

UNIVERSITÉ DE MONTRÉAL

UNE HEURISTIQUE EN DEUX PHASES POUR LA CONFECTION D'HORAIRE DE
PERSONNEL AVEC TRANSFERTS INTER-DÉPARTEMENTAUX D'EMPLOYÉS

EMELYNE MUNEZERO
DÉPARTEMENT DE MATHÉMATIQUES ET DE GÉNIE INDUSTRIEL
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(MATHÉMATIQUES APPLIQUÉES)
DÉCEMBRE 2014

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé :

UNE HEURISTIQUE EN DEUX PHASES POUR LA CONFECTION D'HORAIRE DE
PERSONNEL AVEC TRANSFERTS INTER-DÉPARTEMENTAUX D'EMPLOYÉS

présenté par : MUNEZERO Emelyne

en vue de l'obtention du diplôme de : Maîtrise ès sciences appliquées

a été dûment accepté par le jury d'examen constitué de :

M. SOUMIS François, Ph. D., président

M. DESAULNIERS Guy, Ph. D., membre et directeur de recherche

M. CONTARDO Claudio, Ph. D., membre et codirecteur de recherche

M. ROUSSEAU Louis-Martin, Ph. D., membre

DÉDICACE

*À mon père et ma mère,
merci pour votre amour et votre soutien.*

REMERCIEMENTS

Je souhaite remercier vivement mon directeur de recherche, le professeur Guy Desaulniers. Merci pour le support tant moral que financier qui m'ont permis d'arriver à terme de ce projet de recherche.

Mes sincères remerciements vont aussi à mon codirecteur de recherche, le professeur Claudio Contardo pour ses conseils qui ont grandement contribué à l'accomplissement de mon travail.

Je vous remercie infiniment pour votre disponibilité et votre patience à mon égard. J'ai beaucoup appris à vos côtés ; vos conseils et votre encadrement m'ont été d'une utilité inestimable tout au long de mon projet de maîtrise.

Mes remerciements s'adressent aussi aux membres du jury qui ont accepté d'évaluer ce travail.

Bien que je ne puisse tous les citer, je tiens à remercier du fond du cœur tous mes amis qui, par leur soutien et leurs encouragements, m'ont permis d'arriver à cette étape de ma vie. Je ne vous remercierai jamais assez.

Finalement, je tiens à exprimer ma gratitude à ma famille, spécialement mon père et ma mère. Merci pour votre amour inconditionnel, vos conseils et tous les sacrifices que vous ne cessez de faire à mon égard. Et à mon frère et mes sœurs, je vous dis tout simplement MERCI.

RÉSUMÉ

L'un des problèmes classiques de la recherche opérationnelle est la confection des horaires de travail personnalisés. L'optimisation des horaires de travail vise à spécifier, pour chaque jour de l'horizon de planification, la séquence des jours de travail et de repos de chaque employé, à construire des quarts de travail ainsi qu'à affecter ces quarts aux employés. Dans un contexte multi-activités, cela inclura de générer également les séquences d'activités exécutées dans chaque quart de travail.

La plupart des grandes entreprises sont divisées en départements. La gestion de la main d'œuvre se fait en rattachant un ensemble d'employés à chacun de ces départements. De plus, en tenant compte de l'évolution professionnelle des employés, la plupart se retrouvent rattachés à plus d'un département et acquièrent ainsi de l'expérience dans plusieurs d'entre eux tout au long de leur carrière. Les employés se retrouvent donc qualifiés à travailler dans plusieurs départements au sein d'une même entreprise.

Dans ce mémoire de maîtrise, nous nous intéressons à la confection d'horaires de travail personnalisés en tenant compte de la possibilité de transférer des employés entre les départements. Cela se présente notamment dans les magasins à grande surface où un employé peut passer d'un département à un autre durant son quart de travail. Il s'agit donc d'un contexte multi-départements où chaque département est mono-activité, c'est-à-dire que nous considérons une seule activité par département. De plus, chaque employé peut travailler au plus dans deux départements par quart. Quand il doit travailler dans deux départements, l'un d'eux doit nécessairement être son département d'origine.

Chaque employé est qualifié pour un ensemble donné de départements en dehors du département auquel il est rattaché à l'origine. Par ailleurs, les courbes de demandes pour chaque département ainsi que les qualifications des employés sont connus à l'avance. Aussi, les règles de la convention collective relativement aux horaires ainsi que d'autres règles entreront en jeu pour générer des horaires réalisables.

L'objectif principal du problème est de minimiser la somme pondérée des coûts totaux. Ces derniers sont principalement composés de quatre types de coûts différents : (1) le coût associé aux heures travaillées, (2) des pénalités pour le transfert de personnel entre départements, (3) le coût total des périodes en sous-couverture et (4) le coût total des périodes en

sur-couverture. Le coût de travail est calculé pour chaque employé en fonction de la durée de son quart de travail. Le coût de transfert, quant à lui, est calculé en fonction de la durée du transfert. Pour une période de transfert donnée, ce coût s'ajoute ainsi au coût de travail de la même période. Par ailleurs, la satisfaction de la demande en main-d'œuvre n'est pas toujours possible notamment dans le cas où la demande excède le nombre total d'employés disponibles. Pour cela, on permet la sous-couverture et la sur-couverture de la demande moyennant des pénalités sous forme de coûts : il s'agit des coûts de sous et sur-couverture. Alors que les coûts de travail et de transfert dépendent de l'employé effectuant le quart et de la durée des périodes considérées, les coûts de sous et sur-couverture dépendent quant à elles du département au sein duquel on fait face à la sous-couverture ou à la sur-couverture.

L'objectif principal de ce projet de recherche est donc de proposer une méthodologie de confection d'horaires de travail personnalisés afin de résoudre efficacement ce problème. Des jeux de données réels de grande taille seront utilisés pour tester et valider les différents modèles et algorithmes proposés.

En pratique, les transferts sont planifiés séquentiellement à la main après avoir fait l'affectation par département. Nous commençons d'abord par automatiser ce processus en développant un algorithme glouton. Nous proposons ensuite une amélioration qui consiste à remplacer l'algorithme glouton par un modèle global résolu directement par Xpress-MP qui permet de traiter le problème au complet comme un problème mixte en nombres entiers en ayant une vue globale pour le calcul des transferts. La résolution du modèle global permet d'assigner à chaque employé, durant l'horizon de planification, un ensemble de quarts personnalisés admissibles. Pour cela, nous fournissons une énumération explicite des quarts de travail notamment l'heure de début, la durée du quart de travail, la durée de transfert et le placement du transfert pour chaque quart. Par conséquent, on se retrouve avec un très grand nombre de possibilités ce qui augmente la complexité du modèle et le temps d'exécution.

Par la suite, nous proposons une approche à deux phases pour résoudre le problème de confection d'horaires avec transferts entre les départements pour lequel la résolution d'un modèle mathématique global échoue sur les instances de grande taille. En effet, formuler explicitement tous les quarts de travail en portant une certaine flexibilité sur la définition des heures de début, des durées ou du placement des périodes de transferts produit un très grand nombre de possibilités et augmente la complexité du modèle.

Dans l'approche à deux phases, un modèle séquentiel département par département est

utilisé dans la première phase sans permettre les transferts. La solution obtenue est alors ré-optimisée dans la seconde phase en permettant les transferts entre les départements. Nous présentons les résultats pour les différents modèles proposés dans la deuxième phase. Le modèle séquentiel département par département vise à assigner à chaque employé un quart de travail dans son département d'origine. Un modèle mathématique en nombres entiers résolu par Xpress-MP est alors utilisé. Différentes approches de résolution sont proposées pour la seconde phase de réoptimisation incluant un algorithme glouton et des modèles linéaires en nombres entiers aussi résolus par Xpress-MP.

Les expérimentations numériques réalisées sur différents jeux de données de grande taille montrent que les temps de calcul sont nettement moindres avec la méthode en deux phases. De plus, l'heuristique assure une couverture de la demande totale à plus de 99.8%. Elle améliore aussi les résultats par rapport à la pratique, soit l'algorithme glouton. Par contre, la qualité des solutions est grandement détériorée (d'environ 30%) pour les instances de petite taille. Par ailleurs, des tests montrent que l'heuristique pourrait s'améliorer encore plus en permettant de remettre en cause les quarts internes. Pour les instances de grande taille rencontrées en pratique, l'heuristique permet d'obtenir des solutions en des temps plus qu'acceptables, soit moins de 2 minutes pour des instances avec 8 départements et 191 employés.

ABSTRACT

One of the classic problems of operational research is building customized work schedules. The optimization of working hours is intended to specify, for each day of the planning horizon, the sequence of the days of work and rest for each employee, to build shifts and to assign these shifts to employees. In a multi-activity context, this will also include the generation of activity sequences carried out in each shift.

Most large companies are divided into departments. The management of the labor is done by assigning a set of employees in each of these departments. In addition, taking into account the professional development of employees, most are found assigned to more than one department and thus gain experience in several of them throughout their careers. Employees are therefore qualified to work in several departments within the same company.

In this master's project, we focus on the computation of customized work schedules taking into account the possibility of transferring employees between departments. This occurs especially in large retail stores where employees can move from one department to another during their shift. It is therefore a multi-department environment where each department is mono-activity, that is, we consider a single activity by department. In addition, each employee can work at least in two departments in a shift. When assigned to two departments in the same shift, one of them must be his origin department.

Each employee is qualified for a given set of departments outside the department to which he is originally attached. Moreover, the demand curves for each department and the qualifications of employees are known in advance. Also, the rules of the collective agreement as well as other rules will come into play to generate feasible schedules.

The main goal of the problem is to minimize the weighted sum of the total costs. These are mainly composed of four types of costs: (1) the cost associated with worked hours, (2) penalties for the transfer of staff between departments, (3) the total cost of under-coverage periods and (4) the total cost of over-coverage periods. Labor cost is calculated for each employee on the basis of the duration of his shift. The cost of transfer, meanwhile, is calculated on the basis of the duration of the transfer. For a given transfer period, this cost adds to the cost of work for the same period. Moreover, the satisfaction of the demand for labor is not always possible, especially in cases where demand exceeds the total number of employees

available. For this, we allow under-coverage and over-coverage subject to penalties in the form of costs: these are under and over-coverage costs. While working and transfer costs depend on the employee making the shift and the length of the relevant periods, the costs of under and over-coverage depend on the department in which the under-coverage or over-coverage occurs.

The main objective of this research project is therefore to propose a methodology for customized work schedules in order to effectively solve this problem. Real data sets of large size will be used to test and validate the various models and algorithms proposed.

In practice, transfers are scheduled sequentially by hand after assigning departments. First this process is automated using a greedy algorithm. Then we suggest an improvement by replacing the greedy algorithm by a global model solved using Xpress-MP that process the whole problem as a MIP while keeping a global view during transfers calculations. The resolution of the global model allows to assign to each employee during the planning horizon, a set of feasible shifts. To do this, we provide an explicit enumeration of shifts including the start time, the duration of the shift, the duration of transfer and placement of transfer for each shift. As a result, we are left with a very large number of possibilities that increases the complexity of the model and the solution time.

Subsequently, we propose a two-phase heuristic approach to solve the problem when the solution of the global model fails for large instances. Indeed, enumerating explicitly all shifts can produce a very large number of options given high flexibility on the shift start times, duration, transfer period placement.

In the two-phase approach, a sequential model department by department is used in the first phase without allowing transfers. The resulting solution is then re-optimized in the second phase by allowing transfers between departments. We present the results for different models in the second phase. The sequential model department by department aims to assign to each employee a shift in its original department. An integer mathematical model solved by Xpress-MP is used. Different approaches are proposed for the second phase of re-optimization including a greedy algorithm and integer linear models also solved by Xpress -MP.

Numerical experiments performed on various large data sets show that the processing time is significantly lowered with the two-phase heuristic approach. In addition, using this two-phase heuristic approach, we are able to cover the total demand at 99.8%. It also improves the results compared to the greedy algorithm which reflects what is done in practice. However,

the quality of solutions is greatly deteriorated (by about 30%) for smaller sets. Moreover, tests show that the heuristic could be even better if we question the internal shifts. For larger instances encountered in practical case, the two-phase heuristic approach provides solutions in very acceptable time, namely, less than 2 minutes for instances of 8 departments and 191 employees.

TABLE DES MATIÈRES

DÉDICACE	iii
REMERCIEMENTS	iv
RÉSUMÉ	v
ABSTRACT	viii
TABLE DES MATIÈRES	xi
LISTE DES TABLEAUX	xiii
LISTE DES FIGURES	xv
LISTE DES ANNEXES	xvi
CHAPITRE 1 INTRODUCTION	1
1.1 Contexte général	1
1.2 Objectifs de la recherche	3
1.3 Contributions	4
1.4 Organisation du mémoire	4
CHAPITRE 2 REVUE DE LITTÉRATURE	6
2.1 Classification des problèmes d’horaires de personnel	6
2.2 Construction des quarts de travail	9
2.2.1 Modèle de recouvrement de Dantzig	9
2.2.2 Modèles implicites de Bechtold et Jacobs (1990)	12
2.2.3 Grammaires et langages formels	12
2.3 Affectation des activités aux quarts	13
2.4 Cas multi-départements	15
CHAPITRE 3 PROBLÉMATIQUE ET MODÈLE GLOBAL	17
3.1 Problème multi-départements avec transferts	17
3.1.1 Définitions et terminologie	18
3.1.2 Énoncé du problème	23
3.2 Modèle global	24

3.2.1	Construction des quarts de travail	24
3.2.2	Notations	25
3.2.3	Formulation	26
3.2.4	Méthode de résolution	28
CHAPITRE 4 HEURISTIQUE EN DEUX PHASES		29
4.1	Première phase : Problème séquentiel multi-départements sans transferts . . .	29
4.1.1	Définitions et terminologie	30
4.1.2	Formulation du modèle	30
4.1.3	Méthode de résolution	30
4.2	Deuxième phase : Ré-optimisation du problème multi-départements avec transferts	31
4.2.1	Contexte général	31
4.2.2	Méthodes de résolution	35
CHAPITRE 5 RÉSULTATS NUMÉRIQUES		41
5.1	Conditions de simulations	41
5.2	Jeux de données	41
5.3	Critères de performance	43
5.4	Résultats numériques	44
5.4.1	Modèle global	45
5.4.2	Heuristique en deux phases	47
5.4.3	Heuristique en deux phases : autres cas de tests	60
CHAPITRE 6 CONCLUSION		63
6.1	Synthèse des travaux	63
6.2	Améliorations futures	64
RÉFÉRENCES		65
ANNEXES		68

LISTE DES TABLEAUX

Tableau 2.1	Exemple des quarts de travail tel que défini par Dantzig (1954)	11
Tableau 3.1	Exemple d'une courbe de demande pour un département	19
Tableau 3.2	Notations relatives à un quart de travail	21
Tableau 3.3	Exemple de sous et sur-couvertures	23
Tableau 3.4	Notations des paramètres et constantes du problème	26
Tableau 4.1	Exemple de construction des quarts de travail pour un quart interne de 6h	34
Tableau 4.2	Notations utilisées pour l'algorithme glouton	36
Tableau 4.3	Paramètres du modèle exact de la phase de ré-optimisation	39
Tableau 5.1	Instances de données générées aléatoirement	43
Tableau 5.2	Instances de données réelles de grande taille	43
Tableau 5.3	Critères de performance	44
Tableau 5.4	Valeurs par défaut des paramètres définissant le problème	45
Tableau 5.5	Taille du modèle global pour les instances aléatoires	46
Tableau 5.6	Résultats numériques pour le modèle global	47
Tableau 5.7	Résultats pour la première phase séquentielle	51
Tableau 5.8	Résultats pour la seconde phase de réoptimisation avec l'algorithme glouton	52
Tableau 5.9	Résultats numériques de l'heuristique en deux phases avec l'algorithme glouton	53
Tableau 5.10	Résultats pour la seconde phase de réoptimisation avec le modèle exact	53
Tableau 5.11	Résultats numériques de l'heuristique en deux phases avec le modèle exact	54
Tableau 5.12	Résultats numériques pour le modèle global	59
Tableau 5.13	Résultats numériques pour l'heuristique en deux phases avec le modèle exact	61
Tableau 5.14	Résultats numériques de l'heuristique avec l'algorithme glouton	62
Tableau A.1	Résultats pour la première phase séquentielle des instances aléatoire de grande taille	68
Tableau A.2	Résultats pour la seconde phase de réoptimisation avec le modèle exact pour les instances aléatoires de grande taille	68
Tableau A.3	Résultats pour la première phase séquentielle des instances réelles de grande taille	69

Tableau A.4	Résultats pour la seconde phase de réoptimisation avec le modèle exact des instances réelles de grande taille	69
-------------	--	----

LISTE DES FIGURES

Figure 3.1	Types de quarts	21
Figure 4.1	Horaires des employés après la première phase de résolution séquentielle	33
Figure 4.2	Les 3 possibilités de transfert	33
Figure 5.1	Temps de résolution en fonction du nombre de départements pour le modèle global	48
Figure 5.2	Temps de résolution en fonction du nombre d'employés pour le modèle global	49
Figure 5.3	Temps de résolution Modèle global et heuristiques	55
Figure 5.4	Comparaion du coût de la fonction objectif	56
Figure 5.5	Couverture de la demande	57
Figure 5.6	Couverture de la demande	58
Figure 5.7	Nombre de sur-couvertures générées	60

LISTE DES ANNEXES

ANNEXE A : RÉSULTATS NUMÉRIQUES DE LA PHASE SÉQUENTIELLE . . . 68

CHAPITRE 1

INTRODUCTION

Pour la majorité des organisations, les ressources humaines sont parmi les plus cruciales et les plus coûteuses. Il devient donc primordial pour ces organisations de mettre en place de bonnes stratégies pour la gestion de ces ressources. Par ailleurs, l'un des principaux objectifs poursuivis par toute entreprise est la maximisation de ses profits. Dans cette optique, déterminer des horaires efficaces pour les employés constitue un enjeu important pour les moyennes et grandes entreprises. En effet, une grande partie des coûts de fonctionnement d'une entreprise est attribuée à la gestion de la main-d'œuvre. L'optimisation des horaires de travail devient donc important pour réduire les coûts tout en maximisant la satisfaction des employés.

1.1 Contexte général

Les problèmes de planification des horaires de personnel font partie intégrante de la gestion des ressources d'une entreprise. Le défi souvent rencontré reste de trouver un horaire de personnel qui puisse à la fois satisfaire la demande tout en respectant les clauses de la convention collective et aussi assurer un certain degré de satisfaction de la part des employés quant à leur qualité de travail et de vie. En effet, affecter le bon nombre de personnes aux bonnes tâches et au bon moment est un travail très difficile qui peut nécessiter des méthodes plus complexes dans certains cas.

Ces problèmes de planification trouvent leur pratique dans plusieurs domaines d'applications manufacturières ou de services. Nous pouvons citer, entre autres, les domaines du transport (routier, aérien, ferroviaire, etc.), de la santé (planification du personnel médical dans les hôpitaux), magasins à grande surface, l'enseignement, les centres d'appel, la restauration, les services d'urgence (policiers, pompiers, ambulanciers), les services financiers, etc. Dans ce projet de recherche, nous nous intéressons aux problèmes qui trouvent leur application surtout dans les magasins à grande surface.

Par ailleurs, les problèmes de construction des horaires de travail peuvent se présenter sous divers modèles en fonction des particularités de l'entreprise ou de la durée de la période considérée appelée *horizon de planification*. On désigne par horizon de planification, un en-

semble de périodes de temps successives appelées *périodes de planification*. La durée de ces périodes peut varier selon les spécifications du problème ou les besoins de la modélisation. Les périodes de planification sont généralement de durées égales variant de quelques minutes (par exemple, *15 min* chacune) à quelques heures (par exemple, *2h* chacune).

En général, la construction des horaires passe par la détermination du nombre d'employés requis pour assurer le niveau de service établi, leurs affectations à des tâches précises durant l'horizon de planification ou à l'affectation du lieu de travail à chaque employé pour accomplir ces tâches. Ainsi, l'horaire d'un employé peut simplement déterminer les jours travaillés et de repos durant tout l'horizon de planification. Il peut aussi être plus spécifique selon les besoins de l'organisation en indiquant une heure de début, une heure de fin et des périodes durant lesquelles l'employé pourra prendre une ou plusieurs pauses. Dans d'autres cas, il peut s'avérer nécessaire de connaître quels sont les types de tâches qui seront exécutées durant l'horaire de travail tout en précisant où et à quelle heure elles sont planifiées.

Tout cela donne une idée sur la complexité de ces problèmes et du nombre croissant des modèles développés et de leurs méthodes de résolution afin d'améliorer constamment les solutions obtenues. Plusieurs auteurs se sont penchés sur ces problèmes au cours des dernières décennies.

Dans les grandes entreprises, le travail peut se diviser en plusieurs départements qui peuvent être plus ou moins indépendants. En général, un département désigne une division de l'entreprise qui se spécialise à la production d'un bien ou à l'offre d'un service. Un département peut aussi se caractériser par l'ensemble des activités qui y sont effectuées. En effet, pouvoir offrir un bien ou un service peut nécessiter d'effectuer un travail interruptible constitué d'une séquence d'activités différentes, ou d'une seule activité de même nature et continue. On parle des cas *multi-activités* et *mono-activité*, respectivement. C'est ce dernier cas qui nous intéresse dans le cadre de ce mémoire.

Par ailleurs, chaque département présente une courbe de demande reflétant le nombre d'employés requis pour chaque période de planification durant un horizon déterminé. La planification des horaires de personnel se fera donc dans le but de satisfaire au maximum cette demande. La satisfaction de la demande en personnel peut être imposée sous la forme d'une contrainte rigide n'autorisant que le surplus d'employés le long des périodes de planification. Il s'agit de la *sur-couverture* où le nombre d'employés planifiés excède la demande à certaines périodes si cela assure un gain. Or, dans certaines situations, la demande peut

excéder le nombre total d'employés disponibles. L'un des moyens utilisés pour traiter ces cas est de permettre, pour chaque période de planification, une *sous-couverture* en plus de la *sur-couverture* définie précédemment. L'objectif sera donc de minimiser ces déviations dans la fonction objectif.

La gestion de la main-d'œuvre dans une organisation peut se faire en considérant uniquement les employés qui sont rattachés à chaque département et en traitant les départements comme des entités séparés. On parle alors de contexte *mono-département*. Mais dans certains cas, la demande en personnel d'un département ne peut pas être satisfaite par ses employés internes seulement. Cela implique la mise en place d'autres moyens pour combler la demande, que ce soit par l'autorisation des heures supplémentaires ou le recours à la sous-traitance. Un autre moyen souvent employé dans les entreprises est de permettre le transfert des employés entre les départements pour satisfaire la demande : il s'agit alors du contexte *multi-départements* pour lequel l'optimisation des horaires de travail se fera simultanément pour l'ensemble des départements. Cela introduit une nouvelle notion d'*employés externes* qui, à l'opposé des employés internes, peuvent être affectés à travailler dans des départements où ils n'étaient pas rattachés à l'origine. Cela implique que les employés possèdent les qualifications requises pour travailler dans ces départements. Ces qualifications peuvent avoir été acquises tout au long de leur carrière en cumulant assez d'expériences de travail dans plusieurs départements. En outre, ces transferts doivent se conformer aux réglementations de travail pré-établies comme par exemple le nombre maximal de transferts pour un quart donné ou encore le nombre maximal d'heures transférées pour chaque employé durant tout l'horizon de planification.

1.2 Objectifs de la recherche

L'objectif principal de ce projet de recherche est de proposer une méthodologie pour résoudre les problèmes de confection d'horaires de travail personnalisés de grande taille en des temps rapides tout en tenant compte du transfert d'employés entre les départements. En pratique, les transferts sont planifiés séquentiellement à la main après avoir fait l'affectation par département. Un des objectifs du mémoire est d'automatiser ce processus en développant un algorithme glouton. Nous proposons ensuite une amélioration qui consiste à remplacer l'algorithme glouton par un modèle global résolu directement par Xpress-MP qui permettra d'avoir une vue globale pour le calcul des transferts. De part sa nature, ce problème est complexe et pour être résolu, nous proposons une approche de résolution en deux phases qui,

contrairement à un algorithme glouton utilisé en pratique, fournit des solutions d'assez bonne qualité en des temps de calcul très raisonnables.

La première phase de résolution consiste en une approche séquentielle département par département sans permettre les transferts. Ce problème s'apparente au cas mono-département car seuls les employés rattachés au département seront considérés. La deuxième phase de résolution permettra quant à elle de ré-optimiser la solution obtenue dans la phase précédente en permettant cette fois-ci les transferts tout en respectant certaines règles de transfert et de la convention collective du travail.

Notre objectif principal se subdivise par la suite en deux objectifs spécifiques :

1. Développer des modèles de programmation mathématique pour résoudre efficacement le problème de chaque phase ;
2. Tester et comparer les résultats sur les jeux de données réels de grande taille.

Les critères de comparaison seront, entre autres, le temps de traitement, la valeur de la fonction objectif qui indique les nombres de sous et sur-couvertures finaux attestant ainsi la qualité de la solution obtenue.

1.3 Contributions

Dans ce projet de mémoire, nous développons en premier lieu un algorithme glouton permettant d'automatiser le processus des transferts d'employés entre les départements. Nous proposons ensuite un modèle global qui permet d'avoir une vue globale pour le calcul des transferts. Ce dernier est résolu directement par Xpress-MP et permet de traiter le modèle au complet comme un problème mixte en nombres entiers. Ensuite, nous introduisons une nouvelle méthode de résolution en deux phases. Cette approche de résolution à deux niveaux décisionnels s'avère efficace pour les magasins à grandes surface. En effet, nous montrerons dans ce mémoire que les temps de calcul sont nettement moindres avec cette méthode.

1.4 Organisation du mémoire

Le reste du mémoire est présenté comme suit : le chapitre 2 commencera par présenter une revue de littérature portant sur les problèmes de construction des horaires. Plusieurs sujets seront élaborés, entre autres, la construction des quarts de travail, l'affectation des activités aux quarts, le contexte multi-département, etc. Nous terminerons ce chapitre en

faisant ressortir la contribution de ce projet de recherche par rapport à la littérature. La problématique de ce mémoire sera définie dans le chapitre 3 en fournissant une explication détaillée de la problématique étudiée et une présentation du modèle global. L'heuristique en deux phases sera présentée dans le chapitre 4. La première partie de ce chapitre présentera la première phase de résolution sans transferts tandis que la seconde partie illustrera la ré-optimisation effectuée en considérant cette fois-ci les transferts. Le chapitre 5 présentera les jeux de données testés de même que les résultats numériques obtenus. Nous discuterons par la suite de l'avantage de chaque modèle proposé par rapport aux autres. Enfin, le chapitre 6 présentera la conclusion et les travaux futurs.

CHAPITRE 2

REVUE DE LITTÉRATURE

Les problèmes de planification de personnel ont été beaucoup étudiés et de nombreux travaux ont été effectués surtout au cours des dernières décennies. Cet attrait peut s'expliquer par plusieurs facteurs qui sont surtout de nature économique. En effet, le coût de la main-d'œuvre constitue la majeure partie des coûts directs de la plupart des entreprises et par conséquent, une réduction de ce coût peut s'avérer très bénéfique. Cela se réalise notamment par la mise en œuvre d'un nouvel horaire ou par l'optimisation de l'horaire déjà existant. Cette revue de littérature se veut donc un aperçu des travaux relatifs à la confection des horaires de personnel. Étant donné le nombre élevé et diversifié des travaux qui traitent ce sujet, nous allons établir un cadre de travail traitant les sous-problèmes en lien avec notre problématique.

2.1 Classification des problèmes d'horaires de personnel

Selon den Bergh *et al.* (2013) qui ont élaboré une revue de littérature récente, divers axes d'orientation peuvent être utilisés pour classer les travaux liés au problème d'horaire de personnel. Il s'agit : (1) Des caractéristiques du personnel et de la définition des quarts, notamment les informations sur les critères personnels des employés, les préférences du temps plein ou du temps partiel, etc. ; (2) Des contraintes, des mesures de performance et de la flexibilité ; (3) Des méthodes de résolution et de l'intégration de l'incertitude indiquant quel type de solution ou d'évaluation technique est utilisé et enfin ; (4) Des domaines d'application indiquant dans quels domaines se situe le problème ainsi que son implémentation pratique. À cette classification peut s'ajouter d'autres paramètres comme les règles d'ancienneté et la nature de la demande en personnel (déterministe ou stochastique, etc).

D'après ces auteurs, l'une des premières méthodes de classification du problème d'horaire de personnel a été proposée par Baker et Magazine (1977) qui distinguent trois principaux groupes : La construction des quarts anonymes ou « *Shift scheduling* », la planification des jours de congé ou « *Days-off scheduling problem* » et la construction simultanée de cycles et de quarts anonymes, ou encore « *Tour scheduling problem* » combinant les deux premiers. Par la suite, Ernst *et al.* (2004a,b) ont quant à eux proposé une revue de littérature qui présente le processus de construction d'horaires en 6 modules plus spécifiques :

1. La modélisation de la demande « *Demand modeling* »,
2. La planification des jours de congé « *days-off scheduling* »,
3. La construction des quarts de travail « *shift scheduling* »,
4. La construction d'horaires anonymes « *line of work construction* »,
5. L'affectation des tâches aux quarts « *task assignment* » et
6. L'affectation des horaires aux employés « *staff assignment* ».

La modélisation de la demande consiste à établir une prévision du nombre d'employés requis durant l'horizon de planification afin d'assurer un service continu dans l'entreprise. Cette modélisation est importante car elle va déterminer toute la planification des jours de congé et des quarts de travail afin de maintenir le service de façon continue. Dans le cadre de ce projet de maîtrise, les demandes en nombres d'employés ainsi que les jours de congé sont fournis à l'avance.

Le problème de planification des jours de repos ou « *days-off scheduling* » consiste à établir un ensemble de jours travaillés et de repos au cours de l'horizon de planification tout en satisfaisant la demande et en respectant les normes de travail en vigueur. L'horizon de planification s'étale généralement sur une ou plusieurs semaines. Ainsi par exemple, pour un horizon de planification d'une semaine entière (7 jours), une planification des jours de congé établirait une période de repos de 2 jours consécutifs et une période de travail de 5 jours. Ce type de problème est surtout appliqué dans les secteurs où la présence permanente des employés est requise. Il s'agit par exemple des centres hospitaliers ou des postes de police. Dans ces secteurs, les employés ne peuvent pas tous prendre des jours de congé simultanément. D'autres occurrences du problème sont apparues comme la planification cyclique des jours de repos ou l'allocation de 3 ou 4 jours de repos successifs par semaine. Cette dernière approche est particulièrement intéressante car elle permettrait de satisfaire la demande tout en minimisant le nombre d'employés nécessaires.

La construction des quarts de travail ou « *shift scheduling* » quant à elle consiste à expliciter chaque quart possible en donnant son heure de début, son heure de fin ainsi que l'emplacement de une ou de plusieurs périodes de pause à l'intérieur même du quart. Comme mentionné précédemment, Baker et Magazine (1977) ont établi que les deux problèmes précédents pouvaient être combinés en un seul problème global qui consisterait à spécifier les jours de travail et de repos pour un horizon de planification et à définir les heures de début et de fin pour les quarts tout en intégrant les périodes de pause : il s'agit du problème général d'ordonnancement du personnel, ou encore « *Tour scheduling problem* ».

L'affectation des tâches aux quarts ou « *task assignment* » donne lieu à deux contextes différents : le cas mono-activité et le cas multi-activité. En effet, pour le cas mono-activité où le travail à effectuer durant un quart donné est d'une seule nature et continu, on utilise le terme tâche ou activité pour signifier le seul type de tâche effectué durant le quart. Par contre, pour le cas multi-activité, le travail effectué durant un quart donné est un travail interruptible subdivisé en une séquence d'activités différentes. Dans ce cas, le problème d'affectation consistera à désigner les activités qui seront effectuées dans chacun des quarts ainsi que les périodes de début et de fin pour chacune d'entre elles.

Tandis que la construction d'horaires anonymes ou « *line of work construction* » se fait indépendamment des employés, l'affectation des horaires aux employés ou « *staff assignment* » se fera en affectant un horaire à chaque employé. Pour ce dernier, on tiendra alors compte, en plus des contraintes de réglementation des normes du travail et de la demande prévue, de ses préférences ainsi que d'autres contraintes du problème comme par exemple les qualifications pour les différentes tâches, etc.

L'intégration de ces modules diffère selon la complexité et la nature du problème traité. Certains peuvent même être regroupés dans un même processus. Cette classification qui est parmi les plus récentes dans le milieu, est appliquée dans les problèmes d'horaires de personnel de divers milieux : système de transport, centre d'appels et domaine médical.

Une autre classification des problèmes de construction de quarts de travail basée sur les méthodes de résolution a été proposée par Ernst *et al.* (2004a,b). D'après les auteurs, les problèmes peuvent être résolus en utilisant une ou plusieurs des 5 catégories à savoir :

1. Modélisation de la demande,
2. Approches par intelligence artificielle,
3. Programmation par contraintes,
4. Méta-heuristique et
5. Approches par programmation mathématique.

Dans ce mémoire, nous nous intéressons aux problèmes de construction de quarts de travail dans un contexte multi-départements ainsi qu'à l'approche de résolution par programmation mathématique (PM). Dans la section 2.2, nous passons en revue la construction des quarts de travail. Par la suite, nous détaillerons l'affectation des activités dans le cas du travail interruptible dans la section 2.3. Pour finir, nous parlerons dans la section 2.4 du contexte multi-départements qui permet de couvrir les courbes de demande de plusieurs départements.

2.2 Construction des quarts de travail

La construction des quarts de travail porte sur la conception des horaires quotidiens du personnel. Les modèles pour la planification des employés ont une longue histoire dans la littérature de la recherche opérationnelle et figurent parmi les problèmes les plus anciens. Plus précisément, il s'agit d'élaborer des quarts de travail anonymes qui seront par la suite affectés aux employés ou de combiner les deux étapes pour confectionner des quarts personnalisés. De façon générale, la durée d'un quart complet est répartie en plusieurs périodes de planification 15 min, 30 min ou 1h selon les spécifications du problème. Ainsi, la construction des quarts de travail consistera à définir les heures de début et de fin de chaque quart de travail afin de satisfaire la demande en nombre d'employés pour chaque période de planification. Pour aborder ce problème, plusieurs auteurs ont eu recours à diverses méthodes. Parmi les plus anciennes formulations mathématiques figurent celles d'Edie (1954) et Dantzig (1954).

S'adressant à un problème de circulation aux postes de péage, Edie (1954) a utilisé une combinaison d'analyse empirique et des formules pour les systèmes de files d'attente stationnaires pour gérer les besoins en personnel nécessaire pour assurer le niveau de service spécifié. Peu de temps après, Dantzig (1954) se référant au travail précédent d'Edie, montre que ce problème pouvait se formuler comme un problème de recouvrement sous forme d'un programme linéaire en nombres entiers. L'idée générale était d'explicitier chaque combinaison de quart possible en donnant une heure de début, une heure de fin et une période de pause précise. Il montre par ce fait que cette formulation peut fournir un horaire de personnel pour répondre aux exigences spécifiées pour chaque période de planification et cela à moindres coûts.

2.2.1 Modèle de recouvrement de Dantzig

Le modèle explicite développé par Dantzig se présente comme suit. Étant donné un calendrier spécifiant le nombre de postes de péage ouverts nécessaires tout au long de la journée et à partir duquel peut être déterminé le nombre total d'heures-personnes requis, le problème qui se pose est celui de déterminer le nombre de collecteurs de péage nécessaires pour maintenir le nombre prévu de postes ouverts. Un quart de travail pour chaque collecteur de péage a une durée de 7 h. Cela se rapporte facilement à un modèle de $3\frac{1}{2}$ h avant la pause repas, 30 min de pause repas et $3\frac{1}{2}$ h après la pause. Une journée de planification est divisée en périodes d'une durée de 30 min chacune. Comme le montre le tableau 2.1 par exemple, le quart j_1 comprend une pause de 9h30 à 10h et le quart j_6 de 14h30 à 15h.

Soient :

- J : l'ensemble de tous les quarts de travail ;
- I : l'ensemble des périodes de planification ;
- x_j : le nombre de personnes à assigner au quart j ;
- a_{ij} : $\begin{cases} 1 & \text{si } i \text{ est une période travaillée (et non de pause) dans le quart } j \\ 0 & \text{sinon ;} \end{cases}$
- b_i : le nombre de postes de péage ouverts requis pour la période i .

L'affectation serait choisie de telle sorte que :

$$\sum_{j \in J} a_{ij} x_j \geq b_i \quad \forall i \in I \quad (2.1)$$

$$x_j \geq 0 \text{ et entier} \quad \forall j \in J \quad (2.2)$$

L'objectif est donc de minimiser le nombre total de personnes affectées satisfaisant (2.1) :

$$\text{Min} \sum_{j \in J} x_j \quad (x_j \geq 0) \quad (2.3)$$

Selon l'auteur, d'autres périodes de pause peuvent être ajoutées avant ou après la période de pause repas prise au milieu du quart. Cela pourrait par contre nécessiter de subdiviser les périodes de planification aux 15 min au lieu des 30 min précédemment spécifiées. Par conséquent, le nombre de variables serait multiplié par k^2 où k est le nombre de différents moments où peuvent avoir lieu les pauses dans la période de pré ou post-repas durant le quart de travail. Conscient du nombre croissant de variables que cela pourrait engendrer, l'auteur propose une méthode de résolution qui retiendrait en premier lieu les quarts avec une pause repas au milieu et ajouterait les autres variables avec plus de pauses comme raffinement à la solution obtenue par le modèle restreint.

Tableau 2.1 Exemple des quarts de travail tel que défini par Dantzig (1954)

Quart	Heure de début	Durée (h)	Période de pause
j_1	6h	7	9h30-10h
j_2	7h	7	10h30-11h
j_3	8h	7	11h30-12h
j_4	9h	7	12h30-13h
j_5	10h	7	13h30-14h
j_6	11h	7	14h30-15h

Cette représentation de Dantzig engendre une augmentation de la complexité et de la taille du problème notamment en introduisant une flexibilité sur les durées et les emplacements des pauses ou sur les périodes de début ou la longueur des quarts de travail. Le problème devient donc très difficile sinon impossible à résoudre à l'optimalité pour de grandes instances.

Par ailleurs, cette approche de modélisation en un problème de recouvrement ne tient pas compte de la borne supérieure en nombre d'employés, ce qui n'est pas le cas dans la vie réelle où le nombre de ressources est généralement limité.

Dans le même ordre d'idées, la modélisation de la demande faite sous forme de contraintes rigides n'autorise que des surplus d'employés le long des périodes de planification. Or, dans un contexte plus général de la vie réelle, il pourrait être même impossible de toujours satisfaire la demande en nombre d'employés lorsque l'offre en nombre d'employés est inférieure à la demande. Ainsi, Baker (1976) et Easton et Mansour (1999) proposent une formulation basée sur l'approche de programmation par objectifs. Le but est de minimiser les déviations possibles (surplus ou manque d'employés), ce qui se rapproche de la formulation de notre fonction objectif.

Pour contrer ces limitations, de nouvelles formulations dites "implicites" ont été développées notamment par Bechtold et Jacobs (1990) et Aykin (1996). Un des éléments clés de cette approche consiste à utiliser des variables pour représenter les quarts de travail types au lieu d'explicitier chaque quart. Cela a pour avantage de découpler les deux problèmes de construction de quarts et d'affectation des pauses.

2.2.2 Modèles implicites de Bechtold et Jacobs (1990)

Bechtold et Jacobs (1990) définissent une seule variable pour tous les *quarts-types* ayant une même période de début, une même période de fin et une fenêtre de temps "*break window*" durant laquelle la pause devra être prise. Au lieu d'énumérer explicitement toutes les possibilités de placement de pause, ils ont défini deux types de variables : Les variables de types de quart et les variables de pause. Suivant cette logique, les auteurs ont introduit des contraintes dites "*forward*" et "*backward*" ainsi qu'une contrainte d'égalité. Ces contraintes assurent que pour chaque quart présent dans la solution optimale, une pause unique est attribuée à chaque employé et cela à l'intérieur de la fenêtre de temps spécifiée.

Aykin (1996) quant à lui propose une formulation implicite qui se distingue de celle de Bechtold et Jacobs par la possibilité de placer plusieurs types de pause à l'intérieur même des quarts.

Le modèle implicite proposé par Bechtold et Jacobs (1990) suppose l'absence d'extra-chevauchement qui fait que l'intervalle de pause d'un type de quart donné soit strictement inclus dans l'intervalle de pause d'un autre quart. Plus tard, Addou et Soumis (2007) fournissent une généralisation qui est valide sans cette hypothèse en ajoutant un minimum de contraintes. Et aussi, dans le cas où il y a de l'extra-chevauchement, les auteurs réduisent l'ensemble de leurs contraintes à un sous-ensemble dont la cardinalité reste inférieure au nombre de contraintes introduites par Bechtold et Jacobs.

2.2.3 Grammaires et langages formels

Une alternative aux modèles implicites et explicites est l'utilisation des langages formels et des grammaires pour modéliser les contraintes de travail. Ainsi, Côté *et al.* (2007) proposent un modèle en nombres entiers basé sur un langage régulier pour formuler les contraintes définissant un quart de travail et représenter tous les quarts réalisables en utilisant une formulation sous forme de flux dans un réseau. Côté *et al.* (2009) ont étendu ces résultats en utilisant les grammaires contextuelles pour modéliser les problèmes de confection des quarts de travail. À partir d'une telle grammaire, ils génèrent un modèle en nombres entiers basé sur les variables qui définissent implicitement tous les quarts de travail réalisables pour un employé. Cette formulation a pour inconvénient de générer un grand nombre de variables si le nombre d'employés augmente. En outre, ce modèle présente des problèmes de symétrie dans le cas où la plupart des employés se ressemblent.

Côté *et al.* (2011) présentent une nouvelle formulation implicite utilisant une grammaire contextuelle et qui se veut une amélioration des modèles précédents. Résolue par génération de colonnes, cette formulation se sert de la grammaire pour modéliser les contraintes dans le sous-problème. À partir de la grammaire, ils développent un programme en nombres entiers dont la relaxation linéaire est semblable à celle du modèle de recouvrement introduit par Dantzig (1954). Cette formulation fournit de meilleures performances en temps de calcul par rapport aux modèles implicites typiques de la littérature et possède l'avantage de traiter aussi les cas multi-activités qui ne peuvent être modélisés par les formulations implicites existantes.

2.3 Affectation des activités aux quarts

Parmi les auteurs qui ont traité le problème d'affectation des activités aux quarts récemment, nous pouvons citer Lequy *et al.*. Les auteurs distinguent deux catégories pour le travail d'un employé à savoir les tâches et les activités. Une tâche est un travail non interruptible comme par exemple replacer un étalage tandis qu'une activité est un travail interruptible à tout moment en remplaçant un employé par un autre. Dans les magasins à grande surface par exemple, nous pouvons définir le travail de caissier comme une activité. Dans un contexte multi-activités, il est possible d'affecter plusieurs activités à un quart donné.

Lequy *et al.* (2012a,b, 2013) traitent trois problèmes différents. Dans un premier lieu, Lequy *et al.* (2012a) considèrent uniquement les activités et traitent le problème d'affectation des activités aux quarts. Pour cela, trois modèles de résolution en nombre entiers et plusieurs méthodes de résolution ont été proposées. Parmi les méthodes proposées, les auteurs affirment que la plus efficace reste une heuristique de type horizon fuyant basé sur un programme en nombres entiers résolu par génération de colonnes.

Le deuxième problème traité par les auteurs dans Lequy *et al.* (2012b) est celui d'affectation de tâches et d'activités multiples. Ce problème est une extension du précédent où on ajoute le concept de tâche individuelle effectuée par un seul employé pour une durée fixe. Pour résoudre ce problème, les auteurs proposent une résolution en deux phases pour les instances de grande taille car la méthode de résolution globale échoue. Dans la première phase, les tâches sont affectées a priori grâce à un modèle d'approximation résolu heuristiquement. La deuxième phase de résolution quant à elle résout le problème global par génération de colonnes en affectant les activités tout en autorisant la réaffectation des tâches autour des affectations proposées lors de la première phase.

Finalement, le dernier problème traité par les auteurs Lequy *et al.* (2013) et qui est une extension du problème précédent, introduit la notion de travail en équipe. Autrement dit, il s'agit d'affecter des tâches en équipe et des activités multiples. Le concept de tâches en équipe suppose un ensemble de tâches effectués simultanément par plusieurs employés. La méthode de résolution proposée est basée sur deux modules d'affectation impliqués dans une descente à voisinage variable. Le premier module affecte approximativement les tâches tandis que le deuxième module affecte les activités tout en réaffectant les tâches.

Omari (2002) a traité le cas d'affectation des activités dans un contexte multi-activités. L'auteur propose une formulation comme un problème de réseau multi-flots où les unités de flots représentent des quarts de travail. Utilisant une décomposition de Dantzig-Wolfe, le problème maître est résolu par énumération implicite tandis que les sous-problèmes sont résolus par programmation dynamique.

Vatri (2001) a quant à lui étendu le modèle de Omari (2002) pour construire des quarts personnalisés en même temps que l'affectation des activités aux quarts. Il ajoute ainsi la construction des quarts et l'affectation des quarts aux employés tout en permettant une certaine flexibilité dans la définition des quarts.

Bouchard (2004) a proposé une résolution décomposée en trois étapes à savoir la construction des quarts avec pauses, l'assignation de ces quarts aux employés et l'assignation des activités aux quarts. Pour cela, l'auteur a proposé des heuristiques au niveau des sous-problèmes pour pouvoir accélérer la résolution du modèle. Aussi, l'auteur permet de remettre en cause, à chaque itération de génération de colonnes, les débuts et fins des quarts ainsi que l'emplacement des pauses, ce qui a pour inconvénient de générer des temps de calcul élevés.

D'autres auteurs se sont aussi adressés au problème d'affectation des activités aux quarts. Bard et Wan (2006) se sont intéressés à ce problème dans un contexte où les employés se déplacent entre les différents postes de travail (WSGs ou *Workstation group*). Leur méthode de résolution est une combinaison entre un modèle en nombres entiers et une méthode de recherche tabou qui produit des résultats intéressants.

Finalement, Bouchard (2008) s'intéresse au problème d'affectation d'activités aux quarts prédéterminés. Dans un premier lieu, l'auteur présente une méthode de recherche tabou associée à un programme de réduction qui permet de réduire de façon significative la complexité

combinatoire du problème. Par la suite, l'auteur propose quatre différents modèles en nombres entiers pour résoudre le problème d'affectation d'activités. Deux des modèles proposés ont permis de résoudre rapidement les instances.

2.4 Cas multi-départements

Dans le cadre de ce projet de recherche, nous introduisons une nouvelle hiérarchie qui peut être appliquée à l'ensemble de l'organisation. La plupart des entreprises sont subdivisées en plusieurs départements, qui peuvent être plus ou moins indépendants les uns des autres. Un département peut désigner une branche d'activité d'une entreprise. Ainsi par exemple, on peut retrouver le département de production, le département du service à la clientèle, etc. Généralement, la couverture de la demande dans une entreprise se fait séparément pour chaque département en couvrant la demande d'un département par l'ensemble des employés qui lui sont rattachés.

Par ailleurs, une autre approche de résolution de plus en plus considérée consiste à tenir compte du cas multi-départements où la résolution se fait simultanément pour l'ensemble des départements de l'entreprise en permettant un partage d'employés entre les départements.

Les travaux rencontrés dans la littérature traitent les problèmes multi-départements en adoptant une méthode de résolution séquentielle en deux phases. La première phase, semblable au problème mono-département, résout le problème en agrégeant tous les départements en un seul. Ensuite, les horaires obtenus dans la première phase sont désagrégés pour considérer les transferts entre les différents départements.

Dahmen *et al.* (2011) propose un modèle de résolution en deux étapes qui se rapproche de la définition de notre problème : une première étape dite d'agrégation et une seconde étape dite de désagrégation. L'étape d'agrégation permet de convertir le problème initial en un problème dit *agrégé* défini en fonction de périodes plus grossières à l'opposé du problème non-agrégé qui est défini en fonction de périodes plus petites. Une méthode de génération de colonnes avec trois stratégies de branchement différentes est alors utilisée pour résoudre le modèle agrégé de plus petite taille.

L'étape de désagrégation utilise quant à elle la solution de l'étape précédente pour construire un horaire désagrégé conforme aux règles de travail tout en essayant de s'ajuster le plus possible aux courbes de demande initiales. La méthode de résolution est alors de décomposer

le problème sous forme de graphe dont les sommets sont des départements et les arcs les informations agrégées de transfert.

CHAPITRE 3

PROBLÉMATIQUE ET MODÈLE GLOBAL

Le présent chapitre fournit une description détaillée de la problématique étudiée dans ce mémoire. Le problème porte sur la construction d'horaires de personnel dans un contexte multi-départements où chacun des départements est mono-activité. Un tel problème trouve son application dans plusieurs secteurs comme les magasins à grande surface, les hôpitaux, les centres d'appels, etc. Nous procédons en premier lieu à une revue des concepts qui seront utilisés par la suite tout au long de ce mémoire. La section 3.1 aborde le cœur proprement dit de la problématique en présentant le cadre conceptuel du problème traité et les définitions nécessaires à la formulation de ce dernier. Ensuite, nous présentons un modèle global dans la section 3.2.

3.1 Problème multi-départements avec transferts

Dans le jargon des entreprises, un département est une subdivision de services au sein des organisations, administrations ou entreprises. Le problème que nous abordons traite le cas multi-départements où chaque département se spécialise à l'offre d'un service. Cela se retrouve notamment dans les grandes entreprises où le travail peut se diviser en plusieurs départements plus ou moins indépendants. De ce fait, la gestion de la main-d'œuvre dans de telles entreprises se fera en rattachant un ensemble d'employés à chacun des départements de l'entreprise. Chaque employé de l'entreprise se retrouve ainsi rattaché à au moins un département de l'entreprise selon l'expérience acquise au cours de sa carrière.

Chaque département de l'entreprise dispose, en général, du nombre d'employés requis pour chaque période de l'horizon de planification. Aussi, chaque département peut présenter un ou plusieurs goulots d'étranglement correspondant à des périodes où l'offre en nombre d'employés ne peut pas répondre à la demande. Comme ces goulots se déplacent partout d'un département à un autre durant la journée, les transferts inter-départementaux d'employés ayant les qualifications requises s'avèrent être une solution de choix pour répondre à la demande. Dans les lignes qui vont suivre, nous allons décrire le problème et définir la terminologie utilisée.

3.1.1 Définitions et terminologie

Le problème se décrit comme suit. Soit une entreprise subdivisée en plusieurs départements et faisant appel à un ensemble d'employés. L'ensemble des départements est noté D tandis que celui de tous les employés de l'entreprise est noté E . Un employé possède des qualifications qui lui permettent d'effectuer des activités dans différents départements. Durant son quart, l'employé ne pourra travailler que dans les départements pour lesquels il est qualifié. Chaque employé $e \in E$ est qualifié à l'origine pour un département spécifique mais peut aussi posséder d'autres qualifications qui lui permettront d'être transféré. Cela nous amène à définir deux types de qualifications.

Qualification primaire : Nous désignons par qualification primaire, le département auquel l'employé est rattaché à l'origine. Il s'agit donc de son *département d'origine*. Nous considérons que chaque employé e a une qualification primaire d_e qui est connue à l'avance. Cela introduit l'ensemble d'*employés internes* E_d^{int} pour chaque département $d \in D$. Nous définissons E_d^{int} comme étant l'offre en qualifications primaires du département d , i.e., l'ensemble des employés ayant d comme département d'origine.

Qualification(s) secondaire(s) : C'est l'ensemble de départements auxquels un employé est rattaché en dehors de son département d'origine. Cet ensemble de départements sera noté D_e . L'employé peut être affecté à un département seulement s'il est qualifié pour ce dernier. Contrairement à la qualification primaire, un employé peut être qualifié pour plusieurs départements secondaires aussi appelés *départements d'accueil*. Ces derniers sont aussi connus d'avance. Pour chaque département d , nous définissons l'ensemble E_d^{ext} comme étant l'ensemble d'*employés externes* ayant le département d comme qualification secondaire. C'est à partir de ces ensembles que seront établis les transferts inter-départementaux.

On désigne par *horizon de planification*, l'ensemble de périodes de temps *successives* appelées *périodes de planification* et indicées dans l'ordre chronologique par des entiers positifs. L'ensemble de ces entiers est noté I . La durée de ces périodes peut varier selon les spécifications du problème ou les besoins de la modélisation. Les périodes de planification sont généralement de durées égales et sont représentées par l'heure de début et leur durée.

Soit J l'ensemble des jours constituant l'horizon de planification. Chaque jour $j \in J$ est constitué par son ensemble de périodes de planification. Par exemple, si l'horizon de planification est d'une semaine entière ($|J| = 7$) répartie en périodes de planification de 15 minutes

chacune, on aura $I = \{1, 2, \dots, 96\}$ pour le premier jour où 96 correspond à 24 fois le nombre de périodes de planification dans une heure, soit 4.

Dans le reste du document, l'horizon de planification considéré sera continu et égal à une semaine entière, soit du lundi au dimanche. Comme pour les périodes de planification, les jours sont indicées dans l'ordre chronologique par des entiers positifs. Ainsi par exemple, j_1 représente le premier jour de l'horizon de planification soit le lundi tandis que j_7 représente le dernier jour de l'horizon de planification soit le dimanche. Dans la plupart des magasins à grande surface ou dans les hôpitaux, un quart de travail peut s'étaler sur deux jours consécutifs, ce qui augmente la complexité du problème car le problème ne peut pas être résolu un jour à la fois mais doit être considéré dans son ensemble durant tout l'horizon. Aussi, les périodes de planification auront une durée égale à 15 minutes. Par conséquent, la première période de l'horizon de planification $i_1 \in j_1$ est définie sur l'intervalle $[0h : 00, 0h : 15[$ du lundi tandis que la dernière période $i_{672} \in j_7$ est définie sur l'intervalle $[23h : 45, 0h : 00[$ du dimanche.

Bien qu'un employé peut travailler tous les jours durant l'horizon, nous supposons que chaque employé possède a priori une séquence de jours de travail notée J_e qui définit un ensemble de jours travaillés et de congés pour l'employé $e \in E$. Par exemple, l'employé e_1 peut avoir une séquence $J_{e_1} = \{j_1, j_2, j_5, j_6, j_7\}$ qui signifie qu'il a deux jours de repos à savoir j_3 et j_4 .

Chaque département $d \in D$ possède une courbe de demande. Elle représente la demande en nombre d'employés pour chaque période de planification de l'horizon traité. Pour chaque département $d \in D$ et chaque période $i \in I$, nous supposons que q_d^i employés sont requis pour assurer le niveau de service visé. Dans l'exemple du tableau 3.1, q_d^i est défini pour chaque période $i \in \{1, 2, \dots, 12\}$ (i.e., les trois premières heures de l'horizon de planification).

Tableau 3.1 Exemple d'une courbe de demande pour un département

i	1	2	3	4	5	6	7	8	9	10	11	12
q_d^i	12	12	12	12	8	8	8	8	4	4	4	2

Un quart de travail désigne l'ensemble de périodes de travail continues assignées à chaque employé. Soit Q l'ensemble des quarts de travail. Dans ce qui suit, nous allons définir formellement cet ensemble. Pour cela, il convient de définir premièrement la notion de *bloc de*

travail qui désigne un ensemble de périodes continues effectuées par un employé dans un département donné. Chaque bloc de travail est défini par son heure de début, sa durée et le département au sein duquel le bloc est exécuté. Durant un quart de travail, on dira qu'un employé est transféré s'il est amené à travailler dans un département d'accueil. Un transfert se définit alors comme étant le bloc de travail où l'employé est affecté dans un département autre que son département d'origine. Par ailleurs, un transfert peut soit commencer en début du quart de travail et se terminer à n'importe quel moment du quart selon le quart considéré, soit commencer pendant le quart et se terminer à la fin du quart.

Un ensemble de règles appelés *règles de transfert* régit les transferts inter-départementaux. Le but est de faire travailler chaque employé dans son département d'origine tant qu'on a besoin de lui. Si l'employé devient improductif dans son département d'origine et que l'un des départements d'accueil présente une demande de ressources, il peut être transféré pour combler la demande dans le département d'accueil. Le but des règles de transfert est donc de fournir un cadre conceptuel pour les transferts dans le but de garantir au maximum la satisfaction des employés tout en respectant les règles de la convention collective du travail.

Durant son quart, chaque employé ne peut être transféré qu'au plus une fois durant le même jour. En d'autres termes, un employé peut travailler dans au plus deux départements par quart. Quand il doit travailler dans deux départements, l'un d'eux doit nécessairement être son département d'origine. Quatre cas peuvent ainsi se présenter pour un employé $e \in E$:

- L'employé travaille dans son département d'origine d_e durant son quart au complet,
- L'employé commence son quart dans son département d'origine d_e et est transféré par la suite dans un département d'accueil $d \in D_e$,
- L'employé commence son quart dans un département $d \in D_e$ et le termine dans son département d'origine d_e ,
- L'employé fait tout son quart dans un département $d \in D_e$.

De plus, chaque employé doit passer au minimum un certain pourcentage de sa charge horaire totale durant l'horizon de planification dans son département d'origine. En effet, bien que les transferts peuvent avoir des effets positifs sur les employés en leur permettant de découvrir d'autres activités et acquérir plus d'expériences dans d'autres départements, un nombre élevé de transferts peut s'avérer moins bénéfique pour la productivité de l'entreprise lorsque l'employé est transféré pour une grande partie de sa charge horaire durant l'horizon de planification ou plusieurs fois durant son quart.

Un quart peut donc être constitué par un seul bloc de travail ou deux blocs de travail

selon que l'employé est transféré ou pas vers un département d'accueil. Comme le montre la figure 3.1, un quart de travail peut être soit complètement interne lorsque l'employé qui l'effectue reste dans son département d'origine, soit complètement externe lorsqu'il est effectué au complet dans un département d'accueil ou mixte lorsqu'il est constitué de deux blocs dont l'un est effectué dans le département d'origine et l'autre dans un département d'accueil.

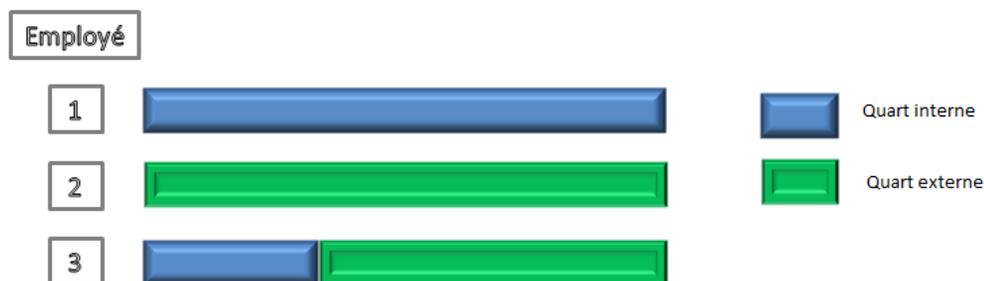


Figure 3.1 Types de quarts

Le tableau 3.2 présente les notations relatives à un quart de travail.

Tableau 3.2 Notations relatives à un quart de travail

Notation	Signification
s_q	Heure de début du quart q
h_q	Durée du quart q
h_q^b	Durée du bloc de début
h_q^f	Durée du bloc de fin
d_q	Département de réalisation du quart q
d_q^b	Département de réalisation du bloc de début
d_q^f	Département de réalisation du bloc de fin

Trois cas peuvent alors se présenter pour désigner un quart personnalisé q assigné à un employé e :

1. Un quart interne q désigné par le triplet (s_q, h_q, d_q) avec $d_q = d_e$
2. Un quart externe q désigné par le triplet (s_q, h_q, d_q) avec $d_q \neq d_e$
3. Un quart mixte q désigné par le quintuplet $(s_q, h_q^b, h_q^f, d_q^b, d_q^f)$ où l'un des départements d_q^b ou d_q^f est le département d'origine.

Pour des fins de simplification, le quintuplet $(s_q, h_q^b, h_q^f, d_q^b, d_q^f)$ sera aussi utilisé pour les deux premiers cas en posant $h_q^f = 0$ et en supposant que d_q^f représente un département fictif. En plus des règles de transfert citées précédemment, deux règles de la convention collective relatives aux horaires des employés s'appliquent. Ces règles sont liées en particulier à la définition des quarts de travail notamment l'heure de début et la durée. La première règle limite la charge totale travaillée pour chaque employé. En effet, la plupart des environnements de travail imposent une limite supérieure quant au nombre d'heures total effectuées par un employé durant tout l'horizon de planification. Cette limite sera notée L_{max} . Par exemple, une compagnie pour imposer une limite de 40 heures de travail par semaine. C'est cette limite qui sera imposée dans la formulation des modèles proposés. La deuxième règle impose une durée de repos minimale entre deux quarts consécutifs pour un employé donné. Ce repos est noté R_{min} . Supposons par exemple un quart de travail de $8h$ qui débute à $21h$ le jour j_1 pour finir à $5h$ du matin le jour j_2 . En considérant une durée de repos minimale de $10h$, le début du quart suivant ne pourra se faire qu'à partir de $15h$ le jour j_2 .

Soit Q_e^j l'ensemble des quarts admissibles pour un employé e le jour j . Un quart de travail est dit admissible pour un employé e durant le jour travaillé j si et seulement si l'heure le quart commence le jour j , sa durée respecte la charge totale maximale imposée par l'entreprise, s'il est réalisé au complet dans un département $d \in D_e$ ou si un des blocs du quart est réalisé dans le département d'origine d_e et l'autre bloc dans un département admissible $d \in D_e$.

Étant donné que la satisfaction de la demande en personnel ne peut pas toujours être satisfaite, on permet la sous-couverture et la sur-couverture moyennant des pénalités. Pour une période de planification donnée, si l'horaire généré offre plus d'employés que nécessaire d'après la courbe de demande, on parlera de période en sur-couverture, contrairement à une période en sous-couverture où la demande n'est pas complètement couverte. Dans l'exemple illustré au tableau 3.3, les périodes 5, 6 et 7 sont en sur-couverture tandis que les périodes 8, 9 et 10 sont en sous-couverture.

Les pénalités aussi appelés coûts de sous-couverture et de sur-couverture dépendent du département au sein duquel ils ont lieu. En effet, un manque de personnel peut être plus coûteux dans un département que dans un autre et dans certains cas, il peut être préférable d'avoir un surplus d'employés que d'en manquer.

Tableau 3.3 Exemple de sous et sur-couvertures

i	1	2	3	4	5	6	7	8	9	10
q_d^i	12	12	12	12	8	8	8	8	4	4
<i>Employés en service</i>	12	12	12	12	10	10	10	5	2	2

Un quart de travail implique deux types de coûts différents : le *coût de travail* et le *coût de transfert*. Ces coûts sont spécifiques à chaque employé. Alors que le coût de travail est proportionnel à la durée de tout le quart de travail, le coût de transfert est proportionnel uniquement à celle du bloc de travail où l'employé est transféré dans un département d'accueil. Il dépend de ce fait du département d'accueil. Soit c_e le coût horaire de travail pour l'employé e . Ce coût peut varier d'un employé à un autre en fonction de la politique salariale de l'entreprise, son expérience ou de son ancienneté. Le coût horaire de transfert c_e^d reflète la baisse de la productivité pouvant survenir lorsque l'employé e est transféré dans le département d . Les coûts de transferts ne sont pas réels mais servent plutôt d'incitatifs pour éviter les transferts. Considérons par exemple un coût de travail horaire de 12\$ pour un employé effectuant un quart de 8h comprenant un transfert de 2h. Si le coût horaire de transfert est de 5\$, le coût de travail total sera de 96\$ et celui de transfert sera de 10\$. Le coût de travail du quart considéré sera alors de 106\$.

3.1.2 Énoncé du problème

En se basant sur les définitions précédentes, le problème de construction des quarts de travail avec transferts inter-départementaux d'employés peut s'énoncer comme suit : Soit une entreprise composée d'un ensemble D de départements et faisant appel à un ensemble E d'employés. Chaque employé $e \in E$ de l'entreprise est rattaché à un département d'origine d_e et possède aussi des qualifications pour travailler dans d'autres départements. Chaque département $d \in D$ de l'entreprise dispose de l'ensemble E_d^{int} d'employés internes et peut accueillir d'autres employés E_d^{ext} s'ils sont qualifiés pour ce dernier. L'objectif du problème est d'assigner à chaque employé $e \in E$ de l'entreprise et pour chaque jour de sa séquence de jours travaillés $j \in J_e$, un quart choisi parmi son ensemble de quarts admissibles Q_e^j le jour j . L'objectif consiste à minimiser la somme pondérée des coûts de travail et de transfert ainsi que les coûts de sous et sur-couverture durant tout l'horizon de planification considéré. Les quarts assignés devront satisfaire les règles de la convention collective et les règles de transfert définies précédemment.

L'énumération explicite de tous les quarts de travail en permettant une grande flexibilité sur les heures de début, les durées et le placement des blocs de transfert augmente la complexité de ce problème. La difficulté à le résoudre dans des temps raisonnables croît considérablement lorsque la taille du problème augmente et cela constitue une difficulté majeure lorsque le problème est résolu dans son intégralité. Dans un autre sens, le modèle global permet de résoudre le problème à l'optimalité lorsque la taille du problème est suffisamment petite.

3.2 Modèle global

Dans cette section, nous proposons un modèle global qui permet de traiter le problème au complet comme un problème mixte en nombres entiers. Ce modèle a pour avantage de fournir des solutions optimales lorsqu'il parvient à résoudre les instances.

3.2.1 Construction des quarts de travail

La construction des quarts de travail se fera explicitement en fournissant l'heure de début, la durée et le département considéré pour chaque bloc constituant le quart. Cette modélisation se veut une généralisation du modèle de recouvrement introduit par Dantzing [1954] et appliqué au modèle multi-départements. Le modèle choisira un ensemble de quarts optimaux parmi l'ensemble de quarts de travail énumérés.

Une énumération explicite consiste donc à définir pour chaque quart q le quintuplet $(s_q, h_q^b, h_q^f, d_q^b, d_q^f)$ où $h_f = 0$ et d_q^f représente un département fictif pour les quarts à un bloc. Pour chaque employé, on énumère l'ensemble de quarts internes et externes personnalisés admissibles. Rappelons que pour qu'un quart soit admissible pour un employé e , sa durée doit respecter la charge totale maximale imposée par l'entreprise, il doit être réalisé au complet dans un département $d \in D_e$ ou un des blocs du quart doit être réalisé dans le département d'origine d_e et l'autre bloc dans un département admissible $d \in D_e$.

Nous considérons ce qui suit dans le cadre de ce projet :

- La durée minimale des quarts est de 6h et la durée maximale de 10h avec une discrétisation de la durée aux heures,
- La durée minimale d'un bloc interne ou externe est de 1h avec une discrétisation de la durée aux heures,
- Les heures de début des quarts sont aux 15min à l'intérieur de l'intervalle de début possible considéré.

3.2.2 Notations

En se référant aux notations définies à la section 3.1, définissons les variables de décision du problème ainsi que ses constantes avant de passer à la formulation proprement dite du modèle.

Variables de décision du problème On distingue trois types de variables de décision associés au problème considéré :

$$X_{eq}^j : \begin{cases} 1 & \text{si le quart } q \text{ est sélectionné pour l'employé } e \text{ le jour } j \\ 0 & \text{sinon} \end{cases}$$

U_d^i : nombre total de sous-couvertures dans la tranche i du département d

O_d^i : nombre total de sur-couvertures dans la tranche i du département d .

Paramètres et constantes du problème Le tableau 4.3 résume toutes les notations des ensembles considérés.

Tableau 3.4 Notations des paramètres et constantes du problème

Notation	Signification
h_q	Nombre d'heures travaillées dans le quart q
h_q^t	Nombre d'heures transférées dans le quart q
s_q	Heure de début du quart q
θ_{dq}^i	Vaut 1 si le quart q ayant d comme département d_q^b ou d_q^f couvre la période i , 0 sinon
R_{min}	Durée de repos minimale entre deux quarts consécutifs
L_{max}	Charge horaire maximale pour un employé durant l'horizon de planification
K_{min}	Pourcentage minimale de la charge horaire totale à faire dans le département d'origine
f_q^i	Vaut 1 si le quart q finit à la période i , 0 sinon
b_q^i	Vaut 1 si le quart q commence à la période i , 0 sinon
v_q^i	Vaut 1 si le quart q contient la période i , 0 sinon
r_d^i	Nombre d'employés requis au département d durant la période i
c_e	Coût d'une heure travaillée de l'employé e
c_{ed}^t	Coût d'une heure transférée de l'employé e dans le département d
c_d^u	Coût d'une sous-couverture dans le département d
c_d^o	Coût d'une sur-couverture dans le département d
δ	Coefficient de pondération du coût de travail
δ^t	Coefficient de pondération du coût de transfert
δ^u	Coefficient de pondération du coût de la sous-couverture
δ^o	Coefficient de pondération du coût de la sur-couverture

3.2.3 Formulation

Nous pouvons maintenant présenter une formulation pour notre problème.

$$Min \sum_{e \in E} \sum_{j \in J_e} \sum_{q \in Q_e^j} (c_e h_q X_{eq}^j + \delta^t c_{ed}^t h_q^t X_{eq}^j) + \sum_{d \in D} \sum_{i \in I} (\delta^u c_d^u U_d^i + \delta^o c_d^o O_d^i) \quad (3.1)$$

$$s.c. \sum_{e \in E_d} \sum_{j \in J_e} \sum_{q \in Q_e^j} \theta_{dq}^i X_{eq}^j + U_d^i - O_d^i = r_d^i \quad \forall d \in D, i \in I \quad (3.2)$$

$$\sum_{q \in Q_e^j} X_{eq}^j \leq 1 \quad \forall e \in E, j \in J_e \quad (3.3)$$

$$\sum_{j \in J_e} \sum_{q \in Q_e^j} h_q X_{eq}^j \leq L_{max} \quad \forall e \in E \quad (3.4)$$

$$\sum_{j \in J_e} \sum_{q \in Q_e^j} f_q^i X_{eq}^j + \sum_{j \in J_e} \sum_{q \in Q_e^j} \sum_{i' \in [i, i + R_{min}]} \left(b_q^{i'} X_{eq}^j + v_q^{i'} X_{eq}^j \right) \leq 1 \quad \forall i \in I, e \in E \quad (3.5)$$

$$\sum_{j \in J_e} \sum_{q \in Q_e^j} (h_q - h_q^t) X_{eq}^j \geq K_{min} \sum_{j \in J_e} \sum_{q \in Q_e^j} h_q X_{eq}^j \quad \forall e \in E \quad (3.6)$$

$$X_{eq}^j \in \{0, 1\} \quad \forall e \in E, j \in J_e, q \in Q_e^j \quad (3.7)$$

$$U_d^i \geq 0 \quad \forall d \in D, i \in I \quad (3.8)$$

$$O_d^i \geq 0 \quad \forall d \in D, i \in I \quad (3.9)$$

Le modèle global (3.1)-(3.9) traite tous les départements simultanément. Les transferts inter-départementaux sont modélisés à l'aide des quarts externes alors que les affectations à l'interne sont modélisés à l'aide des quarts internes.

L'objectif (3.1) est de minimiser la somme pondérée des quatre types de coûts à savoir les coûts de sous-couverture, les coûts de sur-couverture, les coûts des périodes travaillées et les coûts de transfert. Les contraintes (3.2) imposent la couverture de la demande pour tout département et toute période de l'horizon de planification. Ces périodes assurent que le nombre d'employés en service dans chaque département et pour chaque période soit égal au nombre requis plus le nombre d'employés en sur-couverture et moins le nombre d'employés en sous-couverture. Les contraintes (3.3) permettent d'affecter à chaque employé au plus un quart de travail durant une journée de travail. Les règles de la convention collective relatives aux horaires des employés sont modélisés à l'aide des contraintes (3.4) et (3.5). Les contraintes (3.4) assurent que la charge horaire maximale $L_{max} = 40h$ soit respectée pour chaque employé durant l'horizon de planification. Les contraintes (3.5) imposent qu'il y ait un repos minimal R_{min} attribué à chaque employé entre deux quarts consécutifs. Plus précisément, elles interdisent d'affecter à un employé un quart débutant dans la période $[i, i + R_{min}]$ si celui-ci s'est vu affecté un quart finissant à la période i . Chaque employé doit faire un certain pourcentage de sa charge horaire totale dans son département d'origine. Les contraintes (3.6) modélisent cette règle en permettant à chaque employé de passer au moins 70% de son temps travaillé dans son département d'origine. Finalement, les contraintes (3.7), (3.8) et (3.9) assurent l'intégrité et la non-négativité des variables de décision.

3.2.4 Méthode de résolution

Le modèle global formulé précédemment peut être résolu a priori par n'importe quel solveur de programmation linéaire mixte (MIP). Dans le cadre de ce projet de recherche, nous l'avons résolu directement en utilisant le solveur Xpress-MP. Comme nous le verrons dans le chapitre 5, l'immense taille du modèle pouvant résulter de la taille des instances de données rend ce modèle presque impossible à résoudre par un tel solveur dans des temps raisonnables.

Dans le chapitre 4, nous proposons une heuristique de résolution en deux phases qui fournit des solutions de bonne qualité pour des instances de grande taille pour lesquelles le modèle global échoue.

CHAPITRE 4

HEURISTIQUE EN DEUX PHASES

Le but de ce chapitre est de présenter une heuristique en deux phases pour résoudre le problème de confection d'horaires de personnel avec transferts inter-départementaux d'employés. Dans la section 3.2 du chapitre précédent, nous avons proposé un modèle global qui permet de traiter ce problème au complet comme un problème mixte en nombres entiers. L'heuristique en deux phases que nous proposons permet de résoudre les problèmes de confection d'horaires de travail personnalisés de grande taille en des temps rapides. Cette heuristique permettra de traiter les problèmes peu importe la taille des instances considérées en fournissant des solutions en des temps raisonnables.

La première phase de résolution consiste en une approche séquentielle département par département sans permettre les transferts. Il s'agit d'un problème mono-département car seuls les employés internes rattachés au département seront considérés. Pour résoudre ce problème, nous proposons un modèle en nombres entiers qui résout les instances à l'optimalité. Ensuite, la deuxième phase de résolution permettra quant à elle de ré-optimiser la solution obtenue dans la première phase en permettant cette fois-ci les transferts. Un algorithme glouton qui correspond à ce qui se fait en pratique et un modèle exact en nombres entiers seront présentés pour cette phase de ré-optimisation.

Ce chapitre s'organise comme suit. Dans la section 4.1, nous traitons la première phase de résolution à savoir la résolution séquentielle département par département. Nous commencerons par fournir le contexte général et les notations utilisées. Nous présentons la formulation du modèle dans la section 4.1.2. Ensuite, la deuxième phase de ré-optimisation sera élaborée dans la section 4.2 en présentant notamment l'algorithme glouton et le modèle exact dans la section 4.2.2.

4.1 Première phase : Problème séquentiel multi-départements sans transferts

La première phase de résolution consiste en une approche séquentielle département par département sans permettre les transferts. Contrairement au modèle global où on tient compte des qualifications secondaires pour chaque employé, cette première phase de résolution ne retient que l'ensemble des employés internes E_d^{int} attachés à chaque département $d \in D$.

Autrement dit, les départements sont traités d'une manière séquentielle un par un en ne considérant que l'ensemble d'employés ayant d comme département d'origine.

4.1.1 Définitions et terminologie

Le but de cette section est de formuler le problème de construction d'horaires de personnel restreint au département $d \in D$. Fort des définitions de la section 3.1, nous cherchons tout d'abord à définir l'ensemble Q des quarts de travail ainsi que l'ensemble Q_e^j des quarts admissibles pour chaque employé $e \in E_d^{int}$ et chaque jour $j \in J_e$. L'objectif sera d'affecter par la suite à chaque employé $e \in E_d^{int}$ et pour chaque jour de sa séquence de jours travaillés $j \in J_e$, un quart choisi parmi son ensemble de quarts admissibles Q_e^j .

Soit T l'ensemble des types de quarts qui servent à catégoriser les quarts quant aux heures de début admissibles. Chaque type de quart $t \in T$ est défini par un ensemble d'heures de début noté s_t . Nous considérons trois types de quarts avec une discrétisation de l'heure de début à toutes les 15 minutes :

1. Quarts du matin : $s_1 = [6h, 9h]$;
2. Quarts du soir : $s_2 = [12h, 14h]$;
3. Quarts de nuit $s_3 = [21h, 22h]$.

L'énumération explicite des quarts de travail Q se fera donc pour chaque type de quart $t \in T$, chaque durée admissible prise dans l'ensemble $\{6h, 7h, 8h, 9h, 10h\}$ et chaque département $d \in D$. Par ailleurs, les règles de la convention collective relatives aux horaires des employés définies dans le chapitre précédent s'appliquent également dans le contexte mono-département.

4.1.2 Formulation du modèle

Le problème séquentiel département par département peut donc se formuler comme le modèle global (3.1)-(3.9) restreint à un seul département $d \in D$ auquel on considère uniquement les employés internes E_d^{int} et on enlève les contraintes (3.6) imposant à chaque employé de passer un certain pourcentage de la charge horaire totale dans son département d'origine.

4.1.3 Méthode de résolution

Pour chaque département, nous résolvons le modèle (3.1)-(3.9) tel que défini à la section 4.1.2 avec Xpress-MP. La résolution de ce premier modèle donne une solution optimale au

problème d'affectation de quarts de travail aux employés internes de chaque département. Comme le montrent les résultats rapportés dans le chapitre 5, les temps d'exécution sont très rapides. Cela est dû au fait que, contrairement au modèle global où tous les départements sont traités simultanément en considérant les transferts, cette première phase de l'heuristique est restreint uniquement à un département et sans allouer les transferts. Le nombre de variables se trouve ainsi diminué.

La solution obtenue à cette première phase de résolution servira de point de départ à la seconde phase de résolution que nous nous proposons de définir dans les paragraphes qui vont suivre.

4.2 Deuxième phase : Ré-optimisation du problème multi-départements avec transferts

Le but principal de cette deuxième phase est de ré-optimiser la solution de la phase précédente en allouant cette fois-ci des transferts. Un employé peut donc être transféré de son département d'origine vers un autre département d'accueil s'il est qualifié pour ce dernier. Dans la section 4.2.1, nous allons définir le contexte général et les règles qui vont régir les transferts inter-départementaux. Par la suite, nous présenterons l'algorithme glouton ainsi que le modèle exacte en nombres entiers développés pour résoudre ce problème dans la section 4.2.2.

4.2.1 Contexte général

La ré-optimisation de la solution obtenue lors de la première phase de résolution se fait en ajoutant des transferts d'employés entre les départements et en traitant tous les départements simultanément. Après la résolution séquentielle des départements, les périodes en sous et sur-couverture sur tout l'horizon de planification sont connues pour chaque département. Les variables de décision X_{eq}^j permettent de connaître le quart q fait par chaque employé e le jour j . Cette deuxième phase se veut donc une continuité de la première car les employés seront transférés dans les départements d'accueil en se référant aux quarts effectués dans leurs départements d'origine.

Contrairement à la première phase de résolution où les quarts étaient juste internes, les notions d'employés externes E_d^{ext} , de quarts externes et de quarts mixtes seront ré-introduites dans cette deuxième phase. Rappelons qu'un quart est dit externe lorsqu'il est effectué au complet dans un département d'accueil et mixte lorsqu'il est constitué de deux blocs dont

l'un est effectué dans le département d'origine et l'autre dans un département d'accueil.

Pour être transféré, l'employé doit être qualifié pour le département d'accueil et devra satisfaire les règles de transfert établies.

Maximum d'un transfert par employé et par jour Chaque employé ne peut être transféré qu'une seule fois durant la journée. Ainsi, un employé ayant effectué son quart interne ne pourra pas être transféré dans un département d'accueil et revenir dans son département d'origine ou être transféré dans un deuxième département d'accueil par la suite.

Durée minimale d'un transfert La durée minimale d'un transfert est d'une heure. Comme les périodes de planification sont aux $15min$ chacune, on évite ainsi de transférer un employé pour une période de $15min$. L'ajout des transferts se fait donc par bloc d'une heure. Compte tenu de la durée des quarts, les transferts peuvent être de durées égales à $1h, 2h, 3h, \dots, 10h$.

Limite maximale du temps de travail alloué aux transferts Chaque employé doit passer au minimum un certain pourcentage de sa charge horaire totale durant l'horizon de planification dans son département d'origine. Cela permet à chaque employé de travailler dans son département d'origine le plus possible, ce qui correspond le plus souvent aux préférences des employés. Même si l'employé n'a pas atteint sa charge horaire totale, cette contrainte empêchera que l'employé soit transféré si la limite maximale a été atteinte.

Pénalité des transferts Chaque transfert est légèrement pénalisé. Comme pour le modèle global, on associera une pénalité C_{ed}^t pour chaque transfert de l'employé e dans le département d'accueil d .

Le but de la deuxième phase de résolution est de transférer des employés vers les départements présentant encore des sous-couvertures. On traite donc uniquement les périodes encore en sous-couverture pour chaque département. Soit i^* une période présentant une sous-couverture dans le département d^* après la résolution séquentielle. Pour combler cette sous-couverture, nous devons considérer les facteurs suivants :

1. L'employé à transférer
 - doit être qualifié pour le département d^* ;
 - ne doit pas avoir atteint sa limite maximale du temps alloué aux transferts.
2. La période en sous-couverture

- doit être située avant le début ou après la fin du quart q effectué par l'employé e si un quart interne lui avait été assigné dans la phase précédente.

Illustrons cela par un exemple. Considérons la figure 4.1. Soient trois employés *employé 1*, *employé 2* et *employé 3* et une période encore en sous-couverture i^* dans le département d^* le jour j après la première phase de résolution séquentielle. Tous les trois employés sont qualifiés pour le département d^* et peuvent y être transférés. L'*employé 1* et l'*employé 2* ont reçu des quarts internes dans leurs départements d'origine le jour j tandis que l'*employé 3* était au repos.

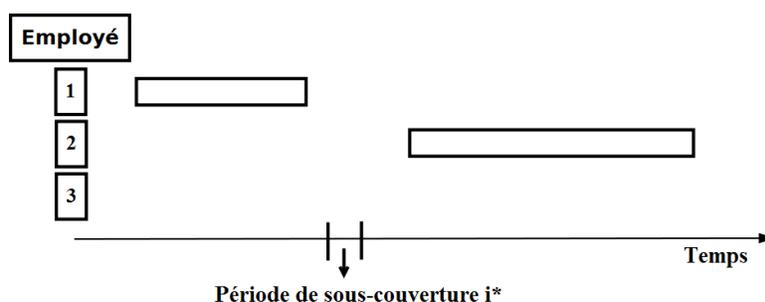


Figure 4.1 Horaires des employés après la première phase de résolution séquentielle

Pour couvrir la période en sous-couverture i^* , les employés peuvent être transférés vers le département d^* selon l'une ou l'autre des trois possibilités suivantes comme le montre la figure 4.2 :

- l'*employé 1* peut être transféré *après* son quart de travail interne pour couvrir la période en sous-couverture ;
- l'*employé 2* peut être transféré *avant* son quart de travail interne et faire son quart interne par la suite ;
- l'*employé 3* peut être transféré pour effectuer tout son quart dans le département d^* .

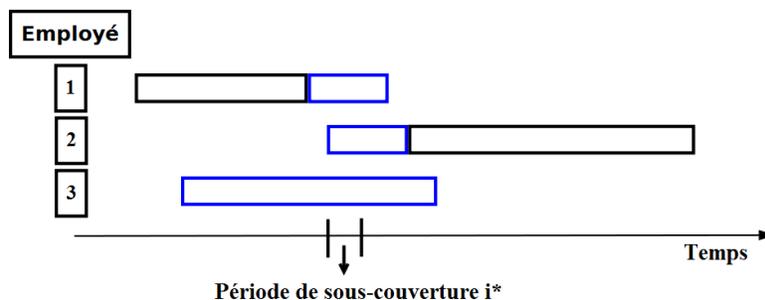


Figure 4.2 Les 3 possibilités de transfert

Par ailleurs, notons qu'en plus de diminuer la sous-couverture de la période considérée,

un transfert peut engendrer également des sur-couvertures dans les périodes adjacentes et couvrir d'autres sous-couvertures. Considérons par exemple une sous-couverture dans la période i_{45} du département d . Si les périodes i_{44} et i_{46} ne présentent pas de sous-couverture et qu'un employé est transféré afin de couvrir cette sous-couverture, ce transfert peut engendrer des sur-couvertures dans les périodes i_{44} et i_{46} . Comme on ne traite que les périodes de sous-couvertures, chaque quart aura donc un coût associé c_q appelé *coût du quart* q qui tient compte des sur-couvertures qu'engendrera le quart q s'il est sélectionné. Pour l'exemple précédent, si le coût c_d^o d'une sur-couverture dans le département d est de 5\$, on aura $c_q = 2 * 5 = 10\$$. Ce coût déterminera s'il est profitable de transférer ou pas un employé selon les coûts engendrés.

La construction des quarts de travail pour la deuxième phase de résolution se fait en ajoutant des blocs de transfert à chaque quart défini dans la première phase. Cela permettra donc de lier les deux phases et de *prolonger* le quart de l'employé selon ce qui lui a été affecté dans la phase précédente. Pour bien comprendre la construction des quarts dans cette phase, considérons le tableau 4.1 qui résume la construction des quarts pour un quart interne de 6h. Comme la durée maximale d'un quart de travail est de 10h, les transferts ne pourront durer que 4h au maximum pour ne pas dépasser cette limite. Il s'ensuit que les autres quarts seront mixtes par ajout d'un bloc de transfert avant ou après la réalisation du bloc interne associé.

Tableau 4.1 Exemple de construction des quarts de travail pour un quart interne de 6h

Durée du quart interne	Durée de transfert	Durée et type du quart résultant
0h	6h	6h (quart externe)
6h	1h	7h (quart mixte)
6h	2h	8h (quart mixte)
6h	3h	9h (quart mixte)
6h	4h	10h (quart mixte)

Ce tableau peut s'appliquer à tous les autres quarts en faisant varier la durée du quart interne. Le bloc de transfert sera représenté par le triplet (s_q, h_q, d_q) . s_q représente l'heure de début du bloc de transfert qui peut être soit au début du quart interne, soit à la fin du quart interne ou à n'importe quelle heure si aucun quart interne n'a été assigné. h_q et d_q représentent respectivement la durée du bloc de transfert et le département au sein duquel le transfert est exécuté.

Un quart q sera admissible pour un employé e le jour $j \in J$ si et seulement s'il respecte les conditions suivantes :

- Sa durée est supérieure ou égale à la durée du quart interne effectué dans le département d’origine le cas échéant ;
- Sa durée respecte la charge totale maximale L_{max} imposée par l’entreprise ;
- Son heure de début respecte le repos minimal R_{min} imposé entre deux quarts consécutifs ;
- S’il est réalisé dans un département admissible $d \in D_e$.

4.2.2 Méthodes de résolution

Pour résoudre le problème de deuxième phase, deux approches ont été développées : un algorithme glouton et un modèle en nombres entiers qui sera résolu par Xpress-MP. En pratique, les transferts sont planifiés séquentiellement à la main après la première phase de l’heuristique. Dans le but d’automatiser ce processus, nous avons développé un algorithme glouton. N’ayant pas une vue globale pour le calcul des transferts, l’algorithme glouton développé fournit des solutions moins bonnes par rapport à un modèle exact qui aurait pour avantage de considérer le problème au complet en ayant une vue globale. Nous proposons donc par la suite une amélioration en remplaçant l’algorithme glouton par un modèle exact.

Algorithme glouton

L’algorithme glouton traite les périodes en sous-couverture de la solution séquentielle une à la fois en commençant par la sous-couverture la plus élevée. Les sous-couvertures sont donc ordonnées dans l’ordre décroissant de leurs valeurs en brisant les égalités par l’ordre lexicographique des indices de leurs périodes respectives. Nous supposons que si une période donnée présente une sous-couverture élevée, les périodes adjacentes auront une grande probabilité d’être aussi en sous-couverture. De ce fait, en transférant un employé pour couvrir cette sous-couverture, d’autres sous-couvertures seront couvertes par la même occasion.

Pour chaque sous-couverture, les possibilités de transfert sont évaluées et la meilleure est choisie sans remise en question. Ainsi, si une sous-couverture est traitée, les transferts retenus sont définitifs et on passe à la sous-couverture suivante. Plusieurs itérations du même algorithme sont faites pour arriver à une solution à partir de laquelle aucune amélioration n’est possible.

Notations Le tableau 4.2 résume toutes les notations qui seront utilisées.

Tableau 4.2 Notations utilisées pour l'algorithme glouton

Notation	Signification
PU	Ensemble des périodes u en sous-couverture
UA	Ensemble des sous-couverture actives
E_d^{ext}	Ensemble des employés externes du département d
s_q	Heure de début du quart q
n_u	Valeur de la sous-couverture u
c_p	Coût de la possibilité de transfert p
H_e	Nombre total d'heures faites par l'employé e dans la solution séquentielle
s_q	Heure de début du quart q
h_q	Durée du quart q
ΔP_u	Ensemble des possibilités de transfert pour la période u en sous-couverture
ΔC	Ensemble des possibilités de transfert retenues par l'algorithme glouton
Q_e^j	Ensemble des quarts admissibles pour l'employé e le jour j
w_e^j	Vaut 1 si l'employé e travaille le jour j dans la solution séquentielle, 0 sinon
T_e^j	Vaut 1 si l'employé e est transféré le jour j , 0 sinon
c_{best}	Coût de la solution courante
c_e	Coût d'une heure travaillée de l'employé e
c_{ed}^t	Coût d'une heure transférée de l'employé e dans le département d
c_d^u	Coût d'une sous-couverture dans le département d
c_d^o	Coût d'une sur-couverture dans le département d

Pour chaque sous-couverture considérée, toutes les possibilités de transfert pouvant couvrir cette période en sous-couverture sont évaluées pour ne retenir que celles ayant un coût inférieur à celui la solution courante notée c_{best} . Le calcul de toutes les possibilités est décrit dans l'algorithme 1 comme une fonction. Cette fonction prend comme arguments la période $u \in PU$ en sous-couverture et le coût c_{best} de la solution courante et retourne l'ensemble ΔP_u des possibilités de transfert et le coût de la solution actualisé.

L'ensemble ΔP_u est constitué par les paires (e, q^*) correspondant à une possibilité de transfert de l'employé e en faisant le quart q^* . Le coût de cette possibilité sera notée c_p et est fournie par l'équation (4.1) où Nu représente le nombre de sous-couvertures qui seront couvertes si le quart q^* est retenu et No représente le nombre de sur-couvertures qui seront créés dans le même cas.

Algorithme 1 Pseudo-code de la fonction *PossibilitésTransfert*(u, c_{best})

Entrée: $u \in PU, c_{best}$

Sortie: $\Delta P_u, c_{best}$

- 1: Soient (d, j) le département et la journée liés à la sous-couverture $u \in PU$
- 2: **Pour tout** $e \in E_d^{ext}$ **faire**
- 3: **Pour tout** $q^* \in Q_e^j$ **faire**
- 4: **Si** $w_e^j = 1$ **ET** $h_{q^*} > h_q$ **alors**
- 5: **Si** $T_e^j = 0$ **ET** $H_e + h_{q^*} - h_q \leq L_{max}$ **ET** ($s_{q^*} \leq s_q$ **OU** $s_{q^*} \geq s_q + h_q$) **alors**
- 6: $p \leftarrow (e, q^*)$
- 7: **fin Si**
- 8: **Sinon Si** $w_e^j = 0$ **ET** $T_e^j = 0$ **ET** $H_e + h_{q^*} - h_q \leq L_{max}$ **alors**
- 9: $p \leftarrow (e, q^*)$
- 10: **fin Si**
- 11: Calculer le coût c_p à l'aide de l'équation (4.1)
- 12: **Si** $c_p < c_{best}$ **alors**
- 13: $\Delta P_u \leftarrow \Delta P_u \cup \{p\}$
- 14: $c_{best} \leftarrow c_p$
- 15: $T_e^j \leftarrow 1$
- 16: **fin Si**
- 17: **fin Pour**
- 18: **fin Pour**

Algorithme 2 Pseudo-code de l'algorithme glouton

Entrée: PU, c_{seq}, ni

Sortie: $\Delta C, c_{best}$

- 1: $c_{best} \leftarrow c_{seq}$
- 2: **Tant que** $ni > 0$ **faire**
- 3: $UA \leftarrow PU$
- 4: Classer les sous-couvertures UA en ordre décroissant
- 5: **Pour tout** $u \in UA$ **faire**
- 6: $\Delta P_u \leftarrow PossibilitesTransfert(u, c_{best})$
- 7: ΔP_u en ordre décroissant
- 8: $n \leftarrow |\Delta P_u|$
- 9: $\Delta C \leftarrow \Delta C \cup \Delta P_u[0..n]$ { on prend les n meilleurs possibilités de transfert }
- 10: $n_u \leftarrow n_u - n$ { le nombre de sous-couvertures diminue }
- 11: **fin Pour**
- 12: $ni \leftarrow ni - 1$
- 13: **fin Tant que**

$$c_p = c_{best} - (c_d^u * Nu) + (c_d^o * No) + (c_{ed}^t * h_{q^*}) \quad (4.1)$$

La fonction $PossibilitésTransfert(u, c_{best})$ a besoin du coût de la solution actuelle c_{best} pour commencer le calcul. La sous-couverture u a lieu dans un département spécifique d le jour j . La ligne 1 permet de retenir ces informations. Par la suite, chaque possibilité de transfert pour chaque employé $e \in E_d^{ext}$ et chaque quart admissible $q^* \in Q_e^j$ est visitée. La ligne 12 vérifie si la possibilité de transfert offre un coût inférieur au coût de la solution actuelle pour qu'elle soit ajoutée dans l'ensemble des possibilités trouvées. La détermination de cet ensemble comporte deux cas. Si l'employé e travaille le jour j dans la solution séquentielle (voir ligne 4), le quart q^* doit être une *prolongation* du quart q effectué dans son département d'origine. Ainsi, la durée h_{q^*} du quart q^* doit être strictement supérieure à celle du quart q (i.e. $h_{q^*} > h_q$). Sinon, si l'employé e ne travaille pas le jour j dans la solution séquentielle (voir ligne 8), il s'agit d'un quart externe et il n'y a pas de restriction sur la durée du quart q^* .

Comme chaque employé ne peut être transféré qu'une fois durant la journée, T_e^j doit être égal à zéro pour pouvoir considérer la possibilité. Dans les lignes 5 et 8 du pseudo-code précédent, l'inégalité $H_e + h_{q^*} - h_q \leq L_{max}$ permet de vérifier le respect de la règle relative aux horaires pour empêcher l'employé de dépasser la charge horaire maximale permise L_{max} en ajoutant la durée du bloc transféré. Par ailleurs, si un quart a été assigné à l'employé dans la solution séquentielle, les conditions ($s_{q^*} \leq s_q$ OU $s_{q^*} \geq s_q + h_q$) permettent de s'assurer que le transfert se fait soit au début ou à la fin du quart pour éviter les transferts au milieu. Si une possibilité est retenue, le coût de la solution courante est mis à jour et la variable T_e^j prend la valeur 1 pour l'employé e le jour j .

L'algorithme glouton proposé prend en entrée l'ensemble de toutes les périodes en sous-couverture PU , le coût de la solution séquentielle noté c_{seq} ainsi que le nombre d'itérations ni à faire. Comme sortie, l'algorithme glouton fournira l'ensemble ΔC de transfert retenues ainsi que le coût de la solution finale. Pour chaque sous-couverture, les possibilités de transfert sont évaluées par appel à l'algorithme 1. Notons que l'algorithme s'exécute en des temps très rapide et que 99% des résultats se retrouvent dans la première itération.

Modèle exact en nombres entiers

En plus des définitions utilisées dans les sections précédentes, définissons les paramètres suivants qui s'appliquent au modèle exact de ré-optimisation.

Tableau 4.3 Paramètres du modèle exact de la phase de ré-optimisation

Notation	Signification
H_e	Nombre total d'heures faites par l'employé e dans la solution séquentielle
PU	Ensemble des périodes u en sous-couverture
r_d^i	Nombre de sous-couvertures de la période $i \in PU$ dans le département d
PO	Ensemble des périodes u en sur-couverture
c_q	coût associé aux sur-couvertures engendrés par le quart q

La formulation P_3 du modèle exact de la phase de ré-optimisation se présente comme suit.

$$Min \quad \sum_{e \in E} \sum_{j \in J_e} \sum_{q \in Q_e^j} (\delta^t c_{ed}^t h_q^t X_{eq}^j + c_q X_{eq}^j) + \delta^u \sum_{d \in D} \sum_{i \in PU} c_d^u U_d^i + \delta^o \sum_{d \in D} \sum_{i \in PO} c_d^o O_d^i \quad (4.2)$$

$$s.c. \quad \sum_{e \in E_d} \sum_{j \in J_e} \sum_{q \in Q_e^j} \theta_{dq}^i X_{eq}^j + U_d^i - O_d^i = r_d^i \quad \forall d \in D, i \in PU \quad (4.3)$$

$$\sum_{q \in Q_e^j} X_{eq}^j \leq 1 \quad \forall e \in E, j \in J_e \quad (4.4)$$

$$\sum_{j \in J_e} \sum_{q \in Q_e^j} h_q X_{eq}^j + H_e \leq L_{max} \quad \forall e \in E \quad (4.5)$$

$$\sum_{j \in J_e} \sum_{q \in Q_e^j} f_q^i X_{eq}^j + \sum_{j \in J_e} \sum_{q \in Q_e^j} \sum_{i' \in [i, i+R_{min}]} (b_q^{i'} X_{eq}^j + v_q^{i'} X_{eq}^j) \leq 1 \quad \forall i \in I, e \in E \quad (4.6)$$

$$\sum_{j \in J_e} \sum_{q \in Q_e^j} (h_q - h_q^t) X_{eq}^j \geq K_{min} \sum_{j \in J_e} \sum_{q \in Q_e^j} h_q X_{eq}^j \quad \forall e \in E \quad (4.7)$$

$$X_{eq}^j \in \{0, 1\} \quad \forall e \in E, j \in J_e, q \in Q_e^j \quad (4.8)$$

$$U_d^i \geq 0 \quad \forall d \in D, i \in I \quad (4.9)$$

$$O_d^i \geq 0 \quad \forall d \in D, i \in I \quad (4.10)$$

Le coût c_q du quart q est obtenu en multipliant le coût c_d^o d'une sur-couverture dans le département d où le bloc de transfert du quart q est effectué par le nombre de sur-couvertures engendrées. L'objectif est de minimiser la somme pondérée des quatre types de coûts à savoir les coûts de sous-couverture, les coûts de sur-couverture, les coûts des quarts et les coûts de transfert. Les contraintes (4.3) imposent la couverture de la demande uniquement pour les périodes en sous-couverture. Les contraintes (4.5) assurent un ajustement du nombre total

d'heures par rapport à la solution séquentielle. Pour cela, les heures faites par l'employé dans son département d'origine sont prises en compte.

Le modèle exact (4.2)-(4.10) de la phase de ré-optimisation est résolu en utilisant le solveur Xpress-MP.

CHAPITRE 5

RÉSULTATS NUMÉRIQUES

Le présent chapitre présente les jeux de données testés de même que les résultats numériques obtenus à l'aide d'implémentations des modèles et algorithmes présentés au chapitre précédent. Les conditions de simulation dans lesquelles les résultats numériques ont été obtenus seront décrites en premier lieu dans la section 5.1. Par la suite, la section 5.2 décrira sommairement les problèmes-tests utilisés dans la partie expérimentale. Nous présenterons dans la section 5.3 les critères qui visent à évaluer la performance des différentes approches implémentées les unes par rapport aux autres. La section 5.4 présentera finalement les résultats numériques obtenus ainsi qu'une comparaison des différentes approches.

5.1 Conditions de simulations

Les modèles présentés dans ce mémoire ont été résolus par Xpress-MP et implémentés en C++ en utilisant la librairie Xpress-BCL *Builder Component Library* qui fournit une bibliothèque de fonctions permettant de formuler et résoudre les modèles linéaires, mixtes et quadratiques. La compilation a été faite avec GNU G++.

La machine utilisée est dotée d'un processeur Intel(R) Xeon(R) CPU X5670 à six cœurs et présente une fréquence d'horloge de 2.93GHz. Un seul cœur est utilisé à la fois pour rouler les tests. Le système d'exploitation est GNU/Linux *x86_64*.

5.2 Jeux de données

Dans cette section, on passe à la description proprement dite des jeux de données utilisés. De façon générale, la structure de données d'un problème-test est constitué de l'ensemble de données suivant :

- un ensemble de départements D
- un ensemble d'employés E
- un ensemble de séquence de travail S . Chaque séquence $s \in S$ est caractérisée par un ensemble de jours de travail et de repos durant l'horizon de planification.
- un ensemble de types de quart T

Chaque département $d \in D$ est décrit par :

- une courbe de demande spécifiant le nombre d’employés requis r_d^i pour chaque période de l’horizon de planification $i \in I$
- un ensemble d’employés internes E_d^{int}
- un ensemble d’employés externes E_d^{ext}
- le coût d’une période de sous-couverture C_d^u
- le coût d’une période de sur-couverture C_d^o
- un ensemble de types de quart T

Chaque type de quart $t \in T$ est caractérisé par un ensemble d’heures de début possibles ainsi qu’une durée minimale et maximale admissible.

Chaque employé $e \in E$ est décrit par :

- son département d’origine d_e
- un ensemble de départements d’accueil D_e
- un coût horaire de travail c_e
- un coût horaire c_{ed}^t mesurant la pénalité de transfert vers un département d’accueil $d \in D_e$
- une séquence de travail J_e assignée à l’avance et qui détermine ses jours de travail et de repos

Nous allons d’abord nous intéresser aux jeux de données de petite taille générés aléatoirement par Dahmen (2014) où il sera possible de comparer les performances du modèle global et de l’heuristique en deux phases. Par la suite, nous présenterons les résultats obtenus par l’heuristique pour les instances de grande taille.

La génération aléatoire des jeux de tests s’est faite de façon à reproduire la structure de celle des problèmes réels. Pour cela, les départements sont classés en trois catégories distinctes selon leur taille (nombre d’employés internes) : les grands départements, les départements moyens et les petits départements. Aussi, trois niveaux d’employés ont été pris en compte selon leur nombre de qualifications : les employés hautement qualifiés, les employés moyennement qualifiés et les employés peu qualifiés. Pour chacune des instances, différents profils de demande ont été définis. On entend par profil de demande l’ensemble des courbes de demande pour tous les départements de l’organisation. Chaque profil a été généré de façon à induire une charge de travail s’ajustant parfaitement à la capacité interne des départements et en introduisant un nombre de sous-couvertures variant d’un profil à un autre. Finalement, étant donné que les goulots d’étranglement se déplacent entre les départements, les hausses et les baisses de la demande ont été créés artificiellement dans les différents départements.

Onze instances différentes avec les caractéristiques décrites dans le tableau 5.1 ont été testées. Dans ce qui suit, chaque instance associée à un nombre de départements $|D|$ et à un nombre d'employés $|E|$ et ayant le profil p sera notée par $IA(|D|,|E|,p)$.

Tableau 5.1 Instances de données générées aléatoirement

Instance	# Employés	# Départements	Profil	Horizon(jours)
IA(2,20,1)	20	2	1	7
IA(2,20,4)	20	2	4	7
IA(2,50,3)	50	2	3	7
IA(3,20,1)	20	3	1	7
IA(3,20,2)	20	3	2	7
IA(3,50,1)	50	3	1	7
IA(3,50,3)	50	3	3	7
IA(5,20,1)	20	5	1	7
IA(5,20,3)	20	5	3	7
IA(5,200,1)	200	5	1	7
IA(10,400,1)	400	10	1	7

Pour les jeux de données réelles de grande taille, nous avons utilisé deux différentes instances décrites dans le tableau 5.2. Chaque instance de données réelles sera notée par $IR(|D|,|E|)$ où $|D|$ représente le nombre de départements et $|E|$ celui d'employés constituant le jeu de test. Pour les jeux de données aléatoires, nous utilisons la même notation que celle utilisée pour les instances de petite taille.

Tableau 5.2 Instances de données réelles de grande taille

Instance	Employés	Départements	Horizon(jours)
IA(5,200,2)	200	5	7
IA(10,400,1)	400	10	7
IR(8,191)	191	8	7
IR(11,191)	300	13	7

5.3 Critères de performance

Pour chaque instance, on peut déterminer le nombre de sous-couvertures d'une instance I qui correspond au nombre total de sous-couvertures pour l'ensemble de départements et

durant tout l'horizon de planification, soit

$$U(I) = \sum_{d \in D} \sum_{i \in I} U_d^i \quad (5.1)$$

De même, le nombre de sur-couvertures total générées $O(I)$ pour une instance I peut être obtenue par l'équation

$$O(I) = \sum_{d \in D} \sum_{i \in I} O_d^i \quad (5.2)$$

Ainsi, pour chaque instance, on pourra quantifier la demande couverte ainsi que le nombre de sur-couvertures générées. D'autres critères entreront aussi en jeu pour évaluer et comparer les résultats numériques obtenus comme le présente le tableau 5.3.

Tableau 5.3 Critères de performance

Critère	Unité	Description
Temps de résolution	secondes	Temps d'exécution total
Objectif	coût(\$)	Coût de la fonction objectif obtenu
$U(I)$	sous-couvertures (entier)	Cardinalité des sous-couvertures restantes dans la solution obtenue
$O(I)$	sur-couvertures (entier)	Cardinalité des sur-couvertures générées dans la solution obtenue
Proportion de couverture pour l'instance	%	Pourcentage de couverture de la demande pour chaque instance présentée dans les tableaux 5.2 et 5.1
Nombre de transferts	Entier	Cardinalité des transferts effectués dans la solution obtenue

5.4 Résultats numériques

Dans cette section, nous présentons les résultats numériques obtenus pour chaque critère de performance listée dans le tableau 5.3. Pour ce faire, nous subdivisons les jeux de tests en

deux grandes catégories à savoir : les jeux de données de petite taille et ceux de grande taille. En effet, pour les instances de petite taille, le modèle global parvient à donner une solution optimale. Ainsi, il sera possible de comparer le modèle global et celui en deux phases pour les instances de petite taille. Par conséquent, les jeux de données aléatoires de 2, 3 et 5 départements avec 20 et 50 employés chacun seront utilisés pour tester et comparer le modèle global avec l'heuristique en deux phases. Les instances de données réelles de grande taille ainsi que les instances aléatoires IA(5,200,1) et IA(10,400,1) seront quand à elles testées avec l'heuristique en deux phases car le modèle global échoue. En effet, pour les instances avec 5 départements et 200 employés, le modèle global n'était pas terminé après 10h de calcul, ce qui constitue un temps inacceptable pour les instances d'une telle taille.

Le tableau 5.4 donne les valeurs par défaut pour chacun des paramètres considérés lors des simulations numériques.

Tableau 5.4 Valeurs par défaut des paramètres définissant le problème

Paramètre	Valeur
R_{min}	12h
L_{max}	40h
K_{min}	70%

5.4.1 Modèle global

Dans un premier temps, les instances IA(2,20,1), IA(2,20,4), IA(2,20,1), IA(2,50,3), IA(3,20,1), IA(3,20,2), IA(3,50,1), IA(3,50,3), IA(5,20,1) et IA(5,20,3) ont été résolus à l'optimalité grâce au modèle global. Les tailles de ces instances sont rapportées dans le tableau 5.5.

Tableau 5.5 Taille du modèle global pour les instances aléatoires

Instance	Variables	Contraintes	Quarts internes	Quarts externes
IA(2,20,1)	130460	1942	1890	28350
IA(2,20,4)	143040	2079	1890	28350
IA(2,50,3)	396056	4097	1890	28350
IA(3,20,1)	216603	2334	2835	42525
IA(3,20,2)	207676	2276	2835	42525
IA(3,50,1)	628282	4664	1890	42525
IA(3,50,3)	630933	4646	1890	42525
IA(5,20,1)	352576	2929	4725	70875
IA(5,20,3)	327951	2823	4725	70875

Pour les instances avec le même nombre de départements, les courbes de demande et le nombre de qualifications par employé varient d'un profil à l'autre. Cela explique la variation du nombre de variables et de contraintes du modèle indépendamment de celui des quarts générés.

Les résultats obtenus sont présentés dans le tableau 5.6 où pour chaque instance de données, les critères de performance sont indiqués. La demande totale correspond au nombre total d'employés requis dans tous les départements durant tout l'horizon de planification pour l'instance considérée. Elle est donnée par l'équation suivante où D^* représente l'ensemble des départements constituant l'instance considérée :

$$\text{Demande totale} = \sum_{d \in D^*} \sum_{i \in I} q_d^i \quad (5.3)$$

Tableau 5.6 Résultats numériques pour le modèle global

Instance	Temps (sec)	Objectif	Demande totale	U(I)	O(I)	Nombre trans-ferts	Nombre quarts	Demande couverte (%)
IA(2,20,1)	33	13555.1	3140	2	2	2	100	99.9
IA(2,20,4)	56	15659.7	3052	2	10	3	100	99.9
IA(2,50,3)	392	34146.5	7604	5	5	9	250	99.9
IA(3,20,1)	62	13950.7	2968	1	5	1	100	99.9
IA(3,20,2)	80	14907.2	2956	3	3	6	100	99.85
IA(3,50,1)	614	32839.6	7564	0	8	3	252	100
IA(3,50,3)	667	32405.3	7632	2	2	5	250	99.9
IA(5,20,1)	107	16337.1	3112	0	12	8	100	100
IA(5,20,3)	121	15435	3048	4	4	3	101	99.85

La résolution du modèle global nous donne une idée du temps nécessaire à la résolution optimale du problème et fournit du même coup la solution optimale de chaque instance qui servira de point de départ pour comparer les solutions obtenues par l'heuristique en deux phases.

Nous remarquons une couverture de la demande supérieure à 99.8% pour toutes les instances. De plus, nous constatons que Xpress-MP est efficace, comme prévu, lorsque la taille de l'instance est assez petite. La figure 5.1 représente l'évolution du temps de résolution en fonction du nombre de départements. On observe que le temps croît linéairement avec le nombre de départements.

Par contre, en observant la figure 5.2 qui représente l'évolution du temps en fonction du nombre d'employés, nous remarquons que le temps croît très rapidement lorsque le nombre d'employés augmente. Ainsi, pour des instances avec 50 employés, les temps de 392 secondes pour 2 départements et 667 secondes pour 3 départements sont très élevés et inacceptables pour notre partenaire industriel où le nombre d'employés peut parfois se chiffrer en milliers.

5.4.2 Heuristique en deux phases

Dans cette section, nous présentons les résultats numériques obtenus pour chacune des deux phases. Les mêmes critères de performance proposés dans la section 5.3 seront utilisés

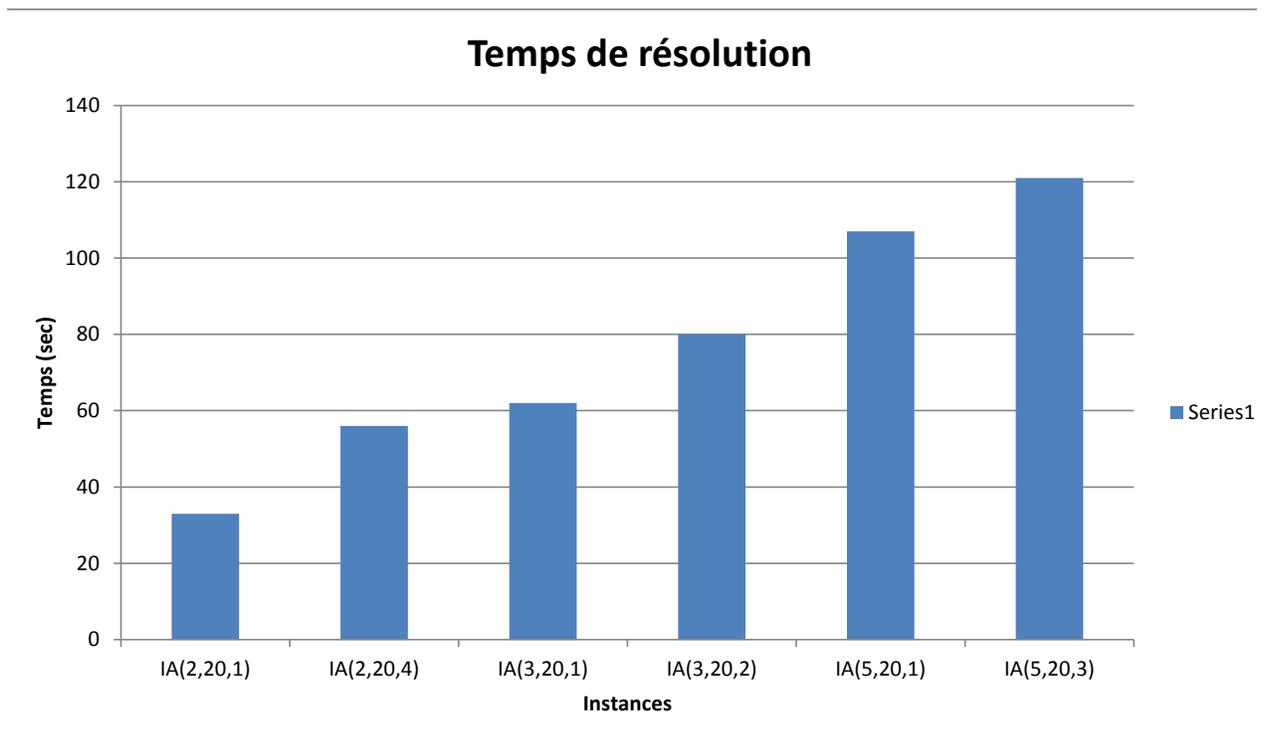


Figure 5.1 Temps de résolution en fonction du nombre de départements pour le modèle global

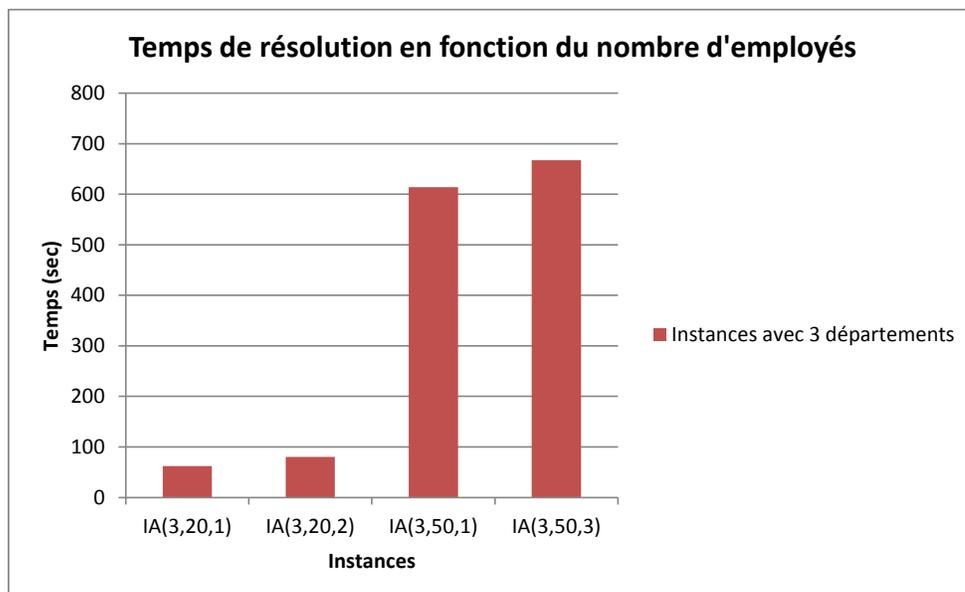
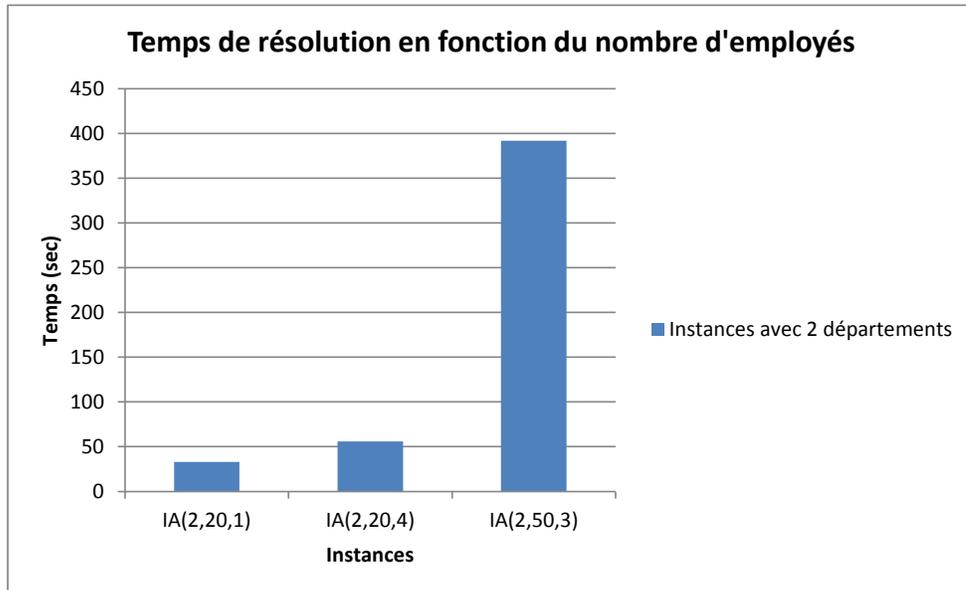


Figure 5.2 Temps de résolution en fonction du nombre d'employés pour le modèle global

pour évaluer les solutions obtenues. En plus de ces critères, nous faisons ressortir, après la première phase de résolution séquentielle, le *coût total travaillé* C_1 donné par l'équation (5.4). Ce coût va nous permettre de calculer la valeur de la fonction objectif après la deuxième phase de résolution pour pouvoir la comparer avec celle de la solution optimale fournie par le modèle global. Pour cela, après la deuxième phase de réoptimisation, le *coût total transféré* C_2 sera calculé et ajouté au coût total des sous et sur-couvertures pour donner le coût total C_t de la deuxième phase qui pourra être comparé à celui du modèle global. Les équations (5.4) à (5.6) montrent les calculs effectués.

$$C_1 = \sum_{e \in E} \sum_{j \in J_e} \sum_{q \in Q_e^j} c_e h_q X_{eq}^j \quad (5.4)$$

$$C_2 = \sum_{e \in E} \sum_{j \in J_e} \sum_{q \in Q_e^j} (c_e + c_{ed}^t) h_q^t X_{eq}^j \quad (5.5)$$

$$C_t = C_1 + C_2 + \sum_{d \in D} c_d^u U_d + c_d^o O_d \quad (5.6)$$

Le tableau 5.7 présente les résultats des calculs pour la première phase de l'heuristique.

Tableau 5.7 Résultats pour la première phase séquentielle

Instance	Dép.	Temps (sec)	Objectif	Demande	U(I)	O(I)
IA(2,20,1)	1	1	14132	1894	14	0
	2	<1	5698	1246	2	0
IA(2,20,4)	1	<1	18763	1862	23	1
	2	<1	10181	1190	0	22
IA(2,50,3)	1	2	28674	4409	25	0
	2	3	27758	3195	4	53
IA(3,20,1)	1	1	7531	1403	0	9
	2	<1	2990	688	0	0
	3	<1	10204	877	13	0
IA(3,20,2)	1	<1	16798	1367	23	0
	2	<1	3058	732	0	0
	3	1	11602	857	3	26
IA(3,50,1)	1	1	20523	3016	16	0
	2	1	10223	2262	0	6
	3	2	10405	2286	0	6
IA(3,50,3)	1	1	14654	2975	25	9
	2	1	16262	2354	4	3
	3	1	11138	2303	3	9
IA(5,20,1)	1	<1	3329	450	0	6
	2	<1	18206	806	30	0
	3	<1	1800	480	0	0
	4	<1	5129	930	0	6
	5	<1	3232	446	0	6
IA(5,20,3)	1	<1	1790	456	0	0
	2	<1	3138	736	0	0
	3	<1	5766	462	2	12
	4	<1	3680	960	0	0
	5	<1	10706	434	18	0

Dans les sections qui suivent, nous présentons les résultats de la réoptimisation par l'algorithme glouton et le modèle exact pour les instances.

Algorithme glouton

Un algorithme glouton a été développé pour réoptimiser la solution séquentielle obtenue dans la première phase de résolution. Cet algorithme reflète la pratique et son but est de résoudre les instances dans des temps de résolution inférieurs à ceux du modèle exact même si la solution obtenue est dans la plupart des cas moins bonne. Le tableau 5.8 fournit les résultats de l'algorithme glouton. Pour les instances de petite taille, on observe une réduction de sous-couvertures de 25% en moyenne. L'algorithme glouton trouvera aussi son intérêt pour les instances de taille réelle en considérant le rapport temps/coût.

Tableau 5.8 Résultats pour la seconde phase de réoptimisation avec l'algorithme glouton

Instance	Temps (sec)	Objectif	U(I) initiales	U(I) finales	O(I) finales	Nombre de transferts
IA(2,20,1)	0.01	8000	16	16	0	0
IA(2,20,4)	0.22	8290	23	10	34	1
IA(2,50,3)	0.06	14500	29	29	53	0
IA(3,20,1)	0.19	4602	13	9	9	1
IA(3,20,2)	0.20	9862	26	13	37	1
IA(3,50,1)	0.30	7204	16	12	16	1
IA(3,50,3)	0.25	6500	13	13	21	0
IA(5,20,1)	0.38	28969	30	25	18	1
IA(5,20,3)	0.01	23922.5	20	17	13	1

Nous rapportons dans le tableau 5.9 les résultats de l'heuristique en combinant les deux phases.

Tableau 5.9 Résultats numériques de l'heuristique en deux phases avec l'algorithme glouton

Instance	Temps (sec)	Objectif	Demande totale	U(I)	O(I)	Nombre de transferts	Nombre de quarts
IA(2,20,1)	<2.01	19830	3140	16	0	0	100
IA(2,20,4)	<2.22	25082	3052	10	34	1	92
IA(2,50,3)	5.06	54571	4409	29	53	0	228
IA(3,20,1)	<3.19	18827	2968	9	9	1	99
IA(3,20,2)	<3.20	27544	2956	13	37	1	91
IA(3,50,1)	4.30	37764	3052	12	16	1	224
IA(3,50,3)	3.25	40918	7632	13	21	0	237
IA(5,20,1)	<5.38	28969	3112	25	18	1	100
IA(5,20,3)	<5.01	23922.5	3048	17	13	1	99

Modèle exact

Le tableau 5.10 présente les résultats des calculs pour la seconde phase de l'heuristique avec le modèle exact.

Tableau 5.10 Résultats pour la seconde phase de réoptimisation avec le modèle exact

Instance	Temps (sec)	Objectif	U(I) initiales	U(I) finales	O(I)	Nombre de transferts
IA(2,20,1)	<1	8000	16	16	0	0
IA(2,20,4)	<1	6337.4	23	6	34	2
IA(2,50,3)	<1	14500	29	29	53	0
IA(3,20,1)	<1	2704	13	5	9	1
IA(3,20,2)	<1	9862	26	13	37	1
IA(3,50,1)	<1	7138.2	16	12	16	1
IA(3,50,3)	<1	6500	13	13	21	0
IA(5,20,1)	<1	9267.3	30	18	18	3
IA(5,20,3)	<1	8842.5	20	17	13	1

Remarque Les temps de résolution marqués par <1 sont en réalité égaux à 0 sec dans la trace d'exécution fournie par Xpress-MP. Dans le calcul du temps total, nous allons considérer la borne supérieure égale à 1 pour chaque département présentant un temps de résolution

inférieur à 1.

L'algorithme glouton et le modèle exact assurent aussi une couverture de la demande supérieure à 99.8% pour toutes les instances. On constate que l'heuristique en deux phases diminue fortement les temps de résolution. Afin de pouvoir comparer les résultats, nous rapportons dans le tableau 5.11 les résultats de l'heuristique en combinant les deux phases et en utilisant les équations (5.4) à (5.6) pour calculer la valeur de la fonction objectif.

Tableau 5.11 Résultats numériques de l'heuristique en deux phases avec le modèle exact

Instance	Temps (sec)	Objectif	Demande totale	U(I)	O(I)	Nombre de transferts	Nombre de quarts
IA(2,20,1)	<3	19830	3140	16	0	0	100
IA(2,20,4)	<3	23129.4	3052	6	34	2	92
IA(2,50,3)	<6	54571	7604	29	53	0	228
IA(3,20,1)	<4	16929	2968	5	9	1	99
IA(3,20,2)	<4	27544	2956	13	37	1	91
IA(3,50,1)	<5	37698.2	7564	12	16	1	224
IA(3,50,3)	<4	40918	7632	13	21	0	237
IA(5,20,1)	<6	25963.3	3112	18	18	3	100
IA(5,20,3)	<6	23922.5	3048	17	13	1	99

Comparaison : Modèle global et heuristique en deux phases

La figure 5.3 indique les différents temps de résolution pour le modèle global, l'heuristique en deux phases avec d'une part le modèle exact et d'autre part l'algorithme glouton comme méthodes de réoptimisation. Pour les instances de petite taille, l'heuristique en deux phases est nettement plus rapide que le modèle exact. En effet, nous remarquons une diminution moyenne du temps d'exécution de 94%. En considérant le temps d'exécution, nous déduisons que l'heuristique en deux phases est nettement plus efficace que le modèle global résolu par Xpress-MP.

Pour bien comparer les résultats, voyons maintenant les autres critères de performance à savoir la couverture de la demande (i.e le nombre de sous-couvertures) et la valeur de la fonction objectif. La figure 5.4 montre une comparaison du coût de la fonction objectif. Au premier coup d'œil, nous remarquons que les heuristiques fournissent des solutions moins

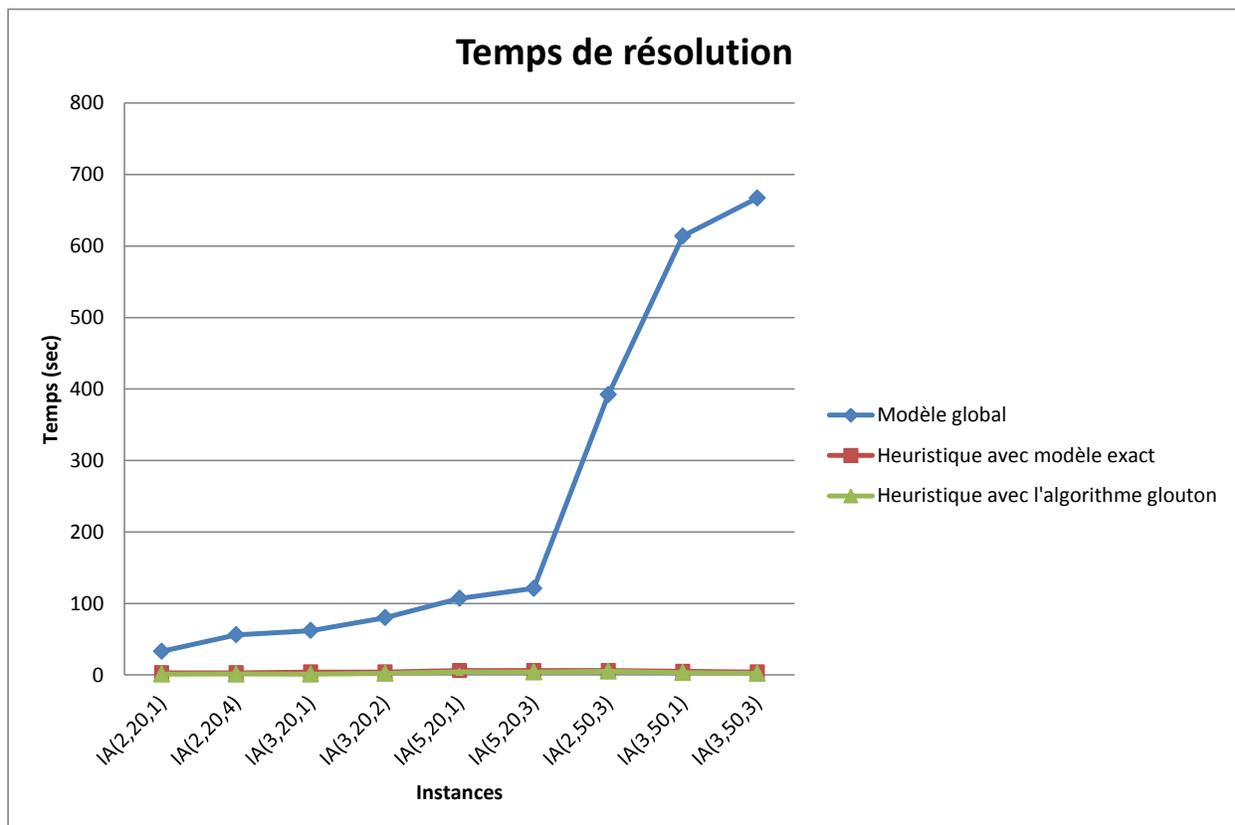


Figure 5.3 Temps de résolution Modèle global et heuristiques

bonnes que le modèle global, ce qui est normal car ce dernier est résolu à l'optimalité. Pour toutes les instances, on voit que la valeur de la fonction objectif présente un écart d'environ 30% par rapport à l'optimalité. Cela s'explique par le fait que les heuristiques fournissent des solutions avec plus de sous-couvertures et/ou de sur-couvertures, même si le nombre de transferts est relativement le même dans les deux cas.

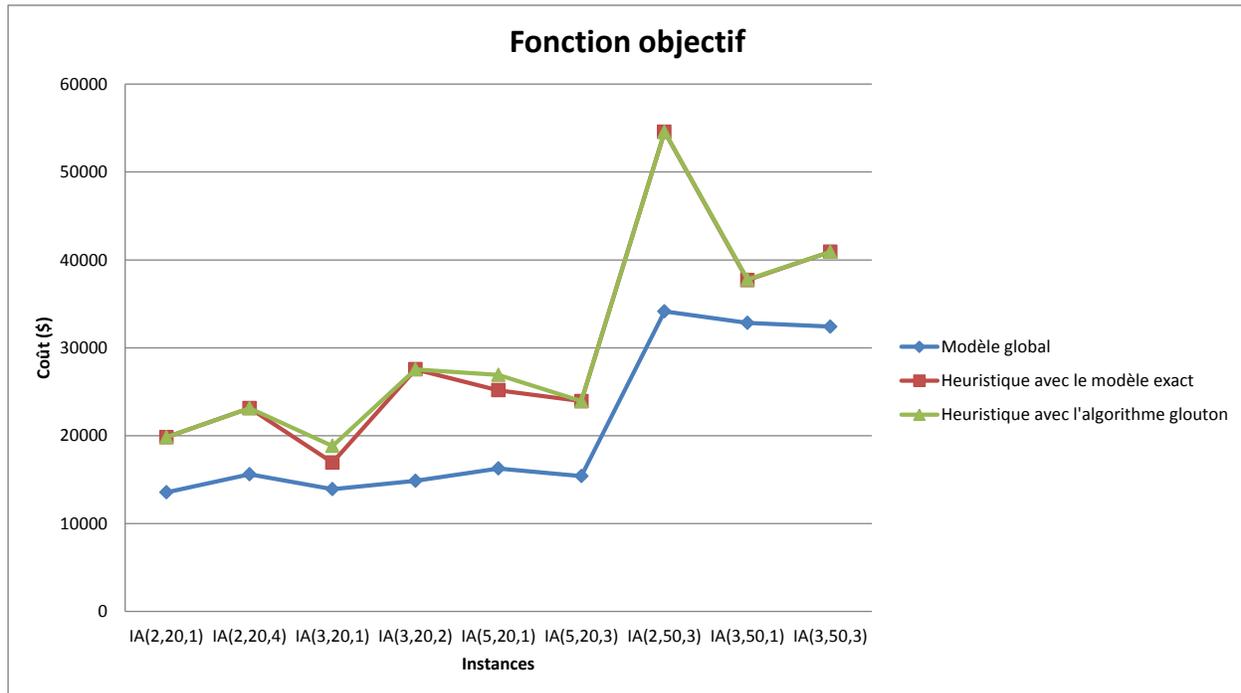


Figure 5.4 Comparaison du coût de la fonction objectif

Par ailleurs, pour 5 des 9 instances, l'algorithme glouton trouve la même solution qu'avec le modèle exact. Cela peut s'expliquer par le fait que pour ces instances, les possibilités de transferts sont limitées et l'algorithme glouton parvient à les trouver du premier coup. En effet, 3 des 5 instances ne permettent pas des transferts même avec le modèle exact. Par contre, s'il y a plusieurs possibilités, l'algorithme choisira la première réalisable mais qui ne donne pas nécessairement le meilleur coût. C'est le cas des 4 instances où l'algorithme glouton et le modèle exact présentent un écart pouvant atteindre 13%. Ainsi, plus la taille des instances sera grande et le nombre de possibilités de transferts élevé, plus le modèle exact fournira une meilleure solution que l'algorithme glouton.

Comme prévu, nous voyons sur les figures 5.5 et 5.6 que le modèle global assure une meilleure couverture de la demande par rapport à l'heuristique en deux phases. Il faut ce-

pendant noter que la différence est en moyenne de 0.45% de demande non couverte de plus par rapport au modèle global. Cela entraîne par contre un nombre élevé de sur-couvertures générées, ce qui donne à la fin une meilleure solution avec le modèle global.

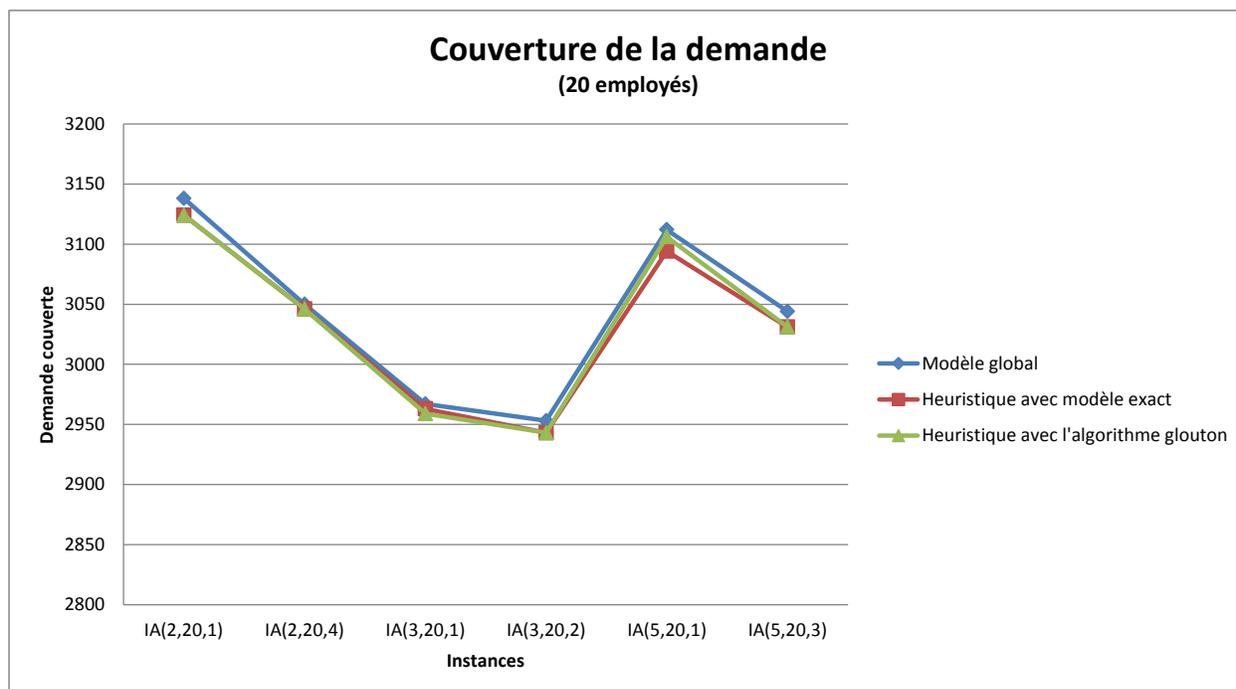


Figure 5.5 Couverture de la demande

Redéfinition des quarts internes du modèle global

Les quarts internes dans le modèle global tels que définis dans la section 3.2.2 ont une durée minimale d'une heure avec une discrétisation de la durée aux heures. Cela peut faire en sorte qu'un employé peut se voir attribuer un quart interne d'une durée de $2h$ avec un quart externe d'une durée de $4h$, pourvu que la durée minimale du quart soit de $6h$. Par contre, l'heuristique en deux phases dans sa première phase de réoptimisation permet des quarts internes d'une durée minimale de $6h$ avec une discrétisation de la durée aux heures.

Comme nous l'avons remarqué, l'heuristique en deux phases présente un écart assez élevé par rapport à l'optimalité. Nous avons donc remis en cause la définition des quarts internes du modèle global en fixant la durée minimale à $6h$. Le tableau 5.12 montre les résultats

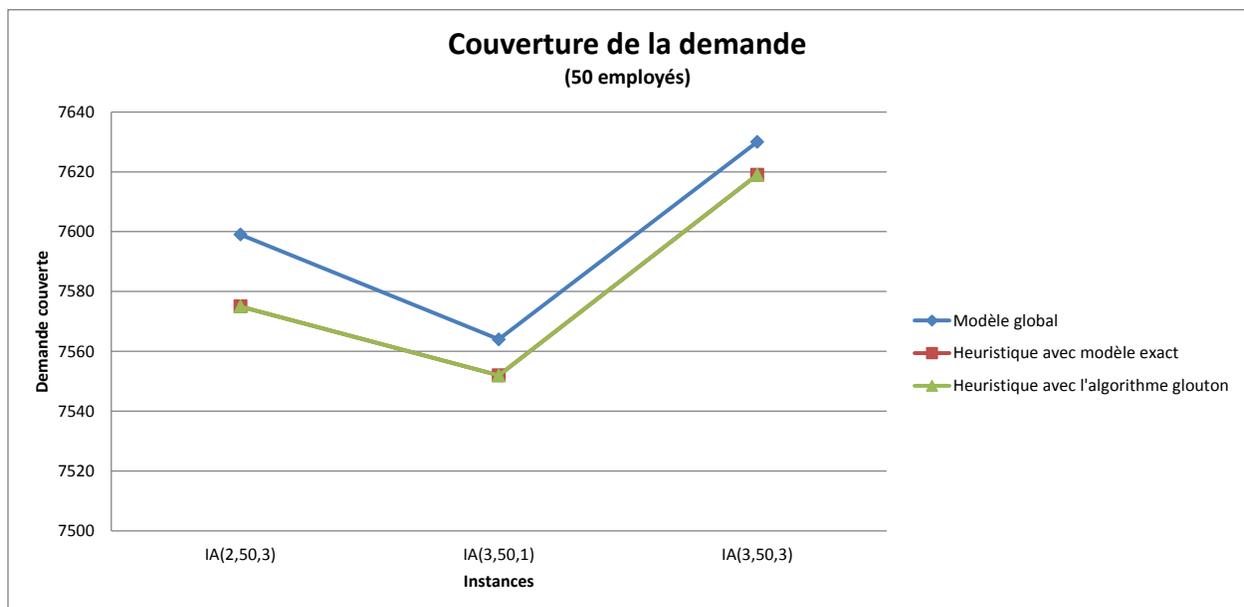


Figure 5.6 Couverture de la demande

numériques obtenus avec le modèle global en redéfinissant la durée des quarts internes pour cinq instances.

Tableau 5.12 Résultats numériques pour le modèle global

Instance	Temps (sec)	Objectif	Demande totale	U(I)	O(I)	Nombre de trans-ferts	Nombre de quarts	Demande couverte (%)
IA(2,20,4)	15	15809.7	3052	0	9	6	139	100
IA(3,20,2)	17	16020.1	2956	3	3	10	100	99.9
IA(5,20,1)	41	25165.9	3112	9	29	7	100	99.7
IA(2,50,3)	113	35968.5	7604	5	5	14	250	99.9
IA(3,50,3)	129	34452.4	7632	2	10	6	252	99.9

Nous remarquons que la couverture de la demande reste supérieure à 99.7% pour toutes ces instances. Comme le montre la figure 5.7, la diminution de l'écart entre le modèle global et l'heuristique en deux phases pour les instances avec 2 départements plafonne à 5% tandis que celle avec 3 départements plafonne à 6%.

Pour l'instance IA(5,20,1) par contre, nous remarquons une forte diminution qui améliore l'heuristique à plus de 90%. En effet, en redéfinissant les quarts internes pour cette instance, l'heuristique en deux phases fournit une solution à 3% de l'optimalité. Nous pouvons ainsi dire que plus le nombre de départements est élevé, plus l'heuristique s'améliore de façon significative.

Nous pouvons ainsi dire que la performance de l'heuristique s'améliore si on fixe la durée des quarts internes à 6h pour le modèle global. Elle pourrait s'améliorer encore plus en permettant de remettre en cause les quarts internes. Par ailleurs, nous remarquons que même si les temps d'exécution augmentent linéairement en fonction du nombre de départements, ils deviennent très élevés si le nombre d'employés augmente. On peut ainsi s'imaginer l'impact sur les données réelles de grande taille où le nombre de départements peut atteindre des dizaines avec des milliers d'employés.

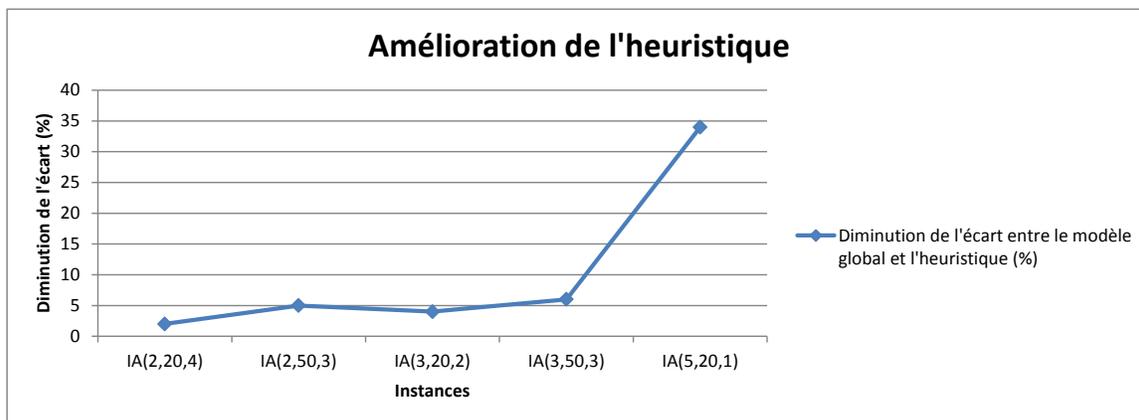


Figure 5.7 Nombre de sur-couvertures générées

5.4.3 Heuristique en deux phases : autres cas de tests

Nous avons testé notre heuristique en deux phases sur d'autres jeux de données de grande taille afin de ne pas restreindre notre étude sur les jeux de données de petite taille. Les jeux de données utilisées peuvent être regroupées en deux catégories :

1. Deux jeux de données de grande taille générés aléatoirement
2. Deux jeux de données réels de grande taille.

Pour les instances réelles avec 8 et 11 départements, chaque employé est qualifié pour l'ensemble des départements. L'instance IR(8,191) présente un maximum de 20 employés internes par département tandis que l'instance IR(11,191) en présente 30. Étant donnée la grande taille de ces instances, nous avons limité la durée des quarts à 6, 8 et 10h.

Pour tous les jeux de données de grande taille, nous avons défini les quarts de travail aux 15 minutes pour tout l'horizon de planification. Cela donne plus de quarts que ceux considérés pour les données de petite taille testées précédemment. Pour les instances réelles, les coûts de sous et sur-couverture n'étaient pas fournis. Nous les avons fixé à 30\$ et 10\$ respectivement. Dans ce qui suit, nous fournissons directement les résultats de la phase séquentielle en faisant la somme des temps d'exécution de tous les départements. Le lecteur trouvera les détails d'exécution de chaque département pour la phase séquentielle dans les annexes A et B.

Modèle exact

Le tableau 5.13 montre les temps d'exécution et la valeur de la fonction objectif pour chacune des 4 instances de grande taille. Les différentes instances se résolvent très rapidement par le solveur Xpress-MP. Pour les instances aléatoires, nous voyons que la demande a été couverte à 93% dans les deux cas. Par contre, nous voyons que le nombre de sur-couvertures générées est assez élevé. Pour les instances réelles, nous remarquons également une couverture de la demande de 81%. Le grand nombre de sous-couvertures résulte du fait que les courbes de la demande pour ces instances ne sont pas lisses, i.e., la demande pour cinq périodes consécutives peut être de 5, 3, 6, 3 et 4, ce qui peut rendre coûteux la couverture complète de la demande égale à 6.

Pour les instances aléatoires de taille réelle IA(5,200,2) et IA(10,400,1), les temps d'exécution de 14 et 46 secondes sont plus qu'acceptables. Pour les instances réelles IR(8,191) et IR(11,191), un temps de 91 secondes est acceptable mais un temps de 628 secondes est limite pour notre partenaire industriel. Comme le montrent les tableaux A.3 et A.4, la phase séquentielle département par département prend 242 secondes dont 189 secondes prises juste par le département 3, tandis que la réoptimisation avec le modèle exact prend 386 secondes.

Tableau 5.13 Résultats numériques pour l'heuristique en deux phases avec le modèle exact

Instance	Temps (sec)	Objectif	Demande totale	U(I)	O(I)	Nombre de transferts	Nombre de quarts
IA(5,200,2)	14	447202	3731	257	930	15	143
IA(10,400,1)	46	972436	7452	549	1989	38	285
IR(8,191)	91	5221300	34087	6343	1157	97	911
IR(11,191)	628	5196330	34087	6303	1144	67	883

Algorithme glouton

Nous avons aussi roulé l'algorithme glouton sur les instances de grande taille. Le tableau 5.14 montre les temps d'exécution et la valeur de la fonction objectif pour chacune des instances. Au premier coup d'œil, nous remarquons que les temps d'exécution deviennent plus stables avec l'algorithme glouton. En effet, pour les instances aléatoires, les temps de 12.08 et 38.4 secondes sont plus qu'acceptables. Aussi, pour les instances IR(8,191) et IR(11,191), les temps d'exécution de l'algorithme glouton sont relativement moins élevées (7.3 et 11.5

secondes respectivement). Mais comme la phase séquentielle prend plus de temps surtout pour l'instance IR(11,191), le temps d'exécution de cette instance demeure assez élevé.

Pour ce qui est de la qualité des solutions, nous remarquons que le taux de couverture de l'algorithme glouton pour les instances de grande taille reste très semblable à celui du modèle exact. Par contre, le nombre de transferts diminue d'environ 30%.

Tableau 5.14 Résultats numériques de l'heuristique avec l'algorithme glouton

Instance	Temps (sec)	Objectif	Demande totale	U(I)	O(I)	Nombre de transferts	Nombre de quarts
IA(5,200,2)	12.08	449027	3731	260	929	6	143
IA(10,400,1)	38.4	976716	7452	568	1952	38	285
IR(8,191)	76.3	5287018	34087	6364	1154	64	911
IR(11,191)	253.5	5270230	34087	6325	1152	38	883

L'heuristique en deux phases assure une couverture de la demande totale à plus de 99.8% et diminue considérablement les temps de calcul d'environ 94% pour les instances de petite taille. Même si la qualité des solutions est détériorée d'environ 30% pour ces instances, nous avons montré que l'heuristique peut s'améliorer encore plus en permettant de remettre en cause les quarts internes dans la première phase de résolution. Dans ce cas, l'heuristique arrive à fournir une solution à 3% de l'optimalité comme nous l'avons vu pour l'instance IA(5,20,1). Pour les instances de grande taille, l'heuristique permet d'obtenir des solutions en moins d'une minute pour les instances aléatoires avec 10 départements et 400 employés, et moins de 2 minutes pour les instances réelles avec 8 départements et 191 employés.

CHAPITRE 6

CONCLUSION

Dans ce chapitre, nous présentons une conclusion sur les travaux effectués dans le cadre de ce projet de mémoire. Après avoir élaboré la synthèse des travaux effectués dans la section 6.1, nous proposeront des possibles améliorations futures dans la section 6.2.

6.1 Synthèse des travaux

Dans le présent document, on s'intéresse à la confection d'horaires de travail personnalisés en tenant compte de la possibilité de transférer les employés entre les départements. Pour cela, nous avons développé un modèle global permettant de traiter le problème au complet comme un problème mixte en nombres entiers. Par la suite, nous proposons une approche à deux phases pour la résolution du problème car le modèle global échoue sur les instances de grande taille.

Le modèle global proposé permet d'assigner à chaque employé, durant l'horizon de planification, un ensemble de quarts personnalisés admissibles. Pour ce faire, une énumération explicite des quarts de travail est réalisé, ce qui tend à complexifier le modèle si la taille des instances augmente. Pour les instances de petite taille, ce modèle a pour avantage de fournir une solution optimale qui servira de point de départ pour l'évaluation de l'heuristique en deux phases. Par contre, on a montré que le temps d'exécution croit rapidement lorsque le nombre de départements et/ou d'employés augmente, ce qui empêche la résolution des instances de grande taille dans des temps raisonnables.

La première phase de l'heuristique dite séquentielle permet de traiter les départements un à un sans permettre les transferts. Pour cela, seuls les employés internes aux départements seront considérés. Nous avons montré que les différentes instances se résolvent très rapidement par le solveur Xpress-MP.

La deuxième phase de l'heuristique est une phase de réoptimisation où la solution obtenue dans la phase séquentielle est améliorée en ajoutant cette fois-ci des transferts mais sans remettre en question les quarts générés dans la solution séquentielle. Deux approches différentes ont été proposées, notamment, un modèle en nombres entiers résolu par Xpress-MP et

un algorithme glouton qui correspond à ce qui se fait en pratique. Les jeux de données testés ont montré un gain du temps d'exécution de plus de 94% par rapport au modèle global.

6.2 Améliorations futures

La phase de réoptimisation telle que présentée ne remet pas en question les quarts obtenus de la solution séquentielle. Une remise en question de la solution obtenue dans la première phase de résolution pourrait être envisagée, ce qui permettrait d'améliorer la solution obtenue et de générer plus de transferts.

Dans ce projet de recherche, nous nous intéressons au cas mono-activité, c'est-à-dire à la présence d'une seule activité par département. On pourrait aussi étendre l'heuristique en deux phases pour le cas multi-activités dans lequel une ou plusieurs activités sont réalisées dans chaque département. Dans ce cas, les transferts se feront en fonction du département et de l'activité visés, et non seulement du département comme c'est le cas présentement.

RÉFÉRENCES

- ADDOU, I. et SOUMIS, F. (2007). Bechtold-jacobs generalized model for shift scheduling with extraordinary overlap. *Annals of Operations Research*, 155, 177–205.
- AYKIN, T. (1996). Optimal shift scheduling with multiple break windows. *Management Science*, 42, 591–602.
- BAKER, K. R. (1976). Workforce allocation in cyclical scheduling problems : A survey. *Operational Research Society*, 27, 155–167.
- BAKER, K. R. et MAGAZINE, M. J. (1977). Workforce scheduling with cyclic demands and day-off constraints. *Management Science*, 24, 161–167.
- BAPTISTE, P., GIARD, V., HAÏT, A. et SOUMIS, F. (2005). *Gestion de production et ressources humaines : méthodes de planification dans les systèmes productifs*. Presses Internationales Polytechniques, Montreal, Canada.
- BARD, J. et WAN, L. (2006). The task assignment problem for unrestricted movement between workstation groups. *Journal of Scheduling*, 9, 315–341.
- BARD, J. F., BINICI, C. et DESILVA, A. H. (2003). Staff scheduling at the united states postal service. *Computers & Operations Research*, 30, 745 – 771.
- BARD, J. F. et WAN, L. (2008). Workforce design with movement restrictions between workstation groups. *Manufacturing & Service Operations Management*, 10, 24–42.
- BECHTOLD, S. E. et JACOBS, L. W. (1990). Implicit modeling of flexible break assignments in optimal shift scheduling. *Management Science*, 36, 1339–1351.
- BECHTOLD, S. E. et JACOBS, L. W. (1996). The equivalence of general set-covering and implicit integer programming formulations for shift scheduling. *Naval Research Logistics (NRL)*, 43, 233–249.
- BOUCHARD, M. (2004). *Optimisation des pauses dans le problème de fabrication d’horaires avec quarts de travail*. École polytechnique.
- BOUCHARD, M. (2008). *Coloration de graphes et attribution d’activités dans des quarts de travail*. École polytechnique.
- BRUSCO, M. J. et JACOBS, L. W. (2000). Optimal models for meal-break and start-time flexibility in continuous tour scheduling. *Management Science*, 46, 1630–1641.
- CÔTÉ, M.-C., GENDRON, B., QUIMPER, C.-G. et ROUSSEAU, L.-M. (2009). Formal languages for integer programming modeling of shift scheduling problems. *Constraints*, 16.

- CÔTÉ, M.-C., GENDRON, B., QUIMPER, C.-G. et ROUSSEAU, L.-M. (2011). Formal languages for integer programming modeling of shift scheduling problems. *Constraints*, 16, 54–76.
- CÔTÉ, M.-C., GENDRON, B. et ROUSSEAU, L.-M. (2007). Modeling the regular constraint with integer programming. P. Van Hentenryck et L. Wolsey, éditeurs, *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, Springer Berlin Heidelberg, vol. 4510 de *Lecture Notes in Computer Science*. 29–43.
- DAHMEN, S., REKIK, M. et HADJ-ALOUANE, A. (2011). Solving multi-activity personnel scheduling problems with a hybrid heuristic. *12e congrès annuel de la ROADEF*. Saint-Etienne, 87–88.
- DANTZIG, G. B. (1954). A comment on edie’s “traffic delays at toll booths”. *Journal of the Operations Research Society of America*, 2, 339–341.
- DANTZIG, G. B. et WOLFE, P. (1960). Decomposition principle for linear programs. *Operations Research*, 8, 101–111.
- DEN BERGH, J. V., BELIËN, J., BRUECKER, P. D., DEMEULEMEESTER, E. et BOECK, L. D. (2013). Personnel scheduling : A literature review. *European Journal of Operational Research*, 226, 367 – 385.
- EASTON, F. F. et MANSOUR, N. (1999). A distributed genetic algorithm for deterministic and stochastic labor scheduling problems. *European Journal of Operational Research*, 118, 505 – 523.
- EASTON, F. F. et ROSSIN, D. F. (1996). A stochastic goal program for employee scheduling*. *Decision Sciences*, 27, 541–568.
- EDIE, L. C. (1954). Traffic delays at toll booths. *Journal of the Operations Research Society of America*, 2, 107–138.
- ERNST, A., JIANG, H., KRISHNAMOORTHY, M., OWENS, B. et SIER, D. (2004a). An annotated bibliography of personnel scheduling and rostering. *Annals of Operations Research*, 127, 21–144.
- ERNST, A., JIANG, H., KRISHNAMOORTHY, M. et SIER, D. (2004b). Staff scheduling and rostering : A review of applications, methods and models. *European Journal of Operational Research*, 153, 3 – 27. Timetabling and Rostering.
- LEQUY, Q., BOUCHARD, M., DESAULNIERS, G., SOUMIS, F. et TACHEFINE, B. (2012a). Assigning multiple activities to work shifts. *Journal of Scheduling*, 15, 239–251.
- LEQUY, Q., DESAULNIERS, G. et SOLOMON, M. M. (2012b). A two-stage heuristic for multi-activity and task assignment to work shifts. *Computers & Industrial Engineering*, 63, 831 – 841.

LEQUY, Q., DESAULNIERS, G. et SOLOMON, M. M. (2013). Assigning team tasks and multiple activities to fixed work shifts. *Information Systems and Operational Research*, 51, 64 – 75.

OMARI, Z. (2002). *Attribution des activités aux employés travaillant sur des quarts*. École polytechnique.

REKIK, M., CORDEAU, J.-F. et SOUMIS, F. (2010). Implicit shift scheduling with multiple breaks and work stretch duration restrictions. *Journal of Scheduling*, 13, 49–75.

VATRI, É. (2001). *Intégration de la génération de quarts de travail et de l'attribution d'activités*. École polytechnique.

WARNER, D. M. et PRAWDA, J. (1972). A mathematical programming model for scheduling nursing personnel in a hospital. *Management Science*, 19, 411–422.

ANNEXE A : RÉSULTATS NUMÉRIQUES DE LA PHASE SÉQUENTIELLE

Tableau A.1 Résultats pour la première phase séquentielle des instances aléatoire de grande taille

Instance	Dép.	Temps (sec)	Objectif	Demande	U(I)	O(I)
IA(5,200,1)	1	2	90370	757	64	153
	2	3	132540	1117	44	327
	3	2	108530	755	69	208
	4	0	50600	372	56	82
	5	1	76290	730	56	120
IA(10,400,1)	1	1	106525	759	75	184
	2	1	45550	370	49	75
	3	1	102580	750	77	163
	4	1	44490	376	36	96
	5	2	130500	1134	53	283
	6	1	99565	738	75	157
	7	2	105910	741	78	171
	8	3	155090	1116	64	336
	9	2	149175	1096	70	290
	10	1	56200	372	54	108

Tableau A.2 Résultats pour la seconde phase de réoptimisation avec le modèle exact pour les instances aléatoires de grande taille

Instance	Temps (Sec)	Objectif	Demande initiale	U(I)	O(I)	Nombre de transferts
IA(5,200,2)	6	202622	289	257	930	15
IA(10,400,1)	31	462101	631	549	1989	38

Tableau A.3 Résultats pour la première phase séquentielle des instances réelles de grande taille

Instance	Dép.	Temps (sec)	Objectif	Demande	U(I)	O(I)
IR(8,191)	1	4	881300	3718	1158	0
	2	13	898610	4051	1179	0
	3	6	2046910	6529	2702	5
	4	7	256210	3439	149	497
	5	8	953560	5504	1227	38
	6	3	76110	761	44	155
	7	14	677450	5471	856	29
	8	14	413700	4614	384	406
IR(11,191)	1	4	881300	3718	1158	0
	2	14	898610	4051	1179	0
	3	189	841260	3654	1104	2
	4	5	849800	3676	1116	0
	5	4	268300	2638	275	197
	6	8	822700	4074	1063	32
	7	1	118200	692	128	76
	8	5	490600	3280	616	54
	9	4	247600	2848	266	138
	10	3	536850	2623	672	0
	11	5	293910	2833	229	428

Tableau A.4 Résultats pour la seconde phase de réoptimisation avec le modèle exact des instances réelles de grande taille

Instance	Temps (Sec)	Objectif	Demande	U(I)	O(I)	Nombre de transferts
IR(8,191)	22	4802770	7699	6343	1157	97
IR(11,191)	386	4833310	7806	6303	1144	67