporellement (c'est-à-dire qu'il n'existe aucune date pour laquelle les deux tâches ont lieu simultanément). Pour rappel, les informations temporelles connues sur une tâche sont :

- t_i^+ : sa date de début (l'intervention commence)
- t_i^- : sa date de fin (la boîte de DM est stérilisée et stockée)

Notons alors respectivement t_1^+ et t_1^- les dates de début de fin de la tâche 1 et t_2^+ et t_2^- celles de la tâche 2. La figure (4.5) représente alors tous les placements temporels relatifs possibles entre les deux tâches. Sur la figure, les cas de chevauchement sont marqués

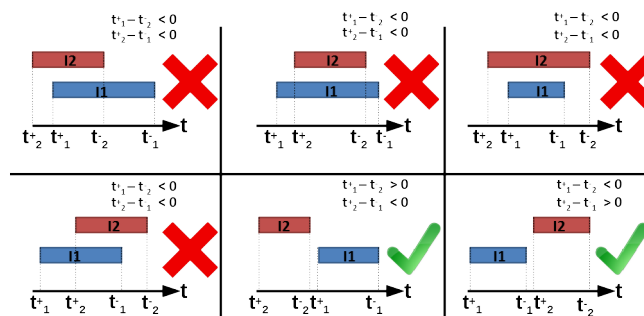


FIGURE 4.5 – Placements relatifs possibles de deux tâches

d'une croix rouge et les autres d'une coche verte. En étudiant le signe des expressions " $t_1^+ - t_2^-$ " et " $t_2^+ - t_1^-$ " pour chacune des configurations on remarque que les cas de non-chevauchement sont caractérisés par le fait que ces deux expressions sont de même signe (ce qui n'est pas le cas des autres possibilités de placement). Concrètement, cela

signifie que les deux tâches se chevauchent si et seulement si $(t_{i_1}^+ - t_{i_2}^-) \times (t_{i_1}^- - t_{i_2}^+) \geq 0$. Sachant que deux tâches sont contraintes de ne pas se chevaucher uniquement si elles utilisent la même boîte le même jour (dit autrement si et seulement si $x_{i_1,j,k} \times x_{i_2,j,k} = 1$) on obtient l'équation (4.7) qui nous donne la contrainte de non-chevauchement.

Une fois le modèle MILP défini, il est nécessaire de mettre en place des méthodes de résolution pour déterminer des solutions efficaces en un temps raisonnable.

4.5/ MÉTHODES DE RÉOLUTION

Grâce à la modélisation MILP proposée précédemment, il est possible de résoudre le problème de façon exacte à l'aide d'un solveur (voir définition (14)). Comme nous l'avons vu lors des chapitres précédents, cette solution peut facilement nécessiter un important temps de calcul. En effet, après quelques tests préliminaires il apparaît que lorsque l'on traite des instances de problèmes dérivées de cas concrets, l'obtention de la solution optimale peut demander plusieurs dizaines d'heures (voir plusieurs jours pour un grand hôpital). Or, les plannings d'interventions doivent être déterminés au plus vite pour permettre aux chirurgiens une plus grande marge de manœuvre.

Afin de régler ce problème, il est possible de mettre en place des méthodes de calcul alternatives pour obtenir de bons résultats sans certitude d'optimalité, mais dans un temps raisonnable.

4.5.1/ CHOIX DE LA MÉTHODE

Pour résoudre rapidement le problème présenté, nous proposons d'utiliser une métaheuristique (voir définition 15). Le choix de la métaheuristique et à la fois important et délicat, car il en existe de nombreuses (et même parfois en différentes versions) et chacune sera plus ou moins efficace dans un contexte donné. Afin de faire le choix le plus judicieux possible sans pour autant tester exhaustivement toutes les possibilités, nous avons sélectionné trois candidats potentiels. Tout d'abord la recherche Tabou, car cette dernière a déjà fait ses preuves pour résoudre ce genre de problème (voir [Hanset et al., 2007]). Ensuite, les algorithmes génétiques et l'optimisation par essaim particulière pour leur capacité à généralement mieux résoudre les problèmes possédant plusieurs solutions optimales.

La recherche tabou est une métaheuristique proposé par [Glover, 1989]. Elle consiste à choisir aléatoirement une solution et de tenter d'améliorer cette dernière en sectionnant une meilleure dans son voisinage. On répète alors ce processus pendant un nombre de cycles déterminé. De plus, pour éviter de rester bloqué dans un extremum local, on

détermine une liste de mouvements interdits (mouvements tabous), à chaque fois que l'on explore un voisinage d'une solution courante on ajoute ce mouvement à la liste. La liste possède une taille paramétrable de telle façon que lorsqu'elle est pleine, le dernier mouvement enregistré est supprimé selon une politique dite FIFO (**F**irst **I**n **F**irst **O**ut). Cette méthode permet d'éviter les retours en arrière et donc de rester bloqué en un point de l'espace de solution.

L'optimisation par essaim particulaire (abrégé en OEP) est une métaheuristique évolutionnaire inventé par Kennedy et Eberhart ([Kennedy, 2011, Shi et al., 1998a, Kennedy, 1995]), son principe est basé sur l'étude du comportement de groupes d'oiseaux ou de bancs de poissons. Des particules représentant des solutions sont créées dans l'espace de solutions et communiquent entre elles pour se déplacer. Les déplacements prennent en compte plusieurs facteurs comme la vitesse propre de la particule, l'emplacement de la meilleure solution qu'elle ait trouvée et celui de la meilleure solution trouvée par le groupe. Cette méthode a l'avantage de permettre de plus facilement repérer une ou plusieurs bonnes solutions.

Les algorithmes génétiques font eux aussi partis de la catégorie des métaheuristiques évolutionnaires, le concept fut inventée en 1960 par John Holland et popularisée par David Goldberg ([Golberg, 1989]). Comme nous l'avons vu précédemment, le principe est d'utiliser les mécanismes évolutionnaires d'une population d'individus (au sens darwinien du terme). Pour ce faire, une population composée de solutions initiales est tout d'abord créée. Cette population est considérée comme la première génération. Puis, génération après génération, de nouvelles solutions sont créées à partir des précédentes en effectuant des croisements et des mutations sur les solutions.

NB : Les algorithmes génétiques ne seront pas définis ici puisque cela a déjà été fait plus tôt dans ce mémoire (voir section 3.3.3).

Afin de faire un choix entre ces trois méthodes, il semble pertinent de mettre en place un algorithme simple pour chacune en utilisant des paramètres classiques. On définit alors 16 scénarii possibles représentant autant d'instances du problème. Les scénarii sont chacun constitués d'un ensemble d'interventions chirurgicales à ordonnancer allant de 5 (pour le premier) à 21 (pour le dernier). Chaque scénario est résolu de façon approchée grâce à chacune des trois méthodes citées précédemment (grâce à des algorithmes classiques) puis de façon exacte grâce au solveur Gurobi (voir section (2.3.3)) qui résout le modèle MILP proposé plus tôt. Les résultats obtenus par chaque méthode sont récapitulés par la table (4.1). Ces derniers sont donnés en nombre de boîtes nécessaires pour réaliser la meilleure solution. Il est important de noter que les résultats des trois méthodes approchées sont des moyennes calculées sur 10 exécutions limitées à 1h de calcul alors que ceux obtenus avec Gurobi correspondent à des solutions optimales.

TABLE 4.1 – Evaluation de la solution obtenue par plusieurs méthodes (en nombre moyen de boîtes de DM)

scénario	nb. interventions	AG	OEP	Tabou	MILP
1	6	2.00	2.00	2.00	2
2	7	2.00	2.00	3.00	2
3	8	2.00	2.00	3.00	2
4	9	2.58	2.00	3.00	2
5	10	2.98	2.00	3.16	2
6	11	3.00	3.00	4.00	3
7	12	3.00	3.00	4.00	3
8	13	3.16	3.00	4.00	3
9	14	3.92	3.00	4.94	3
10	15	4.00	3.20	5.78	3
11	16	4.00	4.00	5.80	4
12	17	4.20	4.00	5.76	4
13	18	4.78	4.00	5.00	4
14	19	4.96	4.00	6.02	4
15	20	5.04	4.96	6.60	4
16	21	5.46	5.00	9.14	5

En étudiant ces résultats préliminaires, on peut remarquer que l'optimisation par essai particulaire donne systématiquement de meilleurs résultats que les deux autres méthodes (en plus de donner des solutions très proches de l'optimale). Bien que, ces résultats ne soient pas exhaustifs, ils tendent à montrer que pour résoudre le problème posé dans ce chapitre, l'OEP semble être la méthode la plus indiquée. Pour produire la meilleure solution possible, il est donc nécessaire de se pencher sur les particularités de cette méthode ainsi que son paramétrage pour que ces derniers soient le plus possible en adéquation avec le problème traité.

4.5.2/ L'OPTIMISATION PAR ESSAI PARTICULAIRE

Il existe de nombreux articles traitant de l'OEP et notamment des façons d'améliorer son fonctionnement ([Robinson et al., 2004, Pandey et al., 2010, Omran et al., 2006]). [Clerc et al., 2002] ont proposé une méthode spécifique pour déterminer un bon paramétrage de l'OEP afin d'améliorer la convergence de la solution en étudiant l'impact de chaque paramètre. Il apparaît, en effet, que l'importance du choix des paramètres ne doit pas être négligée, car ces derniers influent grandement sur la qualité de la solution. En conséquence, certains groupes de recherches ont étudié l'impact de ces paramètres de façon mathématique ([Trelea, 2003]) et empirique ([Shi et al., 1998b]). C'est pourquoi cette section est dédiée aux détails de la modélisation de l'OEP ainsi qu'au choix des paramètres.

4.5.2.1/ MODÉLISATION

Implémenter un algorithme d'OEP implique de déterminer une représentation des particules qui exploreront l'ensemble des solutions. Le but de ce chapitre est de déterminer un planning d'interventions chirurgicales sur une semaine. Or, pour se faire, il faut uniquement déterminer la date de début de chaque intervention. En effet, la durée des tâches engendrées n'est pas une variable décisionnelle et dépend en grande partie de facteurs externes comme l'état du patient, les habitudes du chirurgien ou même le hasard. C'est pourquoi, pour la résolution du problème, les durées des interventions et des cycles de stérilisation sont fixes (et de préférence assez grande pour éviter les blocages en cas de retard). En conséquence, une particule (et donc une solution) sera uniquement constituée d'un ensemble de dates de début d'intervention. On définit alors les différents paramètres de l'OEP (certains seront précisément détaillés dans la suite du présent chapitre).

- m : nombre de particules générées par l'OEP.
- l : nombre de cycles accomplis par l'OEP.
- X_j^k : vecteur position de la particule j à l'étape k .
- V_j^k : vecteur vitesse de la particule j à l'étape k .
- L_j^k : meilleure solution trouvée par la particule j à l'étape k .
- G_j^k : meilleure solution trouvée par les particules voisines de la particule j à l'étape k .
- D : distance minimum pour que deux particules soient voisine.
- $d_{i,j}^k$: distance entre la particule i et j à l'étape k .
- ω : facteur d'inertie.
- ϕ_1 : facteur cognitif.
- ϕ_2 : facteur social.
- Ω_1 : espace de solution.
- Ω_2 : espace de calcul.
- r_1^k, r_2^k : vecteur de nombre réels aléatoire compris en 0 et 1.
- Σ : ensemble des scenarii à traiter.

Lors de l'initialisation de l'algorithme d'OEP, m particules sont créées et placées aléatoirement dans l'espace de solutions. Une particule j sera placée à la position X_j^0 et animée d'une vitesse V_j^0 elle aussi déterminée aléatoirement. Comme nous l'avons vu précédemment, la position d'une particule représente une solution du problème et une solution est complètement connue grâce aux dates de débuts de chaque intervention. Une position est un vecteur de taille n comme présenté dans l'équation (4.13). Par

commodité, X_j^k peut autant représenter une position qu'une particule en elle-même.

$$X_j^k = (t_1, t_2, \dots, t_n) \quad (4.13)$$

Les particules évoluent donc dans un espace de dimension n .

Une fois initialisée, l'OEP se découpe alors en un nombre déterminé de cycles (ou étapes), lors de chaque cycle les particules se déplacent suivant une vitesse qui dépend de leur vitesse lors du cycle précédent, de la position de la meilleure solution déjà visitée par la particule et de la position de la meilleure solution déjà visitée par les particules de son voisinage. Chacune de ces valeurs est modulée par des paramètres tels que montrés dans l'équation (4.14).

$$V_j^{k+1} = \omega V_j^k + \phi_1 r_1^k (L_j^k - X_j^k) + \phi_2 r_2^k (G_j^k - X_j^k) \quad (4.14)$$

Une fois sa nouvelle vitesse calculée, chaque particule se déplace. Ainsi sa nouvelle position est donnée par l'équation (4.15)

$$X_j^{k+1} = X_j^k + V_j^{k+1} \quad (4.15)$$

Une illustration du calcul de la vitesse est donnée par la figure (4.6). Notons que la vitesse ainsi calculée n'est en réalité pas homogène à une vitesse, il serait plus juste de la voir comme un écart entre deux positions (l'actuelle et la suivante). Néanmoins, dans la littérature on parle de vitesse pour des raisons de clarté et pour rendre l'analogie avec un essaim plus parlante.

Comme nous l'avons vu précédemment, la vitesse d'une particule est constituée de trois

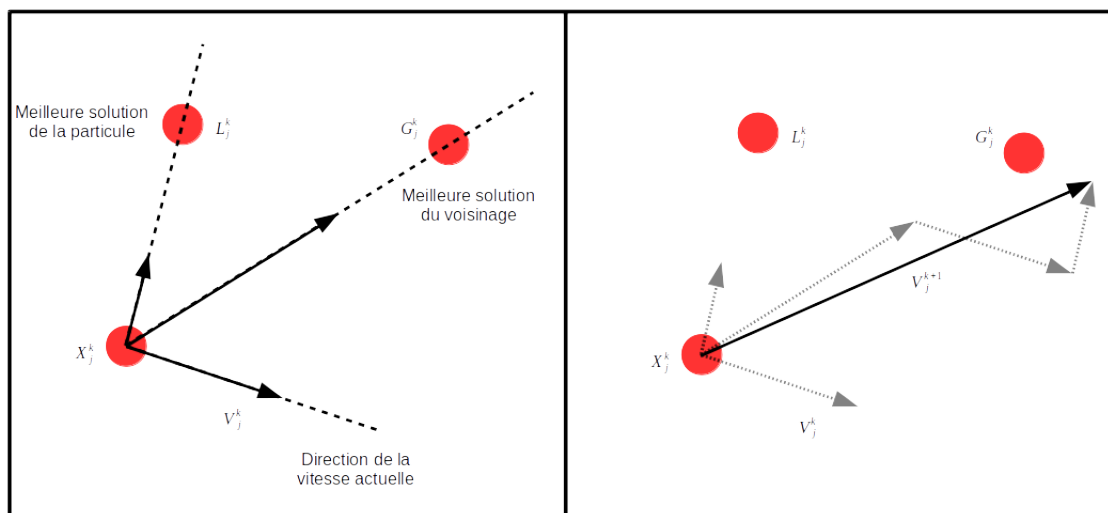


FIGURE 4.6 – Calcul de la vitesse d'une particule

parties.

Tout d'abord, il y a la vitesse d'inertie représentée par l'expression ωV_j^k . Cette composante simule le fait qu'une particule est emportée par sa propre vitesse et qu'elle ne peut pas effectuer de changement "brusque" de direction. ω est appelé le facteur d'inertie, plus il est grand et plus les particules seront soumises à leur inertie.

Ensuite, il y a la vitesse cognitive représentée par l'expression $\phi_1 r_1^k (L_j^k - X_j^k)$. Cette composante représente la capacité d'une particule à être attirée par la meilleure solution qu'elle ait rencontrée jusque-là en espérant trouver mieux dans son voisinage. L_j^k représente la position de cette meilleure solution donc plus cette position est éloignée plus la valeur de $(L_j^k - X_j^k)$ est importante et donc plus la particule est attirée par cette solution. ϕ_1 est appelé le facteur cognitif, plus il est élevé plus les particules sont attirées par leur meilleure solution. r_1^k est un vecteur de nombres aléatoires compris entre 0 et 1 dont le but est d'assurer la part chaotique du déplacement pour mieux explorer l'espace de solutions.

Enfin, il y a la vitesse sociale représentée par l'expression $\phi_2 r_2^k (G_j^k - X_j^k)$. Cette composante fonctionne globalement comme la précédente, mais représente, pour sa part, la capacité d'une particule à être attirée par la meilleure solution rencontrée par les particules de son voisinage. Le voisinage d'une particule est déterminé comme étant l'ensemble des particules qui se trouvent à une distance inférieure à D . Les positions des particules pouvant être représentées par des vecteurs, la distance utilisée est la distance euclidienne (voir équation (4.16)). ϕ_2 est appelé le facteur social, plus il est élevé plus les particules sont attirées par la meilleure solution de leur voisinage. r_2^k , quant à lui, est un vecteur construit selon le même principe et pour la même raison que r_1^k . Il est important de noter que dans certains cas, on simplifie le problème en considérant que toutes les particules sont voisines. Dans ce cas, G_j^k devient alors la meilleure solution rencontrée par l'essaim à l'étape k et s'écrit alors simplement G^k .

$$d_{i,j}^k = \sqrt{\sum_{\eta=1}^m (X_j^k(\eta) - X_i^k(\eta))^2} \quad (4.16)$$

Les cycles se succèdent et les particules se déplacent selon les règles précédemment fixées jusqu'à ce que la condition d'arrêt soit remplie. Dans le cas de la présente étude, cette condition est l'accomplissement d'un nombre déterminé de cycles ou le dépassement d'une limite de temps. Une fois l'algorithme terminé, la meilleure solution rencontrée par l'essaim (le groupe de particules) est considérée comme le résultat de l'OEP.

L'efficacité de l'OEP dépend de nombreux facteurs. Parmi ces derniers, on trouve la topologie de l'espace de solutions. Pour ce problème, cet espace contient tous les plannings réalisables, c'est-à-dire tous les vecteurs de taille n tels que chacune de leurs compo-

santes corresponde à une date de début possible. Cet espace est défini par l'équation (4.17)

$$\Omega_1 = \{(t_1, t_2, \dots, t_n) | \forall i \in \llbracket 1, n \rrbracket, 0 \leq t_i \leq 5 \times T, P \leq t_i \bmod (T) < P'\} \quad (4.17)$$

Ω_1 est un sous-ensemble discontinu de \mathbb{R}^n . Cette discontinuité, due aux horaires d'ouvertures du bloc opératoire, empêche les particules de se déplacer correctement. La figure (4.7) montre un exemple de représentation graphique d'un espace de solutions à 2 dimensions (donc avec seulement 2 interventions à planifier). Sur cette figure, l'espace

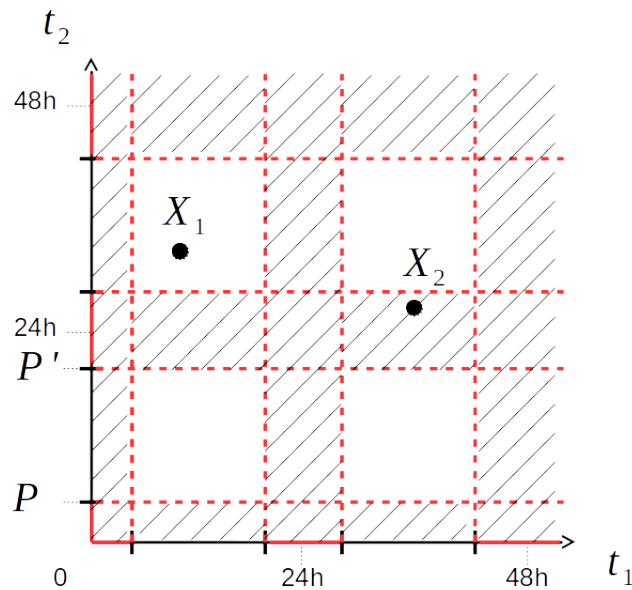


FIGURE 4.7 – Représentation de deux particules dans l'espace de solutions (à 2 dimensions)

de solutions correspond à la partie non hachurée du schéma. Le problème est que, lors d'une étape, lorsqu'une particule se déplace selon les règles données précédemment, il est possible qu'à l'étape suivante la position de cette dernière corresponde à un planning impossible (espace hachuré). En effet, la zone hachurée du schéma représente des solutions pour lesquelles au moins une intervention est programmée alors que le bloc opératoire est fermé. Par exemple, la particule X_1 correspond à une solution faisable ce qui n'est pas le cas de la particule X_2 puisque, pour cette dernière, l'horaire de la première intervention (t_1) se trouve en dehors des horaires d'ouverture du bloc opératoire.

Cette topographie particulière n'empêche pas d'utiliser l'OEP. En effet, chaque solution est évaluée, comme nous l'avons vu plus tôt, par le nombre de boîtes de DM nécessaire pour la réaliser. Dans ce cas, lorsqu'une particule se trouve dans une zone correspondant à un planning irréalisable son évaluation est considérée comme étant $n + 1$ (et donc forcément supérieur à une solution faisable). Le problème est que la présence de toutes ces zones "mortes" rend le calcul plus long et moins précis. De plus, plus il y a d'in-

terventions à ordonnancer, plus les chances qu'une particule se trouve à une position irréalisable est grande.

Pour pallier à ce problème, nous créons un nouvel espace de solutions ne contenant que des solutions valides. L'équation (4.18) définit cet espace que nous appelons espace de calcul et noterons Ω_2 .

$$\Omega_2 = \{(t_1, t_2, \dots, t_n) | \forall i \in \llbracket 1, n \rrbracket, 0 \leq t_i < 5 \times (P' - P)\} \quad (4.18)$$

On peut imaginer Ω_2 comme étant Ω_1 sans les zones "mortes". Cet espace ne représente plus de solutions concrètes, mais permet de faciliter les calculs et surtout de fluidifier le déplacement des particules. Lors de l'initialisation de l'OEP, il est alors possible de créer des particules aléatoirement réparties dans l'espace des solutions (en évitant les solutions irréalisables) puis de "transférer" ces dernières dans l'espace de calcul. Ensuite, les cycles de l'OEP se déroulent jusqu'à atteindre la condition d'arrêt. Dès lors, les particules sont "ramenées" dans l'espace des solutions dans lequel elles sont facilement interprétables comme des plannings.

Pour passer d'un espace à l'autre une transformation doit être appliquée aux particules. La fonction f définie par les équations (4.19) et (4.20) permet de passer une particule de l'espace de solutions vers l'espace de calcul. Une fois l'OEP terminée, la fonction f^{-1} permet de faire l'opération inverse, c'est à dire de ramener une particule depuis l'espace de calcul dans l'espace de solutions.

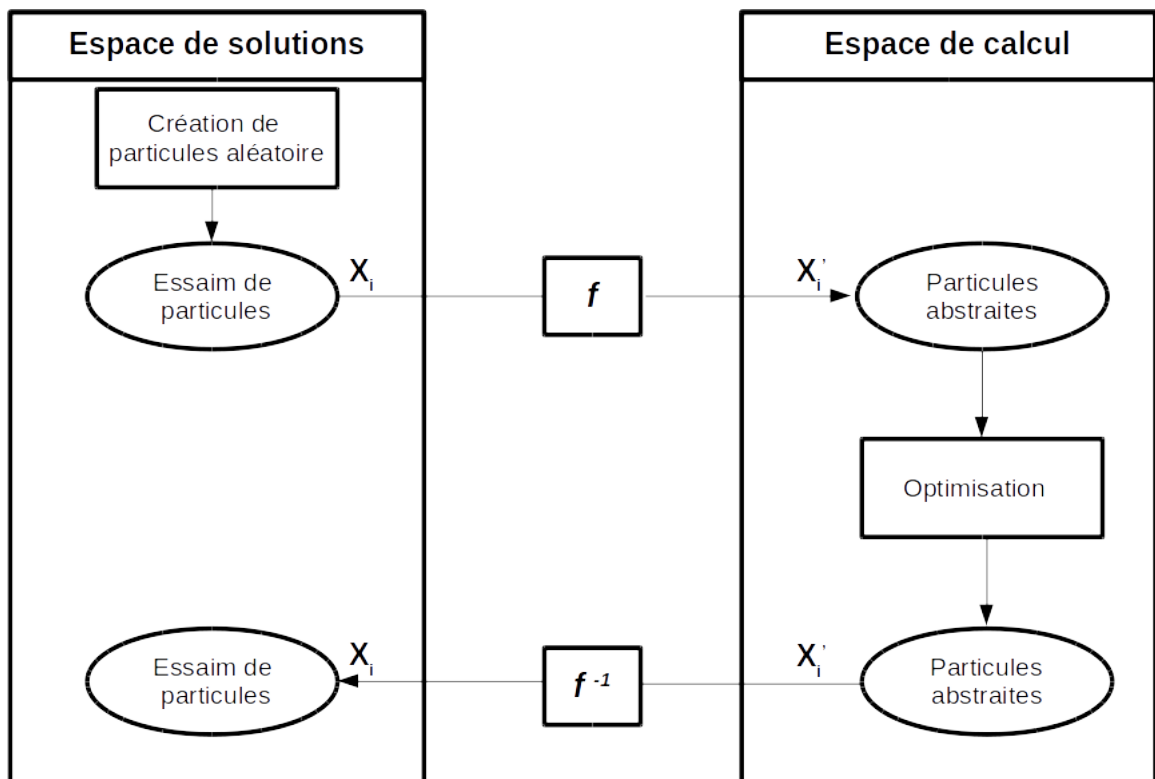
$$\begin{aligned} f : \quad \Omega_1 &\rightarrow \Omega_2 \\ (t_1^-, t_2^-, \dots, t_n^-) &\mapsto f((t_1^-, t_2^-, \dots, t_n^-)) = (t_1'^-, t_2'^-, \dots, t_n'^-) \end{aligned} \quad (4.19)$$

$$t_i = (t_i - P) - \left\lfloor \frac{t_i}{T} \right\rfloor \times (T + P - P') \quad (4.20)$$

$$\begin{aligned} f^{-1} : \quad \Omega_2 &\rightarrow \Omega_1 \\ (t_1'^-, t_2'^-, \dots, t_n'^-) &\mapsto f^{-1}((t_1'^-, t_2'^-, \dots, t_n'^-)) = (t_1^-, t_2^-, \dots, t_n^-) \end{aligned} \quad (4.21)$$

$$t_i^- = (t_i'^- + T^-) + (T + T^- - T^+) \times (t_i'^- \bmod (T^+ - T^-)) \quad (4.22)$$

La figure (4.8) donne une représentation graphique de l'utilisation de ces fonctions pour passer d'un espace à l'autre pendant le processus d'OEP. Une fois les mécanismes de fonctionnement de l'OEP correctement définis il est important de se pencher sur la ques-

FIGURE 4.8 – déroulement de l'OEP en utilisant Ω_1 et Ω_2

tion du choix des paramètres.

4.5.2.2/ CHOIX DES PARAMÈTRES

Comme nous l'avons vu dans la section précédente, une OEP demande à être paramétrée. Le paramétrage dépend fortement de la nature du problème à résoudre, en effet, un paramétrage optimal pour un problème pourrait donner de mauvais résultats pour en résoudre un autre, et ce bien qu'il existe certains jeux de paramètres reconnus comme étant généralement meilleurs ([Clerc et al., 2002]). Les articles de la littérature se penchent généralement sur les valeurs de ω , ϕ_1 et ϕ_2 , car il s'agit des paramètres qui définissent le comportement de l'essaim de particules pendant l'OEP. Afin de choisir au mieux ces valeurs, nous mettons en place un algorithme testant l'effet de différentes combinaisons du triplet (ω, ϕ_1, ϕ_2) sur les résultats obtenus (voir algorithme (6)). Pour chaque triplet, cet algorithme exécute plusieurs fois l'OEP via la fonction "EvaluationSolutionOEP(i, j, k, Σ)" qui lance une OEP avec les paramètres $(\omega, \phi_1, \phi_2) = (i, j, k)$ pour résoudre le problème Σ et renvoie l'évaluation de la meilleure solution. La variable "NbIter" permet de savoir combien de fois cette fonction sera appelée pour chaque triplet, puis, la moyenne des résultats est stockée dans le tableau de nombres réels à

trois dimension "F_s".

Les paramètres évalués avec cet algorithme sont les triplets (ω, ϕ_1, ϕ_2) appartenant à l'ensemble P (défini par l'équation (4.23))

$$P = P_\omega \times P_{\phi_1} \times P_{\phi_2} \quad (4.23)$$

$$P_\omega = \{\omega \in \mathbb{R} \mid \exists i \in \mathbb{N}, \omega = \omega_{start} + i \times \omega_{step}, \omega \leq \omega_{end}\} \quad (4.24)$$

$$P_{\phi_1} = \{\phi_1 \in \mathbb{R} \mid \exists i \in \mathbb{N}, \phi_1 = \phi_{1start} + i \times \phi_{1step}, \phi_1 \leq \phi_{1end}\} \quad (4.25)$$

$$P_{\phi_2} = \{\phi_2 \in \mathbb{R} \mid \exists i \in \mathbb{N}, \phi_2 = \phi_{2start} + i \times \phi_{2step}, \phi_2 \leq \phi_{2end}\} \quad (4.26)$$

Il est alors possible de créer un ensemble de scénarii $\Sigma = \{\sigma_1, \sigma_2, \dots, \sigma_v\}$. L'algorithme (6)

Inputs :

NbIter : nombre d'OEP lancées pour chaque triplets

ω_{start} ω_{step} ω_{end}

ϕ_{1start} ϕ_{1step} ϕ_{1end}

ϕ_{2start} ϕ_{2step} ϕ_{2end}

$F_s \leftarrow$ nouveau tableau de reals

$i \leftarrow \omega_{start}$

$j \leftarrow \phi_{1start}$

$k \leftarrow \phi_{2start}$

tant que $i < \omega_{end}$ **faire**

tant que $j < \phi_{1end}$ **faire**

tant que $k < \phi_{2end}$ **faire**

$F_s[i, j, k] \leftarrow 0$

pour $it \leftarrow 1$ **à** $NbIter$ **faire**

$F_s \leftarrow F_s + EvaluationSolutionOEP(i, j, k, \Sigma)$

fin

$F_s \leftarrow F_s / NbIter$ $k \leftarrow i + \phi_{2step}$

fin

$j \leftarrow i + \phi_{1step}$

fin

$i \leftarrow i + \omega_{step}$

fin

retourner F_s

Algorithme 6 : algorithme de détermination du meilleur triplet (ω, ϕ_1, ϕ_2)

permet de déterminer le meilleur triplet de paramètres (en moyenne) pour résoudre les scénarii proposés (voir équation (4.27)).

$$(\omega_{best}, \phi_{1best}, \phi_{2best}) = \arg \min_{\sigma \in \Sigma} \left(\sum F_\sigma(i, j, k) \right) \quad (4.27)$$

La précision de la solution est dépendante des scenarii sélectionnés ainsi que des valeurs de paramètres testées. Dans le cadre de cette étude, l'ensemble Σ est formé de scenarii inspirés de cas réel de planifications d'interventions de l'hôpital Emile Muller. Pour ce qui est des valeurs des paramètres testées, elles sont définies comme suit :

$$\omega_{start} = \phi_{1\ start} = \phi_{2\ start} = 0.2,$$

$$\omega_{step} = \phi_{1\ step} = \phi_{2\ step} = 0.2,$$

$$\omega_{end} = \phi_{1\ end} = \phi_{2\ end} = 2.0,$$

$$NbIter = 50$$

Une fois les calculs terminés avec ces paramétrages, on obtient les résultats donnés par

l'équation (4.28).

$$(\omega_{best}, \phi_{1\ best}, \phi_{2\ best}) = (0.2, 1.6, 1.6) \quad (4.28)$$

Ce triplet de paramètres est donc celui qui, empiriquement, donne les meilleurs résultats possibles (en moyenne) pour le problème traité ici. Il est intéressant de comparer ces valeurs à celles théoriquement attendues.

En effet, [Clerc et al., 2004] ont mené des études théoriques menant à définir les paramètres de l'OEP de façon à ce que $\phi_1 = \phi_2 = \phi$ et $\phi = \omega \times (2/0.97725)$. C'est pour cette raison que nous avons initialement prévu d'utiliser le triplet $(\omega, \phi_1, \phi_2) = (0.8, 1.64, 1.64)$. Or, les résultats empiriques obtenus nous mènent à deux constatations. Tout d'abord, les valeurs de ϕ_1 et ϕ_2 sont effectivement identiques et approximativement égale à 1.6. Par contre, les valeurs théoriques et pratiques de ω ne coïncident pas. D'un côté, la théorie nous amènerait (pour les mêmes valeurs de ϕ_1 et ϕ_2) à considérer une inertie proche de 1 alors que de l'autre la pratique semble nous diriger vers une valeur très faible de l'inertie ($\omega = 0.2$). Afin de bien comprendre cette contradiction, il est nécessaire de ne pas perdre de vue la signification du facteur d'inertie ω . Un facteur d'inertie élevé indique que les particules sont plus facilement emportées par leurs propres vitesses ce qui rend les changements de direction plus laborieux. Pour vulgariser, plus l'inertie est faible plus les particules peuvent rapidement changer de directions. En contrepartie, une faible valeur du paramètre ω rend plus difficile une exploration complète de l'espace de solutions en laissant d'avantage de zones non explorées. Or, comme nous l'avons vu précédemment, l'espace de solutions exploré ici contient plusieurs solutions optimales non voisines. La conséquence de cet état de fait est qu'il est plus facile pour les particules de converger vers une solution optimale même en explorant peu. Cette particularité permet d'expliquer les différences entre les valeurs théoriques attendues des paramètres (ω, ϕ_1, ϕ_2) et leurs valeurs réelles.

4.6/ RÉSULTATS EXPÉRIMENTAUX

Le but de cette section est de comparer les performances de l'algorithme d'OEP avec les valeurs classiques de paramètres avec celles obtenues par les paramètres déterminés dans la section précédente. De plus, le modèle MILP défini dans la section (4.4) permet d'évaluer ces méthodes en comparant les résultats obtenus avec les solutions optimales. Tout d'abord, on définit un ensemble Σ de scénarii représentatifs composés de différents nombres d'interventions chirurgicales à ordonnancer. Chaque scénario est inspiré de cas réel des archives de l'hôpital Emile Muller. Au total, Σ est composé de 32 scénarii comprenant de 6 à 37 interventions, le nombre d'interventions détermine la complexité de création du planning correspondant et donc le temps nécessaire pour l'obtenir. Une fois l'ensemble des scénarii déterminé, nous définissons les méthodes de résolution par OEP comme suit :

- OEP_1 est l'algorithme qui utilise des paramètres classiques sélectionnés à partir des recommandations de la littérature à savoir $(\omega, \phi_1, \phi_2) = (0.8, 1.64, 1.64)$.
- OEP_2 est l'algorithme qui utilise les paramètres déterminés dans la section précédente à savoir $(\omega, \phi_1, \phi_2) = (0.2, 1.6, 1.6)$

Pour chaque scénario de Σ le problème d'ordonnancement des interventions chirurgicales décrit dans ce chapitre est résolu avec OEP_1 , OEP_2 et le solveur GUROBI™. Ce dernier utilise le modèle MILP pour résoudre de problème et ainsi obtenir la solution optimale. Notons que la valeur du paramètre D (voir section (4.5.2.1)), qui détermine la distance maximale pour que deux particules soient considérées comme voisine, est ici fixée à $D = 5h$ (en effet la distance entre deux solutions se compte tout logiquement en heure). On pose ainsi les valeurs de paramètres suivantes (valable autant pour OEP_1 que OEP_2) :

- $D = 5h$
- $m = 20$
- $l = 1000$

Notons enfin que pour chaque résolution une limite de temps de 1 heure est imposée.

Les résultats obtenus sont résumés par la table (4.2). Cette table permet pour chaque scénario de connaître le nombre d'interventions associées ainsi que les performances des différents algorithmes décrits plus haut en termes de temps et de qualité de la solution. Les valeurs présentées sont, dans les cas des OEP, des moyennes effectuées sur 10 itérations. De plus, l'erreur permet de connaître l'écart entre l'évaluation de la solution trouvée et celle de la solution optimale (le détail est donné par l'équation (4.29)).

$$erreur = \frac{evaluation\ solution\ OEP - evaluation\ solution\ MILP}{evaluation\ solution\ MILP} \times 100 \quad (4.29)$$

TABLE 4.2 – Performances (en nombre de boîtes) des solutions obtenues avec chaque méthode

scenario	interv.	OEP_1			OEP_2			MILP	
		perf..	temps(s)	erreur	perf.	temps(s)	erreur	perf.	temps(s)
1	6	2.00	7.9	0.0 %	2.00	7.9	0.0 %	2	0.09
2	7	2.00	9.0	0.0 %	2.00	8.9	0.0 %	2	0.3
3	8	2.00	10.0	0.0 %	2.00	10.0	0.0 %	2	0.7
4	9	2.06	11.1	3.0 %	2.00	11.1	0.0 %	2	2.17
5	10	2.87	12.3	43.5 %	2.71	12.4	35.5 %	2	1.23
6	11	3.00	13.8	0.0 %	3.00	13.8	0.0 %	3	26.64
7	12	3.00	15.3	0.0 %	3.00	15.3	0.0 %	3	34.43
8	13	3.00	16.9	0.0 %	3.00	17.0	0.0 %	3	10.05
9	14	3.43	18.4	14.3 %	3.15	18.7	5.0%	3	84.36
10	15	3.98	20.4	32.7 %	3.84	20.7	28.0 %	3	80.1
11	16	4.00	22.7	0.0 %	4.00	23.0	0.0 %	4	113.1
12	17	4.00	24.6	0.0 %	4.00	25.1	0.0 %	4	36.47
13	18	4.16	26.2	4.0 %	4.00	26.3	0.0 %	4	152.39
14	19	4.64	28.4	16.0 %	4.41	28.9	10.25 %	4	210.98
15	20	5.0	30.2	25.0 %	4.99	30.7	24.75 %	4	58.66
16	21	5.01	32.3	0.02 %	5.00	33.0	0.0 %	5	143.28
17	22	5.13	34.3	2.6 %	5.02	35.0	0.4 %	5	78.35
18	23	5.72	36.3	14.4 %	5.39	37.5	7.8 %	5	495.25
19	24	5.96	38.2	19.2 %	5.92	39.3	18.4 %	5	558.51
20	25	6.00	40.9	20.0 %	6.00	41.8	20.0 %	5	591.50
21	26	6.38	43.0	1.7 %	6.02	44.0	0.3 %	6	808.80
22	27	6.80	45.4	6.3 %	6.09	46.7	1.5 %	6	820.41
23	28	7.00	47.6	13.3 %	6.52	49.0	8.7 %	6	2413.1
24	29	7.08	50.2	16.7 %	7.00	51.9	16.7 %	6	2842.1
25	30	7.35	52.5	22.5 %	7.01	54.3	18.3 %	6	3109.5
26	31	7.82	55.1	-	7.17	57.0	-	-	≥3600
27	32	7.95	58.0	-	7.63	59.8	-	-	≥3600
28	33	8.0	60.6	-	7.91	62.4	-	-	≥3600
29	34	8.26	63.3	-	8.00	65.7	-	-	≥3600
30	35	8.57	66.1	-	8.07	68.6	-	-	≥3600
31	36	8.87	69.1	-	8.24	71.4	-	-	≥3600
32	37	9.03	72.0	-	8.74	74.7	-	-	≥3600

En observant les résultats, on remarque que l'algorithme OEP_2 donne systématiquement des résultats au moins aussi bons que OEP_1 pour des temps de calcul équivalents, ce qui tend à valider le choix des paramètres discuté dans la section précédente. De plus, on remarque que les cas où les deux algorithmes donnent les mêmes résultats correspondent aux scénarii assez simples pour être résolu de façon exacte. Ces cas se produisent de façon régulière (quand le nombre d'interventions est divisible par le nombre de jours) ce qui explique l'évolution "en dents de scie" de la valeur de l'erreur. Plus globalement, les résultats de OEP_2 sont proches de la solution optimale (en terme de performances) tout

en demandant un temps de calcul plus que raisonnable.

4.7/ CONCLUSION

La synchronisation entre le bloc opératoire et le service de stérilisation est un point crucial pour l'amélioration de la logistique hospitalière, en effet les activités de chacune de ces entités ont un but commun, mais ne communiquent généralement pas assez. La planification des interventions est l'un des champs de réflexions majeures de ce domaine puisque ce sont les plannings qui régissent le rythme de travail des agents. Les méthodes présentées tout au long de ce chapitre ont donc pour but de mettre en place des planifications plus efficaces en tentant de minimiser le nombre de boîtes de DM nécessaires et donc de lisser la charge de travail du service de stérilisation. Le problème étant trop complexe pour être résolu de façon exacte pour des hôpitaux de grande taille (et notamment pour l'hôpital Emile Muller), nous nous sommes rabattus sur des méthodes approchées. Grâce à une étude préliminaire et des conseils recueillis dans la littérature, nous avons finalement choisi l'optimisation par essaim particulaire dont les principes ont déjà fait leurs preuves pour résoudre ce genre de problèmes. Une étude approfondie de l'influence des paramètres de cette méthode a permis de déterminer les meilleures valeurs possible pour ces derniers. Ces valeurs divergent légèrement de celles recommandées par la littérature pour des raisons de topographie de l'espace de solution. Enfin, des résultats expérimentaux ont permis de valider le choix de ces paramètres par rapport à ceux généralement conseillés tout en mettant en évidence que l'OEP ainsi mise en place permet d'obtenir de bons résultats en un temps raisonnable.

AFFECTATION DES DISPOSITIFS MÉDICAUX DANS UN CONTEXTE INTERSITE

Lors du chapitre précédent, nous avons présenté une méthode d'ordonnancement d'interventions chirurgicales au sein d'un hôpital. Cette méthode nécessite de définir une approximation du temps de transport des DM depuis le service de stérilisation jusqu'au bloc opératoire. Ce temps estimé est d'une précision suffisante pour créer un planning d'interventions pour un hôpital unique, mais devient insuffisant lorsque la planification concerne un groupement entier dépendant d'une pharmacie centrale (comme c'est le cas du GHRMSA).

Le présent chapitre présente une méthode dont le but est de mettre en place une logistique de transport de DM intersites pour respecter, au mieux, un planning donné. Cette méthode consiste à créer un premier planning d'interventions grâce à l'algorithme d'optimisation par essais particuliers détaillé lors du précédent chapitre. Puis, à partir de cette solution, nous proposons un algorithme permettant de construire un nouveau planning prenant cette fois en compte les allers et retours des transporteurs. Il est possible que la solution initiale ne permette pas d'obtenir un tel planning. Dans ce cas, il est nécessaire d'identifier les contraintes trop restrictives et de modifier la méthode de résolution en conséquence. Un modèle MILP d'une partie du problème complet basé sur les travaux de [Hammoudan et al., 2016] est alors détaillé, ce dernier étant trop complexe pour être résolu en un temps respectable, un algorithme de Branch & Bound est utilisé pour en déterminer une solution.

Enfin des exemples d'application de la méthode sont présentées ainsi que des résultats expérimentaux sur un ensemble représentatif de scénarii. Ces résultats montrent que la méthode détaillée, bien que complexe, est applicable pour déterminer des planifications de tailles modérées en un temps de calcul raisonnable.

5.1/ ÉTAT DE L'ART

La problématique traitée dans ce chapitre est liée à deux problèmes d'optimisation classiques. D'un côté le **Capacitated Lot Sizing Problem** (abrégé en CLSP) ou "problème de dimensionnement de lots avec capacité" en français et de l'autre le **Batch Scheduling Problem** (abrégé en BSP) ou "problème d'ordonnancement de lots" en français.

Le CLSP (voir définition (21)) est un problème dont le but est de prévoir la production d'articles en fonction de demandes clients. Il fut proposé, dans une version plus simple (connue sous le nom de LSP), par [Harris, 1913] qui cherchaient à réduire les coûts de production d'ateliers industriels soumis à des demandes continues. Le problème n'était alors pas précisément nommé, l'appellation courante du LSP telle qu'utilisée aujourd'hui par de nombreux chercheurs fut pour la première fois employée dans sa forme actuelle (et donc sous le nom que l'on connaît aujourd'hui) par [Van Hoesel et al., 1963] qui tentèrent de le résoudre en utilisant des considérations géométriques. Il est à noter qu'avant ça il existait déjà des études sur des problèmes similaires comme [Wagner et al., 1958] qui choisit de considérer des demandes dynamiques sur un horizon de temps limité, mais sans prendre en compte de contraintes. Ils résolurent le problème de façon exacte grâce à un algorithme de programmation dynamique qui servit de base pour de nombreux travaux de recherche. Au fil du temps, plusieurs variantes de ce problème virent le jour pour être appliquées à des situations de plus en plus précises. Parmi elles on trouve donc le CLSP ([Fleischmann, 1990]) qui prend en compte des contraintes de capacité, mais aussi le DLSP ([Salomon, 1991]) qui est une version discrète du problème dans laquelle les temps de préparation des machines n'est pas considéré comme continu ou encore le GLSP ([Drexel et al., 1997]) qui est une version plus générale qui n'autorise qu'un unique changement de gamme par période de temps.

Le BSP quant à lui consiste en l'ordonnancement de la production pour répondre à la demande d'un client (éventuellement en respectant les lots prévus par un LSP), la solution dépend de la quantité à produire ([Quadt, 2004]). En effet, les deux problèmes sont souvent utilisés conjointement. [Relvas et al., 2013] ont étudié le LSP et le BSP en utilisant ces derniers pour résoudre des problèmes de gestion d'inventaire dans le cadre d'entreprises de distribution de carburant.

Définition 21 : Problème multi-objets de dimensionnement de lots capacitif

Le problème de dimensionnement de lots avec capacité (ici présenté dans une version multi-objets) est un problème d'optimisation qui, étant donné une demande de produits (émise par un client), sépare lesdits produits en lots pour minimiser la somme des coûts de production (réglage des machines outils) et de stockage. Voici une formulation mathématique possible de ce problème ([Gicquel et al., 2008]).

Paramètres :

- N : nombre d'articles à produire.
- T : nombre de périodes.
- $D_{i,t}$: demande de l'article i sur la période t .
- P_t : temps de disponibilité de la ressource pendant la période t .
- $v_{i,t}$: temps de production d'une unité de l'article i lors de la période t .
- h_i : coût de stockage de l'article i par période.
- $c_{i,t}$: coût de préparation de la machine outils de l'article i à la période t ;

Variables :

- $I_{i,t}$: quantité d'article i stocké à la fin de la période t .
- $x_{i,t}$: quantité d'article i produit lors de la période t .
- $y_{i,t}$: variable binaire égale à 1 si et seulement si la machine de production de l'article i est mise en place pour produire à la période t .

La fonction objectif est donnée par :

$$\text{Minimiser } \sum_{i=1}^N \sum_{t=1}^T (h_{i,t} I_{i,t} + c_{i,t} y_{i,t}) \quad (5.1)$$

sujette aux contraintes :

$$I_{i,t} = I_{i,t-1} + x_{i,t} - D_{i,t} \quad \forall 1 \leq i \leq N \quad 1 \leq t \leq T \quad (5.2)$$

$$v_{i,t} x_{i,t} \leq P_t y_{i,t} \quad \forall 1 \leq i \leq N \quad 1 \leq t \leq T \quad (5.3)$$

$$\sum_{i=1}^N v_{i,t} x_{i,t} \leq P_t \quad \forall 1 \leq t \leq T \quad (5.4)$$

La fonction objectif (5.1) implique de trouver la solution qui minimise les coûts globaux (stockage et réglage machines). la contrainte (5.2) assure de correctement calculer la quantité d'articles stockés. La contrainte (5.3), quant à elle, s'assure que les articles ne peuvent être produit qu'à la condition que la machine correspondante soit prête. Enfin la contrainte (5.4) force le respect de la capacité de production.

Définition 22 : Problème d'ordonnement des lots

Le problème d'ordonnement des lots est un problème d'optimisation qui consiste à déterminer les périodes de production de produits et la quantité produite par période pour répondre à une demande en minimisant les coûts de production. Voici une formulation mathématique de ce problème ([Yusriski et al., 2015]).

Paramètres :

- n : nombre d'articles à produire.
- N : nombre de lots.
- s : temps de préparation de la machine outils produisant les articles
- t : durée de production d'un article
- d : date due commune

Variables :

- Q_i : taille du lot i
- B_i : date de début d'utilisation de la machine outils pour le lot i

La fonction objectif est donnée par :

$$\text{Minimiser } \sum_{i=1}^N \left(\sum_{j=1}^i (s + t \times Q_j) - s \right) \times Q_i \quad (5.5)$$

sujette à un certains nombre de contraintes :

$$\sum_{i=1}^N Q_i = n \quad (5.6)$$

$$\sum_{i=1}^N t \times Q_i + (N - 1) \times s \leq d \quad (5.7)$$

$$B_1 + t \times Q_1 = d \quad (5.8)$$

L'objectif (5.5) est donc de minimiser le temps d'utilisation de la machine. La contrainte (5.6) s'assure que la quantité d'article produit correspond bien à la demande, la contrainte (5.7), quant à elle, oblige toute les articles à être produit entre le temps origine et la date due commune. Enfin, la contrainte 5.8) oblige de prévoir la production de façon à livrer les lots à la date due (pas de stockage).

5.2/ SYSTÈME ÉTUDIÉ

Lors du chapitre précédent, nous avons vu comment créer un planning d'interventions chirurgicales en réduisant de plus possible l'utilisation de nouvelles boîtes de DM. Or, comme nous l'avons vu dans l'introduction de ce mémoire, l'hôpital de Mulhouse appartient à un groupement de centres de soin dont la PUI est amenée à devenir la pharmacie centrale. En d'autres termes, cela signifie que si des interventions sont programmées sur un site distant, les boîtes de DM nécessaires à leur réalisation devront être envoyées depuis la PUI grâce à des transporteurs afin être utilisées puis renvoyées pour être stérilisées (la figure (5.1) donne une illustration de cette situation).

La grande différence avec le cas étudié précédemment (avec un seul hôpital) réside dans

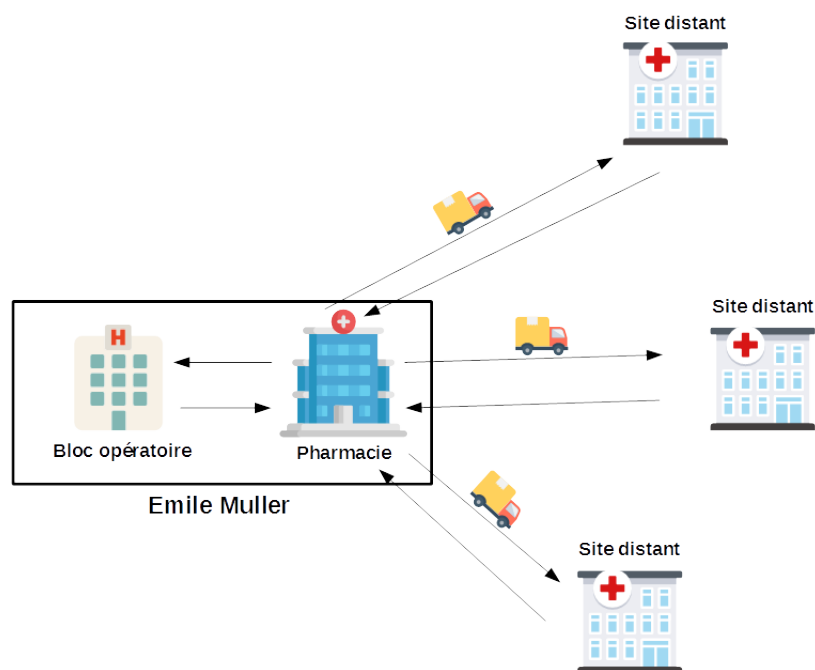


FIGURE 5.1 – livraison de DM dans un groupement hospitalier

l'utilisation de transporteurs intersites. En effet, il n'est plus possible d'envoyer les boîtes de DM au fur et à mesure que les armoires roulantes se remplissent. Afin de s'assurer de disposer d'un nombre suffisant de transporteurs et de minimiser les coûts de transport, il est nécessaire de séparer les boîtes en lots et de planifier l'envoi de ces lots par les transporteurs aux différents centres de soins. Le but de ce chapitre est donc de proposer une méthode de planification des transports de DM au sein du GHRMSA qui soit en adéquation avec les plannings d'interventions déterminés lors du chapitre précédent. Le schéma présenté par la figure (5.2) présente les différentes étapes envisagées pour y parvenir.

Les données d'entrée du problème sont un ensemble d'interventions chirurgicales, pro-

venant des différents sites du groupement, à ordonnancer. On utilise alors la méthode basée sur l'OEP (Optimisation par Essaim Particulaire) vu lors du chapitre précédent pour obtenir un planning qui ne prend pas en compte la logistique de transport à mettre en place. Une fois le planning obtenu, on tente de réunir les boîtes de DM par lots et de mettre en place une planification des envois vers les sites distants en respectant les contraintes des transporteurs. Cette planification s'inspire du LSP et BSP et constitue la problématique du présent chapitre.

Si la planification des interventions préalablement établie ne permet pas de livrer à temps les boîtes de DM, les contraintes trop restrictives sont alors étudiées pour obtenir une nouvelle planification proche de la précédente. On modifie ainsi le planning original jusqu'à obtenir une logistique de transport possible.

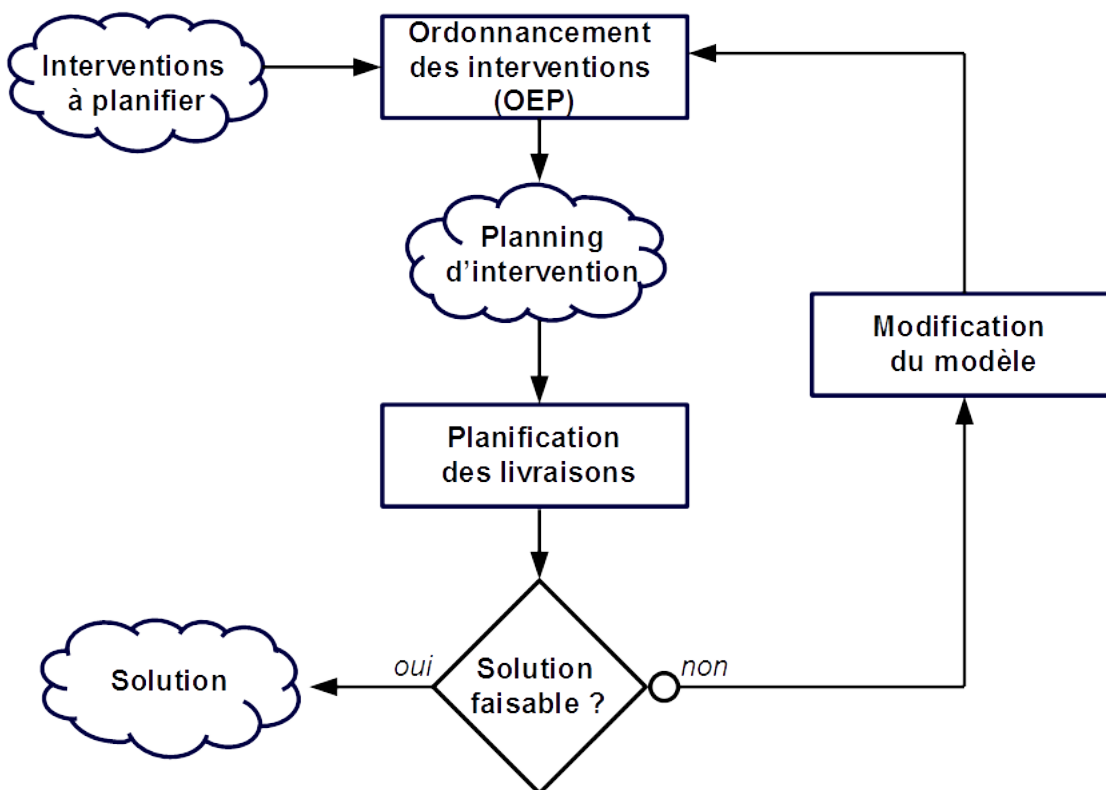


FIGURE 5.2 – Etapes du processus global de planification

5.3/ MODÉLISATION

Dans un premier temps, nous cherchons donc à obtenir une logistique de transport de boîtes de DM à partir de dates dues des interventions déterminées en amont. Pour ce faire, nous nous basons sur le travail de [Hammoudan et al., 2016] qui propose un

modèle mathématique pour résoudre un problème nommé le MCTLSDSP pour "Multiple Customers-Transporters Lot Sizing Delivery Scheduling Problem" que l'on peut traduire en français par "problème de dimensionnement et d'ordonnement de livraisons de lots dans un contexte multiclient et multitransporteur". Le modèle proposé définit un certain nombre de paramètres.

- m : nombre de sites à livrer
- n : nombre d'interventions
- p : nombre de transporteurs
- $J = \{1, 2, \dots, n\}$: ensemble des interventions
- $H = \{1, 2, \dots, m\}$: ensemble des sites à livrer
- $T = \{1, 2, \dots, p\}$: ensemble des transporteurs
- j : index interventions
- h : index des sites à livrer
- t : index des transporteurs
- k : index pour les lots (de boites de DM)
- n_h : nombre d'interventions par site
- d_j : date due de l'intervention j
- cl_j : site de déroulement de l'intervention j
- c_t : capacité du transporteur t
- $\tau_{b_k,h,t}$: temps nécessaire au transporteur t pour livrer un lot de taille b_k vers le site h et revenir
- $\eta_{b_k,h,t}$: coût engendré par le transporteur t pour livrer un lot de taille b_k vers le site h et revenir
- $\alpha(k)$: coût de livraison du lot k
- β_j : fonction de coût supplémentaire appliqué en cas de livraison en avance des DM nécessaire à l'intervention j

Ce qui nous permet d'introduire les variables.

principales

- $\delta_{j,k}^1 = 1$ si la boîte de l'intervention j appartient au lot k et 0 sinon
- $\delta_{k,h}^2 = 1$ si le lot k doit être livré au site h et 0 sinon
- $\delta_{k,h,t}^3 = 1$ si le lot k est livré au site h par le transporteur t et 0 sinon

secondaires

- C_j : date d'arrivée de la boîte de l'intervention j sur son site de destination
- B_k : date d'arrivée du lot k sur son site de destination
- b_k : taille du lot k (en nombre de boites)

- $y_k = 1$ si le lot k existe et n'est pas vide et 0 sinon
- u_h : nombre de lots livré au site h

le MCTLSDSP est alors défini par une fonction objectif (équation (5.9)).

$$\sum_{t \in T} \sum_{h=1}^m \sum_{k=1}^{u_h} \eta_{h,t} \times \delta_{k,h,t}^3 + \sum_{j=1}^n \beta_j (d_j - C_j) \quad (5.9)$$

Cette fonction est soumise à un certain nombre de contraintes.

$$\sum_{k=1}^n \delta_{j,k}^1 = 1 \quad 1 \leq j \leq n \quad (5.10)$$

$$\sum_{h=1}^m \sum_{t=1}^p \delta_{k,h,t}^3 \leq 1 \quad 1 \leq k \leq n \quad (5.11)$$

$$\sum_{h=1}^m \sum_{t=1}^p \delta_{k,cl_j,t}^3 \leq \delta_{j,k}^1 \quad 1 \leq j \leq n \quad (5.12)$$

$$1 \leq k \leq n$$

$$u_h = \sum_{k=1}^n \sum_{t=1}^p \delta_{k,h,t}^3 \quad 1 \leq h \leq m \quad (5.13)$$

$$y_k \leq y_{k+1} \quad 1 \leq k \leq n-1 \quad (5.14)$$

$$\sum_{j=1}^n \delta_{j,k}^1 \leq \sum_{h=1}^m \sum_{t=1}^p c_t \times \delta_{k,h,t}^3 \quad 1 \leq k \leq n \quad (5.15)$$

$$C_j \leq d_j \quad 1 \leq j \leq n \quad (5.16)$$

$$C_j = \sum_{k=1}^n B_k \times \delta_{j,k}^1 \quad 1 \leq j \leq n \quad (5.17)$$

$$B_{k2} - B_{k1} - \sum_{h=1}^m (\tau_{h,t} + M) \times (\delta_{k2,h,t}^3 + \delta_{k1,h,t}^3) \leq -2 \times M \quad 1 \leq k1 \leq n \quad (5.18)$$

$$1 \leq k2 \leq n$$

$$1 \leq t \leq p$$

$$k2 \leq k1$$

$$C_j \leq 0 \quad 1 \leq j \leq n \quad (5.19)$$

La fonction objectif du MCTLSDSP (équation (5.9)) assure de trouver la solution qui minimise les coûts globaux, à savoir les coûts de transport ($\eta_{h,t} \times \delta_{k,h,t}^3$) et les coûts liés aux livraisons trop en avance ($\beta_j (d_j - C_j)$). Cet objectif est sujet à un certain nombre de

contraintes. La contrainte (5.10) garantit que chaque boîte appartient à un unique lot. Les contraintes (5.11) et (5.12) assurent que chaque lot est bien assigné à un unique transporteur pour être livré à un unique site. La contrainte (5.13) permet de calculer la valeur de la variable u_h , à savoir le nombre de lots délivrés pour chaque site h . La contrainte (5.14) évite de créer des lots vides (c'est à dire, ne contenant aucune boîte). La contrainte (5.15) assure que la capacité maximale de chaque transporteur est bien respectée, c'est-à-dire que le nombre de boîtes contenues dans un transporteur n'excède jamais le maximum autorisé. La contrainte (5.16) force chaque boîte à être livrée au plus tard à sa date due. En effet, dans ce modèle l'avance de livraison est sanctionnée (grâce à la fonction β_j), mais le retard est interdit. La contrainte (5.17) permet de faire la corrélation entre la date d'arrivée de chaque boîte (C_j) et celles des lots (B_k). Pour que la solution soit réalisable, il faut s'assurer que les transporteurs soient bien disponibles pour effectuer chaque livraison, c'est pourquoi la contrainte (5.18) évite que les plages horaires d'utilisation d'un même transporteur se chevauchent (ce qui reviendrait à effectuer deux livraisons différentes au même moment). L'équation (5.19) permet de définir la positivité de la variable C_j (on note qu'il n'est pas nécessaire de faire de même pour B_k , car la positivité de cette variable est assurée par l'équation (5.17)).

Il est important de noter que le modèle proposé ici n'est pas linéaire, et ce à cause de l'équation (5.17). Cette particularité ne pose pas de problème ici, car l'auteur du modèle en question propose de résoudre le problème grâce à un algorithme de branch and bound (voir section suivante).

Si l'on se penche sur les paramètres d'entrée du modèle proposé (sites, transporteurs, boîtes...), on remarque que dans le cas de notre problème ils sont tous connus par avance à l'exception des dates dues de chaque intervention (d_j). Ces valeurs seront en effet déterminées grâce au planning mis en place par les méthodes du chapitre précédent (toujours en adéquation avec les étapes décrites par le schéma de la figure (5.2)).

5.4/ MÉTHODE DE RÉOLUTION

Il est théoriquement possible de définir un modèle complet du problème et de le linéariser pour le résoudre grâce à un solveur, mais cette méthode ne permettrait pas d'obtenir de solutions en un temps raisonnable pour des instances de tailles importantes. Afin de pallier à cet état de fait, le problème présenté ici se découpe donc en plusieurs étapes. Nous avons vu précédemment comment créer les planning d'interventions. Pour ce qui est de la résolution du MCTLSDSP [Hammoudan et al., 2016] propose un algorithme de B&B pour déterminer la solution optimale.

La borne inférieure d'une solution partielle définit pour ce B&B est donnée par l'équation

(5.20).

$$LB(w) = z(w) + \sum_{h=1}^m \left\lceil \frac{r_h(w)}{\max_{t \in T} (c_t)} \right\rceil \times \min_{t \in T} (\eta_{0,h,t}) \quad (5.20)$$

Dans cette équation, w représente la solution partielle à évaluer. Une solution partielle contient zéro ou plusieurs boîtes de DM déjà associées à des lots et des transporteurs. $z(w)$ correspond alors à l'évaluation de cette solution, c'est à dire aux coûts globaux (transport et pénalité d'avance de livraison) engendrés par la livraison des boîtes déjà prises en compte par la solution w . $r_h(w)$ correspond aux nombres de boîtes que la solution partielle w n'associe à aucun lot. La deuxième partie de l'équation représente donc la somme des coûts de transport des boîtes restantes pour chaque site en partant du principe qu'elles sont toutes livrées par le transporteur le moins cher rempli au maximum. L'utilisation du B&B permet alors de résoudre le problème d'ordonnement des transports de boîtes de DM en l'initialisant avec les dates dues obtenues grâce au planning d'interventions.

La difficulté est que ce planning ne prenant pas en compte le transport à la base, il est possible que le problème soit insoluble. Par exemple, si le nombre de transporteurs est trop faible pour livrer toutes les boîtes de DM en temps et en heure.

De plus, le MCTLSDSP étant originalement prévu pour être appliqué au domaine industriel, il ne prend pas en compte les retours. En effet, dans le cas de livraisons intragroupement hospitalier, il est nécessaire qu'une fois utilisées, les boîtes de DM soient renvoyées à la pharmacie pour y être stérilisées et utilisées à nouveau pour d'autres interventions.

5.4.1/ MODIFICATION DU MODÈLE

Comme nous l'avons vu précédemment, la méthode pour obtenir une solution se découpe en trois étapes, à savoir l'ordonnement des interventions, la planification de livraison et la relaxation des contraintes de transport. Nous avons d'ores et déjà étudié les deux premières, c'est pourquoi cette section est dédiée à la troisième. En effet, lorsqu'un planning d'interventions est déterminé, il est possible qu'aucune planification des livraisons ne permette de le respecter au vu des ressources disponibles (nombre de transporteurs, capacité, temps de trajet). Dans ce cas, il est nécessaire de modifier les résultats obtenus lors de la 1ère étape.

Pour ce faire, on commence par identifier, sur la solution finale, les plages horaires représentant un partie de planning irréalisable. On considère qu'une partie de planning est irréalisable si pour être respecté elle nécessite de posséder plus de boîtes de DM que le maximum disponible (des exemple d'identification de plage horaires critiques sont présentés dans la suite de ce chapitre).

L'identification des plages horaires critique permet de déterminer un ensemble d'interventions critiques (les interventions qui se trouve dans ces plages horaires). On peut alors relancer l'OEP en modifiant la fonction d'évaluation des solutions pour ajouter une nouvelle contrainte. En effet, les solutions contenant les interventions précédemment identifier planifiées de la même façon seront considéré comme infaisable. On réitère alors le processus en cumulant les nouvelles contraintes ajoutés lors de chaque cycle jusqu'à trouver une solution faisable.

5.4.2/ GESTION DES RETOURS

La méthode décrite précédemment permet donc d'obtenir un ordonnancement des livraisons de boites de DM. Malheureusement, cet ordonnancement ne prend pas en compte le trajet retour (du site distant vers la pharmacie) des boites pour qu'elles soient stérilisées. Il est donc nécessaire de mettre en place une méthode pour créer une solution complète (avec les retours) à partir d'une solution incomplète (sans les retours). Pour y parvenir sans complexifier le problème à outrance, nous proposons d'utiliser les mêmes lots pour la livraison des boites et leurs retours. Concrètement, cela signifie que si un ensemble de boites est livré par lot vers un site distant, ce même ensemble sera retourné à la pharmacie après utilisation. Pour ce faire, nous procédons en plusieurs étapes résumées par l'algorithme (7).

- **Étape 0 :**

Un nouveau planning vierge est créé sous la forme d'un ensemble vide de lots. Chaque lot est identifié par un ensemble de paramètres tels que :

- **dateDue** : date de livraison (arrivée sur site) ;
- **taille** : la taille du lot (le nombre de boite) ;
- **transporteur** : le transporteur associé pour la livraison ;
- **dest** : le site destinataire de la boite ;
- **finInt** : la date à partir de laquelle toutes les boites du lot sont disponibles pour être renvoyées ;
- **estUnretour** : permet de savoir s'il s'agit d'une livraison (**faux**) ou d'un retour (**vrai**)

Ensuite, pour chaque lot du planning original, les étapes 1 à 3 sont effectuées

- **Étape 1 :**

On vérifie que le lot courant peut être ajouté au nouveau planning, c'est le rôle de la fonction "*estAjoutable(lot, planning)*". Cette dernière renvoie "*vrai*" si et seulement si en ajoutant "*lot*" dans "*planning*" il est possible de faire les livraisons correspondantes. Dans le cas contraire, on augmente la date due jusqu'à pouvoir créer le

Inputs :*plan* : ensemble de lots de boites ordonnancés (sans les retours)*nbLots* : nombre de lots contenus dans l'ensemble *plan**i* : variable servant d'index

```

/* ----- */
/* etape 0 */
nouvPlan ← ∅
pour i ← 1 à nbLots faire
|
| /* ----- */
| /* etape 1 */
| tant que estAjoutable(plan[i], nouvPlan) = faux faire
| | plan[i].dateDue ← plan[i].dateDue + 1
| fin
|
| /* ----- */
| /* etape 2 */
| nouvPlan.ajouter(plan[i])
| retour ← plan[i]
| retour.dateDue ← retour.transporteur.tpsTrajetVersSite[retour.dest] + retour.finInt
| retour.estUnRetour ← vrai
|
| /* ----- */
| /* etape 3 */
| tant que estAjoutable(retour, nouvPlan) = faux faire
| | retour.dateDue ← retour.dateDue + 1
| fin
| nouvPlan.ajouter(retour)
fin
retourner nouvPlan

```

Algorithme 7 : algorithme de création de plannings avec retours

nouveau planning.

- **Étape 2** :

Une fois le lot placé dans le nouveau planning, il faut prendre en compte le retour correspondant, ce dernier étant placé au plus tôt. Pour ce faire on utilise la date à partir de laquelle toutes les boites du lot sont disponibles pour être renvoyées ("*finIntervention*"). Les lots de retour étant considérés d'un point de vue mathématique comme des livraisons vers un site distant (avec un transporteur vide à l'aller et chargé au retour) la nouvelle date due est donc $retour.transporteur.tpsTrajetVersSite[retour.dest] \times 2 + retour.dureeIntervention$ (la durée d'intervention + deux fois la durée de transport).

- **Étape 3** :

Une fois que les paramètres du lot de retour sont fixés, on vérifie que ce dernier

peut être ajouté au nouveau planning en procédant de façon analogue à l'étape 1. Une fois que la bonne date due est déterminée, le lot est ajouté au nouveau planning.

Grâce à cette méthode, il est possible de créer un ordonnancement global des lots de boîtes (avec retours) en un temps de calcul raisonnable en prenant en compte à la fois la problématique de disponibilité des boîtes de DM et celle des coûts de transport et de stockage. Il est théoriquement possible de se concentrer uniquement sur la minimisation du nombre de boîtes (donc sans prendre en compte les coûts engendrés), en tentant de résoudre le problème de planification des interventions et des transporteurs non pas l'un après l'autre, mais en même temps. Pour ce faire, il est possible de "fusionner" les deux modèles mathématiques vus précédemment en un seul. Malheureusement, des tests préliminaires ont montré que la résolution d'un tel modèle par un solveur ou par d'autres méthodes exactes (comme le B&B) demanderait bien trop de temps de calcul pour traiter des cas réels.

La figure 5.3 donne un exemple des résultats obtenus à chaque étape du processus de planification. Pour cet exemple, on imagine dix interventions de même nature réparties entre deux sites distants. Pour effectuer les livraisons, on dispose de deux transporteurs identiques d'une capacité de 4 boîtes et qui peuvent rejoindre chaque site en 5h. Le premier tableau résume les résultats de la méthode d'ordonnancement vu au chapitre précédent. Cette planification ne prend pas en compte la disponibilité et le coût des transports et part donc du principe que dès qu'une boîte est prête elle peut être expédiée vers le site correspondant. La figure 5.4 donne une représentation graphique du planning ainsi obtenue. Sur cette figure, chaque intervention est représentée par un rectangle dont la longueur permet de connaître le temps d'indisponibilité de la boîte (comme nous l'avons vu au chapitre précédent), la différence de temps avec les valeurs du tableau est due à la prise en compte du temps de transport et de stérilisation des boîtes. On note alors que pour cet exemple, les dix interventions peuvent être effectuées en utilisant seulement 4 boîtes.

On utilise ensuite les méthodes de planification de transports tirées des travaux de [Hammoudan et al., 2016] pour obtenir un planning prenant en compte le transport des boîtes de la pharmacie centrale vers les sites distants (mais pas leurs retours) en fonction des capacités des transporteurs et de leur disponibilité. Les résultats sont synthétisés dans le deuxième tableau de la figure (5.3). Pour que le planning soit complet, on utilise ensuite la méthode décrite par l'algorithme (7) permettant d'ajouter les retours et d'obtenir les résultats du troisième tableau. La figure (5.5) donne une représentation graphique du planning final ainsi obtenue (toujours avec prise en compte de l'étape de stérilisation). On

Ordonnancement des interventions

Site 1			Site 2		
#	début	durée	#	début	durée
1.1	Jeu di 8h15	1h30	2.1	Mardi 12h45	2h
1.2	Vendredi 11h45	1h30	2.2	Vendredi 8h15	1h30
1.3	Lundi 12h30	1h30	2.3	Jeu di 13h15	3h
1.4	Mercredi 8h45	1h	2.4	Mardi 13h15	4h
1.5	Vendredi 13h45	3h	2.5	Lundi 12h00	3h

Planification des livraisons

	boîtes	date livraison	Transp.
Lot 1	[2.1]	Mardi 12h45	Transporteur 2
Lot 2	[2.2/2.3]	Jeu di 13h15	Transporteur 2
Lot 3	[1.1/1.2/1.3]	Lundi 7h	Transporteur 1
Lot 4	[2.4/2.5]	Lundi 12h	Transporteur 1
Lot 5	[1.4/1.5]	Mercredi 8h45	Transporteur 2

Ajouts des retours

	boîtes	Date livraison	Date de retour
Lot 1	[2.1]	Mardi 12h45	Mardi 19h45
Lot 2	[2.2/2.3]	Jeu di 13h15	Vendredi 14h45
Lot 3	[1.1/1.2/1.3]	Lundi 7h	Vendredi 18h15
Lot 4	[2.4/2.5]	Lundi 12h	Mardi 22h15
Lot 5	[1.4/1.5]	Mercredi 8h45	Vendredi 21h45

FIGURE 5.3 – Exemple complet de planification

peut observer qu'une fois le processus achevé, le nombre de boîtes de DM nécessaires pour respecter le nouveau planning passe de 4 à 7. Cette augmentation est bien sûr due à la prise en charge des boîtes par les transporteurs disponibles. Une fois le processus terminé, il est possible d'établir le coût global du planning obtenu en considérant la somme des coûts de transport, des pénalités d'avances et des coûts associés au nombre de boîtes utilisées.

De plus, il est important de noter que le choix des paramètres du problème, notamment en termes de coûts, influe significativement sur le résultat. En effet, en définissant des coûts de transport ainsi que des pénalités d'avance (β_j) basses il est possible de

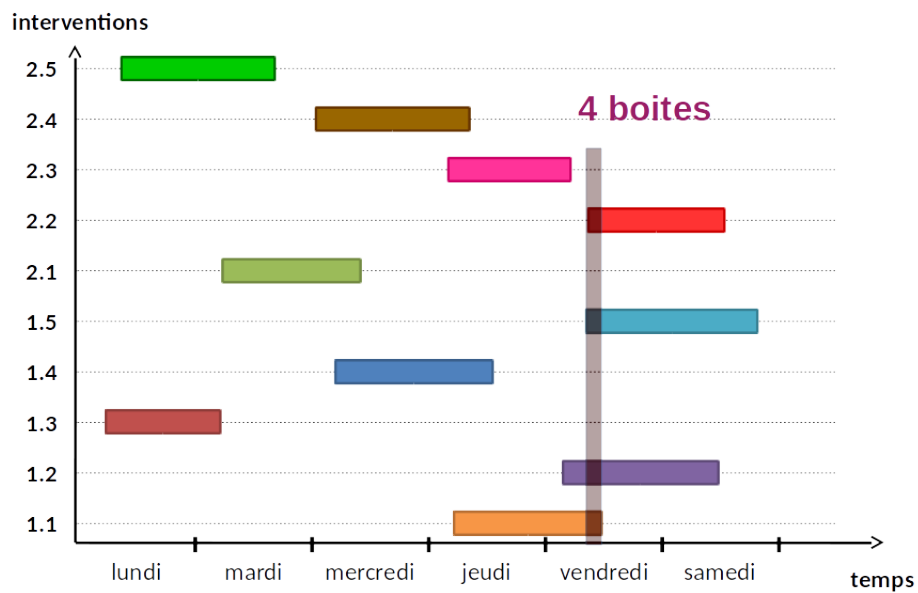


FIGURE 5.4 – planning de l'étape 1 de la figure 5.3

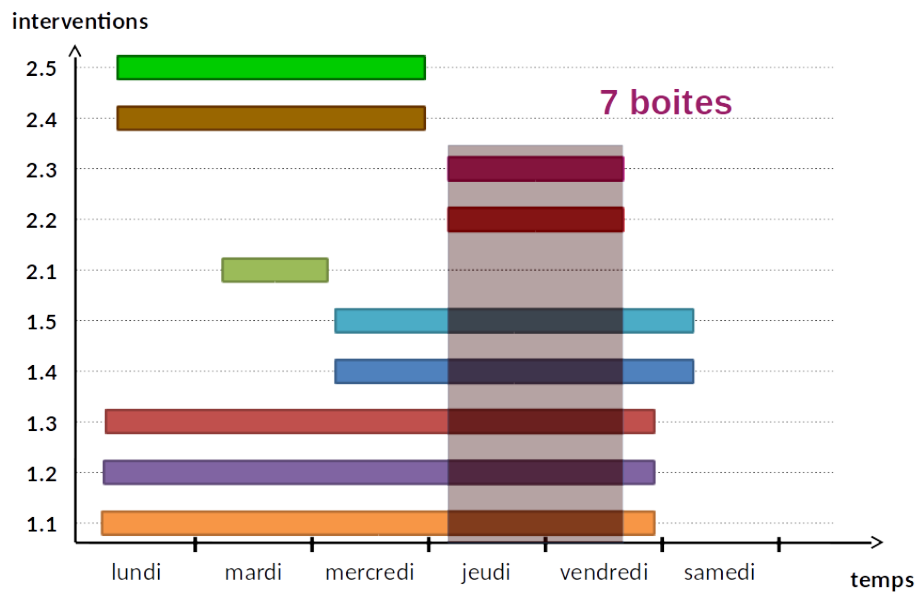


FIGURE 5.5 – planning de l'étape 3 de la figure 5.3

diminuer le nombre de boites nécessaires, mais d'augmenter le coût réel de l'opération. C'est pourquoi les paramètres relatifs aux coûts doivent être choisis avec minutie et de préférence par l'ensemble des acteurs décisionnels impliqués dans la logistique du groupement concerné.

5.4.3/ HEURISTIQUE DÉDIÉE

Afin de pouvoir comparer les performances de la méthode présentée dans ce chapitre, nous proposons une heuristique dédiée permettant de créer des plannings d'interventions de façon intuitive. Il faut, pour ce faire, utiliser les dates dues déterminées par l'OEP présentée lors du chapitre précédent. L'objectif est alors de livrer chaque boîte de DM une à une (une boîte par transporteur) de façon à les livrer juste à temps (sans stockage). Si cela est impossible, la livraison de ladite boîte est retardée d'autant de temps que nécessaire. Cette heuristique permet d'obtenir une planification de façon quasi-instantanée sans chercher à créer des lots de boîtes et ce quelque soit le nombre d'interventions à traiter.

5.4.4/ RÉSULTATS EXPÉRIMENTAUX

Afin de mettre en évidence l'efficacité de la méthode décrite dans ce chapitre, il est nécessaire de tester cette dernière dans différents cas de figure. Pour ce faire, on définit un groupement représentant le GHRMSA pour mener les expérimentations (voir figure (5.6)). Ce groupement est constitué de trois sites à pourvoir en boîtes de DM. Le site 1 correspond à l'hôpital auquel appartient la pharmacie centrale du groupement (l'hôpital de Mulhouse), aucun transporteur n'est donc nécessaire pour l'approvisionner. Les deux autres (site 2 et site 3) correspondent aux sites distants, c'est à dire, des établissements de santé (hôpitaux, cliniques...) appartenant au groupement et pratiquant des interventions chirurgicales nécessitant un approvisionnement en boîtes de DM. Dans le cas du GRHSMA, nous considérerons qu'il s'agit des sites de Colmar et de Saint-Louis. Pour pratiquer ces approvisionnements, un nombre variable de transporteurs est disponible. Le trajet vers de site 2 demande 2h à un transporteur (aller et retour) et 1h pour le site 3. De plus, chaque transporteur ne peut transporter plus de 5 boîtes par trajet.

Les plannings des transports sont alors déterminés en utilisant les paramètres suivants :

- $m = 3$
- $c_t = 5$ ($\forall 1 \leq t \leq p$)
- $\tau_{b_k, h, t}$
 - $\tau_{b_k, 0, t} = 0\text{h}$ ($\forall 1 \leq t \leq p, \forall 1 \leq b_k \leq c_t$)
 - $\tau_{b_k, 1, t} = 2\text{h}$ ($\forall 1 \leq t \leq p, \forall 1 \leq b_k \leq c_t$)
 - $\tau_{b_k, 2, t} = 1\text{h}$ ($\forall 1 \leq t \leq p, \forall 1 \leq b_k \leq c_t$)
- $\eta_{b_k, h, t}$
 - $\eta_{b_k, 0, t} = 0$ ($\forall 1 \leq t \leq p, \forall 1 \leq b_k \leq c_t$)
 - $\eta_{b_k, 1, t} = 200$ ($\forall 1 \leq t \leq p, \forall 1 \leq b_k \leq c_t$)
 - $\eta_{b_k, 2, t} = 100$ ($\forall 1 \leq t \leq p, \forall 1 \leq b_k \leq c_t$)

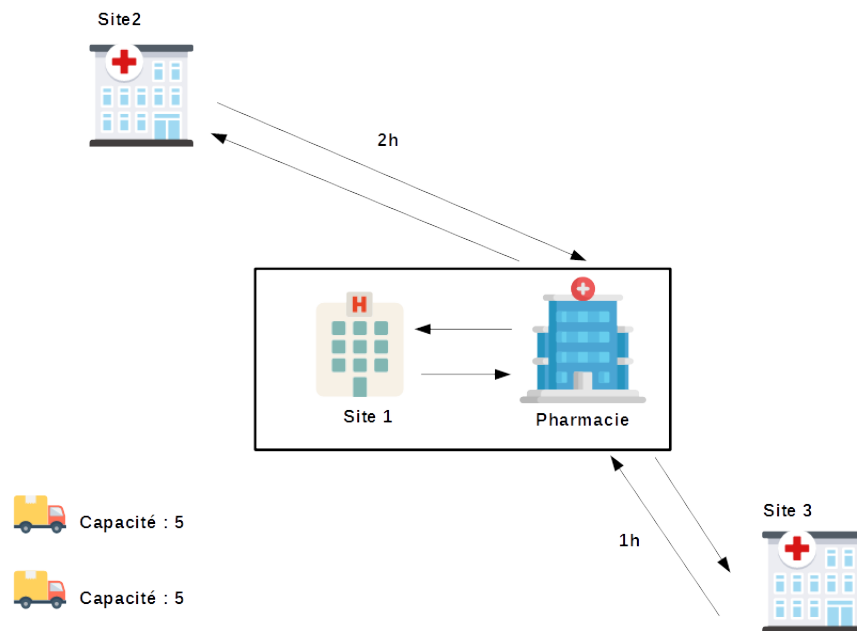


FIGURE 5.6 – disposition des sites du groupement pour l'expérimentation

- $\beta_j = 1$ (par seconde)

NB : Pour ce qui est du paramétrage de l'OEP, il est identique à celui proposé dans le chapitre précédent.

Les tests sont menés pour différentes valeurs de nombre d'intervention (n) et de nombre de transporteurs (p) afin d'obtenir des résultats couvrant un large panel de situations.

- n varie de 5 à 15
- p varie de 2 à 4

On définit un scénario comme étant un ensemble d'interventions prévues sur différents sites du groupement hospitalier et dont les boîtes de DM correspondantes doivent être livrées grâce à un ensemble variable de transporteurs. On cherche alors à définir un planning de livraison. Comme chaque scénario se définit pour des valeurs différentes du couple (n,p) , on en dénombre un total de 30. Notons que les durées des interventions sont calculés sur les durées réels des interventions pratiquées par le GRHMSA.

Il est important de noter que pour des questions de simplicité et de clarté des résultats, les transporteurs partagent toujours les mêmes caractéristiques (capacité, temps et coût de transport...) et ce quelque soit leur nombre. De plus, pour mener à bien chaque test, un ensemble de 20 interventions, dont les caractéristiques (durée et site de destination) sont choisies aléatoirement, est créé. Chaque scénario utilise alors tout ou partie de cet ensemble afin que les résultats puissent être comparables. Notons que les durées en question sont toujours situées entre 30min à 6h.

Les résultats de l'application de la méthode proposée sur les différents scénarii sont

synthétisés par la table (5.1). Pour chaque scénario, 10 tests sont effectués afin d'obtenir des moyennes représentatives. Les tables présentent alors les performances moyennes des 10 meilleures solutions trouvées avec la méthode proposée qu'on nous nommons MCTLSP+ ainsi que celle de la meilleure solution parmi les 10. Ces performances sont comparées à celles de l'heuristique dédié détaillé plus tôt afin de calculer l'écart de performance moyen entre ces deux méthodes. Le temps de calcul moyen nécessaire est aussi précisé dans chaque cas. les resultat sont exprimé en terme de coût global (transport, pénalité d'avance et coût d'une boite) en considérant que l'utilisation d'une boite reviens à un coût de 500.

Ces résultats expérimentaux nous montrent que plus le nombre de transporteurs est faible, plus la planification des interventions est complexe à déterminer. De plus la performances se dégradent avec le nombre d'interventions considérées, il semble d'ailleurs impossible de planifier plus d'une quinzaine d'interventions d'un même type en un temps raisonnable. Les temps de calcul élevés obtenue pour les plannings de plus de 10 interventions reflètent la complexité du problème à traiter. Notons qu'il est envisageable de n'utiliser qu'une partie des méthodes proposées. Par exemple, en déterminant un planning ne prenant pas en compte toute les interventions et de le compléter ou encore de diviser en deux groupes les transporteurs pour effectuer la procédure deux fois ou même de l'appliquer à des plannings fictifs pour estimer l'augmentation ou la diminution du nombre de boites induit par un changement. Le résultat obtenu peu aussi être utiliser comme base par le comité de planification pour être modifier à volonté.

Afin d'obtenir des plannings complets pour tout les types d'interventions, il est possible d'utiliser la méthode décrite dans ce chapitre plusieurs fois. Il faut alors créer les plannings préliminaires (sans les transports) pour chaque type d'intervention grâce à une succession d'OEP. Puis, de fusionner tout les plannings ainsi obtenue pour créer un unique planning auquel on ajoute les transports grâce à la procédure détaillé précédemment.

5.5/ CONCLUSION

Le chapitre précédent présentait une méthode de planification des interventions chirurgicales minimisant le nombre de boites de DM utilisé sans prise en compte du transport grâce à un algorithme d'OEP. Le présent chapitre propose une extension de cette méthode afin d'ajouter une logistique de transport pour livrer les boites dans les différents

services des différents sites d'un groupement hospitalier.

La méthode présentée consiste à améliorer les solutions primaires (sans le transport) grâce à une formulation MILP du MCTLSDSP. À l'aide d'un solveur exacte, il devient alors possible de générer un planning de livraison minimisant les coûts en fonction d'un certain nombre de paramètres (tel que le nombre d'interventions, la distance de chaque site, les caractéristiques des transporteurs, etc.). Malheureusement, l'utilisation de cette méthode n'est pas suffisante puisqu'elle ne prend pas en compte les livraisons retours. De plus, les différents tests ont montré qu'elle nécessitait un temps de calcul important. Pour pallier ce problème, nous avons mis en place une heuristique permettant, à partir d'un planning de livraison sans retour, de déterminer un planning de livraison complet (donc avec les retours). Le résultat global de ce chapitre est donc une méthode de mise en place de planning complet d'interventions avec prise en compte des livraisons de boîtes de DM et des retours. Elle se décompose en trois étapes : un algorithme d'OEP pour planifier les interventions, un solveur pour mettre en place les livraisons et une heuristique dédiée pour ajouter les retours. On cherche alors tout d'abord à minimiser le nombre de boîtes de DM nécessaire puis à minimiser les différents coûts liés au transport. Le choix de cette décomposition permet d'éviter d'utiliser une méthode exacte trop gourmande en terme de temps de calcul tout en à exploiter les meilleurs aspects de chaque méthode.

Les résultats expérimentaux ont montré que la méthode permet d'obtenir des solutions valides en un temps raisonnable pour une dizaine d'interventions de mêmes types, ces résultats reflètent à nouveau la complexité du problème traité dans ce chapitre. Afin d'obtenir un planning complet, il est alors possible d'utiliser l'algorithme d'OEP pour chaque type d'interventions et de réunir les plannings ainsi formés pour leur appliquer le MCTLSDSP et l'heuristique de détermination des retours créant ainsi un planning complet. Il est important de noter que la mise en place d'une telle procédure implique un certain nombre d'aménagements (surtout d'un point de vue informatique) et ne peut prendre en compte certaines particularités du fonctionnement du groupement dont une liste exhaustive ne saurait être dressée (indisponibilité exceptionnelle d'un agent, manières de faire différente d'un site à un autre, dispositions particulières pour un patient, etc.). Pour autant les solutions obtenues peuvent servir de base pour créer un planning respectant ces particularités ou encore pour évaluer les investissements nécessaires en terme de boîtes de DM lors d'une modification à venir du fonctionnement du groupement.

TABLE 5.1 – évaluation (en coût) des solutions obtenue avec 2,3 et 4 transporteurs

Transp.	Nombre inter.	Heuristique	MCTLSP+		Ecart	Tps. calcul moyen MCTLSP+
			moyenne	meilleur		
2	4	1720	1720	1600	0.00%	3.50 s.
	5	5361	5271	2500	1.00%	4.25 s.
	6	7754	6912	2500	11.0%	5.45 s.
	7	7662	5709	2500	25.5%	6.50 s.
	8	19061	16327	9000	14.3%	12.5 s.
	9	19866	16526	9500	16.9%	25.4 s.
	10	25156	18899	14400	24.8%	6.50 h.
	11	25820	19244	14400	25.4%	35.7 h.
	12	38845	28674	24300	26.2%	82.4 h.
	13	-	>150h	>150h	-	-
3	4	1700	1700	1600	0.00%	2.10 s.
	5	5320	5031	2000	5.40%	2.65 s.
	6	7465	6732	2500	9.81%	3.35 s.
	7	7345	5532	2500	25.7%	4.35 s.
	8	17654	15320	8500	13.2%	8.50 s.
	9	18453	15665	8000	15.1%	16.5 s.
	10	23654	16455	12300	25.3%	5.6 h.
	11	23854	17654	12400	26.0%	31.7 h.
	12	35644	26455	21300	25.8%	62.4 h.
	13	-	>150h	>150h	-	-
4	4	1750	1750	1600	0.00%	1.95 s.
	5	3456	2654	2000	23.2%	2.40 s.
	6	4568	3455	2000	24.3%	2.95 s.
	7	6125	5532	2500	9.68%	3.40 s.
	8	16235	13455	8200	17.2%	3.84 s.
	9	17654	13878	7900	21.3%	4.35 s.
	10	22154	15685	11500	29.2%	5.25 s.
	11	21548	16458	11900	23.6%	5.48 h.
	12	32548	22548	19500	30.7%	12.5 h.
	13	-	>150h	>150h	-	52.8 h.

CONCLUSION GÉNÉRALE ET PERSPECTIVES

Le but de cette thèse menée sur 3 ans était de fournir des outils logistiques à destination de groupements hospitaliers. Ce travail a pris place dans le cadre de la fusion de plusieurs centres de soins alsaciens centrés sur l'hôpital de Mulhouse, créant ainsi le **Groupe Hospitalier de la Région Mulhouse et Sud-Alsace (GHRMSA)**.

Le but de cette section est de revenir sur les éléments détaillés dans ce mémoire en portant un intérêt tout particulier aux méthodes développées et à leurs implications dans la logistique d'un groupement hospitalier. Nous verrons ensuite les perspectives envisagées pour ce travail en termes d'améliorations ou de nouvelles pistes à explorer.

BILAN

Comme nous l'avons vu lors de l'introduction de ce mémoire, ce travail de thèse était axé sur deux points distincts.

- I- **L'utilisation des particularités intrinsèques au milieu hospitalier pour établir un agencement optimal des stocks sur palettes de médicaments et dispositifs médicaux.**

- II- **La création d'une méthode de planification des interventions chirurgicales cherchant à lisser la charge de travail du service de stérilisation et la mise en place d'une logistique de transport associée.**

Pour rappel, le but des chapitres 2 et 3 était de répondre au premier point tandis que le deuxième point était couvert par les chapitres 4 et 5.

Dans un premier temps, nous avons envisagé la possibilité d'utiliser un véhicule lors des opérations de picking au sein du stock de dispositifs médicaux (DM). En effet, l'uti-

lisation d'un véhicule permet, en plus d'accélérer la vitesse des déplacements, d'utiliser des étagères afin de pouvoir stocker des produits sur plusieurs niveaux (et ce même à des hauteurs élevées) pour ainsi réduire la surface de stockage. Cet aspect est intéressant dans le cadre d'un groupement hospitalier puisque la pharmacie centrale est amenée à devoir stocker de nouvelles références (ou une plus importante quantité d'une même référence) pour chaque nouvelle inclusion d'un centre de soins. Nous avons alors identifié 3 contraintes principales liées à l'utilisation de véhicules et d'un stockage multi niveaux.

- 1** Le convoi ne peut pas faire demi-tour, car les allées ne sont pas assez larges pour que le conducteur puisse facilement manœuvrer.

- 2** Les véhicules utilisés dans les entrepôts peuvent être munis de barrières de sécurité pour protéger le conducteur de chutes d'objets éventuels. La barrière est positionnée d'un certain côté et le picking se fait de l'autre ce qui implique que le véhicule roule toujours du même côté des allées. Il existe aussi certains véhicules qui ne peuvent prendre en charge que les articles se trouvant d'un certain côté pour éviter d'avoir à manœuvrer.

- 3** Le convoi permet de transporter un certain nombre d'articles, mais sa capacité reste limitée. Si la quantité d'articles à collecter dépasse la capacité du véhicule, un système de tournée doit être mis en place pour effectuer le picking en plusieurs fois (ou avec plusieurs véhicules).

Le but étant alors de déterminer un plus court chemin (avec d'éventuelles tournées) à travers le stock pour collecter l'intégralité d'une liste de picking donnée en respectant ces contraintes. Nous avons alors montré que les contraintes 1 et 2 pouvaient être respectées grâce à une modélisation graphique astucieuse du problème faisant intervenir la notion de " carrefours " et de " nœuds orientés ". La troisième contrainte peut être respectée en considérant le problème comme un VRP avec capacité limitée du véhicule. En combinant ces différentes méthodes, nous sommes finalement parvenues à mettre en place une méthode exacte afin de déterminer le plus court chemin (avec tournées) que le véhicule doit parcourir pour collecter chaque référence en respectant les contraintes énoncées plus haut. Malheureusement cette méthode ne permet pas de traiter des listes de taille trop importante (plus d'une quarantaine de références). Pour pallier ce problème, nous nous sommes penchés sur la réalisation d'un algorithme génétique. L'utilisation de cette métaheuristique a permis de déterminer de bonnes solutions, et ce même pour des tailles de listes élevées.

Un tel processus peut être utilisé en condition réelle assez simplement. En effet, il n'est nul besoin de créer un logiciel dédié puisque les listes de picking d'un centre hospitalier restent sensiblement les mêmes au fil du temps. Il est donc possible de déterminer les différentes tournées pour chaque liste et d'ordonner directement les listes utilisées par les agents collecteurs. Le processus devra alors être réitéré à chaque changement (ou ajout) d'une liste de picking.

Le précédent travail nous a permis de mettre en évidence que les listes de picking sont peu modifiées au cours du temps contrairement à leurs homologues industriels (les bons de commande). De cette particularité est née l'idée d'organiser le stock de façon à naturellement minimiser les distances parcourues lors des opérations de collecte. Ce problème étant moins assimilable à un problème connu de la littérature, nous avons fait le choix de tester un certain nombre de méthodes pour le résoudre.

Tout d'abord, nous avons défini un modèle MILP du problème afin de résoudre ce dernier de façon exacte. Malheureusement, il s'est vite avéré que l'utilisation d'un solveur exacte ne permettrait pas d'obtenir de résultat en un temps raisonnable. C'est pourquoi nous avons par la suite défini une méthode de résolution par algorithme génétique et une autre basée sur des mécanismes de branch & bound (B&B).

De plus nous avons mis au point une heuristique dédiée nommée MFIF (pour **M**ost **F**requent **I**tem **F**irst) et dont le principe est de placer les références les plus utilisées le plus proche possible du dépôt. Enfin, dans le but de tirer parti des performances des métaheuristiques et de la rapidité du MFIF, nous avons défini des méthodes mixtes consistant à initialiser l'algorithme génétique et le B&B avec la solution initiale donnée par le MFIF.

Les résultats expérimentaux ont montré que même si les avantages du B&B semblent ne pas être significatifs dans ce contexte, l'algorithme génétique mixte, quant à lui, permet d'obtenir de très bons résultats en un temps de calcul raisonnable.

Ces méthodes de résolutions peuvent être utilisées facilement dans un contexte réel. En effet il suffit de déterminer le meilleur agencement du stock à un instant t puis de conserver ce dernier tant qu'aucune importante modification des listes de picking n'est constatée. Il pourrait éventuellement être intéressant de réitérer le processus tous les ans (ou sur n'importe quelle période de temps qui semble approprié).

Il est bien sûr possible d'utiliser les méthodes détaillées par les chapitres 2 et 3 conjointement ou non pour répondre au mieux aux besoins du groupement concerné.

La seconde partie de ce travail porte sur l'ordonnancement des interventions chirurgicales. Le but ici est de mettre en place une méthode capable de créer des plannings d'interventions qui minimisent le nombre de boîtes de DM total nécessaire. Le but est d'améliorer la sécurité en s'assurant d'avoir toujours du matériel disponible pour traiter les urgences ou les imprévus, mais aussi, de lisser la charge de travail du service de stérilisation.

Pour y parvenir, nous avons à nouveau défini un modèle MILP du problème basé sur le problème de bin-packing en y ajoutant des contraintes temporelles supplémentaires. Les résultats obtenus par la résolution par un solveur exact de ce modèle sont de plutôt bonne qualité, mais ne permettent pas de créer des plannings pour un très grand nombre d'interventions de même type. Il est ici important de noter que les performances obtenues sont déjà suffisantes pour la plupart des centres hospitaliers, mais dans le chapitre 5 nous réutilisons cette procédure pour l'incorporer à une autre procédure plus importante nous obligeant à réduire le temps de calcul autant que possible.

Pour ce faire nous avons choisi d'utiliser un algorithme d'Optimisation par **Essaim Particulaire** (EOP) après avoir comparé ses performances à celles d'un algorithme génétique et d'une recherche tabou. Nous avons alors cherché à obtenir le meilleur paramétrage possible de l'OEP dans le cadre de notre problème puisque les caractéristiques propres à ce dernier impliquaient d'utiliser des valeurs légèrement différentes de celles proposées dans la littérature.

Nous avons donc comparé les différents plannings obtenus avec la méthode exacte ainsi qu'avec l'OEP. Il apparaît que l'OEP donne des résultats satisfaisants bien que la valeur de l'écart entre les deux méthodes suive une évolution en dents de scie. Cette particularité est due au fait que le problème est plus complexe à résoudre pour les cas limites qui apparaissent de façon régulière.

L'utilisation concrète des méthodes décrites dans ce chapitre sont plus complexes que pour les chapitres précédents. En effet ici les plannings doivent être recalculés chaque semaine ce qui implique, dans le meilleur des cas, que l'intégralité des prescripteurs du groupement soient informatisés en conséquence pour qu'une fois toutes les prescriptions d'une période donnée recueillies, le planning correspondant puisse être créé. Il serait même envisageable de mettre en place un système automatique pour prévenir les patients de leur horaire de passage (via SMS par exemple). Il est aussi envisageable d'incorporer ces méthodes dans un progiciel permettant de visualiser plusieurs propositions de plannings et éventuellement de faire des modifications directement ou d'ajouter des contraintes précises liées au fonctionnement du groupement.

Il semble aussi possible dans une moindre mesure de déterminer un planning idéal chaque semaine pour le fournir au comité décisionnel qui pourra s'en servir de base pour établir le planning réel.

Il nous a fallu, pour finir, traiter d'un problème complexe. En effet les méthodes de planifications du chapitre précédent ne prennent pas en compte le transport (aller et retour) des boîtes de DM nécessaires pour pratiquer les interventions sur chaque site. Le problème de planification des transports étant très complexe, nous avons choisi de procéder en trois étapes. Tout d'abord, nous utilisons l'OEP défini lors du chapitre précédent pour obtenir un planning sans transport. Puis, nous utilisons un B&B pour résoudre le **Multiple Customers-Transporters Lot Sizing Delivery Scheduling Problem** (ou MCTLSDSP) à partir de cette solution incomplète pour obtenir un nouveau planning prenant en compte les livraisons de la pharmacie centrale vers les sites. Enfin nous avons défini une heuristique dédiée pour ajouter des retours à ce planning créant ainsi un planning complet. S'il apparaît que la solution finale n'est pas réalisable, l'OEP est relancée avec de nouvelles contraintes pour s'assurer de ne pas retrouver la même solution.

Ce problème est assez complexe et des résultats peuvent être obtenus pour des plannings ne dépassant pas une quinzaine d'interventions d'un même type. De plus, on note que plus le centre hospitalier dispose de transporteurs plus la solution est rapide à obtenir.

Cette méthode est plus délicate à utiliser que les autres. En effet elle implique une importante informatisation du groupement hospitalier concerné liant les prescripteurs, la pharmacie centrale et le pôle de gestion des transporteurs. Cela dit il est toujours possible, comme nous l'avons vu précédemment, de l'utiliser pour conseiller le comité décisionnel du groupement ou pour déterminer le nombre de boîtes de DM nécessaire pour maintenir le fonctionnement du groupement après un important changement structurel.

PERSPECTIVES

Ce travail est donc divisé en deux grandes parties dont chacune est riche en perspectives.

Pour ce qui est des méthodes d'amélioration de gestion du stock, elles sont bien évidemment complémentaires et gagneraient à être associées à des logiciels de modalisation de stocks. En effet, il s'avérerait intéressant de pouvoir déterminer l'agencement global d'un stock (par exemple dans un nouvel entrepôt) en prenant en compte les considérations apportées par ces méthodes. De plus, il serait possible d'imaginer un stockage non fixe dans lequel les emplacements des références varieraient en fonction des périodes de l'année ou des imprévus de certains services, bien que cela impliquerait à priori une informatisation importante du groupement.

Pour ce qui est de la partie planification, son utilisation est plus délicate. En effet, il apparaît clair que pour pouvoir exploiter le potentiel des méthodes détaillées dans les chapitres 4 et 5 il serait idéal de les inclure dans un progiciel de planification d'interventions qui serait utilisé par les différents sites du groupement. De plus, la création de plannings complets (avec transports) est une tâche ardue qui mériterait davantage de considérations pour tenter de déterminer des procédures alternatives en fonction des caractéristiques des différents groupements.

Enfin il me semble important de préciser une chose. S'il est vrai que les méthodes discutées au sein de ces pages peuvent apporter des améliorations certaines au fonctionnement d'un groupement hospitalier telles que le GHRMSA, leurs applications concrètes peuvent facilement s'avérer être une d'une difficulté tout autre. En effet, les mécanismes hiérarchiques et décisionnels des hôpitaux français sont des éléments complexes et bien souvent difficiles à ébranler. Lors de ces 3 ans de thèse, je me suis avant tout penché sur les moyens techniques à mettre en place pour répondre aux problématiques posées sans réellement prendre en compte ces différents aspects. Ces considérations humaines et organisationnelles sont loin d'être négligeables, mais relèvent avant tout d'une compétence managériale voir même d'une évolution des mentalités de travail et d'organisation pour certains cas.

NOTATIONS

CHAPITRE 2

- n : Nombre de références (et donc d'emplacements).
- $V = v_1, \dots, v_n$: ensemble des articles (nœud) à collecter.
- Q : Volume maximum qu'un convoi peut transporter.
- q_i : Volume de l'article v_i .
- $d_{i,j}$: Distance à parcourir pour aller de l'emplacement de l'article v_i à celui de l'article v_j .
- $x_{i,j,k}$: variable binaire égale à 1 si et seulement si lors de la tournée k l'agent doit aller directement de v_i vers v_j .
- $u_{i,k}$: L'ordre dans lequel l'article v_i sera collecté lors de la tournée k .
- n_k : nombre d'articles collectés lors de la tournée k .

CHAPITRE 3

- n : Nombre de références (et donc d'emplacements).
- m : Nombre de listes de dotation fournies par l'hôpital.
- C_t : Ensemble des objets contenus dans la liste t .
- $d_{k,l}$: Distance entre l'emplacement k et l'emplacement l .
- $x_{i,k}$: Variable binaire égale à 1 seulement si l'objet i se trouve à l'emplacement k .
- $l_{i,j,t}$: Variable binaire égale à 1 seulement si dans la liste t l'objet i est directement suivi par l'objet j .
- $u_{i,t}$: L'ordre de l'objet i dans la liste t .

CHAPITRE 4

MILP

- n : nombre d'interventions.
- P_i : durée de l'intervention i et du transport de la boîte correspondante.
- S_i : durée du processus de stérilisation complet de la boîte de l'intervention i .

- S : date ouverture du service de stérilisation.
- S' : date de fermeture du service de stérilisation.
- P : date d'ouverture du bloc opératoire.
- P' : date de fermeture du bloc opératoire.
- T : durée d'ouverture du service de stérilisation sur une journée ($T = S' - S$).

On considère ensuite les indices d'indexation suivant

- i : indice des interventions ($1 \leq i \leq n$).
- j : indice des boîtes de DM ($1 \leq j \leq n$).
- k : indice des jours de la semaine ($1 \leq k \leq 5$).

Ce qui nous permet d'introduire les variables.

principales :

- $x_{i,j,k}$: variable binaire égale à 1 si et seulement si l'intervention i utilise la boîte de DM j pendant le jour k .
- y_j : variable binaire égale à 1 si et seulement si la boîte de DM j est utilisée au moins une fois dans la semaine.

secondaires :

- t_i, t'_i : respectivement date de début et de fin de la tâche i .
- a_i, b_i, c_i : variables binaires sans signification particulière, utilisées pour créer les contraintes temporelles.

OEP

- m : nombre de particules générées par l'OEP.
- l : nombre de cycles accomplis par l'OEP.
- X_j^k : vecteur position de la particule j à l'étape k .
- V_j^k : vecteur vitesse de la particule j à l'étape k .
- L_j^k : meilleure solution trouvée par la particule j à l'étape k .
- G_j^k : meilleure solution trouvée par les particules voisines de la particule j à l'étape k .
- D : distance minimum pour que deux particules soient voisines.
- $d_{i,j}^k$: distance entre la particule i et j à l'étape k .
- ω : facteur d'inertie.
- ϕ_1 : facteur cognitif.
- ϕ_2 : facteur social.

- Ω_1 : espace de solution.
- Ω_2 : espace de calcul.
- r_1^k, r_2^k : vecteur de nombre réels aléatoire compris en 0 et 1.
- Σ : ensemble des scenarii à traiter.

CHAPITRE 5

- m : nombre de sites à livrer
- n : nombre d'interventions
- p : nombre de transporteur
- $J = \{1, 2, \dots, n\}$: ensemble des interventions
- $H = \{1, 2, \dots, m\}$: ensemble des sites à livrer
- $T = \{1, 2, \dots, p\}$: ensemble des transporteurs
- j : index interventions
- h : index des sites à livrer
- t : index des transporteurs
- k : index pour les lots (de boites de DM)
- n_h : nombre d'interventions par site
- d_j : date due de l'intervention j
- cl_j : site de déroulement de l'intervention j
- c_t : capacité du transporteur t
- $\tau_{b_k, h, t}$: temps nécessaire au transporteur t pour livrer un lot de taille b_k vers le site h et revenir
- $\eta_{b_k, h, t}$ coût engendré par le transporteur t pour livrer un lot de taille b_k vers le site h et revenir
- $\alpha(k)$: coût de livraison du lot k
- β_j : fonction de coût supplémentaire appliqué en cas de livraison en avance des DM nécessaire à l'intervention j

Ce qui nous permet d'introduire les variables.

principales

- $\delta_{j,k}^1 = 1$ si la boite de l'intervention j appartient au lot k et 0 sinon
- $\delta_{k,h}^2 = 1$ si le lot k doit être livré au site h et 0 sinon
- $\delta_{k,h,t}^3 = 1$ si le lot k est livré au site h par le transporteur t et 0 sinon

secondaire

- C_j : date d'arrivée de la boîte de l'intervention j sur son site de destination
- B_k : date d'arrivée du lot k sur son site de destination
- b_k : taille du lot k (en nombre de boîtes)
- $y_k = 1$ si le lot k existe et n'est pas vide et 0 sinon
- u_h : nombre de lots livré au site h

RAPPELS DES ACRONYMES

PHARMACIE

- **PUI** - Pharmacie à Usage Interne
- **BPPH** - Bonnes Pratiques de Pharmacie Hospitalière
- **DM/DMR** - Dispositif Médical/ Dispositif Médical Réutilisable
- **CAMSP** - Centrale d'Achat de Matériel Stérile et Pansements
- **PH** - Praticien Hospitalier
- **GCS** - Groupement de Coopération Sanitaire
- **GHRMSA** - Groupement Régional de Mulhouse et Sud Alsace
- **ETP** - Équivalent Temps Plein
- **MeaH** : Mission national d'expertise et d'audit Hospitalier
- **IDE** : Infirmier Diplômé d'Etat
- **ATU** : Autorisation Temporaire d'Utilisation
- **AVG** - Automatic Guided Vehicles

OPTIMISATION

- **AG** - Algorithme Génétique
- **OEP** - Optimisation par Essaim Particulaire
- **PSO** - Particle Swarm Optimization
- **MILP** Mixed Integer Linear Programming
- **UML** - Unified Modeling Language
- **LSP** - Lot Sizing Problem
- **BSP** - Batch Scheduling Problem
- **TSP** - Traveling Salesman Problem
- **VRP** - Vehicle Routing Problem
- **CVRP** - Capacity Vehicle Routing Problem
- **LRP** - Location Routing Problem
- **MCTLSDSP** - Multiple Customers-Transporters Lot Sizing Delivery Scheduling Problem
- **MFIF** - Most Frequency Item First

BIBLIOGRAPHIE

- [Anact, 2012] Anact (2012). **Lean. la performance visée s’obtiendrait-elle au détriment des conditions de travail ?**
- [Aslog, 2002] Aslog (2002). **Commission logistique hospitalière, ” résultats de l’enquête nationale ”**. *Hôpital Expo Conférence*.
- [Beaulieu et al., 2010] Beaulieu, M., et Landry., S. (2010). **Réflexion sur la gestion de la logistique et de l’approvisionnement dans le réseaux québécois de la santé : Pistes de solution et conditions de succès**. Rapport technique, Association québécoise d’établissements de santé et de services sociaux.
- [Beaulieu et al., 2004] Beaulieu, M., et Patenaude, G. (2004). **La gestion des approvisionnements en réseau : le cas du centre hospitalier notre-dame de la merci**. *Logistique & Management*, 12(sup1) :59–64.
- [Ben-Tovim et al., 2007] Ben-Tovim, D. I., Bassham, J. E., Bolch, D., Martin, M. A., Dougherty, M., et Szwarcbord, M. (2007). **Lean thinking across a hospital : redesigning care at the flinders medical centre**. *Australian Health Review*, 31(1) :10–15.
- [Beroule et al., 2017] Beroule, B., Grunder, O., Barakat, O., et Aujoulat, O. (2017). **Order picking problem in a warehouse hospital pharmacy**. *IFAC-PapersOnLine*, 50(1) :5017–5022.
- [Beroule et al.,] Beroule, B., Grunder, O., Barakat, O., Aujoulat, O., et Lustig, H. **Ordonancement des interventions chirurgicales d’un hopital avec prise en compte de l’étape de stérilisation dans un contexte multi-sites**.
- [Beroule et al., 2016a] Beroule, B., Grunder, O., Barakat, O., Aujoulat, O., et Lustig, H. (2016a). **Operating room scheduling including medical devices sterilization : towards a transverse logistic**. *IFAC-PapersOnLine*, 49(12) :1146–1151.
- [Beroule et al., 2016b] Beroule, B., Grunder, O., Barakat, O., Aujoulat, O., et Lustig, H. (2016b). **Particle swarm optimization for operating theater scheduling considering medical devices sterilization**. Dans *International Conference on Swarm Intelligence Based Optimization*, pages 41–56. Springer.
- [Böhme et al., 2016] Böhme, T., Williams, S. J., Childerhouse, P., Deakins, E., et Towill, D. (2016). **Causes, effects and mitigation of unreliable healthcare supplies**. *Production Planning & Control*, 27(4) :249–262.

- [**Boldy et al., 1982**] Boldy, D. P., et O’Kane, P. C. (1982). **Health operational research—a selective overview**. *European Journal of Operational Research*, 10(1) :1–9.
- [**Booch et al., 1996**] Booch, G., Jacobson, I., Rumbaugh, J., et others (1996). **The unified modeling language**, volume 14. Addison Wesley.
- [**Canter, 1991**] Canter, L. P. (1991). **Is materiel management the beast in the basement ?** *Hospital materiel management quarterly*, 12(3) :80.
- [**Cardoen et al., 2009a**] Cardoen, B., Demeulemeester, E., et Beliën, J. (2009a). **Optimizing a multiple objective surgical case sequencing problem**. *International Journal of Production Economics*, 119(2) :354–366.
- [**Cardoen et al., 2009b**] Cardoen, B., Demeulemeester, E., et Beliën, J. (2009b). **Sequencing surgical cases in a day-care environment : an exact branch-and-price approach**. *Computers & Operations Research*, 36(9) :2660–2669.
- [**Cardoen et al., 2010**] Cardoen, B., Demeulemeester, E., et Beliën, J. (2010). **Operating room planning and scheduling : A literature review**. *European Journal of Operational Research*, 201(3) :921–932.
- [**Clerc et al., 2002**] Clerc, M., et Kennedy, J. (2002). **The particle swarm-explosion, stability, and convergence in a multidimensional complex space**. *IEEE transactions on Evolutionary Computation*, 6(1) :58–73.
- [**Clerc et al., 2004**] Clerc, M., et Siarry, P. (2004). **Une nouvelle métaheuristique pour l’optimisation difficile : la méthode des essais particuliers**. *J3eA*, 3 :007.
- [**Conti et al., 2006**] Conti, R., Angelis, J., Cooper, C., Faragher, B., et Gill, C. (2006). **The effects of lean production on worker job stress**. *International journal of operations & production management*, 26(9) :1013–1038.
- [**Coyle et al., 1996**] Coyle, J. J., Bardi, E. J., Langley, C. J., et others (1996). **The management of business logistics**, volume 6. West Publishing Company St Paul, MN.
- [**CPLEX, 2009**] CPLEX, I. I. (2009). **V12. 1 : User’s manual for cplex**. *International Business Machines Corporation*, 46(53) :157.
- [**Cunningham et al., 1992**] Cunningham, W., et Wilson, J. (1992). **Stockless inventory systems for the health care provider : Three successful applications**. *Journal of health care marketing*, 12(2) :39–45.
- [**Danas et al., 2002**] Danas, K., Ketikidis, P., et Roudsari, A. (2002). **A virtual hospital pharmacy inventory : An approach to support unexpected demand**. *Journal of Medical Marketing*, 2(2) :125–129.
- [**Dantzig et al., 1959**] Dantzig, G. B., et Ramser, J. H. (1959). **The truck dispatching problem**. *Management science*, 6(1) :80–91.

- [De Koster et al., 2007] De Koster, R., Le-Duc, T., et Roodbergen, K. J. (2007). **Design and control of warehouse order picking : A literature review**. *European Journal of Operational Research*, 182(2) :481–501.
- [DeJohn, 2009] DeJohn, P. (2009). **Moving up... to the c-suite**. *Materials management in health care*, 18(7) :18.
- [Dexter et al., 2002] Dexter, F., Blake, J. T., Penning, D. H., Sloan, B., Chung, P., et Lubarsky, D. A. (2002). **Use of linear programming to estimate impact of changes in a hospital's operating room time allocation on perioperative variable costs**. *Anesthesiology : The Journal of the American Society of Anesthesiologists*, 96(3) :718–724.
- [Dexter et al., 2003] Dexter, F., et Epstein, R. H. (2003). **Scheduling of cases in an ambulatory center**. *Anesthesiology Clinics of North America*, 21(2) :387–402.
- [Dexter et al., 2004] Dexter, F., et Macario, A. (2004). **When to release allocated operating room time to increase operating room efficiency**. *Anesthesia & Analgesia*, 98(3) :758–762.
- [Dexter et al., 1999] Dexter, F., Macario, A., et Traub, R. D. (1999). **Which algorithm for scheduling add-on elective cases maximizes operating room utilization? use of bin packing algorithms and fuzzy constraints in operating room management**. *The Journal of the American Society of Anesthesiologists*, 91(5) :1491–1491.
- [Dombrowski et al., 2017] Dombrowski, U., Wullbrandt, J., et Reimer, A. (2017). **Lean stress sensitization in learning factories**. *Procedia Manufacturing*, 9 :339–346.
- [Dorigo et al., 1997] Dorigo, M., et Gambardella, L. M. (1997). **Ant colony system : a cooperative learning approach to the traveling salesman problem**. *IEEE Transactions on evolutionary computation*, 1(1) :53–66.
- [Drexler et al., 1997] Drexler, A., et Kimms, A. (1997). **Lot sizing and scheduling—survey and extensions**. *European Journal of Operational Research*, 99(2) :221–235.
- [Engel et al., 2000] Engel, F., Kletz, F., Moisdon, J.-C., et Tonneau, D. (2000). **La démarche gestionnaire à l'hôpital**. *Le PMSI*, 2.
- [Enterprises, 2012] Enterprises, S. (2012). **Scilab : Le logiciel open source gratuit de calcul numérique**. *Scilab Enterprises, Orsay, France*, 71.
- [Fei et al., 2010] Fei, H., Meskens, N., et Chu, C. (2010). **A planning and scheduling problem for an operating theatre using an open scheduling strategy**. *Computers & Industrial Engineering*, 58(2) :221–230.
- [Fleischmann, 1990] Fleischmann, B. (1990). **The discrete lot-sizing and scheduling problem**. *European Journal of Operational Research*, 44(3) :337–348.
- [Flores et al., 1986] Flores, B. E., et Clay Whybark, D. (1986). **Multiple criteria abc analysis**. *International Journal of Operations & Production Management*, 6(3) :38–46.

- [Fowler, 2004] Fowler, M. (2004). **UML distilled : a brief guide to the standard object modeling language**. Addison-Wesley Professional.
- [Fries, 1976] Fries, B. E. (1976). **Bibliography of operations research in health-care systems**. *Operations Research*, 24(5) :801–814.
- [Gicquel et al., 2008] Gicquel, C., Minoux, M., et Dallery, Y. (2008). **Capacitated lot sizing models : a literature review**.
- [Glover, 1989] Glover, F. (1989). **Tabu search—part i**. *ORSA Journal on computing*, 1(3) :190–206.
- [Golberg, 1989] Golberg, D. E. (1989). **Genetic algorithms in search, optimization, and machine learning**. Addison Wesley, 1989 :102.
- [Graham et al., 1979] Graham, R. L., Lawler, E. L., Lenstra, J. K., et Kan, A. R. (1979). **Optimization and approximation in deterministic sequencing and scheduling : a survey**. *Annals of discrete mathematics*, 5 :287–326.
- [Grefenstette et al., 1985] Grefenstette, J., Gopal, R., Rosmaita, B., et Van Gucht, D. (1985). **Genetic algorithms for the traveling salesman problem**. Dans *Proceedings of the first International Conference on Genetic Algorithms and their Applications*, pages 160–165.
- [Groene et al., 2008] Groene, O., Klazinga, N., Kazandjian, V., Lombrail, P., et Bartels, P. (2008). **The world health organization performance assessment tool for quality improvement in hospitals (path) : an analysis of the pilot implementation in 37 hospitals**. *International Journal for Quality in Health Care*, 20(3) :155–161.
- [Guide, 1998] Guide, M. U. (1998). **The mathworks**. Inc., Natick, MA, 5 :333.
- [Guinet et al., 2003] Guinet, A., et Chaabane, S. (2003). **Operating theatre planning**. *International Journal of Production Economics*, 85(1) :69–81.
- [Hammoudan et al., 2016] Hammoudan, Z., Ahmed, F., Grunder, O., et El-Moudni, A. (2016). **Multiple customers/transporters integrated scheduling problem to minimize the total cost**. *IFAC-PapersOnLine*, 49(12) :892 – 897.
- [Hanset et al., 2007] Hanset, A., Fei, H., Roux, O., Duvivier, D., et Meskens, N. (2007). **Ordonnancement des interventions chirurgicales par une recherche tabou : Exécutions courtes vs longues**. *Logistique et Transport LT'07*.
- [Hargadon, 1999] Hargadon, T. (1999). **Formation of medical and hospital groups**. *Managed care quarterly*, 7(4) :7–10.
- [Harris, 1913] Harris, F. W. (1913). **How many parts to make at once**. *Factory, the magazine of management*, 10(2) :135–136.

- [Hasle et al., 2016] Hasle, P., Nielsen, A. P., et Edwards, K. (2016). **Application of lean manufacturing in hospitals—the need to consider maturity, complexity, and the value concept.** *Human Factors and Ergonomics in Manufacturing & Service Industries*, 26(4) :430–442.
- [Hassan, 2006] Hassan, T. (2006). **Logistique hospitalière : organisation de la chaîne logistique pharmaceutique aval et optimisation des flux de consommables et des matériels à usage unique.** PhD thesis, INSA de Lyon.
- [Hohmann, 2012] Hohmann, C. (2012). **Lean management : outils, méthodes, retours d'expériences, questions-réponses.** Editions Eyrolles.
- [Jacquier, 2013] Jacquier, B. (2013). **Du lean au management maigre.** Lulu. com.
- [JDS, 2017] JDS (2017). **Hôpital emile muller - moenchsberg - mulhouse.**
- [Jean-Claude, 2001] Jean-Claude, C. (2001). **Bonnes pratiques de pharmacie hospitalière.**
- [Jebali et al., 2006] Jebali, A., Alouane, A. B. H., et Ladet, P. (2006). **Operating rooms scheduling.** *International Journal of Production Economics*, 99(1) :52–62.
- [Jobin et al., 2004] Jobin, M.-H., Beaulieu, M., et Boivin, A. (2004). **Gérer la performance de la logistique hospitalière.** *Logistique & Management*, 12(sup1) :21–30.
- [Kearney et al., 2004] Kearney, E., et Kearney, A. (2004). **Excellence in logistics 2004.** *European Logistics Association, Brussels.*
- [Kennedy, 2011] Kennedy, J. (2011). **Particle swarm optimization.** Dans *Encyclopedia of machine learning*, pages 760–766. Springer.
- [Kennedy, 1995] Kennedy, R. (1995). **J. and eberhart, particle swarm optimization.** Dans *Proceedings of IEEE International Conference on Neural Networks IV*, pages, volume 1000.
- [King et al., 2006] King, D. L., Ben-Tovim, D. I., et Bassham, J. (2006). **Redesigning emergency department patient flows : application of lean thinking to health care.** *Emergency Medicine Australasia*, 18(4) :391–397.
- [Krafcik, 1988] Krafcik, J. F. (1988). **Triumph of the lean production system.** *MIT Sloan Management Review*, 30(1) :41.
- [Land et al., 1960] Land, A. H., et Doig, A. G. (1960). **An automatic method of solving discrete programming problems.** *Econometrica : Journal of the Econometric Society*, pages 497–520.
- [Landry et al., 2001] Landry, S., et Beaulieu, M. (2001). **La logistique hospitalière : un remède aux maux du secteur de la santé ?** 26 :34.

- [Marcon et al., 2006] Marcon, E., et Dexter, F. (2006). **Impact of surgical sequencing on post anesthesia care unit staffing**. *Health Care Management Science*, 9(1) :87–98.
- [Meinert et al., 2016] Meinert, T., McGrath, S., et Lason, K. E. (2016). **Pharmaceutical technology in hospital pharmacy—first issue**. *Pharmaceutical Technology in Hospital Pharmacy*, 1(1) :1–1.
- [Miliotis, 1978] Miliotis, P. (1978). **Using cutting planes to solve the symmetric traveling salesman problem**. *Mathematical programming*, 15(1) :177–188.
- [Miller et al., 1994] Miller, R. H., et Luft, H. S. (1994). **Managed care plan performance since 1980 : a literature analysis**. *Jama*, 271(19) :1512–1519.
- [Moeller, 2011] Moeller, K. (2011). **Increasing warehouse order picking performance by sequence optimization**. *Procedia-Social and Behavioral Sciences*, 20 :177–185.
- [Mohsen et al., 2002] Mohsen, et Hassan, M. (2002). **A framework for the design of warehouse layout**. *Facilities*, 20(13/14) :432–440.
- [Morgan, 2000] Morgan, J. (2000). **How effective are your measurement systems**. *Purchasing*, 129(10) :26–28.
- [Omran et al., 2006] Omran, M. G., Salman, A., et Engelbrecht, A. P. (2006). **Dynamic clustering using particle swarm optimization with application in image segmentation**. *Pattern Analysis and Applications*, 8(4) :332.
- [Optimization, 2014] Optimization, G. (2014). **Inc., “gurobi optimizer reference manual,” 2014**. URL : <http://www.gurobi.com>.
- [Organization, 2016] Organization, W. H. (2016). **Global health observatory (gho) data**. <http://www.who.int/gho/database>.
- [Ozkil et al., 2009] Ozkil, A. G., Fan, Z., Dawids, S., Aanes, H., Kristensen, J. K., et Christensen, K. H. (2009). **Service robots for hospitals : A case study of transportation tasks in a hospital**. Dans *Automation and Logistics, 2009. ICAL'09. IEEE International Conference on*, pages 289–294. IEEE.
- [Ozturk, 2012] Ozturk, O. (2012). **Optimisation du chargement des laveurs dans un service de stérilisation hospitalière : ordonnancement, simulation, couplage**. PhD thesis, Université de Grenoble.
- [Pandey et al., 2010] Pandey, S., Wu, L., Guru, S. M., et Buyya, R. (2010). **A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments**. Dans *Advanced information networking and applications (AINA), 2010 24th IEEE international conference on*, pages 400–407. IEEE.
- [Quadt, 2004] Quadt, D. (2004). **Lot-sizing and scheduling for flexible flow lines**, volume 546. Springer Science & Business Media.

- [Rangel-Mayoral et al., 2012] Rangel-Mayoral, J., Martin-Clavo, S., Gemio-Zumalave, P., Romero-Soria, L., Braga-Fuentes, L., et García-Cuevas, L. B. (2012). **Web 2.0 in the hospital pharmacy**. *European Journal of Hospital Pharmacy : Science and Practice*, 19(2) :132–132.
- [Ratliff et al., 1983] Ratliff, H. D., et Rosenthal, A. S. (1983). **Order-picking in a rectangular warehouse : a solvable case of the traveling salesman problem**. *Operations Research*, 31(3) :507–521.
- [Relvas et al., 2013] Relvas, S., Magatão, S. N. B., Barbosa-Póvoa, A. P. F., et Neves, F. (2013). **Integrated scheduling and inventory management of an oil products distribution system**. *Omega*, 41(6) :955–968.
- [Ritchie et al., 2000] Ritchie, L., Burnes, B., Whittle, P., et Hey, R. (2000). **The benefits of reverse logistics : the case of the manchester royal infirmary pharmacy**. *Supply Chain Management : An International Journal*, 5(5) :226–234.
- [Robinson et al., 2004] Robinson, J., et Rahmat-Samii, Y. (2004). **Particle swarm optimization in electromagnetics**. *IEEE transactions on antennas and propagation*, 52(2) :397–407.
- [Romero, 2013] Romero, A. (2013). **Managing medicines in the hospital pharmacy : logistics inefficiencies**. Dans *Proceedings of the World Congress on Engineering and Computer Science*, volume 2, pages 1–6.
- [Romero et al., 2015] Romero, A., et Lefebvre, E. (2015). **Combining barcodes and rfid in a hybrid solution to improve hospital pharmacy logistics processes**. *International Journal of Information Technology and Management*, 14(2-3) :97–123.
- [Rosenhead, 1978] Rosenhead, J. (1978). **Operational research in health services planning**. *European Journal of Operational Research*, 2(2) :75–85.
- [Saadani et al., 2006] Saadani, N., Guinet, A., et Chaabane, S. (2006). **Ordonnancement des blocs opératoires**. Dans *MOSIM : Conference francophone de MOdélisation et SIMulation*, volume 6.
- [Salomon, 1991] Salomon, M. (1991). **Deterministic lotsizing models for production planning**, volume 355. Springer.
- [Sampieri et al., 2000] Sampieri, N., et Bongiovanni-Delarozière, I. (2000). **Enjeux et perspectives des pratiques logistiques : Pour une amelioration globale de la performance—le cas de l’hôpital public français**. Dans *Rencontres internationales de recherche en logistique*.
- [Schoor et al., 2007] Schoor, J., Collins, D., Smith, A., Lisitano, R., Kulkarni, R., et D’Onofrio, G. (2007). **286 : Use of lean techniques to simplify admission procedures and decrease ed process time**. *Annals of Emergency Medicine*, 50(3) :S90.

- [Shi et al., 1998a] Shi, Y., et Eberhart, R. (1998a). **A modified particle swarm optimizer**. Dans *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, pages 69–73. IEEE.
- [Shi et al., 1998b] Shi, Y., et Eberhart, R. C. (1998b). **Parameter selection in particle swarm optimization**. Dans *International conference on evolutionary programming*, pages 591–600. Springer.
- [Snyder et al., 2006] Snyder, L. V., et Daskin, M. S. (2006). **A random-key genetic algorithm for the generalized traveling salesman problem**. *European Journal of Operational Research*, 174(1) :38–53.
- [Talbi, 2009] Talbi, E.-G. (2009). **Population-based metaheuristics**. *Metaheuristics : From Design to Implementation*, pages 190–307.
- [Teghem, 2012] Teghem, J. (2012). **LA RECHERCHE OPÉRATIONNELLE (tome 1)**. Ellipses.
- [Trelea, 2003] Trelea, I. C. (2003). **The particle swarm optimization algorithm : convergence analysis and parameter selection**. *Information processing letters*, 85(6) :317–325.
- [Van Hoesel et al., 1963] Van Hoesel, S., Wagelmans, A., et Moerman, B. (1963). **Using geometric techniques to improve dynamic programming algorithms for the economic lot-sizing problem and extensions. report 9212/a, econometric institute, erasmus university rotterdam, rotterdam, the netherlands, 1992.(to appear in european journal of operational research).(9] af veinott, jr**. *Unpublished class notes. Program in Operations Research, Stanford University, Stanford, California*.
- [Wagner et al., 1958] Wagner, H. M., et Whitin, T. M. (1958). **Dynamic version of the economic lot size model**. *Management science*, 5(1) :89–96.
- [Whitson, 1997] Whitson, D. (1997). **Applying just-in-time systems in health care**. *IIE solutions*, 29(8) :32–38.
- [Womack et al., 2005] Womack, J. P., Byrne, A. P., Fiume, O. J., Kaplan, G. S., et Toussaint, J. (2005). **Going lean in health care**. *Cambridge, MA : Institute for Healthcare Improvement*.
- [Womack et al., 1996] Womack, J. P., et Jones, D. T. (1996). **Beyond toyota : how to root out waste and pursue perfection**. *Harvard business review*, 74(5) :140.
- [Womack et al., 1990] Womack, J. P., Jones, D. T., et Roos, D. (1990). **Machine that changed the world**. Simon and Schuster.
- [Wullink et al., 2007] Wullink, G., Van Houdenhoven, M., Hans, E. W., van Oostrum, J. M., van der Lans, M., et Kazemier, G. (2007). **Closing emergency operating rooms improves efficiency**. *Journal of Medical Systems*, 31(6) :543–546.

- [Yusriski et al., 2015] Yusriski, R., Astuti, B., Samadhi, T. A., Halim, A. H., et others (2015). **Integer batch scheduling problems for a single-machine to minimize total actual flow time**. *Procedia Manufacturing*, 2 :118–123.
- [Zhiqiang et al., 2012] Zhiqiang, Z., Chao, Y., et Xiangquan, L. (2012). **The development of automated pharmacy technology**. Dans *Proceedings of the 2012 Third International Conference on Mechanic Automation and Control Engineering*, pages 1994–1997. IEEE Computer Society.

TABLE DES FIGURES

1.1	Centres membre du GHRMSA	5
1.2	La chaîne logistique interne [Beaulieu et al., 2010]	7
1.3	Cycle de vie d'un DM	11
1.4	Positionnement d'un laveur	12
1.5	Exemple de boite de DM reconstituées	13
1.6	boîtes reconstituées en attente devant un autoclave	14
1.7	Logo de l'UML	14
1.8	Des AGV sur leurs stations de recharge	15
1.9	Diagramme d'activité du service de stérilisation	17
1.10	Logigramme du processus de pré lavage des DM	18
1.11	Logigramme du processus de reconditionnement des DM	19
1.12	Diagramme d'activité de la CAMSP	21
1.13	Diagramme d'activité du service de dispensation	25
2.1	Exemple de stock industriel avec des étagères	31
2.2	Un chariot élévateur dans un entrepôt industriel	31
2.3	un convoi d'articles	32
2.4	Un chariot élévateur collectant uniquement sur le coté	32
2.5	exemple de représentation graphique d'un problème de routing	34
2.6	Carrefour complet (gauche) et carrefour avec une direction détaillée (droite)	34
2.7	représentation graphique avec et sans carrefours	35
2.8	nœud simple – nœud orienté vers la droite – nœud orienté vers la gauche	36
2.9	Représentation graphique avec et sans nœuds orientés	37
2.10	Exemple d'application de l'algorithme de création d'allées - Étape 0	39

2.11 Exemple d'application de l'algorithme de création d'allées - Étape 1	39
2.12 Exemple d'application de l'algorithme de création d'allées - Étape 2	39
2.13 Exemple d'application de l'algorithme de création d'allées - Étape 3	40
2.14 Exemple d'application de l'algorithme de création d'allées - Étape 4	40
2.15 Exemple d'application de l'algorithme de création d'allées - Étape 5	40
2.16 Exemple d'une solution comprenant des sous-tours	44
2.17 Représentation d'une solution	46
2.18 Les étapes de l'algorithme génétique	46
2.19 Exemple de croisement de deux individus	49
2.20 Exemple de mutation d'un individu	50
2.21 Entrepôt organisé en allées	50
2.22 Exemple de représentation graphique d'une liste de picking	52
2.23 Solution optimal du VRP appliqué à la figure (2.22)	53
3.1 Explication de la fonction objectif	62
3.2 Exemple de solution avec sous-tours	64
3.3 Les étapes d'élimination des sous-tours	69
3.4 Les étapes de la méthode basée sur le B&B	72
3.5 Représentation partielle d'un arbre de solution (ordre standard)	73
3.6 Représentation partielle d'un arbre de solution (ordre prédéfinis)	74
3.7 Exemple de calcul de la borne inférieure	77
3.8 construction des méthodes mixtes	80
3.9 stock de 400 emplacements	82
4.1 Exemple de planning de six interventions	93
4.2 Identification du nombre minimum de boites de DM	94
4.3 Exemple de planning avec identification des boites	94
4.4 Nouvel exemple de planning	95
4.5 Placements relatifs possibles de deux tâches	100
4.6 Calcul de la vitesse d'une particule	105

4.7	Représentation de deux particules dans l'espace de solutions (à 2 dimensions)	107
4.8	déroulement de l'OEP en utilisant Ω_1 et Ω_2	109
5.1	livraison de DM dans un groupement hospitalier	119
5.2	Étapes du processus global de planification	120
5.3	Exemple complet de planification	128
5.4	planning de l'étape 1 de la figure 5.3	129
5.5	planning de l'étape 3 de la figure 5.3	129
5.6	disposition des sites du groupement pour l'expérimentation	131

LISTE DES TABLES

2.1	Comparaison de performance entre le MILP et l'algorithme génétique . . .	52
3.1	Temps de calcul du B&B avec la borne inférieure standard et améliorée . .	76
3.2	optimization methods efficiency (GA)	84
3.3	optimization methods efficiency (BB)	85
3.4	gaps analysis	86
4.1	Evaluation de la solution obtenue par plusieurs méthodes (en nombre moyen de boites de DM)	103
4.2	Performances (en nombre de boites) des solutions obtenues avec chaque méthode	113
5.1	évaluation (en coût) des solutions obtenue avec 2,3 et 4 transporteurs . . .	134

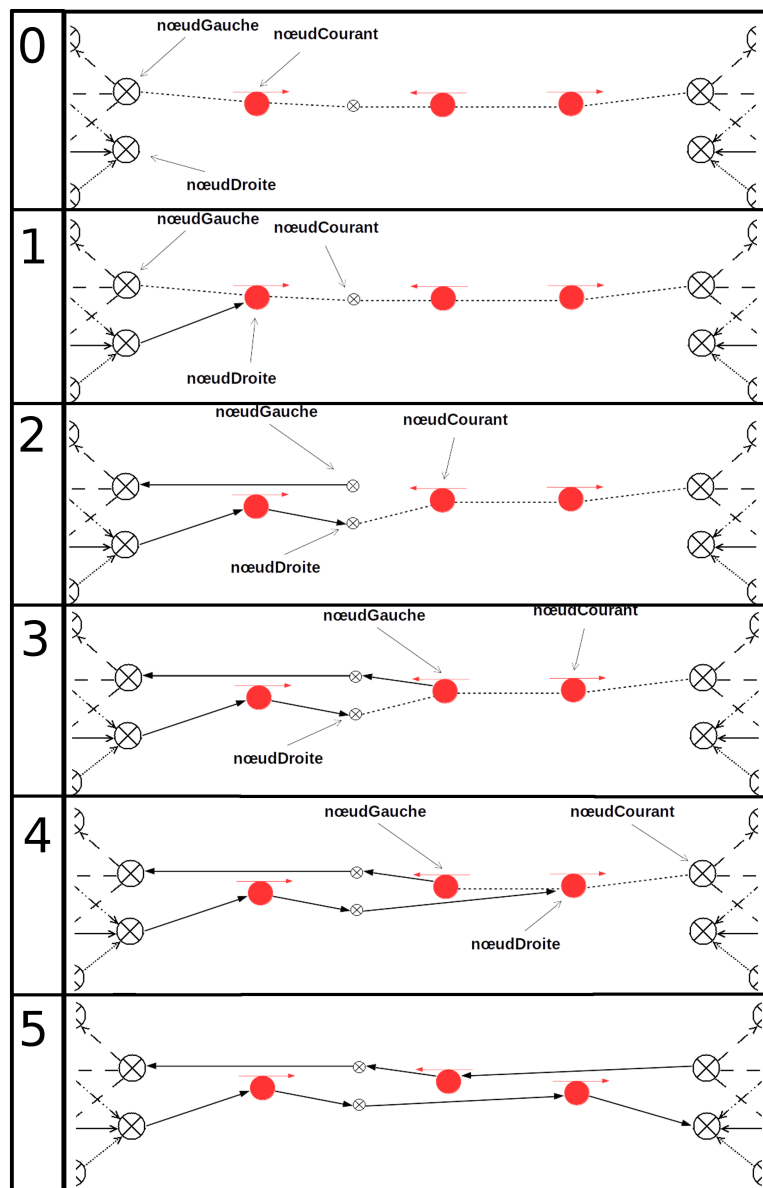
LISTE DES DÉFINITIONS

1	Définition : lean management	7
2	Définition : Dispositif médical	10
3	Définition : Stérilisation - Extrait des BPPH	11
4	Définition : Cloisonnement des tâches - Extrait des BPPH	13
5	Définition : Picking	22
6	Définition : Stockage des stupéfiants – Extrait des BPPH	22
7	Définition : Autorisation temporaire d'utilisation	22
8	Définition : Rétrocession – Extrait des BPPH	23
9	Définition : Pharmacie d'officine	23
10	Définition : Problème du voyageur de commerce	29
11	Définition : Problème de tournées de véhicules	33
12	Définition : modèle LP (L inear P rograming)	41
13	Définition : modèle MILP (M ixed I nteger L inear P rograming)	41
14	Définition : Solveur	44
15	Définition : Métaheuristique	45
16	Définition : Heuristique	45
17	Définition : Location Routing Problem	59
18	Définition : Location Routing Problem - MILP	60
19	Définition : Problème NP-complet	65
20	Définition : Bin packing [Teghem, 2012]	89
21	Définition : Problème multi-objets de dimensionnement de lots capacitif	117
22	Définition : Problème d'ordonnancement des lots	118



ANNEXES

APPLICATION DE L'ALGORITHME (1)



B

MODÈLE MILP D'OPTIMISATION DE STOCK

- n : Nombre de références (et donc d'emplacements).
- m : Nombre de listes de dotation fournies par l'hôpital.
- C_t : Ensemble des objet contenu dans la liste t .
- $d_{k,l}$: Distance entre l'emplacement k et l'emplacement l .
- $x_{i,k}$: Variable binaire égale à 1 seulement s l'objet i se trouve à l'emplacement k .
- $l_{i,j,t}$: Variable binaire égale à 1 seulement si dans la liste t l'objet i et directement suivit par l'objet j .
- $u_{i,t}$: L'ordre de l'objet i dans la liste t
- $X_{i,j,k,l}$: variable binaire représentant $x_{i,k} \times x_{j,l}$
- $Y_{i,j,k,l,t}$: variable binaire représentant $X_{i,j,k,l} \times l_{i,j,t}$

$$\text{Minimize } z = \sum_{i=1}^n \sum_{j=1}^n \sum_{k=1}^n \sum_{l=1}^n \sum_{t=1}^m Y_{i,j,k,l,t} \times d_{k,l} \quad (\text{B.1})$$

$$\sum_{j=1}^n x_{i,j} = 1 \quad 1 \leq i \leq n \quad (\text{B.2})$$

$$\sum_{i=1}^n x_{i,j} = 1 \quad 1 \leq j \leq n \quad (\text{B.3})$$

$$x_{1,1} = 1 \quad (\text{B.4})$$

$$\sum_{j \in C_t} l_{i,j,t} = 1 \quad i \in C_t \quad 1 \leq t \leq m \quad (\text{B.5})$$

$$\sum_{i \in C_t} l_{i,j,t} = 1 \quad j \in C_t \quad 1 \leq t \leq m \quad (\text{B.6})$$

$$l_{i,i,t} = 1 \quad 1 \leq t \leq m \quad (\text{B.7})$$

$$u_{1,t} = 1 \quad 1 \leq t \leq m \quad (\text{B.8})$$

$$\begin{aligned} u_{i,t} - u_{j,t} + n \times l_{i,j,t} \leq n - 1 \quad & 1 \leq i \leq n \\ & 2 \leq j \leq n \\ & 1 \leq t \leq m \\ & i \neq j \end{aligned} \quad (\text{B.9})$$

$$\begin{aligned} X_{i,j,k,l} \leq x_{i,k} \quad & 1 \leq i \leq n \\ & 1 \leq j \leq n \\ & 1 \leq k \leq n \\ & 1 \leq l \leq n \end{aligned} \quad (\text{B.10})$$

$$\begin{aligned} X_{i,j,k,l} \leq x_{j,l} \quad & 1 \leq i \leq n \\ & 1 \leq j \leq n \\ & 1 \leq k \leq n \\ & 1 \leq l \leq n \end{aligned} \quad (\text{B.11})$$

$$\begin{aligned} X_{i,j,k,l} \geq x_{i,k} + x_{j,l} - 1 \quad & 1 \leq i \leq n \\ & 1 \leq j \leq n \\ & 1 \leq k \leq n \\ & 1 \leq l \leq n \end{aligned} \quad (\text{B.12})$$

$$\begin{aligned} Y_{i,j,k,l,t} \leq X_{i,j,k,l} \quad & 1 \leq i \leq n \\ & 1 \leq j \leq n \\ & 1 \leq k \leq n \\ & 1 \leq l \leq n \\ & 1 \leq t \leq m \end{aligned} \quad (\text{B.13})$$

$$\begin{aligned}
Y_{i,j,k,l,t} &\leq l_{i,j,l} & 1 \leq i \leq n \\
& & 1 \leq j \leq n \\
& & 1 \leq k \leq n \\
& & 1 \leq l \leq n \\
& & 1 \leq t \leq m
\end{aligned} \tag{B.14}$$

$$\begin{aligned}
Y_{i,j,k,l,t} &\geq X_{i,j,k,l} + l_{i,j,t} - 1 & 1 \leq i \leq n \\
& & 1 \leq j \leq n \\
& & 1 \leq k \leq n \\
& & 1 \leq l \leq n \\
& & 1 \leq t \leq m
\end{aligned} \tag{B.15}$$

MODÈLE MILP D'OPTIMISATION DES PLANNINGS D'INTERVENTIONS

- n : nombre d'interventions.
- P_i : durée de l'intervention i et du transport de la boîte correspondante.
- S_i : durée du processus de stérilisation complet de la boîte de l'intervention i .
- S : date ouverture du service de stérilisation.
- S' : date de fermeture du service de stérilisation.
- P : date d'ouverture du bloc opératoire.
- P' : date de fermeture du bloc opératoire.
- T : durée d'ouverture du service de stérilisation sur une journée ($T = S' - S$).
- M : un grand nombre utilisé pour la linéarisation ($M = 999999$)

*

- i : indice des interventions ($1 \leq i \leq n$).
- j : indice des boîtes de DM ($1 \leq j \leq n$).
- k : indice des jours de la semaine ($1 \leq k \leq 5$).

*

principales :

- $x_{i,j,k}$: variable binaire égale à 1 si et seulement si l'intervention i utilise la boîte de DM j pendant le jour k .
- y_j : variable binaire égale à 1 si et seulement si la boîte de DM j est utilisée au moins une fois dans la semaine.
- $X_{i_1,i_2,j,k}$: variable binaire représentant $x_{i_1,j,k} \times x_{i_2,j,k}$

secondaire :

- t_i, t'_i : respectivement date de début et de fin de la tâche i .

176 ANNEXE C. MODÈLE MILP D'OPTIMISATION DES PLANNINGS D'INTERVENTIONS

- a_i, b_i, c_i : variables binaires sans signification particulière, utilisées pour créer les contraintes temporelles.
- $\alpha_{i_1, i_2}, \beta_{i_1, i_2}$: variables binaires sans signification particulières, utilisé pour linéariser le modèle

$$\text{Minimize } \sum_{j=1}^n y_j \quad (\text{C.1})$$

$$\sum_{i=1}^n x_{i,j,k} \times (P_i + S_i) \leq T \times y_j \quad \forall j \in \llbracket 1, n \rrbracket \quad (\text{C.2})$$

$$\forall k \in \llbracket 1, 5 \rrbracket$$

$$\sum_{j=1}^n \sum_{k=1}^5 x_{i,j,k} = 1 \quad \forall i \in \llbracket 1, n \rrbracket \quad (\text{C.3})$$

$$\begin{aligned} X_{i_1, i_2, j, k} &\leq x_{i_1, j, k} & 1 \leq i_1 \leq n \\ & & 1 \leq j \leq n \\ & & 1 \leq k \leq 5 \end{aligned} \quad (\text{C.4})$$

$$\begin{aligned} X_{i_1, i_2, j, k} &\leq x_{i_2, j, k} & 1 \leq i_2 \leq n \\ & & 1 \leq j \leq n \\ & & 1 \leq k \leq 5 \end{aligned} \quad (\text{C.5})$$

$$\begin{aligned} X_{i_1, i_2, j, k} &\geq x_{i_1, j, k} + x_{i_2, j, k} - 1 & 1 \leq i_1 \leq n \\ & & 1 \leq i_2 \leq n \\ & & 1 \leq j \leq n \\ & & 1 \leq k \leq 5 \end{aligned} \quad (\text{C.6})$$

$$\begin{aligned} t_{i_1} - (t_{i_2} + P_{i_2} + P'_{i_2}) + M \times \alpha_{i_1, i_2} &\geq 0 & 1 \leq i_1 \leq n \\ & & 1 \leq i_2 \leq n \end{aligned} \quad (\text{C.7})$$

$$\begin{aligned}
t_{i_1} - (t_{i_2} + P_{i_2} + P'_{i_2}) + M \times \alpha_{i_1, i_2} \leq M & \quad 1 \leq i_1 \leq n \\
& \quad 1 \leq i_2 \leq n
\end{aligned} \tag{C.8}$$

$$\begin{aligned}
(t_{i_1} + P_{i_1} + P'_{i_1}) - t_{i_2} + M \times \beta_{i_1, i_2} \geq 0 & \quad 1 \leq i_1 \leq n \\
& \quad 1 \leq i_2 \leq n
\end{aligned} \tag{C.9}$$

$$\begin{aligned}
(t_{i_1} + P_{i_1} + P'_{i_1}) - t_{i_2} + M \times \beta_{i_1, i_2} \leq M & \quad 1 \leq i_1 \leq n \\
& \quad 1 \leq i_2 \leq n
\end{aligned} \tag{C.10}$$

$$\begin{aligned}
\alpha_{i_1, i_2} - \beta_{i_1, i_2} \geq X_{i_1, i_2, j, k} - 1 & \quad 1 \leq i_1 \leq n \\
& \quad 1 \leq i_2 \leq n
\end{aligned} \tag{C.11}$$

$$\begin{aligned}
\alpha_{i_1, i_2} - \beta_{i_1, i_2} \leq 1 - X_{i_1, i_2, j, k} & \quad 1 \leq i_1 \leq n \\
& \quad 1 \leq i_2 \leq n
\end{aligned} \tag{C.12}$$

$$\begin{aligned}
x_{i_1, j, k} \times x_{i_2, j, k} \times (t_{i_1} - t'_{i_2}) \times (t'_{i_1} - t_{i_2}) \geq 0 & \quad \forall i_1 \in \llbracket 1, n \rrbracket \\
& \quad \forall i_2 \in \llbracket 1, n \rrbracket \\
& \quad \forall j \in \llbracket 1, n \rrbracket \\
& \quad \forall k \in \llbracket 1, 5 \rrbracket
\end{aligned} \tag{C.13}$$

$$P + a_i \times T \leq t_i \leq P' + a_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \tag{C.14}$$

$$P + a_i \times T \leq t_i + P_i \leq P' + a_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \tag{C.15}$$

$$S + b_i \times T \leq t_i + P_i \leq S' + b_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \tag{C.16}$$

$$S + c_i \times T \leq t'_i \leq S' + c_i \times T \quad \forall i \in \llbracket 1, n \rrbracket \tag{C.17}$$

$$\begin{aligned}
(k-1) \times T \times x_{i, j, k} \leq t_i \leq 5 \times T \times (1-k) \times x_{i, j, k} & \quad \forall i \in \llbracket 1, n \rrbracket \\
& \quad \forall j \in \llbracket 1, n \rrbracket \\
& \quad \forall k \in \llbracket 1, 5 \rrbracket
\end{aligned} \tag{C.18}$$

