



HAL
open science

Modèles et méthodes d'optimisation pour la mutualisation des chaînes logistiques

Juliette Medina

► **To cite this version:**

Juliette Medina. Modèles et méthodes d'optimisation pour la mutualisation des chaînes logistiques. Recherche opérationnelle [math.OC]. Ecole des Mines de Nantes, 2016. Français. NNT : 2016EMNA0299 . tel-01460708

HAL Id: tel-01460708

<https://theses.hal.science/tel-01460708>

Submitted on 1 Mar 2017

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

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

Thèse de Doctorat

Juliette MEDINA

*Mémoire présenté en vue de l'obtention du
grade de Docteur de l'École nationale supérieure des Mines de Nantes
sous le sceau de l'Université Bretagne Loire*

École doctorale : Sciences et Technologies de l'Information, et Mathématiques (STIM)

Discipline : Informatique et applications, section CNU 27

Spécialité : Informatique et recherche opérationnelle

Unité de recherche : Institut de Recherche en Communications et Cybernétique de Nantes (IRCCyN)

Soutenue le 8 décembre 2016

Thèse n° : 2016EMNA0299

Modèles et méthodes d'optimisation pour la mutualisation des chaînes logistiques

JURY

Rapporteurs : **M. Bernard GENDRON**, Professeur, Université de Montréal
M. Dominique FEILLET, Professeur, École des Mines de Saint Étienne

Examineurs : **M. Frédéric SEMET**, Professeur, École Centrale de Lille
M. Alexandre DOLGUI, Professeur, École des Mines de Nantes
M. Marc SEVAUX, Professeur, Université de Bretagne Sud

Invités : **M. Xavier PERRAUDIN**, Président, 4S Network
M. Christian LEROUX, Directeur Général, 4S Network

Directeur de thèse : **M. Olivier PÉTON**, Maître assistant HDR, École des Mines de Nantes

Co-encadrant de thèse : **M. Fabien LEHUÉDÉ**, Maître assistant HDR, École des Mines de Nantes

Table des matières

1	Introduction	9
1.1	La mutualisation du transport : motivations et enjeux	9
1.2	Problématiques de recherche de cette thèse	11
1.3	Plan de la thèse	12
2	Description du réseau logistique mutualisé	15
2.1	4S Network	16
2.2	CRC services	16
2.3	Sites du réseau logistique	18
2.3.1	Réseau amont	18
2.3.2	Réseau aval	20
2.4	Flux de marchandises : commandes et sous-commandes	20
2.4.1	Commandes	20
2.4.2	La préparation de commande	20
2.4.3	Sous-commandes	22
2.5	Transporteurs et tournées de véhicules	22
2.5.1	Types de tournées et structures tarifaires dans le réseau amont	23
2.5.2	Types de tournées et structures tarifaires dans le réseau aval	23
2.6	Itinéraires des commandes et sous-commandes	25
2.6.1	Itinéraire des sous-commandes dans le réseau amont et le réseau direct	25
2.6.2	Itinéraire des commandes dans le réseau aval	25
2.7	Un exemple de fonctionnement global du réseau	25
3	Cadre scientifique de la thèse	27
3.1	Facturation du transport	28
3.2	Fonction linéaire par morceaux	28
3.2.1	Définition	28
3.2.2	Modélisation pour un véhicule	29
3.2.3	Modélisation pour plusieurs véhicules	29
3.3	Modélisation des transports dans un réseau espace temps	31
3.4	Génération de colonnes	32
3.5	La recherche tabou	35

4	État de l'art : problèmes de conception de réseau logistique et tournées riches de véhicules	37
4.1	Conception de réseau de transport	38
4.1.1	Fonctions de coût linéaires par morceaux	38
4.1.2	Merge-in-transit	39
4.2	Planification opérationnelle des transports dans un réseau	39
4.2.1	Équilibre	40
4.2.2	Cycles	40
4.2.3	Coûts	41
4.2.4	Application des problèmes de type SNDP	41
4.2.5	Synthèse	42
4.3	Tournées de véhicules avec transfert	42
4.3.1	Tournées de véhicules avec cross-dock	42
4.3.2	Problèmes de collectes et livraisons avec transferts	43
4.4	Problèmes riches de tournées de véhicules	44
4.4.1	Tournées multi-trip	44
4.4.2	Contraintes temporelles sur les tournées	45
4.4.3	Transport sous-traité	45
4.4.4	Collaboration entre transporteurs	46
4.5	Conclusion	46
5	Modèles mathématiques pour l'estimation des coûts dans le réseau mutualisé	47
5.1	Introduction	48
5.2	Description du problème modélisé	48
5.2.1	Définition des ensembles du réseau logistique	48
5.2.2	Tournées de véhicules et structure tarifaire	49
5.2.3	Flux et structure des marchandises	52
5.3	Modélisation des transferts	53
5.3.1	Structure de transfert complète	53
5.3.2	Structure de transfert en étoile	53
5.4	Modèles mathématiques	54
5.4.1	Itinéraire amont	55
5.4.2	Itinéraire aval	58
5.4.3	Itinéraires directs	58
5.4.4	Lien entre réseau amont et réseau aval	59
5.4.5	Satisfaction de la demande	59
5.4.6	Modélisation des tournées	59
5.4.7	Fonction objectif	60
5.4.8	Deux modèles mathématiques	61
5.5	Tests numériques	63
5.5.1	Instances	63
5.5.2	Résultats	67
5.6	Conclusion	70

6	Distribution depuis les CRC : un problème riche de tournées de véhicules	71
6.1	Description du problème	72
6.1.1	Commandes	72
6.1.2	Modélisation : un sommet pour chaque commande	73
6.1.3	Transport	73
6.1.4	Définition du problème traité	76
6.2	La génération de colonnes pour les problèmes riches de tournées de véhicules	77
6.3	Méthode de résolution	79
6.3.1	Modélisation du problème maître	79
6.3.2	Ensemble de colonnes initiales	81
6.3.3	Résolution des sous-problèmes	81
6.3.4	Tests de réalisabilité d'une insertion	86
6.4	Tests numériques	88
6.4.1	Instances	88
6.4.2	Paramétrage	91
6.4.3	Résultats	92
6.4.4	Analyse des résultats selon le pourcentage d'horaires fixés	94
6.5	Conclusion et perspectives	96
7	Optimisation conjointe des réseaux amont et aval	99
7.1	Introduction	100
7.2	Modélisation du réseau amont	100
7.2.1	Description et modélisation du réseau	100
7.2.2	Coût des tournées dans le réseau amont	102
7.3	Modélisation du réseau aval	102
7.3.1	Description et modélisation du réseau	102
7.3.2	Coûts des tournées dans le réseau aval	103
7.4	Commandes et sous-commandes	104
7.5	Modélisation du problème $\mathcal{F}_{SND RP-CRC}$	105
7.5.1	Modèle $\mathcal{F}_{SND RP-CRC}$	106
7.5.2	Modélisation pour la génération de colonnes $\mathcal{F}^L_{SND RP-CRC}$	107
7.6	Matheuristique pour la résolution du SND RP-CRC	110
7.6.1	Solution initiale	110
7.6.2	SP1 – Génération des itinéraires amont	112
7.6.3	SP2 – Génération des tournées de véhicules aval	114
7.7	Tests numériques	116
7.7.1	Génération des instances	116
7.7.2	Paramétrage de la recherche tabou	116
7.7.3	Écart entre la relaxation linéaire et la solution	119
7.7.4	Influence de la discrétisation	119
7.7.5	Influence du coût de stockage	121
7.8	Conclusion et perspectives	123

8	Conception intégrée de plans de chargement et de tournées de véhicules	125
8.1	Introduction	125
8.2	Description du problème	127
8.3	Formulations sur les tournées et sur les arcs du SNDRP	128
8.3.1	Notations	128
8.3.2	Réseau espace temps	129
8.3.3	Modélisation du SNDRP	129
8.3.4	Formulation \mathcal{F}_R	130
8.3.5	Formulation \mathcal{F}_A	131
8.4	Méthode de résolution	133
8.5	Tests numériques	137
8.5.1	Instances	137
8.5.2	Comparaison des deux formulations	138
8.5.3	Comparaison logistique	138
8.6	Conclusion	142
9	Outils logiciels réalisés	145
9.1	Choix technologiques	146
9.2	Outil de simulation	146
9.2.1	Présentation de l'outil	147
9.2.2	Données d'entrées et paramètres	147
9.2.3	Résultats	148
9.3	Outil opérationnel	150
9.3.1	Présentation de l'outil	150
9.3.2	Utilisation de l'outil	151
9.4	Conclusion et perspectives	155
10	Conclusion	157
	Acronymes	169
	Liste des tableaux	169

Remerciements

Tout d'abord je souhaite remercier chaleureusement mes rapporteurs de thèse, Bernard Gendron et Dominique Feillet, pour avoir accepté de juger mon travail, ainsi que mon jury de thèse Frédéric Semet, Alexandre Dolgui et Marc Sevaux.

Je tiens également à remercier mes encadrants Olivier et Fabien, pour leur disponibilité et leur confiance dans le travail que j'ai réalisé. Vous m'avez encouragée et guidée tout au long de cette thèse et dans le monde de la recherche, dans nos séances de travail comme à la cervoiserie. J'espère pouvoir continuer à collaborer avec vous sur des projets de recherche tous aussi intéressants et challengeants.

Je souhaite également remercier sincèrement les autres chercheurs avec lesquels j'ai travaillé avant ou pendant cette thèse : je remercie particulièrement Sylvain Fournier, rencontre fortuite au bord du lagon de Florianopolis et grâce à qui j'ai pu m'initier à la RO et à la génération de colonnes tout en continuant à profiter des plages paradisiaques de Floripa. Je remercie également Mike Hewitt grâce à qui j'ai effectué un séjour académique à l'université de Loyola de Chicago, pour ses précieux conseils, son expertise scientifique et sa bonne humeur constante.

Je remercie également toute l'équipe de 4S Network, Xavier, Christian, Valérie, Inès, Anthony et Yann, pour leur confiance, leur implication et leur enthousiasme pour mon travail comme pour le projet CRC. Les visites régulières à Rueil me démontraient l'intérêt industriel de mon sujet de recherche, et vos encouragements m'ont permis de rester motivée tout au long de ces trois ans.

Je ne peux oublier dans ces remerciements mes collègues du bureau CO25b, Axel, Thomas, Yuan et Fabrice. J'ai en mémoire tellement de bons moments passés dans ce bureau que j'ai peur d'en oublier : les longues discussions scientifiques (ou pas), 2048 et agar.io, les cours de chinois et la nourriture associée, notre défaite injuste au concours de logos, vos encouragements dans les derniers moments de rédaction ainsi que pour ma soutenance... Merci pour tout. Plus généralement je remercie sincèrement l'ensemble de l'équipe du DAPI pour m'avoir fait passer trois années très agréables au sein de l'école des Mines et en dehors.

Je tiens à remercier ma famille pour leurs encouragements et leur enthousiasme pour tous mes projets professionnels. Les moments que nous passons ensemble me sont très chers et nos échanges ont fortement participé à ma motivation pendant ces trois ans.

J'ai également une forte pensée pour mes amis issus de rencontres Nantaises et (beaucoup) plus lointaines, mais également pour mes amis de l'Insa Lyon qui se sont vus progressivement mêlés à ceux de

l'Insa Rennes, pour des aventures toujours plus folles et audacieuses auxquelles j'adore toujours autant participer.

Enfin, merci Valentin, pour ton soutien inconditionnel, ta patience et notre amour. Merci pour tous ces moments passés et ceux à venir.

Introduction

Sommaire

1.1 La mutualisation du transport : motivations et enjeux	9
1.2 Problématiques de recherche de cette thèse	11
1.3 Plan de la thèse	12

1.1 La mutualisation du transport : motivations et enjeux

La logistique de la grande distribution est depuis des décennies au coeur des préoccupations politiques et sociétales. Elle n'a cessé de se transformer, pour passer du stade artisanal à une organisation aujourd'hui industrielle. La complexification des flux de marchandises peut s'expliquer de manière simpliste par l'augmentation du nombre de points de ventes et des références de produits. Elle s'explique aussi par une certaine course à la performance, qui se décline de multiples façons : mise en place de réseaux logistiques articulés autour de plateformes de stockage ou de cross-docking, diminution des temps de transport, heures de livraison de plus en plus précises, diminution des espaces de stockage en magasin. Ces différents facteurs ont pour effet d'augmenter les fréquences de livraison des commandes. Les politiques de réduction des coûts de transport sont menées à la fois par les fournisseurs et la grande distribution : ces coûts représentent en moyenne 8% du prix d'un produit en magasin, et peuvent monter jusqu'à 15% pour des produits pondéreux comme les eaux minérales¹. Les politiques actuelles visent également à diminuer l'impact environnemental de l'ensemble de la chaîne logistique (conception des produits, transport, impact des sites logistiques). La préoccupation actuelle des entreprises est de diminuer cet impact sans dégrader la performance logistique. En particulier, les émissions liées au transport représentent 27,6% des émissions de gaz à effet de serre en France². Avec un taux moyen de remplissage

¹source : CEREMA, 2008

²source : Ministère de l'Environnement de l'Énergie et de la Mer, 2013

des camions de 65%³, il reste une réelle marge de progression.

Les projets de mutualisation dans les chaînes logistiques, apparus depuis une dizaine d'années, ont justement pour but de trouver de nouveaux moyens de réduire les coûts de transport, et par là même diminuer l'impact environnemental et améliorer la qualité de service aux clients. Ces projets s'inscrivent dans le cadre plus vaste de l'économie collaborative, dont l'émergence a été rendue possible par la révolution numérique.

Dans ce manuscrit, nous étudions la mutualisation du transport entre fournisseurs de la grande distribution. La mutualisation des opérations de transport est celle qui vient en premier à l'esprit lorsqu'on parle de mutualisation⁴. Elle présente un double effet de levier. Lorsque plusieurs petites commandes, provenant de fournisseurs différents, sont co-chargées dans un même véhicule, des économies d'échelle substantielles peuvent être réalisées sur le prix du transport. Si, de plus, le chargement est suffisamment volumineux pour représenter un camion complet, le tarif correspondant s'avère encore plus avantageux. Le bénéfice de la mutualisation est donc évident pour les fournisseurs qui payent le transport. La mutualisation présente également des avantages pour les distributeurs. Les livraisons groupées de commandes provenant de différents fournisseurs permettent de réduire le nombre de rendez-vous à planifier et de véhicules à décharger. En outre, les fournisseurs peuvent plus facilement, grâce à cette organisation, multiplier leurs fréquences de livraison, ce qui améliore la qualité de service. Enfin, ce type de fonctionnement profite également aux transporteurs sélectionnés, qui se voient confier une charge plus volumineuse et régulière. Grâce à un taux de remplissage plus élevé, la rentabilité des véhicules est accrue.

Les entreprises qui engagent des projets de collaboration doivent néanmoins faire face à de nombreux changements dans leur mode de management, leur système d'information et dans la gestion de leur réseau logistique. Les obstacles culturels, organisationnels et techniques ne manquent pas. Aussi, la réussite de projets de mutualisation est étroitement liée à une implication forte de la direction des entreprises concernées et à l'intervention de tiers chargés de conduire ces projets. En phase opérationnelle, l'exploitation d'un système logistique mutualisé requiert souvent l'intervention d'un prestataire de type SPL, en charge d'organiser le transport et de coordonner les différents acteurs de la chaîne logistique.

Ces travaux se placent dans le cas d'un tel prestataire. Les travaux de recherche décrits dans ce manuscrit ont été menés à l'École des Mines de Nantes et dans l'entreprise 4S Network⁵. Le rôle de cette entreprise est de promouvoir et mettre en oeuvre des projets de mutualisation logistique pour les fournisseurs de la grande distribution, ce qui inclut plusieurs tâches :

- définir le fonctionnement du réseau mutualisé ;
- contractualiser les accords de collaboration ;
- rechercher des prestataires logistiques pour le transport et le cross-docking ;
- construire les plans de transport ;
- facturer.

De nombreuses problématiques d'optimisation logistique sont liées à la mise en place de la mutualisation, à tous les niveaux de décision. Au niveau stratégique, on trouve le choix de la localisation des

³source : Ministère des Transports, 2010

⁴<https://mutualisationlogistique.com/>, baromètre 2015

⁵<http://www.4snetwork.com/>

plateformes de mutualisation et leur dimensionnement. Ces aspects ne seront pas abordés dans cette thèse : on suppose que les décisions stratégiques ont été prises au préalable.

Au niveau tactique, la conception et le dimensionnement du réseau de transport est un facteur clé pour l'organisation de la mutualisation. Les modèles d'optimisation aident à la décision sur plusieurs plans : définition des horaires d'ouverture et de fermeture des plateformes du réseau, planification de certaines lignes de transport à horaire fixe (train, bateau), dimensionnement des ressources humaines et techniques.

Enfin, au niveau opérationnel, il s'agit de planifier les opérations logistiques quotidiennes, telles que les tournées de véhicules, les horaires de travail des conducteurs et employés des plateformes de cross-docking, la réaction aux aléas.

Dans cette thèse, nous nous intéressons principalement à la construction de plans de transport, c'est-à-dire au routage des commandes à travers le réseau logistique. Les projets de mutualisation conduits par 4S Network s'appuient sur un réseau de plateformes logistiques appelées Centres de Routage Collaboratifs (CRC)⁶. Les CRC sont des plateformes de cross-docking qui constituent un échelon intermédiaire entre les sites logistiques de la grande distribution (appelés ici *clients*) et leurs fournisseurs. Leur objectif est de consolider les flux marchandises en provenance de fournisseurs multiples, afin de les livrer simultanément à leurs clients. Le réseau logistique est donc naturellement découpé en deux parties.

- Le réseau amont concerne les transports depuis les fournisseurs vers les CRC. Ce réseau contient généralement des liaisons longue distance (à l'échelle d'un pays) ; le transport peut être routier, ferré, aérien ou maritime.
- Le réseau aval concerne la distribution des marchandises depuis un CRC vers un ensemble de clients. L'échelle d'un tel réseau est régionale, le mode de transport privilégié est le camion (tournées de véhicules).

1.2 Problématiques de recherche de cette thèse

Le but de cette thèse est de proposer des modèles mathématiques et des méthodes d'optimisation pour des problématiques de transport mutualisé entre fournisseurs et distributeurs de la grande distribution. La contribution principale de cette thèse est de proposer une approche intégrant le transport dans les réseaux amont et aval. Les problématiques de ces deux types de réseau, considérées indépendamment, constituent deux problèmes bien identifiés dans la littérature :

- les problématiques de transport dans le réseau longues distances correspondent à un problème conception de plan de chargement ([Service Network Design Problem \(SNDP\)](#)) ;
- les problématiques de transport dans chaque réseau local correspondent à des problèmes de tournées de véhicules ([Vehicle Routing Problem \(VRP\)](#)) ;

Le principal verrou scientifique concerne la synchronisation du réseau amont et du réseau aval. Une approche répandue est de contourner les contraintes de synchronisation en formulant certaines hypothèses simplificatrices, par exemple, imposer chaque jour des fenêtres horaires disjointes pour les heures

⁶<http://www.crc-services.com/>

d'arrivée au CRC et la distribution dans le réseau aval. Ce genre d'hypothèse rend l'optimisation particulièrement inefficace lorsque les délais de livraison sont très courts. C'est pourquoi, nous proposons dans cette thèse des modèles et des méthodes de résolution pour une décision intégrée d'un problème de plan de chargement et de plusieurs problèmes de tournées de véhicules. Gardant en vue l'applicabilité immédiate de ces travaux par 4S Network, nous avons inclus dans ces modèles un grand nombre de contraintes métier.

1.3 Plan de la thèse

Le chapitre 2 définit les problématiques industrielles rencontrées par 4S Network. Nous présentons en détail le réseau logistique mutualisé et son fonctionnement. Nous décrivons l'organisation du transport et les itinéraires potentiels des commandes dans le réseau. Le chapitre 3 récapitule des notions scientifiques nécessaires à la bonne compréhension du manuscrit. Le chapitre 4 présente un état de l'art sur les problèmes de conception de réseau logistique et tournées riches de véhicules.

Notre première contribution scientifique apparaît dans le chapitre 5. Celui-ci propose une modélisation du problème de transport dans le réseau logistique de 4S Network, comprenant un réseau longue distance et plusieurs réseaux de distribution locale. Ce modèle sert à estimer les coûts de transport avant et après mutualisation. Les résultats permettent d'observer un gain lié à la mutualisation compris entre 3% et 10%. Ce chapitre propose également une comparaison entre deux modélisations du transfert de marchandises d'un véhicule à un autre.

Le chapitre 6 propose une méthode de résolution pour le problème de tournées de véhicules au départ d'un CRC. Nous proposons une heuristique basée sur la génération de colonnes. Le sous-problème de la génération de colonnes est résolu de manière heuristique grâce à une recherche tabou. Cette recherche tabou intègre un algorithme d'ordonnancement minimisant le temps d'attente de chaque tournée. Des instances de test basées sur des données réelles sont générées et comprennent deux types de commandes : des commandes dont l'horaire de livraison est fixé (comme il est généralement le cas) et des commandes dont la livraison doit respecter une certaine fenêtre horaire. Sans surprise, les résultats montrent que les instances avec le moins d'horaires fixés sont celles de moindre coût. Plus spécifiquement, on remarque que même un petit pourcentage de commandes sans horaires fixes permet de réaliser une économie considérable.

Le chapitre 7 aborde le problème de transport dans l'ensemble du réseau (amont+aval). Nous proposons également une heuristique basée sur la génération de colonnes. Deux types de colonnes sont générées, modélisant respectivement l'itinéraire des commandes dans le réseau amont, et les tournées de véhicules dans le réseau aval. Le premier des deux sous-problèmes posés pour la génération des colonnes est résolu par un algorithme de programmation dynamique ; le second est résolu par une recherche tabou. La synchronisation entre l'amont et l'aval est assurée par des contraintes du problème maître de la génération de colonnes. Des instances basées sur des données réelles sont générées et utilisées afin de montrer l'intérêt d'une prise en compte conjointe de l'amont et de l'aval. Le passage par un CRC occasionne un coût additionnel lié à la manutention et au stockage. Nous montrons que ce surcoût est largement compensé par l'adaptation des itinéraires et l'ordonnancement des tournées.

Le chapitre 8 est le fruit d'un séjour de deux mois à Loyola University à Chicago, en collaboration avec Mike Hewitt. Ces travaux ont une visée plus théorique que les problèmes traités précédemment. Ils concernent la résolution de problèmes de plan de chargement intégrant les tournées de distribution. Le chapitre propose deux formulations qui sont adaptées à un algorithme de résolution d'un **SNDP**. La première formulation permet d'intégrer des contraintes riches aux tournées de véhicules. Elle a pour inconvénient de générer un nombre important de variables et passe difficilement à l'échelle pour la résolution d'instances de taille moyenne. La seconde formulation ne permet pas l'intégration de contraintes riches, mais sa résolution est beaucoup plus rapide et permet de résoudre des instances de plus grande taille.

Description du réseau logistique mutualisé

Sommaire

2.1	4S Network	16
2.2	CRC services	16
2.3	Sites du réseau logistique	18
2.3.1	Réseau amont	18
2.3.2	Réseau aval	20
2.4	Flux de marchandises : commandes et sous-commandes	20
2.4.1	Commandes	20
2.4.2	La préparation de commande	20
2.4.3	Sous-commandes	22
2.5	Transporteurs et tournées de véhicules	22
2.5.1	Types de tournées et structures tarifaires dans le réseau amont	23
2.5.2	Types de tournées et structures tarifaires dans le réseau aval	23
2.6	Itinéraires des commandes et sous-commandes	25
2.6.1	Itinéraire des sous-commandes dans le réseau amont et le réseau direct	25
2.6.2	Itinéraire des commandes dans le réseau aval	25
2.7	Un exemple de fonctionnement global du réseau	25

Ce chapitre présente le contexte de la thèse. Il décrit tout d'abord la société 4S Network et le service que cette société a conçu. Les différentes composantes du réseau de transport proposé à travers ce service sont ensuite détaillées et illustrées.

2.1 4S Network

La société 4S Network accompagne ses clients dans la conception et l'organisation de solutions innovantes dédiées à la supply chain. Ce type de société, initialement appelée **Third-party logistics (3PL)** (logistique tierce partie), gère les flux logistiques de ses clients et coordonne les différents acteurs de leur supply chain. L'objectif est d'améliorer la performance de la chaîne logistique globale. Avec l'évolution des métiers de la logistique, le terme de **Third-party logistics (3PL)** a été étendu : les 4PL sont spécialisés dans le conseil et la mise en place de nouvelles solutions logistiques, tandis que les 5PL offrent en plus de nouveaux services en systèmes d'information améliorant les performances des chaînes logistiques [52]. 4S Network exerce aujourd'hui en tant que 3PL, 4PL ou 5PL. Dans cette thèse, pour plus de clarté, on assimilera ces trois secteurs au métier du 3PL.

4S Network travaille sur trois axes majeurs :

- Une *visualisation de l'empreinte environnementale du transport*, par le développement d'une plateforme web collaborative ¹ servant de tableau de bord pour le calcul des émissions de CO_2 . Cette plateforme permet de piloter une stratégie environnementale d'entreprise dans le domaine du transport de marchandises.
- Un *fonctionnement collaboratif en réseau* par le produit/service KAYPAL MR² permettant de piloter un réseau de palettes carton. Sur une palette, deux couches de produits différents sont séparées par une palette, appelée *palette de séparation*. L'idée est de remplacer ces palettes de séparation en bois par des palettes en carton réutilisables en moyenne 5 fois.
- La *conception et gestion d'un réseau collaboratif*, où l'objectif est de mettre en place des projets de mutualisation du transport. Cette thèse s'inscrit dans ce contexte, et plus particulièrement sur le projet des Centres de Routage Collaboratifs³, appelé CRC services, pour la grande distribution.

2.2 CRC services

Cette thèse a été réalisée dans le contexte de la montée en puissance du projet CRC services, initié en 2012. Son objectif principal est d'intervenir en support à la création du réseau et de produire des outils d'optimisation afin d'en améliorer le fonctionnement. On considère ici le cas d'une coopération entre plusieurs fournisseurs et distributeurs de la grande distribution, autour de plateformes de cross-docking appelées *Centres de Routage Collaboratif (CRC)* ⁴.

Dans un réseau logistique classique de la grande distribution on trouve généralement trois types d'acteurs :

- Les *fournisseurs* – qui possèdent leur propre réseau logistique d'entrepôts et d'usines de production. Ces sites sont les points de départ des expéditions. On appelle *site fournisseur* tout site d'un fournisseur.

¹<http://www.4snetwork.com/activite/environnement/stm/>

²<http://www.4snetwork.com/activite/produits-services/otc-kaypal-mr/>

³<http://www.crc-services.com/>

⁴CRC et CRC services sont des marques déposées par 4S Network

- les *clients* qui reçoivent les commandes. On trouve dans cette catégorie des entrepôts de la grande distribution, des supermarchés ou des enseignes plus petites.
- les *transporteurs* qui délivrent les commandes d'un fournisseur jusqu'à un client.

Une commande d'un client vers un fournisseur est composée d'un ensemble de produits (que l'on définira plus tard par lignes de commandes) qui peuvent être répartis sur différents sites. L'unité logistique est la palette. Des palettes composées de différents produits peuvent être préparées sur des sites fournisseurs.

En France, l'essentiel du transport routier de marchandises est assuré par des PME/TPE (97% des entreprises ont moins de 50 salariés). Ces entreprises ne disposent généralement pas d'un réseau logistique et d'un volume de commandes suffisant pour transporter elles même leurs marchandises vers leurs clients. Il arrive même que le transporteur auquel un fournisseur a confié ses commandes sous-traite une partie des transports par manque de flux aller ou retour dans certaines régions. Cette accumulation d'intermédiaires sans organisation globale ne peut pas être efficace : la répartition des flux sur un nombre élevé de transporteurs engendre un faible taux de remplissage des véhicules. Ainsi, sur le plan économique, les fournisseurs paient plus cher pour de petits volumes de palettes livrés. Concernant la qualité de service, la fréquence de livraison que peut proposer un fournisseur n'est pas aussi élevée que le souhaiteraient les clients. D'un point de vue environnemental, un plus grand nombre de véhicules en sous capacité sont affrétés. Ils ont un impact négatif élevé sur l'environnement.

Pour remédier aux problèmes listés ci-dessus, 4S Network souhaite orchestrer une collaboration plus verticale (entre fournisseurs, transporteurs et clients) sur un secteur d'activité donné : la grande distribution. Contrairement aux approches classiques qui consistent à consolider les flux à proximité des fournisseurs, le service CRC propose un ensemble de points de consolidation proches des clients. Il offre la possibilité aux fournisseurs d'alimenter un CRC en camions pleins sur de longues distances. CRC services se place comme l'interlocuteur des clients pour convenir d'un créneau de livraison pour l'ensemble de leurs fournisseurs et réaliser efficacement des tournées de livraison depuis le CRC. Ce fonctionnement permet à des fournisseurs de toutes tailles de pouvoir transporter leurs marchandises à des prix raisonnables dans le réseau logistique. Enfin, les transporteurs peuvent collaborer avec le service CRC en proposant leurs services de transport pour certaines régions. Ils peuvent également mettre à disposition une partie de leurs plateformes logistiques, afin d'y proposer le service CRC. En échange, le service CRC leur garantit des flux à la fois plus volumineux et plus réguliers.

Ce projet propose également un réseau logistique transparent entre fournisseurs et clients : grâce à un service de traçabilité des palettes passant par les CRC, les fournisseurs peuvent avoir une vision du transit de leurs commandes, ainsi que des coûts associés. Cette prise en compte globale du volume et le traitement global du transport de l'ensemble des couples fournisseur-client permet une optimisation des flux beaucoup plus efficace à tous les niveaux.

On retrouve donc dans cette nouvelle organisation du réseau logistique les trois types d'entités vus précédemment, auxquels viennent s'ajouter les plateformes collaboratives régionales (CRC). Elles permettent de réaliser plus simplement la mutualisation des marchandises en aval des CRC et facilitent la collaboration entre transporteurs. Ce fonctionnement est profitable à tous les acteurs :

- L'organisation du transport des commandes pour les fournisseurs est simplifiée : les commandes à destination de plusieurs clients d'une même région sont envoyées vers un unique CRC.

- Les transporteurs ont en charge un volume de commande suffisant pour assurer de bons taux de remplissage des véhicules : ils ne doivent plus se soucier de sous-traiter une partie de leur transport à d'autres transporteurs.
- Les clients gagnent en qualité de service : l'accroissement des taux de remplissage permet aux fournisseurs d'envoyer de plus petites commandes sans augmenter le coût du transport. Cela permet donc une hausse des fréquences de livraison. Dans le même temps, le nombre de véhicules à réceptionner diminue, car les commandes de différents fournisseurs sont regroupées sur CRC et livrées ensemble aux clients.

En parallèle de la thèse, CRC services a fait l'objet d'un projet Predit avec partenariat avec GS1 et Mines ParisTech à partir de 2013. Un projet pilote a démarré en 2014 avec la mise en place d'une première plateforme CRC dans le sud de la France à Saint Martin De Crau. Elle dessert plus de 100 plateformes de distributeurs tels que Auchan, Métro ou Intermarché dont les marchandises proviennent de plusieurs fournisseurs comme SC Johnson, le Pool GILDE (Mapa Spontex, Bourjois, Wilkinson Sword), Mars, Ferrero. Cette initiative de mutualisation a été récompensée par le "Grand Prix des ROIS de la Supply Chain" en 2016. La même année, cinq nouvelles plateformes CRC ont été ouvertes à Arras, Brumath, Lomme, Nantes et Bordeaux. L'objectif est d'implanter une douzaine de CRC sur toute la France, avec plusieurs centaines de milliers de mouvements de palettes prévus dès 2018.

2.3 Sites du réseau logistique

Nous considérons un réseau logistique tel qu'illustré par la figure 2.1. Il comprend trois fournisseurs avec respectivement 2, 4 et 1 sites logistiques, deux CRC et 18 clients. On distingue un *réseau amont*, allant des fournisseurs aux CRC et un *réseau aval*, allant des CRC aux clients. On considère de plus que des commandes peuvent être expédiées directement du fournisseur au client. On parle alors de *réseau direct*.

2.3.1 Réseau amont

Le réseau amont concerne le transport entre les sites de production ou d'entreposage des fournisseurs et les CRC. Les sites des fournisseurs sont les origines de tous les produits qui transitent dans le réseau. Un fournisseur peut posséder un ou plusieurs sites : certains dédiés à la production (usines) et d'autres au stockage (entrepôts). Ces sites peuvent avoir différentes fonctions logistiques. Afin de généraliser ces fonctions on ne distingue pas les usines des entrepôts, mais on considère que chaque *site fournisseur* peut posséder – ou non – les fonctions logistiques suivantes :

- *Transfert*. On définit le transfert par l'action de décharger des marchandises d'un véhicule et de les recharger dans un véhicule différent.
- *Préparation de commande*. La préparation de commande ou *picking* consiste à regrouper différents produits sur une même palette. Elle implique un transfert pour les produits qui ne sont pas fabriqués sur le site de préparation. Nous reviendrons plus longuement sur cette fonction dans la section 2.4.2.

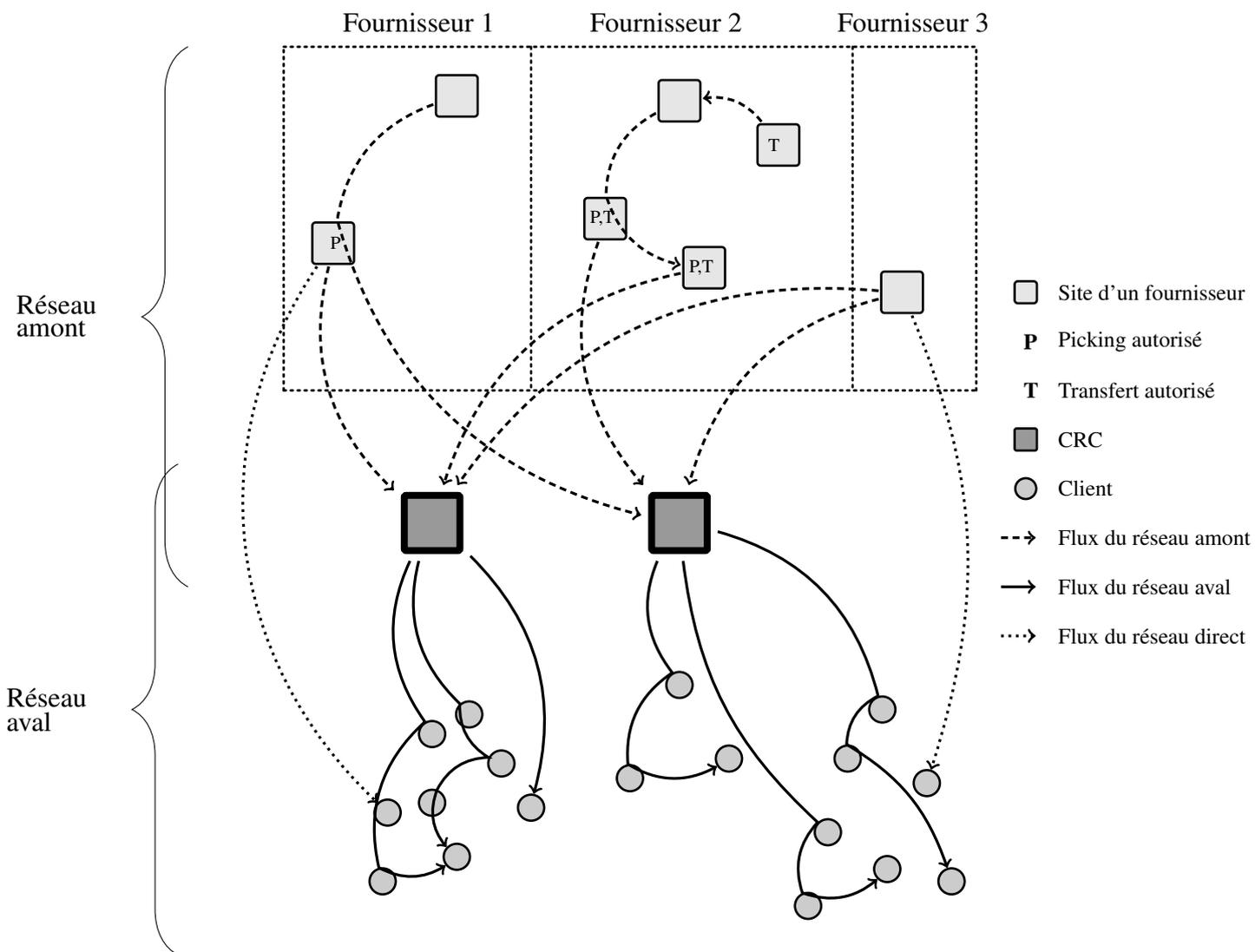


Figure 2.1 – Structure globale du réseau

Ces fonctions logistiques sont applicables aux marchandises du fournisseur qui possède le site en question : les marchandises d'un fournisseur ne peuvent pas être transférées ni être préparées sur un site appartenant à un autre fournisseur.

2.3.2 Réseau aval

Le réseau aval réalise la distribution des marchandises depuis les CRC jusqu'aux clients. Les véhicules partent de chaque CRC et livrent un ou plusieurs clients.

2.4 Flux de marchandises : commandes et sous-commandes

2.4.1 Commandes

Une commande est définie comme un ensemble de produits qu'un fournisseur doit livrer à un client à une date donnée. La taille d'une commande peut-être exprimée suivant différentes dimensions : nombre de palettes et/ou poids. Notons qu'une commande a une destination unique, mais peut avoir plusieurs origines : les produits d'une commande peuvent provenir de différents sites d'un même fournisseur, mais doivent être livrés au client par le même véhicule.

2.4.2 La préparation de commande

On distingue deux types de palettes :

- *Les palettes homogènes* contiennent un seul type et une quantité toujours identique de produits. En sortie de production, toutes les palettes sont supposées homogènes.
- *Les palettes hétérogènes* peuvent contenir plusieurs types de produits et des quantités ajustées pour correspondre à une commande particulière. Le fait de créer une palette hétérogène est appelé *préparation de commandes* ou *picking*. Les usines ne sont pas toujours en mesure d'effectuer cette préparation de commande et les palettes d'une commande doivent parfois être préparées dans un entrepôt ou une autre usine du fournisseur. Lorsque la préparation ne peut-être faite sur le site de production, la préparation se traduit donc par un transfert sur un site de préparation du fournisseur. CRC services ne gère pas les stocks des fournisseurs, qui garde la main sur ses flux internes. Aussi, ne sont communiquées à CRC services que les quantités correspondant aux commandes des clients et les décisions de passage par les sites de préparation de commande.

Exemple 1. La figure 2.2 illustre un exemple de préparation de palettes. Une commande comprend 5 types de produits A,B,C,D et E), avec une demande respective de 1.9, 0.7, 0.6, 0.4 et 0.1 palettes. Le produit A (1.9 palettes) est réparti sur deux palettes : une palette homogène P1 et une palette hétérogène P2 d'où l'on a retiré 10% de produit A. Les autres produits B, C, D et E sont regroupés sur deux palettes hétérogènes P3 et P4. La commande globale est donc composée de quatre palettes. P1 est la seule palette qui ne nécessite pas de préparation de commande.

Considérons que le produit A est fabriqué sur un site fournisseur noté F1 et les produits B, C, D et E sur un autre site, noté F2. Seul le site fournisseur F2 est habilité à préparer les commandes. La palette

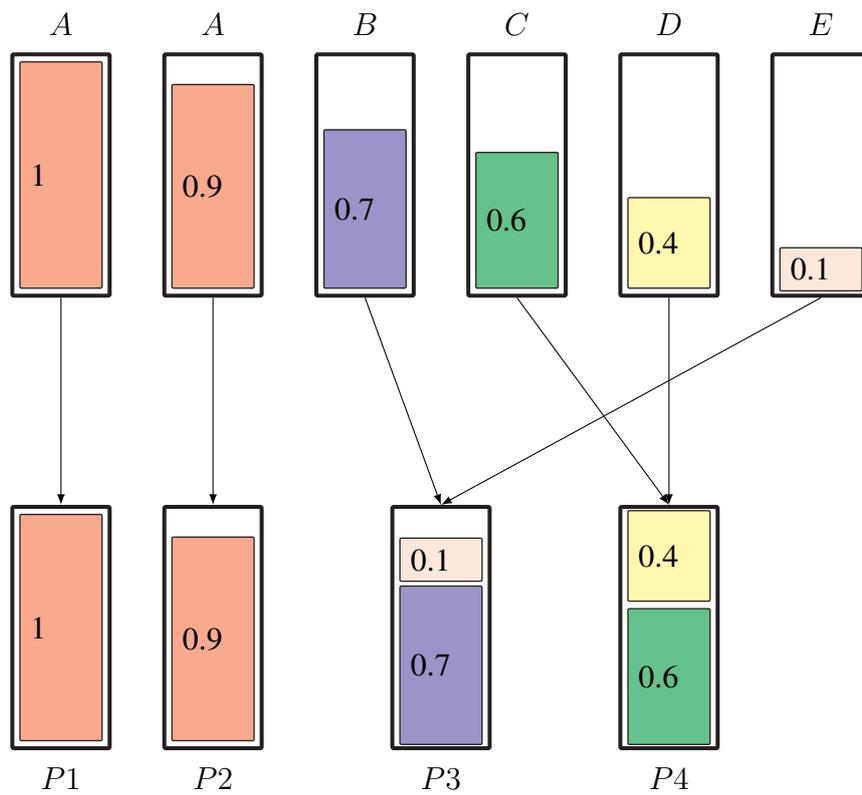


Figure 2.2 – Processus de regroupement de produits d'une palette hétérogène

$P1$ peut être directement envoyée sur un CRC, mais $P2$ doit passer par le site $F2$ pour être préparée. Des itinéraires possibles pour les produits sont représentés sur la figure 2.3.

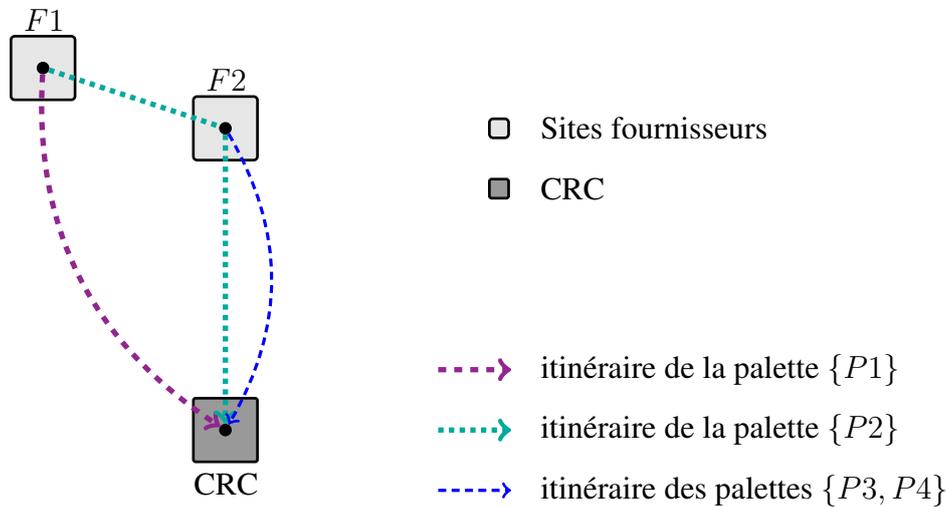


Figure 2.3 – Itin raires des sous-commandes

2.4.3 Sous-commandes

 tant donn es les diff rentes origines possibles des produits d'une m me commande et les contraintes de pr paration des palettes h t rog nes, on d compose une commande en plusieurs sous-commandes. Une sous-commande est d finie comme un ensemble de palettes ayant une seule origine, et ayant besoin – ou non – de pr paration.

Dans l'exemple 1, la commande est alors compos e de trois sous-commandes diff rentes :

- la palette $P1$, ayant pour origine $F1$ et n'ayant pas besoin de pr paration de commande ;
- la palette $P2$ ayant pour origine $F1$ et ayant besoin de pr paration de commande ;
- les palettes $P3$ et $P4$ ayant pour origine $F2$ et ayant besoin de pr paration de commande.

Dans cette th se on consid re que le d coupage en sous-commandes est une donn e d'entr e et non une d cision. Les sous-commandes d'une commande doivent  tre regroup es avant d' tre livr es au client. Elles peuvent se regrouper sur un site de leur fournisseur, ou sur un CRC.

2.5 Transporteurs et tourn es de v hicules

Les transporteurs sont les acteurs qui transportent les marchandises des sites des fournisseurs jusqu'aux clients. On consid re dans cette th se que le transport est sous-trait . 4S Network ne poss de pas de flotte ni de plateforme logistique et confie le transport et les op rations de cross-docking   un ensemble de prestataires logistiques sp cialis . Le r seau logistique fonctionnant gr ce   plusieurs prestataires,

plusieurs structures tarifaires et contraintes métier peuvent y cohabiter. Dans cette section, nous décrivons les différents types de tournées et les structures tarifaires usuellement rencontrées.

On appelle *tournée* le trajet emprunté par un véhicule entre plusieurs sites du réseau logistique mutualisé. Le transport étant sous-traité, on considère des *tournées ouvertes*, c'est-à-dire qui ne prennent pas en compte le voyage des véhicules de leur dépôt jusqu'au premier site visité, ni le retour au dépôt après visite au dernier client. De plus, le nombre de sites visités par tournée est souvent limité à trois. Cette contrainte est courante dans le domaine du transport de la grande distribution française : cela permet de limiter les risques de retard en cascade. On appelle *segment* le trajet direct d'un véhicule entre deux sites. Une tournée est donc composée d'un ou plusieurs segments.

2.5.1 Types de tournées et structures tarifaires dans le réseau amont

Dans le *réseau amont*, on considère une structure tarifaire dite *less-than-truckload* ([Less-than-truckload \(LTL\)](#)). Cette tarification est généralement utilisée en transport routier de marchandises pour facturer le transport d'un petit nombre de palettes dans un véhicule. Pour une région d'origine donnée, elle est représentée par une matrice qui fixe un prix pour chaque destination et chaque valeur possible de chargement. Le tarif [Less-than-truckload \(LTL\)](#) est dégressif : le prix unitaire à la palette diminue (de façon non linéaire) lorsque le nombre de palettes augmente.

Un coût fixe de transfert est appliqué à chaque palette transférée d'un véhicule à un autre. Enfin, un dernier coût appelé *coût d'ouverture de porte*, est facturé pour chaque arrêt d'un véhicule entre son point de départ et son point d'arrivée.

2.5.2 Types de tournées et structures tarifaires dans le réseau aval

Dans le *réseau aval*, on considère une structure tarifaire dite *full truckload* [Full-truckload \(FTL\)](#). En transport routier de marchandises, ce tarif est utilisé pour facturer l'utilisation d'un camion complet sur un itinéraire prédéterminé. Il ne dépend pas du nombre de palettes transportées.

Une tournée dans le *réseau aval* part d'un CRC et visite un ou plusieurs clients. Le nombre maximum de clients visités par une tournée varie entre 3 à 5.

Exemple 2. La figure 2.4 présente un exemple de transport dans le réseau aval avec 2 CRC et 11 clients. Trois tournées partent de chaque CRC. Deux CRC peuvent livrer un même client si celui-ci est assez proche des deux CRC.

Le tarif FTL est calculé comme suit : la zone de chalandise de chaque CRC est divisée en plusieurs zones associées à un coût. Le tarif FTL correspond à celui de la zone traversée la plus chère. De plus, un coût d'ouverture de porte est ajouté à chaque arrêt du véhicule chez un client.

Exemple 3. La figure 2.5 donne un exemple de la répartition des clients dans les zones. Une tournée aval visite 3 clients situés dans les zones 2, 3 et 4.

La figure 2.6 montre le coût associé à chaque zone. La zone 4 étant la plus chère, c'est cette dernière qui est retenue pour obtenir le coût de location du véhicule. Il faut ensuite y ajouter trois fois le coût unitaire d'ouverture de porte pour obtenir le prix total de la tournée.

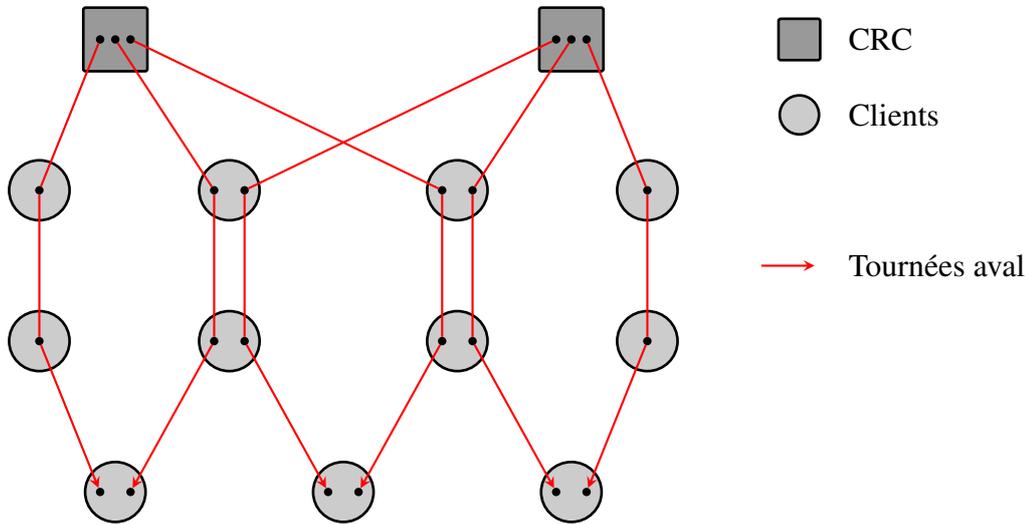


Figure 2.4 – Un exemple de réseau aval

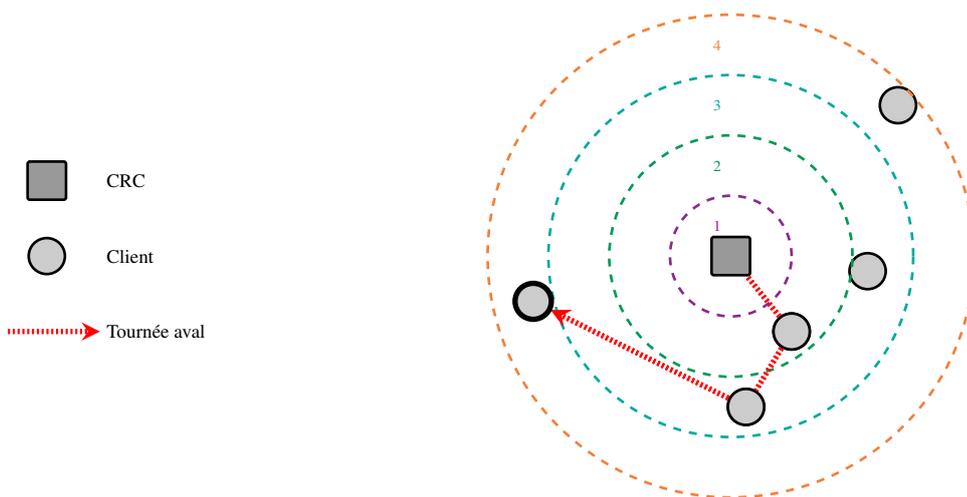


Figure 2.5 – Répartition des clients en zones tarifaires

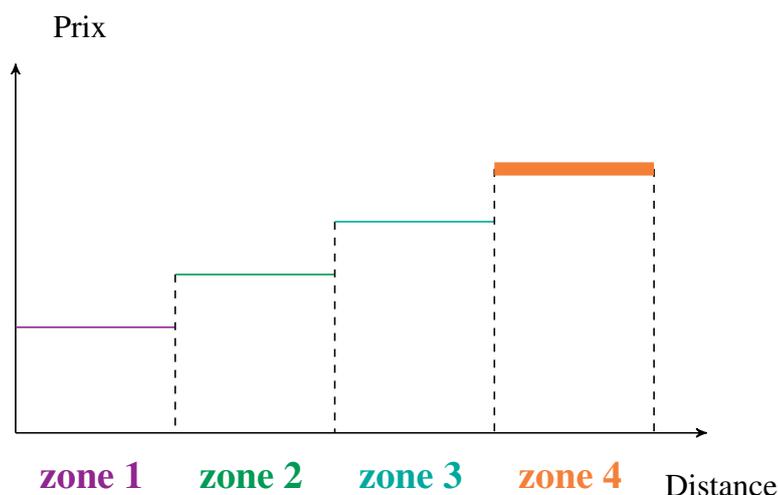


Figure 2.6 – Coût fixe de la tournée selon les zones

2.6 Itinéraires des commandes et sous-commandes

On utilise le terme *itinéraire* uniquement pour désigner le trajet des commandes et des sous-commandes, en opposition avec le terme *tournée* utilisé uniquement pour le trajet des véhicules.

2.6.1 Itinéraire des sous-commandes dans le réseau amont et le réseau direct

Le réseau amont et le réseau direct partagent l'ensemble des sites fournisseurs. Dans le réseau amont, les sous-commandes sont transportées de leur origine jusqu'à un CRC. Dans le réseau direct elles sont expédiées ensemble au client depuis un site fournisseur. On appelle *itinéraire direct* le fait de ne pas passer par un CRC, et *itinéraire CRC* le passage par CRC avant distribution de la commande au client. Un itinéraire peut-être composé d'un ou plusieurs segments de différentes tournées. L'itinéraire d'une sous-commande peut donc comporter un ou plusieurs transferts entre véhicules dans le réseau amont. Une contrainte forte est que l'ensemble des sous-commandes d'une commande doivent être regroupées, au plus tard sur le CRC, en cas d'itinéraire CRC. Dans le cas d'un itinéraire direct ce regroupement aura lieu sur un site fournisseur si besoin. Cette caractéristique est qualifiée de *merge-in-transit* dans la littérature [22].

2.6.2 Itinéraire des commandes dans le réseau aval

En raison de la contrainte de *merge-in-transit*, toutes les sous-commandes d'une commande ont été regroupée sur un CRC lorsqu'elle arrive dans le réseau aval. L'itinéraire d'une commande dans le réseau aval suit donc la tournée d'un un seul véhicule.

2.7 Un exemple de fonctionnement global du réseau

Exemple 4. La figure 2.7 illustre un exemple avec quatre sites de fournisseurs, un CRC et deux clients.

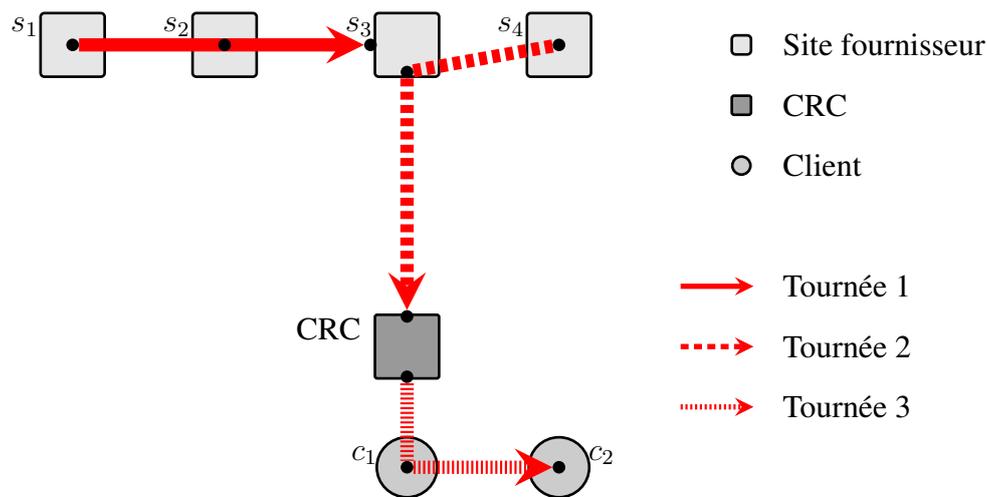


Figure 2.7 – Tournées des véhicules

- La tournée amont 1 débute son trajet au site s_1 , visite ensuite s_2 et termine en s_3 . Cette tournée est donc composée de deux segments $\{s_1; s_2\}$ et $\{s_2; s_3\}$.
- La tournée amont 2 est également composée de deux segments $\{s_4; s_3\}$ et $\{s_3; \text{CRC}\}$.
- La tournée aval 3 comporte deux segments : $\{\text{CRC}; c_1\}$ et $\{c_1; c_2\}$.

La figure 2.8 montre l'itinéraire d'une commande composée d'une sous-commande ayant pour origine s_2 et destination c_1 . Elle voyage dans trois véhicules différents et est transférée deux fois, en s_3 et au CRC.

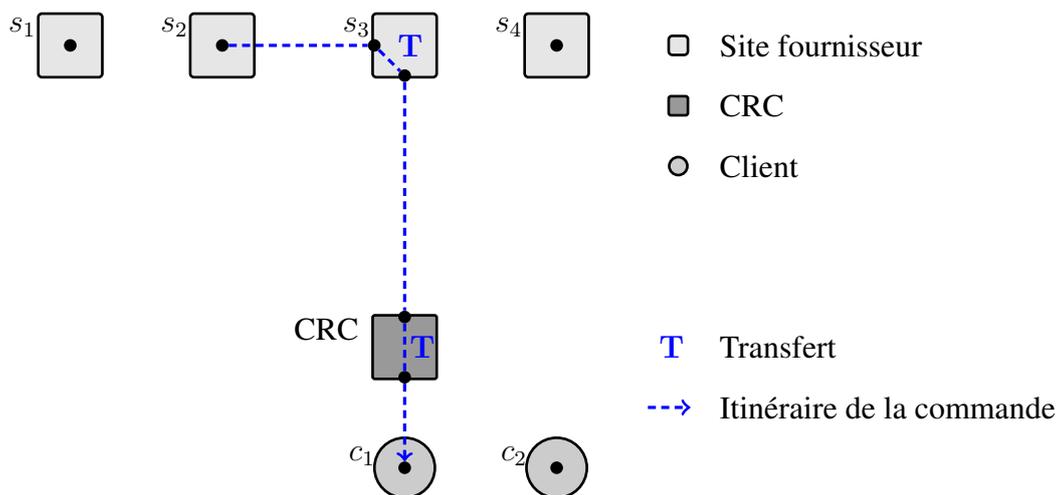


Figure 2.8 – Itinéraire de la commande

Cadre scientifique de la thèse

Sommaire

3.1 Facturation du transport	28
3.2 Fonction linéaire par morceaux	28
3.2.1 Définition	28
3.2.2 Modélisation pour un véhicule	29
3.2.3 Modélisation pour plusieurs véhicules	29
3.3 Modélisation des transports dans un réseau espace temps	31
3.4 Génération de colonnes	32
3.5 La recherche tabou	35

L'objectif de ce chapitre est d'introduire les notions et le cadre scientifique nécessaire à la bonne compréhension de cette thèse. Ce chapitre aborde les cinq thématiques suivantes :

- les différents types de tarif pour un transport sous-traité (section 3.1) ;
- la modélisation d'un tarif **Less-than-truckload (LTL)** par une fonction concave linéaire par morceaux (section 3.2) ;
- la modélisation de la chaîne logistique par un réseau espace temps (section 3.3) ;
- la génération de colonnes (section 3.4).
- la recherche tabou (section 3.5)

3.1 Facturation du transport

4S Network se positionne comme un 5PL. L'entreprise gère l'affrètement des véhicules et confie le transport à des prestataires nationaux (du fournisseur au CRC) ou régionaux (du CRC au client). Ces transporteurs facturent leurs prestations selon des règles tarifaires qui ne correspondent pas à leurs coûts internes et à l'utilisation effective des véhicules. Cette section présente, non pas le mode de calcul du coût interne au transporteur (qui est inconnu), mais le tarif des prestations tel qu'il est facturé. Les deux tarifs les plus fréquemment rencontrés, et repris dans cette thèse, sont le tarif **Full-truckload (FTL)** et le tarif **Less-than-truckload (LTL)**.

- En tarif *full truckload* (**Full-truckload (FTL)**), ou chargement complet, l'ensemble du véhicule est sous-traité. La totalité de la capacité du véhicule est louée pour un trajet donné. Le prix ne dépend que de la distance parcourue par le véhicule. Cette option est généralement choisie par un fournisseur ayant un volume important de palettes à transporter.
- En tarif *less-than-truckload* (**Less-than-truckload (LTL)**), ou chargement partiel, un véhicule peut transporter des marchandises provenant de plusieurs fournisseurs. Le prix dépend de la distance et du volume confié au transporteur.

Dans les deux modes de sous-traitance, *un coût d'ouverture de porte*, correspondant à un coût fixe indépendant de la charge, est facturé pour tout arrêt occasionnant un déchargement. Ce coût est inhérent à la durée additionnelle de réservation du véhicule générée par l'arrêt.

Il est à noter que le mode de tarification est souvent lié à un contrat ou partenariat avec un transporteur sur le moyen ou long terme. Le choix du type de tarif est donc plus souvent une décision tactique qu'opérationnelle.

Exemple 5. *On considère deux commandes non mélangeables, comprenant 27 et 9 palettes au départ du même site fournisseur et à destination du même client. Ces commandes sont transportées par 2 véhicules de capacité égale à 33 palettes.*

*Un transporteur A propose pour le trajet un prix de 100 euros, tarif **FTL**. Un transporteur B propose un tarif **LTL** égal à 6 euros par palette pour un volume confié de 9 palettes, et 5 euros par palette pour un volume confié de 27 palettes.*

*Le transport de la première commande (27 palettes) coûte donc 100 euros avec en **FTL** et $27 \times 5 = 135$ euros en **LTL**. On choisira donc le mode **FTL**. Le transport de la deuxième commande (9 palettes) coûte 100 euros en **FTL** et $6 \times 9 = 54$ euros en **LTL**. Le tarif **LTL** est donc plus avantageux pour cette commande.*

3.2 Fonction linéaire par morceaux

3.2.1 Définition

Les transporteurs utilisant un tarif **LTL** fournissent généralement des *matrices de coûts* 2D dont la 1ère dimension est le trajet réalisé par le véhicule (entre deux points) et la deuxième est la charge à transporter. Cette matrice indique un tarif à la palette selon ces deux dimensions. Ce tarif à la palette est très élevé pour des petits lots et diminue lorsque la charge augmente. L'approche la plus répandue dans

la littérature (voir chapitre 4) est d'approximer ce tarif par une fonction concave, linéaire par morceaux. Nous adoptons cette approche.

La figure 3.1 illustre le cas d'une telle fonction, constituée de 3 segments et passant par l'origine. Q est la capacité totale du véhicule, et q_1 et q_2 représentent les limites de charge des segments de la fonction linéaire par morceaux.

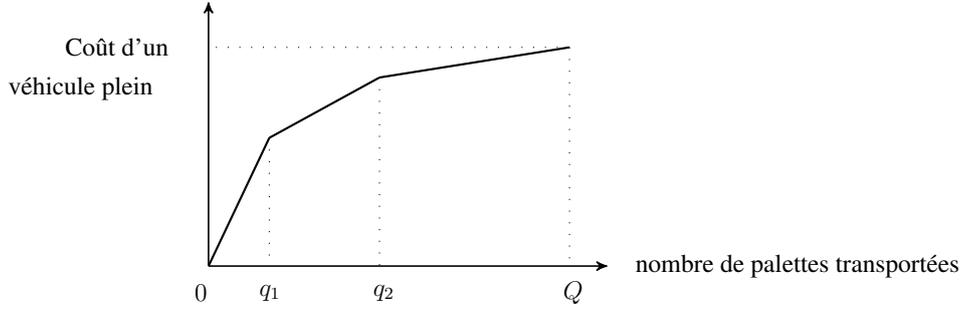


Figure 3.1 – Exemple de tarif LTL

3.2.2 Modélisation pour un véhicule

La fonction linéaire par morceaux est calculée grâce à la fonction $h(x)$ suivante, où $\alpha_1, \alpha_2, \alpha_3$ sont les coefficients directeurs respectifs des trois segments (avec $\alpha_1 > \alpha_2 > \alpha_3 > 0$) et f_1, f_2, f_3 sont les ordonnées à l'origine des trois segments.

$$h(x) = \begin{cases} \alpha_1 x + f_1 & \text{si } 0 \leq x \leq q_1, \\ \alpha_2 x + f_2 & \text{si } 0 \leq x \leq q_2, \\ \alpha_3 x + f_3 & \text{si } 0 \leq x \leq Q. \end{cases} \quad (3.1)$$

3.2.3 Modélisation pour plusieurs véhicules

La fonction linéaire par morceaux utilisée dans cette thèse représente, non pas le tarif pour un seul véhicule, mais plusieurs plusieurs véhicules identiques effectuant le même trajet. Le tarif d'un tel trajet est donc construit en reproduisant la fonction d'un véhicule autant de fois que nécessaire, comme illustré par la figure 3.2.

Les coefficients directeurs $\alpha_1, \alpha_2, \alpha_3$ restent les mêmes utilisés pour chaque véhicule ajouté à la fonction. On appelle S l'ensemble des segments de cette fonction. Les ordonnées à l'origine sont calculées en respectant la continuité de la fonction $f_s, s \in S$:

$$f_s = (\alpha_{s-1} q_{s-1} + f_{s-1}) - q_{s-1} \alpha_s \quad (3.2)$$

On utilise donc pour chaque véhicule la fonction de type :

$$h(x) = \begin{cases} \alpha_1 x + f_s & \text{si } q_{s-1} \leq x \leq q_s, \\ \alpha_2 x + f_{s+1} & \text{si } q_s \leq x \leq q_{s+1}, \\ \alpha_3 x + f_{s+2} & \text{si } q_{s+1} \leq x \leq q_{s+2}. \end{cases} \quad (3.3)$$

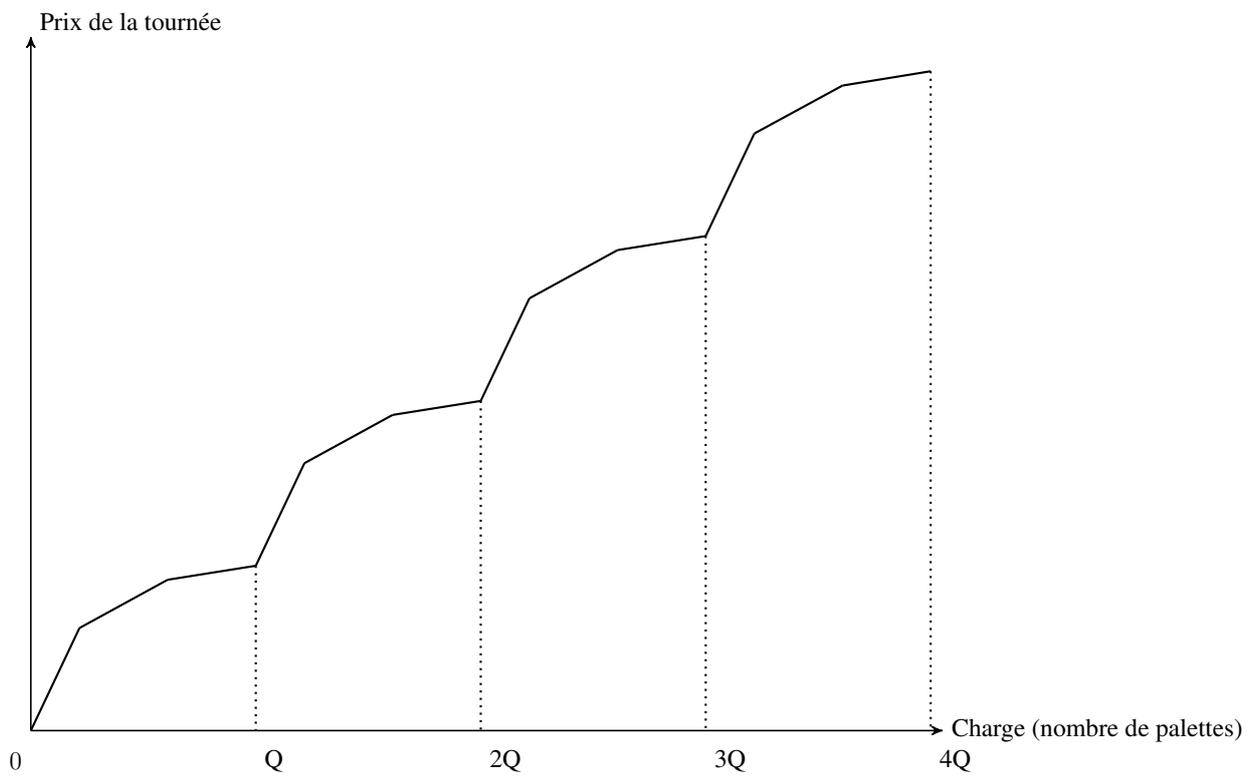


Figure 3.2 – Tarif LTL pour une flotte homogène

Croxton et al [32] proposent trois types de modélisations de la fonction linéaire par morceaux et montrent que leur relaxation linéaire (qui est une mesure importante de la qualité d'une formulation en PLNE) est équivalente.

Nous choisissons la modélisation appelée *multiple choice model* (MCM), où chaque segment $s \in S$ est caractérisé par un intervalle de charge $[q_{s-1}, q_s]$, un coefficient directeur a_s et une ordonnée à l'origine b_s . Soit q une quantité de chargement dont on souhaite évaluer coût de transport grâce à la fonction linéaire par morceaux. On définit les variables suivantes pour chaque segment $s \in S$:

- $\xi_s = 1$ si $q_{s-1} \leq q \leq q_s$, et 0 sinon,
- $\lambda_s = q$ si $q_{s-1} \leq q \leq q_s$, et 0 sinon.

Comme q appartient à un intervalle de charge et un seul, une seule variable par type de variables ξ_s et λ_s est non nulle. Le prix peut être calculé par la somme suivante :

$$\sum_{s \in S} (a_s \lambda_s + b_s \xi_s). \quad (3.4)$$

En outre, pour modéliser la fonction linéaire par morceaux dans le modèle MCM, les variables ξ_s et λ_s doivent satisfaire les contraintes (3.5) et (3.6).

$$\sum_{s \in S} \xi_s \leq 1, \quad (3.5)$$

$$q_{s-1} \xi_s \leq \lambda_s \leq q_s \xi_s \quad \forall s \in S. \quad (3.6)$$

Les contraintes 3.5 garantissent qu'un seul segment est sélectionné et (3.6) permet la sélection de ce segment.

3.3 Modélisation des transports dans un réseau espace temps

Les réseaux de transport sont naturellement modélisés par des graphes où chaque sommet représente un lieu physique. Néanmoins, il est parfois nécessaire de pouvoir représenter également la dimension temporelle du système étudié. Cela est particulièrement vrai dans les réseaux de transport si on veut représenter l'heure de passage d'un véhicule en un point, les temps d'attente, la synchronisation entre deux véhicules, etc. Une approche classique est d'ajouter une dimension temporelle au graphe *physique* représentant le réseau. On appelle réseau espace-temps un graphe dans lequel un sommet modélise à la fois un lieu physique et une période de temps. Cet outil est abondamment utilisé dans le domaine de la conception de lignes de transport public (bus, trains, transports scolaires), sous le nom de graphique en ligne, ou graphique de circulation.

Pour une manipulation plus facile des réseaux utilisés dans cette thèse, on définit tout d'abord la notion de *réseau plat*

Définition 1. On appelle *réseau plat* un graphe $D = (N, A)$ composé d'un ensemble de sommets N et d'arcs A , où les sommets représentent les lieux réels du réseau.

Pour construire un réseau espace temps à partir d'un réseau plat, il est nécessaire de partitionner, ou discrétiser, un horizon fini T de planification en un certain nombre de périodes de temps. Soit $i \in N$ un sommet du réseau plat. On note T_i l'ensemble des périodes temporelles pendant lesquelles i peut être visité. Un sommet du réseau espace temps est noté (i, t) , avec $t \in T_i$.

Définition 2. On appelle *réseau espace-temps*, dérivé du réseau plat $D = (N, A)$, le réseau $D_{\mathcal{T}} = (N_{\mathcal{T}}, H_{\mathcal{T}} \cup A_{\mathcal{T}})$ où l'ensemble des sommets $N_{\mathcal{T}}$ contient un sommet (i, t) pour chaque $i \in N$ et chaque $t \in T_i$.

L'ensemble $A_{\mathcal{T}}$ contient les arcs de la forme $((i, t)(j, \bar{t}))$, où $(i, j) \in A$, $t \in T_i$ et $\bar{t} \in T_j$. Cet ensemble représente le transport sur l'arc $(i, j) \in A$ du réseau plat entre les périodes $t \in T_i$ et $\bar{t} \in T_j$.

L'ensemble $H_{\mathcal{T}}$ contient les arcs de la forme $((i, t), (i, \bar{t}))$, $t, \bar{t} \in T_i$ pour chaque $i \in N$. On les appelle *arcs d'attente* car ils ne modélisent pas un changement de sommet physique, mais une attente sur un sommet donné.

Exemple 6. La figure 3.3 illustre un exemple de réseau plat $D = (N, A)$ comportant 4 sommets et 6 arêtes.

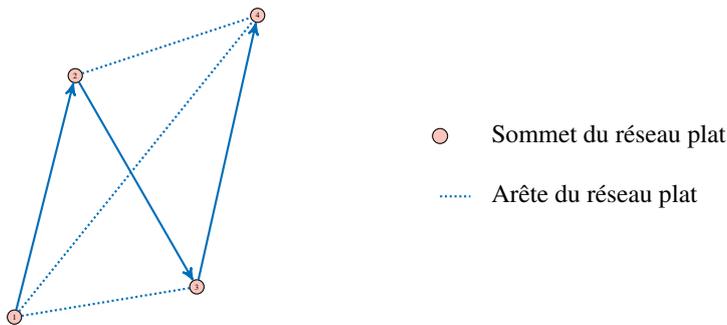


Figure 3.3 – Exemple d'un réseau plat

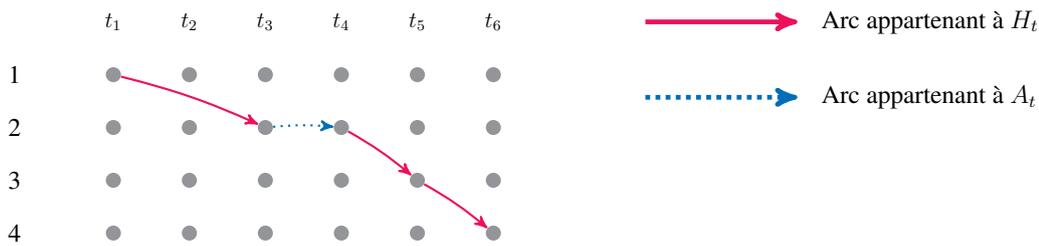
L'horizon temporel est égal à 6h, et le temps est discrétisé par périodes de 1h. 6 périodes temporelles sont donc définies.

La figure 3.4 représente le réseau espace-temps $D_{\mathcal{T}} = (N_{\mathcal{T}}, H_{\mathcal{T}} \cup A_{\mathcal{T}})$ créé à partir de ce réseau plat. Un exemple d'arc appartenant à $H_{\mathcal{T}}$ est représenté en pointillé sur la figure. D'autres exemples d'arcs appartenant à $A_{\mathcal{T}}$ sont représentés en traits pleins.

Cette figure représente un itinéraire dans le réseau espace temps, partant du sommet 1 à la période 1, arrivant au sommet 2 à la période 3, partant du sommet 2 à la période 4 pour arriver au sommet 3 à la période 5 et au sommet 4 à la période 6.

3.4 Génération de colonnes

Nous présentons dans cette section le principe de la génération de colonnes, utilisée dans les chapitres 6 et 7. Le lecteur intéressé par plus de détails concernant cette technique et ses applications aux problèmes de transport peut se référer aux articles de Lübbecke et Desrosiers [69], Barnhart et al. [7], Desaulniers et al. [35] et Feillet [40]. La méthode de génération de colonnes permet de résoudre des programmes linéaires (appelé ici PLM) de grande taille en ajoutant itérativement les variables au modèle.



Les variables du PLM sont appelées *colonnes*. La technique de génération de colonnes est utilisée lorsque l'ensemble des colonnes du PLM est trop grand pour pouvoir résoudre le modèle complet en un temps acceptable. L'idée est qu'une solution optimale d'un PLNE peut souvent être décrite en n'utilisant qu'un sous ensemble très restreint de colonnes. La technique de génération de colonnes est un algorithme itératif qui est initialisé avec un sous-ensemble limité de colonnes, et qui ajoute à chaque itération des colonnes générées dynamiquement par un *sous-problème*. Le modèle mathématique intégrant les colonnes initiales et les colonnes générées est appelé Problème Maître Restreint (PLMR). La génération de colonnes s'arrête lorsque le sous-ensemble de colonnes générées est suffisant pour trouver une solution optimale du PLM. La figure 3.5, issue de la thèse de Simon Spoorendonk [88], montre les relations entre le PLM et le PLMR. Les deux premiers rectangles symbolisent respectivement l'ensemble des colonnes du PLM et du PLMR. Le PLM contient toutes les colonnes alors que le PLMR n'en contient qu'un sous-ensemble. Les deux derniers rectangles symbolisent l'ensemble des colonnes du PLMR à des phases différentes de la génération de colonnes, afin d'illustrer l'ajout dynamique de colonnes dans le PLMR.

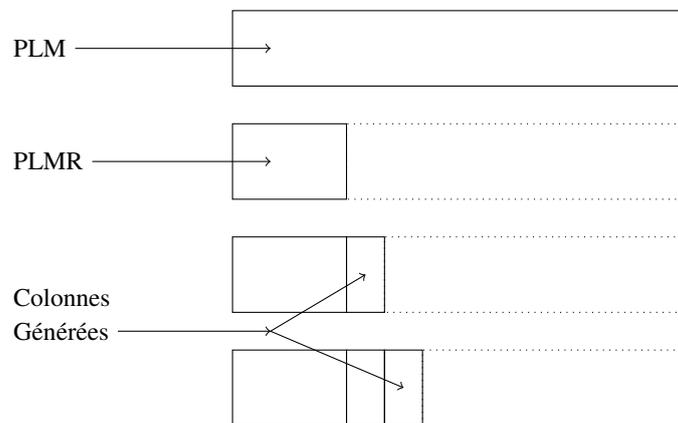


Figure 3.5 – Principe de la génération de colonnes

Les colonnes sont générées par un sous-problème appelé *problème de pricing*. Il a pour but de générer une ou plusieurs colonnes qui améliorent autant que possible la fonction objectif du PLMR, et qui sont ajoutées à l'ensemble de colonnes du PLMR.

Considérons le PLMR constitué de l'ensemble \mathcal{P} de colonnes $x_j \in \mathcal{P}$.

$$\text{Min} \sum_{j \in \mathcal{P}} c_j x_j \tag{3.7}$$

$$s.c. \sum_{j \in \mathcal{P}} a_{ij} x_j \geq n_i \quad \forall i = 1, \dots, m \quad (3.8)$$

$$x_j \geq 0, j \in \mathcal{P}. \quad (3.9)$$

Soient $\pi_i \geq 0, i = 1, \dots, m$ les variables duales associées aux contraintes de ce PLMR. Le problème dual est le suivant :

$$\max \sum_{i=1}^m n_i \pi_i \quad (3.10)$$

$$s.c. \sum_{i=1}^m a_{ij} \pi_i \leq c_j \quad \forall j \in \mathcal{P} \quad (3.11)$$

On note x^* et π^* les solutions optimales respectives du primal et dual du PLMR. On note que ces solutions ne sont pas forcément optimales pour le PLM.

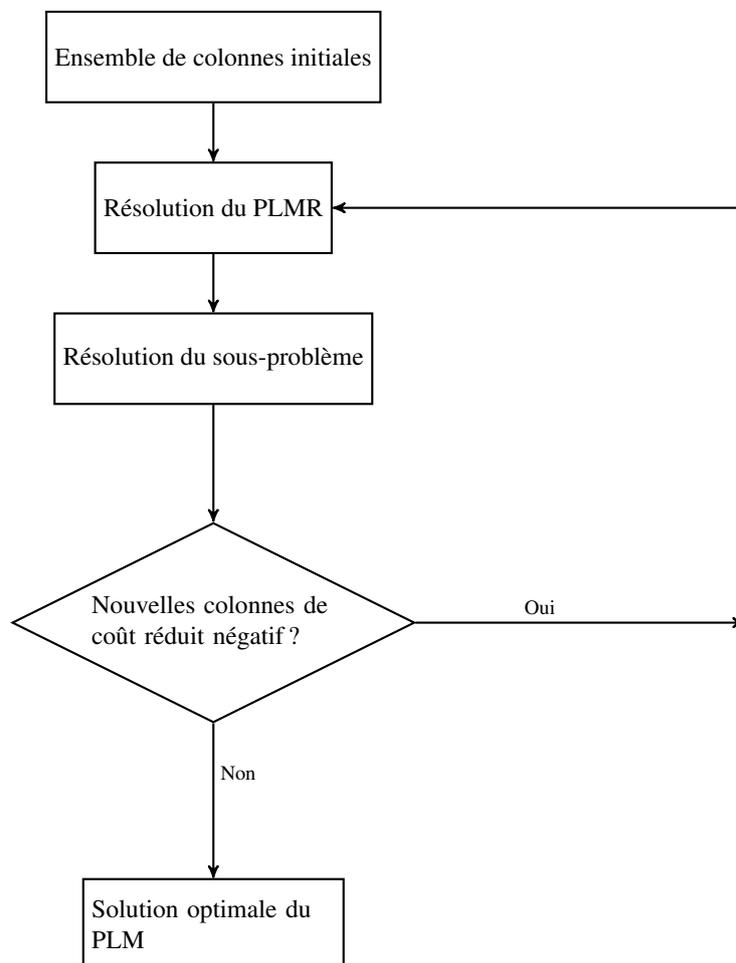


Figure 3.6 – Schéma général de la génération de colonnes

La figure 3.6 représente le diagramme de flux de la technique de génération de colonnes. Au début de chaque itération de la génération de colonnes, le PLMR est résolu. Les valeurs des colonnes π^* extraites

de cette résolution sont utilisées dans le sous-problème qui génère de nouvelles colonnes. Le *coût réduit* d'une colonne mesure le gain marginal qu'apporterait l'introduction de cette colonne dans le PLMR. Le coût réduit d'une colonne j est calculé par la formule suivante :

$$r_j = c_j - \sum_{i=1}^m a_{ij} \pi_i^* \quad (3.12)$$

Une colonne de coût réduit négatif permettra de diminuer la valeur de la fonction objectif lors de sa prochaine résolution. Le sous problème consiste donc à trouver une ou plusieurs colonnes de coût réduit négatif. Le problème d'optimisation correspondant consiste à minimiser le coût réduit.

La résolution du PLMR et du sous-problème sont répétés jusqu'à ce que le sous-problème ne trouve plus de colonnes au coût réduit négatif. Si le sous-problème est résolu de façon optimale, cela signifie que la résolution du x^* du PLMR est optimale pour le PLM. Notons que la génération de colonnes nécessite au préalable un ensemble de colonnes formant une solution réalisable pour le PLMR et permettant une première résolution, afin de récupérer les valeurs des variables duales.

Cette technique est généralement utilisée pour résoudre un [Programme Linéaire en Nombres Entiers \(PLNE\)](#) par branch-and-bound dans lequel la borne inférieure est calculée par génération de colonnes. Cette méthode est appelée *branch-and-price*.

Dans les problèmes d'optimisation des transports où les colonnes représentent des trajets dans le réseau, minimiser le coût réduit revient à trouver un plus court chemin. Selon les caractéristiques du problème, le sous-problème comprend parfois certaines contraintes additionnelles, dites contraintes de ressources, sur le plus court chemin (horaires de visite des sommets, durée des tournées, chargement du véhicule, etc.). C'est le cas dans les chapitres 6 et 7, qui décrivent des méthodes de résolution basées sur la génération de colonnes. Un tutoriel sur la génération de colonnes pour les problèmes de tournées de véhicules est proposé par Feillet [40].

Les méthodes exactes permettant de résoudre de manière optimale un problème de plus court chemin avec ou sans contraintes de ressources sont des algorithmes de programmation dynamique comme par exemple l'algorithme de Bellman-Ford [9] [42], qui a donné lieu à de nombreuses extensions pour l'algorithme de plus court chemin avec contraintes de ressources ([41]).

On classifie en deux catégories les méthodes heuristiques résolvant un sous-problème d'une génération de colonnes : les méthodes tirées d'algorithmes de résolution exacte rendus heuristiques, et les méthodes issues des algorithmes de recherche locale ou métaheuristique. Parmi les métaheuristiques utilisées pour résoudre le sous-problème d'une génération de colonnes, on utilise dans cette thèse une méthode de recherche tabou, méthode fréquemment utilisée dans ce cas (voir Desaulniers et al. [36]). La section suivante décrit le fonctionnement de cette méthode.

3.5 La recherche tabou

La recherche tabou [44][50] est une métaheuristique provenant d'une extension de la recherche locale. Dans une recherche locale, la solution courante est remplacée à chaque itération par une meilleure solution dans son voisinage. La recherche se termine généralement en un optimum local, lorsque la solution ne peut plus être améliorée. Le désavantage de ce type de méthode est le risque qu'elle aboutisse à un optimum local qui ne corresponde pas à l'optimum global. Une amélioration possible est de sélectionner non pas une meilleure solution que la solution courante, mais la meilleure solution du voisinage, tout en

conservant la meilleure solution trouvée en mémoire. Ainsi il devient possible de détériorer la solution courante afin de trouver un autre optimum local. Cependant, cette technique risque de faire cycliser l'algorithme autour d'un minimum local. C'est ici qu'intervient le principe de la recherche tabou : une *liste tabou* garde en mémoire les transformations de la solution récemment effectuées et interdit de réaliser les transformations inverses pendant un certain nombre d'itérations.

État de l'art : problèmes de conception de réseau logistique et tournées riches de véhicules

Ce chapitre d'état de l'art a pour but d'identifier les principales publications scientifiques, modélisations et méthodes de résolutions concernant les problèmes d'optimisation dans les chaînes logistique possédant des caractéristique communes avec notre domaine de recherche, à savoir :

- un réseau de distribution à plusieurs échelons ;
- multiplicité des tarifs, notamment tarification [LTL](#) et [FTL](#) ;
- opérations logistiques complexes : consolidation, transferts et synchronisation entre le réseau amont et le réseau aval.
- contraintes opérationnelles : merge-in-transit, contraintes riches sur les tournées de véhicules, etc.

Crainic et al. [31] classifient les problèmes de transport de marchandises en trois catégories. Les problèmes stratégiques, liés à des décisions à long terme, comme l'implantation d'une nouvelle plateforme dans un réseau. Les problèmes tactiques, qui représentent des décisions à moyen terme comme l'ouverture de lignes de transport dans un réseau de plateformes existantes. Les problèmes opérationnels qui traitent des décisions à court terme comme l'optimisation des tournées de véhicules ou du planning des chauffeurs au jour le jour.

Dans cette thèse, nous abordons principalement des problématiques tactiques ou opérationnelles, à savoir la conception de lignes/d'un réseau de transport (Crainic et Laporte [31]), l'acheminement des commandes dans un réseau et l'optimisation des tournées de véhicules (Toth et Vigo [93]). Cet état de l'art est divisé en quatre parties. La section 4.1 traite des problèmes de conception de réseaux de transport. La section 4.2 étudie les problèmes de planification opérationnelle des transports dans un réseau. Les deux dernières sections abordent deux points particuliers des problèmes de tournées de

véhicules : les problèmes de tournées avec transferts (section 4.3) et les problèmes riches de tournées de véhicules (section 4.4).

4.1 Conception de réseau de transport

Les chaînes logistiques complexes incluent plusieurs types de sites avec des fonctions variées (sites de production, d'assemblage, de stockage longue durée, de préparation de commandes, de cross-docking, centres de distribution, sites clients, etc.). Le réseau logistique prend donc la forme d'un graphe à plusieurs niveaux. La notion de conception de réseau recouvre plusieurs types de décision : localisation et dimensionnement des sites, choix des modes de transport, allocation des produits et clients à chaque site, définition des flux logistiques récurrents, recherche de prestataires.

Concernant les décisions purement stratégiques (localisation, dimensionnement, allocation), on parle de *Supply Chain Network Design*. Nous renvoyons le lecteur à l'état de l'art de Melo et al. [73] ou au chapitre de Alumur et al. [1]. Un état de l'art du problème de localisation et routage d'un réseau à N-échelons est proposé par Gonzalez-Feliu [46]. Dans cette catégorie, Gendron et Semet [43] étudient un problème de localisation et routage à trois échelons pour un problème de livraison rapide. L'objectif est de déterminer les itinéraires des commandes en minimisant les coûts de transport et d'utilisation des plateformes. Les auteurs montrent qu'une modélisation basée sur les chemins est toujours meilleure qu'une modélisation basée sur les arcs.

Nous utilisons la définition proposée par Campbell [15] pour définir les problèmes de conception de réseau avec une tarification *LTL*. Les demandes prises en compte dans ces problèmes peuvent avoir une origine et une destination correspondant à des sites clients mais, le plus souvent, les demandes sont agrégées et se réfèrent à des plateformes d'entrées ou de sorties du réseau (appelées *end-of-line* en anglais). Les sommets du réseau correspondent donc à ces plateformes et les arcs aux trajets les plus courts entre deux sommets. Une caractéristique forte est que les distances entre plateformes généralement élevées : cela encourage la consolidation des commandes de petite taille sur les plateformes et permet de rentabiliser l'usage des véhicules. Le problème consiste à déterminer les capacités de transport à installer sur les arcs de manière à permettre l'acheminement des commandes en minimisant les coûts de transport. Ce coût est constitué d'un coût fixe pour l'utilisation de chaque arc et d'un coût variable lié au passage des commandes dans le réseau. Crainic [31] définit et nomme ce problème *multi commodity network design*. Sa résolution permet de répondre à des questions généralement stratégiques ou tactiques.

Le nombre d'articles traitant de conception de réseau est considérable. C'est pourquoi nous focalisons cette section sur deux caractéristiques étudiées dans cette thèse :

- la modélisation de la tarification *LTL* par une fonction linéaire par morceaux ;
- le *merge-in-transit* qui est une pratique consistant à regrouper les produits d'une commande (ou les produits qui constituent le produit final), au cours de leur itinéraire dans le réseau logistique, sur des plateformes dédiées.

4.1.1 Fonctions de coût linéaires par morceaux

Supposer que le transport est sous-traité nous impose d'utiliser dans les modèles mathématiques les mêmes tarifs que ceux pratiqués par les transporteurs. Ils peuvent être très différents d'un transporteur

à l'autre. Outre les différents tarifs qui dépendent uniquement du nombre de kilomètres, une caractéristique de cette thèse est d'utiliser la tarification *LTL*, qui dépendant de la distance mais également du chargement des véhicules. De nombreux transporteurs proposent des *matrices* qui calculent un tarif selon deux dimensions : le nombre de palettes transportées et le nombre de kilomètres parcourus. Le tarif à la palette diminue avec la charge, de manière non linéaire. Dans cette thèse, comme dans d'autres travaux de la littérature, cette dégressivité est modélisée par une fonction linéaire par morceaux. Chan et al. [18] et Hill et al. [51] utilisent une fonction linéaire par morceaux appelée *modified all-unit discount cost* (MAUD). Avec cette fonction, il est parfois profitable de déclarer un volume de marchandises plus grand pour bénéficier d'un tarif à la palette plus attractif. Lapierre et al. [62] construisent une fonction de coût composite en combinant plusieurs types de fonctions linéaires par morceaux. Muriel et al. [77] étudient un problème de multi-flot dans un réseau où les coûts sont modélisés par une fonction linéaire par morceaux. Ils développent une heuristique basée sur la relaxation linéaire du modèle. Chang [19] résolvent un problème bi-objectif de transport dans un réseau multimodal. Les deux objectifs à minimiser sont le coût de transport, modélisé par une fonction linéaire par morceaux et la durée du transport. La méthode de résolution développée décompose le problème en un ensemble de petits problèmes plus faciles à résoudre.

Croxton et al. [32] comparent trois modèles de coûts linéaire par morceaux. Ils montrent que les relaxations linéaires de ces trois modélisations sont équivalentes et peuvent être approximées par leurs enveloppes convexes inférieures. Un résultat appréciable est que si la fonction linéaire par morceaux est concave, alors son enveloppe convexe inférieure est linéaire, et peut-être représentée par la droite joignant l'origine et le point de la fonction de charge maximale. Moccia et al. [75] étendent cette propriété en montrant que, sous certaines conditions, l'enveloppe convexe inférieure d'une fonction linéaire par morceaux non concave peut-être représentée par cette même droite. Cet article a inspiré les travaux de recherche présentés au chapitre 7.

4.1.2 Merge-in-transit

La contrainte de regroupement est centrale pour les problèmes de transport dans le réseau mutualisé. En effet, les sous-commandes d'une commande n'ont pas toutes la même origine et doivent se regrouper sur une plateforme pour être livrées au client dans le même véhicule. Chopra et al. [22] définissent la notion de *merge-in-transit* comme le regroupement de plusieurs parties d'une commande provenant de différents sites afin de les livrer ensemble au client. Cole et al. [24] proposent un modèle de conception de réseau avec des contraintes de regroupement lié à un système d'information géographique. Ce modèle détermine l'ouverture de centres de regroupement et l'itinéraire des marchandises. Croxton et al. [33] étudient un problème de conception de réseau logistique avec contraintes de regroupement ainsi qu'une structure de coût linéaire par morceaux. Les auteurs proposent un algorithme de plans coupants suivi d'un branch-and-bound. Dans Song et al. [87], la plateforme de cross-docking est utilisée comme un centre de regroupement. Le problème est résolu grâce à une relaxation lagrangienne.

4.2 Planification opérationnelle des transports dans un réseau

Cette section traite d'une famille de problèmes de conception de réseau intégrant des décisions plus opérationnelles que dans la section précédente. Le problème consiste à définir le service (trajet, chargement, horaire) de chaque acteur du réseau mutualisé dans un horizon de quelques jours à quelques semaines.

On parle alors de *Service Network Design* (**Service Network Design Problem (SNDP)**). La modélisation de ce problème repose essentiellement sur le *multicommodity network design* décrit plus haut, mais en y ajoutant de nouvelles considérations d'ordre tactique ou opérationnel. Wieberneit et al. [98] propose une étude des variantes du **SNDP**. Crainic [29] classifient les **SNDP** en deux catégories : le *frequency service network design problem* est utilisé pour répondre à des problématiques tactiques assez proches du niveau opérationnel (par exemple choisir entre plusieurs politiques de transport dans le réseau), le *dynamic service network design problem* répond a des problématiques beaucoup plus opérationnelles, comme déterminer les itinéraires au jour le jour dans un réseau.

Les problèmes de type **Service Network Design Problem (SNDP)** reposent sur des graphes espace-temps décrits au chapitre précédent. L'intégration de composantes opérationnelles dans un problème tactique passe par la gestion de la flotte de véhicules. Une première manière approche pour prendre en considération une flotte de véhicules est d'ajouter des contraintes équilibrant le nombre de véhicules entrant et sortant de chaque plateforme. Ces contraintes d'équilibre sont l'occasion d'introduire un coût de repositionnement des véhicules voyageant à vide. Nous mentionnons quelques articles entrant dans cette catégorie en section 4.2.1.

Une approche encore plus précise pour représenter des décisions censées se répéter au cours du temps (semaines types de travail, lignes de transport régulières) est de recourir à des plans de transports cycliques, c'est-à-dire comportant un arc de retour entre un sommet de la dernière période et un sommet de la première période du graphe espace temps. On définit alors des tournées appelées *cycles*, abordées en section 4.2.2.

4.2.1 Équilibre

Jarrah et al. [54] proposent une modélisation intégrant des contraintes d'équilibre entre le nombre de véhicules entrant et sortant sur chaque sommet du réseau. Ils considèrent une tarification **FTL** et des coûts de manutention de marchandises en cas de transfert d'un véhicule à un autre. Le problème résultant est résolu avec une méthode de *slope scaling*. Erera et al. [38] étudient un problème de plans de chargement avec fenêtres de temps, tarification **FTL** et coûts de transfert à chaque sommet. Ils intègrent une contrainte d'équilibre des flux de véhicules sur chaque sommet. Ce problème est résolu avec une recherche locale basée sur la résolution de PLNE. Ce travail est étendu dans [39] : à partir d'un plan de chargement, le trajet de chaque véhicule et les journées de travail de chaque conducteur sont déterminés. Une approche en trois phases est utilisée : la première phase fait correspondre les expéditions des marchandises avec le départ des véhicules, la deuxième détermine les fenêtres horaires de chaque expédition et la troisième crée les journées de travail des conducteurs.

4.2.2 Cycles

Andersen et al. [4] présentent un modèle de **SNDP** multi-modal reposant sur des cycles dans un graphe espace temps. Andersen et al. [3] proposent des modélisations plus génériques du **SNDP** intégrant des cycles et montrent que les modèles dont les variables représentent des cycles de véhicules surpassent ceux basés sur les arcs. Pedersen et al. [80] proposent un modèle générique ainsi qu'une recherche tabou. Teypaz et al. [92] proposent un algorithme de décomposition où trois sous-problèmes sont résolus séquentiellement : l'itinéraire des commandes, les cycles empruntés par les véhicules, et la planification dans le temps de ces véhicules. Pour ce même type de problème, Andersen et al. [2] proposent un

algorithme de branch-and-price où deux type colonnes sont générées : des tournées de véhicules et des itinéraires marchandises. On y retrouve la co-existence de la vision véhicule (tournées) et de la vision marchandises (itinéraires). Crainic et al. [30] intègrent un troisième type de coût lié à l'utilisation des véhicules. La méthode de résolution est basée sur la génération de colonnes et un algorithme de *slope scaling* qui résout itérativement un problème relaxé.

4.2.3 Coûts

Les problèmes de type **SNDP** considèrent généralement deux types de coûts liés au transport ([92], [2], [10]) : un *coût fixe* lié à l'ouverture d'un service qui représente l'utilisation d'un véhicule pour réaliser un trajet d'un sommet à un autre du réseau et un *coût variable* relatif au flux de marchandises transporté. La méthode exacte développée récemment par Boland et al. [10] prend en compte ces deux types de coûts. Leur algorithme, appelé *Dynamic Discretization Discovery*, permet de trouver des solutions optimales dans un réseau espace-temps discrétisé à la minute, pour des instances comprenant jusqu'à 200 commandes. Au chapitre 8, nous décrirons comment étendre l'utilisation de cet algorithme à un problème combinant *Service Network Design* et tournées de véhicules. Notons enfin les articles qui, tout comme dans cette thèse, utilisent une tarification **LTL** ([54] [38] [75] [66] [67]) ou des coûts de transfert ([54] [38] [75] [66] [67]).

4.2.4 Application des problèmes de type **SNDP**

Les problèmes de types **SNDP** peuvent avoir plusieurs champs d'application, notamment en transport multi-modal. L'article de Moccia et al. [75] considère un **SNDP** pour la conception de plans de transport sur un réseau multimodal en Italie. Le problème est résolu avec une heuristique basée sur la génération de colonnes. Chang [19] étudie un problème de **SNDP** bi-objectif dans un réseau multi-modal. Le premier objectif est de minimiser les coûts de transport, modélisé par des fonctions linéaires par morceaux. Le second objectif est de minimiser la durée du transport. Le problème est résolu par une heuristique basée sur la relaxation lagrangienne.

Un autre champ d'application est le transport express. Lin [66] étudie la planification du transport express pour une entreprise taïwanaise. Le réseau en étoile est constitué de terminaux et de plateformes intermédiaires. Une première modélisation, basée sur la modélisation de **SNDP**, est proposée ainsi que deux algorithmes : une relaxation lagrangienne et une énumération implicite. Cette dernière donne de meilleurs résultats sur des instances réelles allant jusqu'à dix terminaux et trois plateformes intermédiaires. Lin et al. [67] proposent une nouvelle modélisation de ce problème intégrant une contrainte de conservation des flux des véhicules à chaque sommet. Ils proposent également un algorithme d'énumération implicite pour un problème similaire à [66]. L'algorithme d'énumération est testé sur des données provenant d'une compagnie de transport aérien express qui possède 13 avions. Kim et al. [57] étudient un problème de **SNDP** dans un réseau multi-modal pour l'expédition express de marchandises. Leur modèle comprend une contrainte d'équilibre des flux de véhicules. Leur heuristique basée sur la génération de colonnes et de plans coupants permet de trouver de bonnes solutions à des problèmes de grande taille. Armacost et al. [5] modélisation un **SNDP** pour l'expédition express de marchandises avec nouveau type de variables qu'ils nomment *variables composites*. Les auteurs montrent que la relaxation linéaire de leur modèle est meilleure que celle du modèle classique.

Référence	Tournées		Coûts				Méthode de résolution
	Équilibre ressources	Cycles	Fixes	Variables	LM	Transferts	
Andersen et al. [4]		✓	✓	✓		✓	solveur
Andersen et al. [3]		✓	✓	✓		✓	solveur
Pedersen et al. [80]		✓	✓	✓			solveur & TS
Teypez et al. [92]		✓	✓	✓			3 phases
Andersen et al. [4]		✓	✓	✓			B&P
Crainic et al. [30]		✓	✓	✓			CG&SS
Boland et al. [10]			✓	✓			DDD
Jarrah et al. [54]	✓		✓	✓		✓	SS
Erera et al. [38]	✓		✓	✓			SS
Moccia et al. [75]					✓	✓	H-CG
Chang [19]					✓		H-LR
Lin [66]				✓		✓	LR & IE- ϵ
Lin et al. [67]	✓			✓		✓	LR & IE- ϵ
Armacost et al. [57]	✓		✓				solveur
Kim et al. [5]	✓		✓				H-CPCG
Cette thèse	✓		✓	✓	✓	✓	solveur

TS= Recherche Tabou ; H-CG =Heuristique basée sur la Génération de Colonnes ;H-CPCG Heuristique basée sur la génération colonnes et de plans coupants ; H-LR Heuristique basée sur la Relaxation Lagrangienne ; SS = Solver

Tableau 4.1 – Caractéristiques générales du réseau et des transferts

4.2.5 Synthèse

Le tableau 4.1 classe les principales références bibliographiques de cette section selon les plusieurs propriétés : la gestion des véhicules par une contrainte *d'équilibre des ressources* ou en définissant des *cycles*, présence de coûts fixes, variables, tarif *LTL* approximé par une fonction linéaire par morceaux (*LM*), coûts de transferts.

En conclusion de cette section, le *SNDP* a été particulièrement étudié ces dernières années. De nombreuses études ont tenté d'intégrer la gestion des tournées de véhicules et la conception des plans de chargement ([2] [3] [4] [30] [80] [92]). Une autre caractéristique est que le *SNDP* permet de traiter le cas de réseaux multi-modaux ([4] [19] [66] [67] [75]). Notre travail de thèse comprend une grande partie des caractéristiques recensées dans le Tableau 4.1 : contraintes d'équilibre des ressources (sauf en début et fin de tournée, car nous considérons des tournées ouvertes), coûts fixes, coûts variables, tarif *LTL* représentés par une fonction linéaire par morceaux et coûts de transfert. Le problème à résoudre dans la partie amont du réseau correspond à la définition du *SNDP* avec comme particularité d'avoir des tournées contenant peu de sommets. Notons que si le tarif *LTL* a déjà été intégré à plusieurs reprises dans un *SNDP* pour des trajets directs, il n'existe, à notre connaissance, pas d'études appliquant ce tarif à des tournées de véhicules.

4.3 Tournées de véhicules avec transfert

Nous décrivons ici deux cas particuliers de problèmes de tournées de véhicules avec transferts, qui présentent de nombreuses similitudes avec notre recherche : le problème de tournées de véhicules avec cross-dock (*Tournées de véhicule avec cross-dock (VRPCD)*), et le problème de collectes et livraisons avec transferts (*PDPT*).

4.3.1 Tournées de véhicules avec cross-dock

Le problème de tournées de véhicules avec cross-dock (*Tournées de véhicule avec cross-dock (VRPCD)*) est un problème de collecte et livraison qui intègre un unique cross-dock. Un ensemble de tournées de

véhicules partent du cross-dock, collectent les marchandises et reviennent au cross-dock où les marchandises sont transférées vers des véhicules réalisant les tournées de livraison. Ce problème a initialement été introduit par Lee et al. [63]. Les auteurs proposent une modélisation et une recherche tabou pour le **VRPCD** où tous les véhicules réalisant les collectes doivent arriver en même temps au cross-dock. Cette méthode a ensuite été améliorée par Liao et al. [65]. Par la suite, Santos et al. [82], et Santos et al. [83] ont développé deux méthodes exactes basées sur la génération de colonnes. Wen et al. [97] ont enrichi le problème de **VRPCD** avec des fenêtres de temps, en relâchant l'hypothèse d'arrivée simultanée de l'ensemble des véhicules. Tous les véhicules passent par le cross-dock, mais les marchandises ne sont pas obligatoirement toutes déchargées. Ils proposent également une modélisation du problème et développent une recherche tabou couplée à une procédure de mémoire adaptative. Ces travaux sont améliorés par Tarantilis et al. [91] avec une recherche tabou à démarrages multiples. Ils considèrent également le cas des tournées ouvertes, c'est à dire lorsque les véhicules se sont pas obligés de retourner à un dépôt à la fin de leur tournée. Ces résultats sont également améliorés par Morais et al. [76] qui proposent un algorithme ILS combiné à la résolution d'un problème de partition d'ensembles. Enfin, Grangier et al. [48] améliorent les résultats de Wen et al. [97], Tarantilis et al. [91] et Morais et al. [76] en développant une recherche à voisinage large associée à la résolution d'un problème de partition d'ensemble et couplage. Dondo et al. [37] proposent un modèle pour le problème multi-échelon de tournées de véhicules où la distribution directe et indirecte (via un cross-dock) est autorisée. Deux transferts d'une même requête sont autorisés, ainsi que la distribution d'une commande en plusieurs fois. Un *branch-and-price* incomplet résout un cas d'étude comportant 44 requêtes et deux cross-docks.

Les problèmes étudiés dans cette section ne présentant pas de caractéristiques très différentes, nous ne présentons pas de tableau de classification pour cette section.

4.3.2 Problèmes de collectes et livraisons avec transferts

Le problème de collectes et livraisons avec transferts (**PDPT**) est une extension du problème de collecte et livraison (**PDP**) : l'ensemble des demandes doivent être transportées de leur origine jusqu'à leur destination et peuvent éventuellement être transférées sur un ensemble de points de transfert. L'objectif est de minimiser uniquement le coût des tournées de véhicules réalisant le transport, un transfert n'occasionnant pas de coût supplémentaire. Les principales différences avec le **VRPCD** sont qu'un véhicule peut collecter et livrer des marchandises dans une même tournée, sans avoir forcément à passer par un point de transfert. Le **PDPT** a été introduit en 1978 par Stein et al. [89] pour un problème de transport de personnes. Shang et al. [86] définit en 1996 le problème de collecte et livraison avec transfert pour le transport de marchandises. Par la suite, Mitrovic-Minic et al. [74] ont développé une recherche locale pour un **PDPT** sans contraintes de capacité sur les véhicules. Cortes et al. [27] proposent le premier modèle mathématique du **PDPT** et résolvent optimalement des instances comprenant jusqu'à six demandes grâce à un *branch-and-cut*. Masson et al. [72] présentent un *branch-and-cut-and-price* pour un cas spécial du **PDPT** où un véhicule n'est pas autorisé à visiter plus de deux destinations de demande. Des instances comprenant jusqu'à 87 destinations ont pu être résolues en moins d'une heure. Masson et al. [70] et Masson et al. [71] proposent ensuite un **ALNS** respectivement pour le **PDPT** et pour le *dial-a-ride problem* avec transfert.

Rais et al. [81] proposent un nouveau modèle pour le **PDPT** autorisant de multiples transferts. Un solveur est capable de résoudre des instances comprenant jusqu'à 14 demandes. Kerivin et al. [56] étudient un problème de **PDPT** où une requête peuvent être divisée en plusieurs parties transportées séparément

Référence	Plusieurs transferts	Transfert milieu	Méthode
Stein et al. [89]		✓	Analysis
Shang and Cuff [86]		✓	Heuristique
Mitrovic-Minic et Laporte [74]			LS
Cortes et al. [27]	✓	✓	Modèle + B&C
Kerivin et al. [56]	✓	✓	B&C
Masson et al. [70]		✓	B&C
Masson et al. [71]		✓	B&C
Masson et al. [72]			B&C&P
Rais et al. [81]	✓	✓	solveur
Coltin et al. [25]	✓	✓	VNS

LS= Recherche Locale ; ALNS = Adaptive Large Neighborhood Search ; B&C = Branch and Cut ; B&C = Branch and Cut and Price ; VNS = Recherche à voisinage variable

Tableau 4.2 – Problèmes de collectes et livraison avec transferts

afin d'optimiser la capacité des véhicules. Un *branch-and-cut* qui résout des instances incluant jusqu'à 15 requêtes et 7 véhicules dans un réseau comprenant 10 sommets. Plus récemment, Coltin et al. [25] autorisent plusieurs transferts pour une demande. Ils développent un VNS qui résout des instances de 144 requêtes et 10 véhicules.

Le tableau 4.2 regroupe les problèmes de transport avec transfert cités dans cette section. La colonne 2 indique si plusieurs transferts d'une même demande sont autorisés ou non. la colonne 3 indique s'il est possible ou non de transférer une demande en milieu de tournée. Notre étude possède ces deux caractéristiques.

4.4 Problèmes riches de tournées de véhicules

L'évolution des techniques d'optimisation permet aujourd'hui de résoudre des instances de plus en plus grandes et des modèles de plus en plus complets de problèmes de tournées de véhicules, Les problèmes dits *riches* sont ceux qui intègrent un nombre élevé de contraintes réelles, et donc modélisent assez finement les problèmes d'optimisation du monde industriel. Pour une revue complète des problèmes riches de tournées de véhicules, nous renvoyons le lecteur aux états de l'art de Lahyani et al. [61], Caceres-Cruz et al. [14] et Vidal et al. [94].

Cette section reprend les caractéristiques liées au contexte industriel des problèmes de cette thèse : des tournées multi-trip, c'est-à-dire qu'un véhicule peut effectuer plusieurs tournées (section 4.4.1), des contraintes temporelles limitant le temps d'attente ou la durée de la tournée (section 4.4.2), des caractéristiques liées à la sous-traitance du transport (section 4.4.3) et des caractéristiques liées à la collaboration entre transporteurs (section 4.4.4).

4.4.1 Tournées multi-trip

Lorsque l'utilisation d'un véhicule engendre un coût fixe, ou dans le cas des flottes hétérogènes (voir l'état de l'art de Koç et al [58]), le choix du véhicule qui doit effectuer chaque tournée devient une variable de décision du problème. Il peut être alors opportun de confier plusieurs tournées à un même véhicule, on parle alors de tournées multiples (ou multi-trip).

Taillard et al. [90] définissent le problème de tournée de véhicules avec multi-trip en 1996, et proposent une recherche tabou pour des instances jusqu'à 90 clients. Par la suite, de nombreux articles ont intégré les tournées multiples, souvent couplées avec d'autres caractéristiques riches. Nguyen et al. [78] étudient un problème multi-trip dans lequel l'espace géographique est divisé en plusieurs zones. Chaque

zone comporte un site d'approvisionnement et des clients à servir. Un véhicule ne peut pas arriver avant l'ouverture d'un site d'approvisionnement, il doit dans ce cas à attendre dans une station d'attente. Les auteurs proposent une recherche tabou pour résoudre ce problème. Azi et al. [6] présentent un **ALNS** pour la résolution d'un problème de tournées de véhicules avec une flotte limitée et des tournées multi-trip possibles. L'objectif est de maximiser le nombre de clients servis, et dans un second temps de minimiser la distance totale parcourue par les véhicules. Cataruzza et al. [16] proposent une recherche locale itérative pour résoudre le problème de tournées de véhicules avec fenêtres de temps et multi-trip. Grangier et al. [47] présentent également une méthode **ALNS** pour un problème de transport sur deux échelons et à voyages multiples.

4.4.2 Contraintes temporelles sur les tournées

Ceselli et al. [17] présentent un algorithme d'optimisation basé sur la génération de colonnes pour un problème riche de tournées de véhicules. Ce dernier intègre des contraintes de compatibilité entre produits, durée maximale d'une tournée, réglementation des journées des conducteurs, ainsi que la possibilité de sous-traiter certaines livraisons de commandes. Le coût dépend de multiples facteurs comme la distance, la charge du véhicule, ou le nombre d'arrêts de la tournée. Parragh et al. [79] étudient un problème riche de transport de personnes : la flotte est hétérogène et des règles relatives à la législation du travail doivent être respectées. Les auteurs proposent un algorithme basé sur la génération de colonnes ainsi qu'une recherche à voisinage variable.

4.4.3 Transport sous-traité

Cette section traite les problématiques d'optimisation inhérentes à la gestion d'un réseau dont le transport est sous-traité. Cette sous-traitance modifie certaines caractéristiques d'un problème de transport classique, telles que la tarification ou la forme des tournées imposées par les transporteurs.

Basligil et al. [8] étudient un problème de collectes et livraisons du point de vue d'un 3PL qui doit choisir entre trois types de véhicules. Chaque type de véhicule possède sa propre tarification. Le problème est résolu avec une heuristique en deux phases : la première phase attribue les commandes aux véhicules par la résolution d'un **PLNE** ; la seconde phase construit les tournées avec un algorithme génétique. Gunther et al. [49] étudient un problème de planification du transport dans un réseau où le transport est sous-traité. Ils proposent également une approche en deux phases : la première phase construit des plans de chargement de véhicules. Les meilleures consolidations sont ensuite sélectionnées par un solveur dans une seconde phase. Krajewska et al. [60] utilisent quatre tarifs différents qui correspondent au parc de véhicules de l'entreprise et à trois types de sous-traitance : un tarif à la tournée (dépendant de la distance parcourue), un tarif à la journée, et le tarif d'une expédition d'une commande donnée. Le problème est résolu par une recherche tabou évaluée sur des instances générées de 25 à 50 clients ainsi que sur des instances réelles. Pour le même problème, Kopfer et Wang [59] proposent un modèle de **PLNE** résolu avec Cplex pour des instances à 20 clients. Nous nous sommes inspirés de ces articles pour décrire les trois types de tarifs dans le chapitre 6 de cette thèse. Enfin, Lindsey et al. [68] étudient un problème de collectes et livraisons dans un réseau logistique où une commande peut être livrée directement au client avec un tarif **LTL**, ou être consolidée avec d'autres commandes sur une plateforme de consolidation avec un tarif **FTL**. Des instances de petite taille sont résolues avec un solveur. Celles de plus grande taille sont résolues par une décomposition et une résolution séquentielle

de modèles mathématiques contenant un sous-ensemble des variables du modèle global.

4.4.4 Collaboration entre transporteurs

Cette section traite de collaboration entre transporteurs.

Chu [23] étudie un problème où un ensemble de commandes provenant d'un entrepôt unique doivent être distribuées par un transporteur qui peut utiliser sa propre flotte de véhicules ou confier certaines demandes à d'autres transporteurs. Chaque véhicule de sa flotte a un coût fixe à la journée et un coût variable dépendant de la distance parcourue. Des tournées sont réalisées par les véhicules du transporteur et un coût fixe est payé par commande si elle est confiée à d'autres transporteurs. Une heuristique de recherche locale est proposée et résout des instances qui comportent jusqu'à 29 clients. Bolduc et al. [12] améliorent les résultats de Chu [23] avec de nouveaux opérateurs pour la recherche locale, notamment par rapport aux procédures d'échanges de commande entre transporteurs. Par la suite, Bolduc et al. [11] montrent que le problème peut-être formulé comme un problème de tournées de véhicules avec une flotte hétérogène. Les auteurs proposent une métaheuristique qui utilise une procédure de perturbation pour la construction de la solution et les procédures d'échange de commandes. Les résultats sont significativement améliorés et de nouvelles instances comprenant jusqu'à 480 clients sont proposées. Pour le même problème, Côté et al. [28] développent une recherche tabou qui améliore notamment la qualité de la solution et réduit le temps de résolution. Wang et Kopfer [96] travaillent également sur un problème où les transporteurs peuvent échanger leurs commandes. Chaque transporteur possède un ensemble de demandes à livrer et l'objectif est de trouver une meilleure répartition des commandes entre les transporteurs, par un mécanisme d'échange de commandes. Enfin, Vornhusen et al. [95] proposent un modèle mathématique pour le PDPT avec un regroupement de transporteurs, où les demandes sont partagées entre plusieurs transporteurs. Les dépôts ne sont pas partagés, mais des véhicules appartenant à différents transporteurs peuvent passer par les mêmes plateformes de transfert.

4.5 Conclusion

Cet état de l'art présente différents types de problèmes présentés dans quatre sections différentes : les problèmes de conception de réseau logistique, les problèmes de planification opérationnelle du transport dans un réseau logistique, les problèmes de tournées de véhicules avec transfert et enfin les problèmes riches de tournées de véhicules. Ces quatre types de problèmes regroupent des caractéristiques des problèmes étudiés par cette thèse. On remarque que de nombreuses méthodes existent pour la conception d'un réseau logistique, en intégrant une dimension opérationnelle aux problèmes telle que la détermination de tournées de véhicules dans le réseau. Néanmoins, l'aspect le moins abordé de la littérature est l'optimisation intégrée du transport dans un réseau longue distance synchronisé avec le transport dans les réseaux locaux de distribution de marchandises par des tournées de véhicules. Ce dernier point fait l'objet des chapitres 5, 7 et 8 de cette thèse.

Modèles mathématiques pour l'estimation des coûts dans le réseau mutualisé

Sommaire

5.1	Introduction	48
5.2	Description du problème modélisé	48
5.2.1	Définition des ensembles du réseau logistique	48
5.2.2	Tournées de véhicules et structure tarifaire	49
5.2.3	Flux et structure des marchandises	52
5.3	Modélisation des transferts	53
5.3.1	Structure de transfert complète	53
5.3.2	Structure de transfert en étoile	53
5.4	Modèles mathématiques	54
5.4.1	Itinéraire amont	55
5.4.2	Itinéraire aval	58
5.4.3	Itinéraires directs	58
5.4.4	Lien entre réseau amont et réseau aval	59
5.4.5	Satisfaction de la demande	59
5.4.6	Modélisation des tournées	59
5.4.7	Fonction objectif	60
5.4.8	Deux modèles mathématiques	61
5.5	Tests numériques	63
5.5.1	Instances	63

5.5.2 Résultats	67
5.6 Conclusion	70

5.1 Introduction

Le problème traité dans ce chapitre reprend la problématique décrite dans le chapitre 2. On propose deux modélisations d'un problème de plan de chargement dans le réseau mutualisé en intégrant des tournées de véhicules. Ces modèles mathématiques n'ont pas une vocation opérationnelle mais tactique : ils servent à estimer les coûts de transport dans le réseau logistique mutualisé sur un horizon d'un an. En outre, on veut pouvoir comparer ces coûts à des coûts de transport sans mutualisation. L'étude est basée sur un historique de commandes de 52 semaines (pour chaque fournisseur). Pour chaque jour de livraison, nous modélisons un problème d'optimisation correspondant aux hypothèses de mutualisation à tester. L'approche consistant à modéliser par **Programme Linéaire en Nombres Entiers (PLNE)** et résoudre par solveur correspond bien à la problématique tactique. Les temps de calculs ne sont pas la préoccupation première, mais il est nécessaire de pouvoir faire évoluer le modèle pour tester de nouvelles hypothèses. La performance du modèle liée à sa taille apparaît néanmoins comme un critère important afin de pouvoir résoudre toutes les tailles de problèmes auxquels 4S Network fait face. Dans l'optique de réduire la taille des modèles, nous proposons et comparons deux modélisations des transferts.

Ce chapitre présente un modèle qui a été résolu par un solveur sur un grand nombre de périodes, avec plusieurs coalitions de partenaires, et sous différentes hypothèses. Les résultats ont permis d'appuyer 4S Network lors de la recherche des premiers partenaires du projet CRC Services. Ils ont également permis de choisir des prestataires et des modes de fonctionnement pour le réseau.

5.2 Description du problème modélisé

Cette section présente le problème modélisé, ainsi que les hypothèses prises en compte dans le modèle. Nous décrivons le réseau logistique considéré, la structure des tournées de véhicules ainsi que leurs types de coûts, et enfin les différents itinéraires possibles des marchandises.

5.2.1 Définition des ensembles du réseau logistique

Le réseau logistique étudié est modélisé par un ensemble \mathcal{V} de sommets. Parmi ces sommets, on distingue un ensemble $\mathcal{C} \subset \mathcal{V}$ de CRC et un ensemble $D \subset \mathcal{V}$ de clients. On note \mathcal{I} l'ensemble des fournisseurs et $\mathcal{F}_i \subset \mathcal{V}$ l'ensemble des sites de production/entrepotage. On introduit $\mathcal{T}_i \subset \mathcal{F}_i$ et $\mathcal{S}_i \subset \mathcal{T}_i$ les deux sous-ensembles représentant les sites d'un fournisseur i capables respectivement de transférer des marchandises d'un véhicule à un autre, et de réaliser de la préparation de commande.

- Le *réseau amont* comprend les CRC \mathcal{C} et les sites des fournisseurs $\bigcup_{i \in \mathcal{I}} \mathcal{F}_i$. On considère un réseau complet pour le réseau amont : on trouve un arc entre chaque paire de sommets de l'ensemble $\mathcal{C} \cup \bigcup_{i \in \mathcal{I}} \mathcal{F}_i$.
- L'ensemble D des clients compose avec l'ensemble \mathcal{C} des CRC les sites du *réseau aval*. Dans le réseau aval, un CRC ne peut livrer que l'ensemble des clients qui sont situés à une distance

inférieure à sa distance maximale de livraison, appelée *couverture* et noté δ . Par exemple, si la couverture d'un CRC est de 300km, un client situé à 400km du CRC ne peut pas être livré. On considère un réseau complet entre chaque CRC $c \in \mathcal{C}$ et l'ensemble de clients qu'il peut livrer $v \in D_c$.

- Le *réseau direct* comporte l'ensemble $\bigcup_{i \in I} \mathcal{F}_i$ des sites des fournisseurs et l'ensemble D des clients. Le réseau direct est composé des arcs entre chaque site fournisseur et chaque client.

Exemple 7. La figure 5.1 représente un exemple de réseau amont avec trois sites fournisseurs et deux CRC. La figure 5.2 représente un exemple du réseau aval avec deux CRC et six clients. Enfin la figure 5.3 illustre un réseau direct avec trois fournisseurs et six clients.

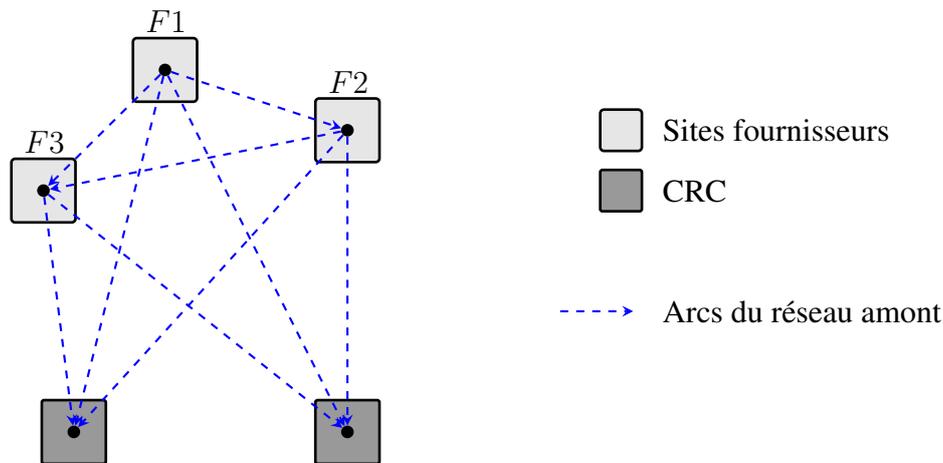


Figure 5.1 – Un exemple de réseau amont

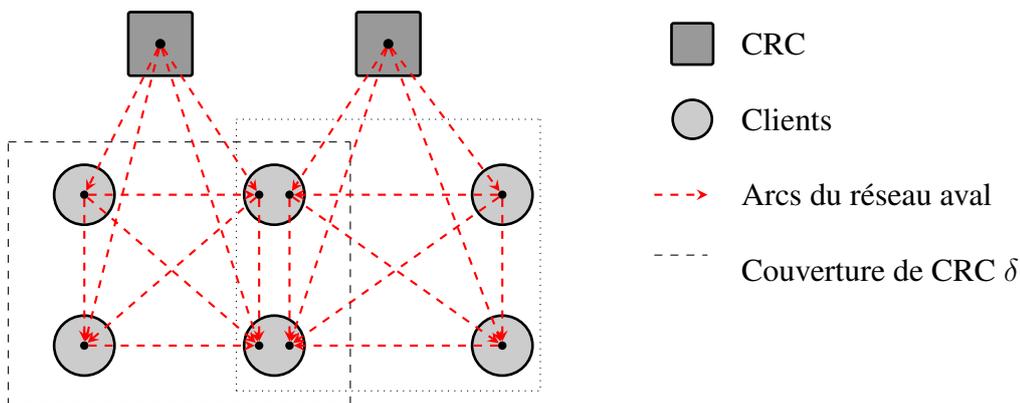


Figure 5.2 – Un exemple de réseau aval

5.2.2 Tournées de véhicules et structure tarifaire

Les opérations de transport dans le réseau de distribution sont réalisées au moyen de tournées ouvertes. Après avoir introduit les notations nécessaires, nous décrivons plus précisément les différentes contraintes et structures de tarif s'appliquant aux tournées.

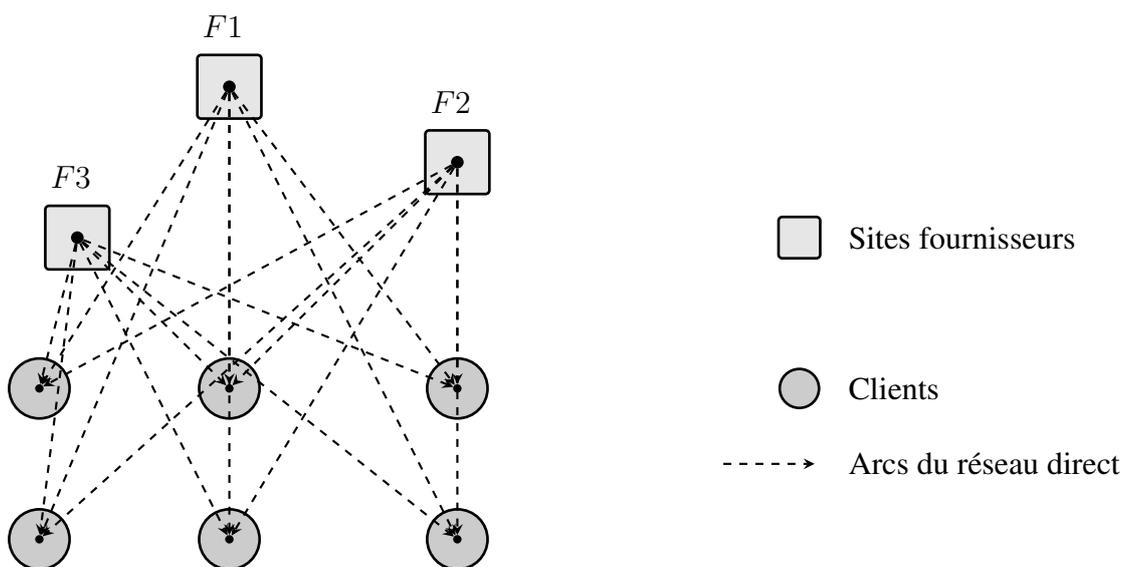


Figure 5.3 – Un exemple de réseau direct

On note \mathcal{R} l'ensemble des tournées dans le réseau. \mathcal{R} peut être décomposé en un sous-ensemble \mathcal{R}^- de tournées dans le réseau amont et le réseau direct et un sous-ensemble \mathcal{R}^+ de tournées dans le réseau aval. Pour tout sommet $v \in \mathcal{R}$, v^- et v^+ désignent les sommets précédant et suivant v dans sa tournée. Notons enfin \mathcal{V}_r l'ensemble des sommets de la tournée $r \in \mathcal{R}$ et \mathcal{R}_v l'ensemble des tournées passant par $v \in \mathcal{V}$.

5.2.2.1 Réseau amont et direct

Une tournée dans le réseau amont $r \in \mathcal{R}^-$ comprend deux à quatre sommets du réseau. Cette contrainte logistique vient de la volonté de limiter l'impact d'un retard : si un véhicule est en retard à cause d'un passage sur un site, ce retard se répercutera sur un nombre limité de sites à visiter. Une tournée du réseau amont est composée d'un ou plusieurs sites de fournisseurs. Un CRC est potentiellement présent uniquement à la fin de la tournée.

Dans le réseau direct, une tournée de véhicule est un trajet direct entre le site d'un fournisseur et un client.

Exemple 8. La figure 5.4 représente 2 tournées dans le réseau amont. La Figure 5.5 représente 5 tournées dans le réseau direct.

Le coût d'une tournée dans le réseau amont et réseau direct a une structure tarifaire LTL c'est à dire que son prix dépend à la fois du nombre de palettes et de la distance. Une fonction linéaire par morceaux modélise cette structure de coût, comme décrit dans la section 3.2 du chapitre 3. Contrairement à la majorité des modèles représentant le coût d'une tournée, le coût total de la tournée n'est pas calculé en sommant le coût de chacun de ses arcs : c'est la charge maximale transportée au cours de la tournée qui est prise en compte pour le calcul du coût. Le prix associé à l'utilisation d'une flotte homogène sur une tournée est calculé en reproduisant autant de fois que nécessaire le tarif LTL d'un véhicule unique, comme illustré par la figure 3.2 dans la section 3.2 du chapitre 3.

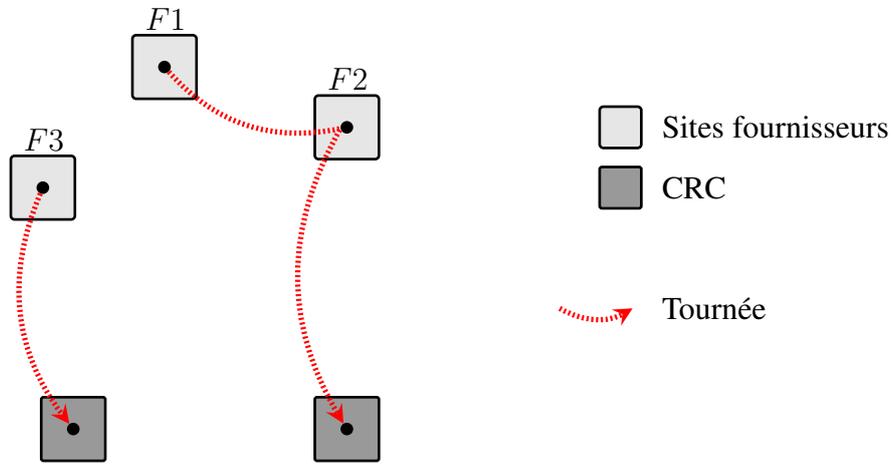


Figure 5.4 – Tournées dans un réseau amont

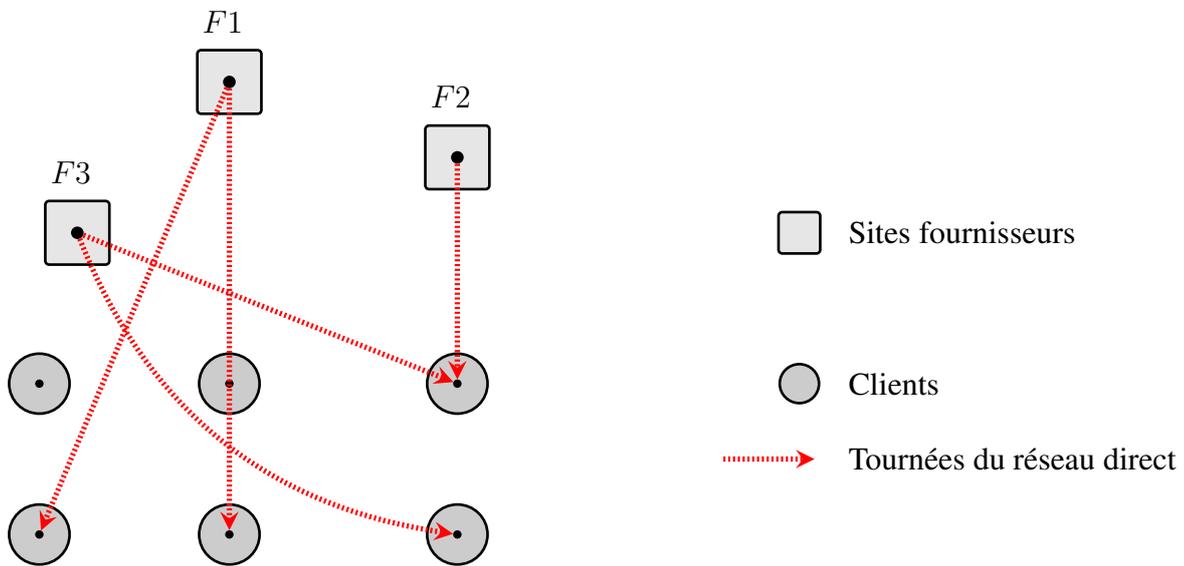


Figure 5.5 – Tournées dans un réseau direct

5.2.2.2 Réseau aval

Une tournée aval $r \in R^+$ part d'un CRC et visite un à trois clients. La figure 5.6 donne un exemple de deux tournées aval.

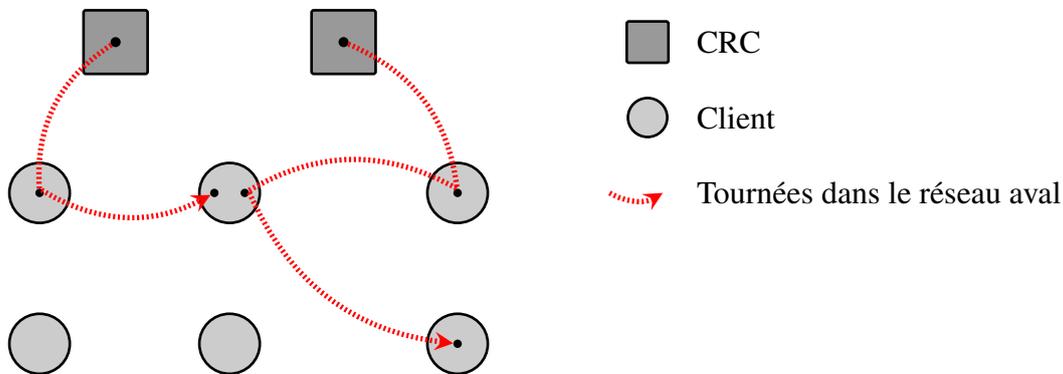


Figure 5.6 – Tournées dans le réseau aval

Dans le réseau aval, le tarif FTL est utilisé. Le coût est dépendant des zones de livraison par lesquelles un véhicule passe et du nombre d'ouverture de portes, comme défini dans la section 2.5.2 du chapitre 2.

5.2.2.3 Énumération des tournées

La contrainte du nombre maximal de sites visités dans les tournées amont et aval limite fortement le nombre total de tournées potentielles : Il est ainsi possible de les énumérer. On réalise une énumération de toutes les tournées de chaque réseau en pré-traitement des données :

- en amont, on énumère toutes les tournées potentielles entre les sites fournisseurs : pour un ensemble de sites fournisseurs, on considère toutes les positions des sommets dans la tournée. Un CRC peut-être visité par un véhicule du réseau amont mais uniquement à la fin de la tournée.
- en aval, on énumère toutes les tournées potentielles. Le départ de la tournée est systématiquement un CRC et le véhicule visite ensuite 1 à 3 clients. Pour un même ensemble de clients visités par un véhicule depuis un CRC on retient uniquement la configuration avec la distance la plus courte.

5.2.3 Flux et structure des marchandises

On note J l'ensemble des commandes et O l'ensemble des sous-commandes. On appelle d_j le client de la commande j . Soit O^j l'ensemble des sous-commandes d'une commande $j \in J$ et O_i l'ensemble des sous-commandes passées à un fournisseur $i \in I$. Le sous ensemble $O_i^P \subset O_i$ désigne l'ensemble des sous-commandes de O_i ayant besoin d'une préparation. Une sous-commande $o \in O^j$ est caractérisée par une origine (point de départ) $p_o \in \mathcal{F}_i$ et une quantité q_o .

5.2.3.1 Itinéraires

Les sous-commandes peuvent emprunter deux types d'itinéraires :

- Un *itinéraire CRC* : les sous-commandes transitent par un CRC avant d'être livrées au client. Les sous-commandes d'une commande doivent toutes passer par le même CRC afin d'y être consolidée avant livraison au client (*merge-in-transit*).
- Un *itinéraire direct* : les sous-commandes ne passent pas par un CRC. Elles peuvent utiliser le réseau direct et amont (sous condition de ne pas passer par un CRC). Toutes les sous-commandes d'une même commande doivent alors être consolidées sur un site fournisseur pour être expédiées ensembles à leur client.

Dans les deux cas, une sous-commande peut-être transférée plusieurs fois dans le réseau amont. De plus, une commande peut être transférée d'une tournée 1 à une tournée 2 avant la fin de la tournée 1. C'est pourquoi on distingue l'itinéraire d'une commande de la tournée d'un véhicule. Lors d'un transfert d'un véhicule à un autre, un coût de transfert noté $\rho_v, v \in \mathcal{V}$ est pris en compte.

5.3 Modélisation des transferts

Dans le domaine de la conception de réseau et plus spécialement pour des réseaux multi-modaux, la modélisation des transferts implique généralement un grand nombre de variables. Les transferts peuvent être modélisés de plusieurs façons. La modélisation des transferts doit tenir compte de deux impératifs. Tout d'abord, le transfert des marchandises au sommet $v \in \mathcal{V}$ occasionne un coût fixe ρ_v . Il faut donc compter le nombre de transferts réalisés en chaque sommet $v \in \mathcal{V}$ afin de calculer le prix total des transferts. Ensuite, certains transferts sont obligatoires. C'est par exemple le cas lorsqu'une sous-commande nécessite une préparation. Il faut donc pouvoir modéliser le forçage d'un transfert.

Les modèles évalués dans cette thèse ont été introduits par Karsten et al. [55] sous les appellations *structure de transfert complète* et *structure de transfert en étoile*. Comme dans ce chapitre, les auteurs comparent leur propre modélisation de transfert à la *structure complète de transfert* pour en évaluer les performances.

5.3.1 Structure de transfert complète

L'idée de la structure de transfert complète est qu'un site de transfert visité par n tournées est modélisé par n sommets distincts dans le réseau associé. Cela permet de représenter un transfert par un arc entre deux de ses sommets. La figure 5.7 donne un exemple de la modélisation de la structure de transfert complète avec quatre tournées passant par un unique sommet physique. Cette modélisation est la plus naturelle pour représenter des transferts dans un réseau. Elle est notamment utilisée dans Rais et al. [81] et plus généralement dans le cas où les modèles ne sont pas résolus avec l'aide d'un solveur (Moccia et al. [75], Lin et al. [67], Brouer et al. [13]). Le principal défaut de cette modélisation est qu'elle induit la création d'un très grand nombre d'arcs : dans les modèles le nombre d'arcs est en $o(|\mathcal{V}|^2)$. La mémoire peut alors être très rapidement saturée en cas de résolution d'instances trop grandes.

5.3.2 Structure de transfert en étoile

La structure de transfert en étoile modélise également un site de transfert par n sommets, mais évite d'introduire un nombre trop important d'arcs entre les n répliques. Pour ce faire, un sommet supplémentaire est créé et relié aux n sommets de transfert, comme illustré par la figure 5.8. Cela réduit le

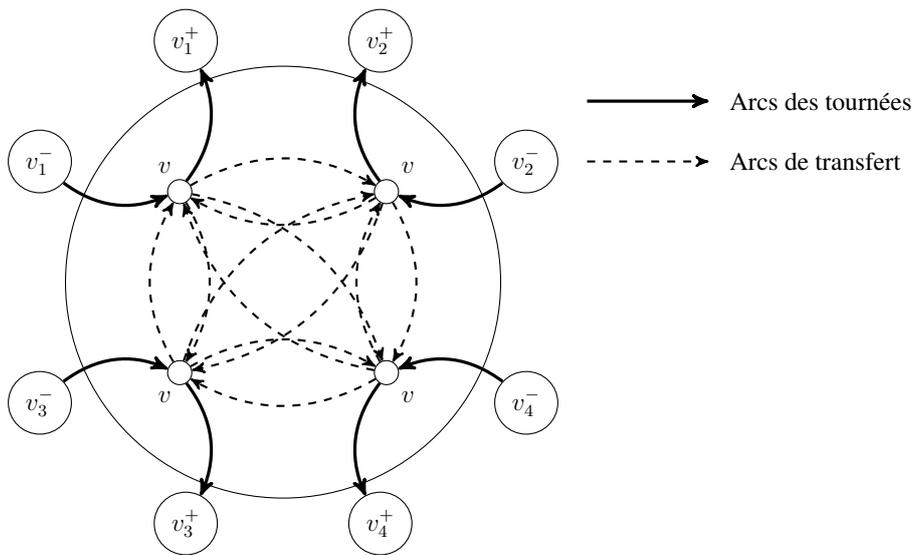


Figure 5.7 – Structure de transfert complète

nombre d'arcs du modèle : $o(V)$ arcs. La figure 5.8 donne un exemple de structure de transfert en étoile avec quatre tournées visitant un même sommet physique. Nous n'avons pas trouvé d'article résolvant cette modélisation à l'aide d'un solveur.

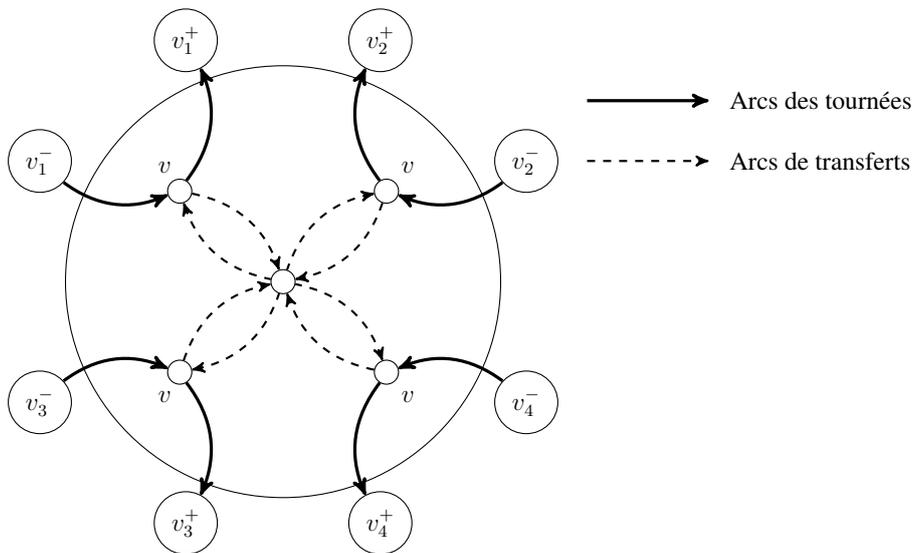


Figure 5.8 – Structure de transfert en étoile

5.4 Modèles mathématiques

Dans cette section, nous proposons une modélisation mathématique du problème de minimisation des coûts de transport dans le réseau collaboratif. Nous proposons deux variantes, utilisant la structure de transfert complète ou en étoile. Notons que dans les sections 5.4.1 à 5.4.7, nous détaillons chaque

contrainte du modèle mathématique. Dans la section 5.4.8, nous réécrivons le modèle de manière plus compacte.

5.4.1 Itinéraire amont

Tout d'abord, on décrit les contraintes liées aux tournées du réseau amont et réseau direct.

5.4.1.1 Contraintes ne concernant pas les points de transferts

L'itinéraire d'une sous-commande o dans le réseau amont est modélisé par la variable binaire $z_{o,r,v}$. Cette variable est égale à 1 si la sous-commande $o \in O$ passe par le segment de la tournée $r \in R^-$ qui arrive au sommet $v \in \mathcal{V}_r$, et 0 sinon. Notons que cette variable représente le passage de la sous-commande o par l'arc de la tournée r entre le sommet prédécesseur de v (dans cette tournée) et v . En associant l'indice de la tournée r à la variable, il n'y a pas besoin de spécifier les deux sommets de l'arc.

Origine des sous-commandes Les contraintes (5.1) indiquent qu'une sous-commande $o \in O$ doit être chargée dans une tournée à sa plateforme de départ p_o .

$$\sum_{r \in R_{p_o}} z_{o,r,p_o^+} = 1, \quad \forall o \in O \quad (5.1)$$

5.4.1.1.1 Conservation des flux dans un véhicule On considère le cas où une sous-commande o est transportée par une tournée r et n'est pas transférée sur le sommet v . La sous-commande reste donc dans la tournée r . On a donc $z_{o,r,v^+} = z_{o,r,v} = 1$. Ces contraintes de conservation des flux sont imposées pour les situations où une sous-commande passe par une plateforme où elle ne peut pas sortir du véhicule. C'est le cas lorsque la plateforme ne peut pas réaliser de transferts, ou qu'elle appartient à un autre fournisseur que celui de la sous-commande considérée.

Une sous-commande ne peut pas être transférée d'un véhicule à un autre sur une plateforme n'appartenant pas à son fournisseur. Ceci est modélisé de la manière suivante :

$$z_{o,r,v^+} = z_{o,r,v}, \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{V} \setminus (\mathcal{T}_i \cup \mathcal{C} \cup \{d_o\}), \forall r \in R_v \quad (5.2)$$

5.4.1.2 Structure de transfert complète

On introduit les variables $u_{or'r'v}$, égales à 1 si la sous-commande o est transférée de la tournée r à la tournée r' sur la plateforme v et 0 sinon.

Les contraintes de transfert (5.3) connectent les variables représentant les tournées $z_{o,r,v}$ et les variables de transfert $u_{or'r'v}$.

$$z_{o,r,v^+} - z_{o,r,v} = \sum_{\substack{r' \in R_v \\ r \neq r'}} (u_{o,r',r,v} - u_{o,r,r',v}) \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v. \quad (5.3)$$

En cas de transfert sur le sommet v d'une sous-commande arrivant par la tournée r alors $z_{o,r,v^+} - z_{o,r,v} = -1$, ce qui force une variable $u_{o,r,r',v}$ à être égale à 1 (et occasionner un coût de transfert dans la fonction objectif). Dans le cas où la sous-commande reste dans la tournée r , alors $z_{o,r,v^+} - z_{o,r,v} = 0$.

Les contraintes (5.4) et (5.5) permettent de modéliser le cas des transferts obligatoires, en cas de préparation de commandes.

Les contraintes de transfert (5.4) obligent une commande o nécessitant une préparation de commandes à être transférée sur au moins une plateforme capable de préparer les palettes. Les contraintes (5.5) connectent les variables $z_{o,r,v}$ et $u_{o,r,r',v}$.

$$\sum_{v \in \mathcal{S}_i} \sum_{r \in R_v} \sum_{\substack{r' \in R_v \\ r' \neq r}} u_{o,r,r',v} \geq 1 \quad \forall i \in \mathcal{I}, \forall o \in O_i^P \quad (5.4)$$

$$z_{o,r,v} \geq \sum_{\substack{r' \in R_v \\ r' \neq r}} u_{o,r,r',v} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.5)$$

Si un transfert obligatoire est défini par $u_{o,r,r',v}$ depuis la route r , alors on fixe $z_{o,r,v}$ à 1.

Exemple 9. La figure 5.9 montre un exemple d'itinéraire d'une sous-commande. La sous-commande o arrive au site de transfert en provenance du sommet v^- prédécesseur de v sur la tournée r et est transférée vers le véhicule effectuant la tournée r' . Les variables égales à 1 sont donc $z_{o,r,v}$, z_{o,r',v^+} et $u_{o,r,r',v}$.

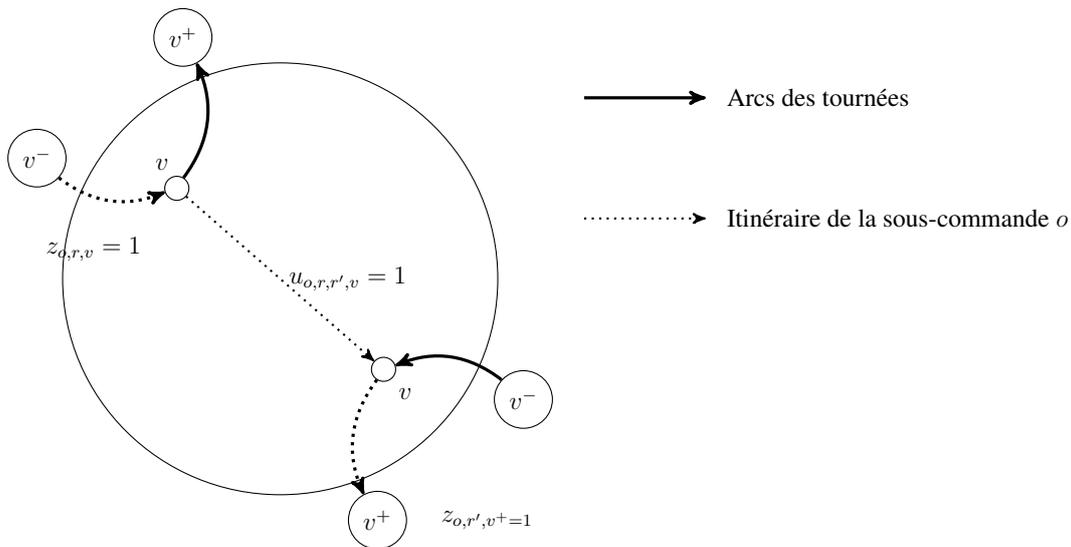


Figure 5.9 – Utilisation de la structure complète de transfert

5.4.1.3 Structure de transfert en étoile

Comme décrit dans la partie 5.3, la structure de transfert en étoile permet de générer un nombre réduit de variables par rapport à la structure de transfert complète.

On définit les variables entières $u_{o,v} \in \{0, 1\}$ égales à un si la sous-commande o est transférée sur la plateforme v , 0 sinon.

On définit les variables entières $a_{o,r,v}^+ \in \{0,1\}$ égales à un si la sous-commande o est transférée depuis la tournée r sur la plateforme physique v et les variables $a_{o,r,v}^- \in \{0,1\}$ égales à un si la sous-commande o est transférée dans le véhicule qui effectue la tournée r sur la plateforme v .

Exemple 10. La figure 5.10 donne un exemple d'un itinéraire d'une sous-commande comprenant un transfert. La sous-commande o arrive sur la plateforme v via la tournée r_1 . Elle est ensuite transférée dans le véhicule qui effectue la tournée r_4 . Les variables égales à 1 et représentant l'arrivée, le transfert, et le départ de la sous commande o sur la plateforme v sont $z_{o,r_1,v}$, $z_{o,r_4,v}$, $a_{o,r_1,v}^+$, $a_{o,r_4,v}^-$ et $u_{o,v}$.

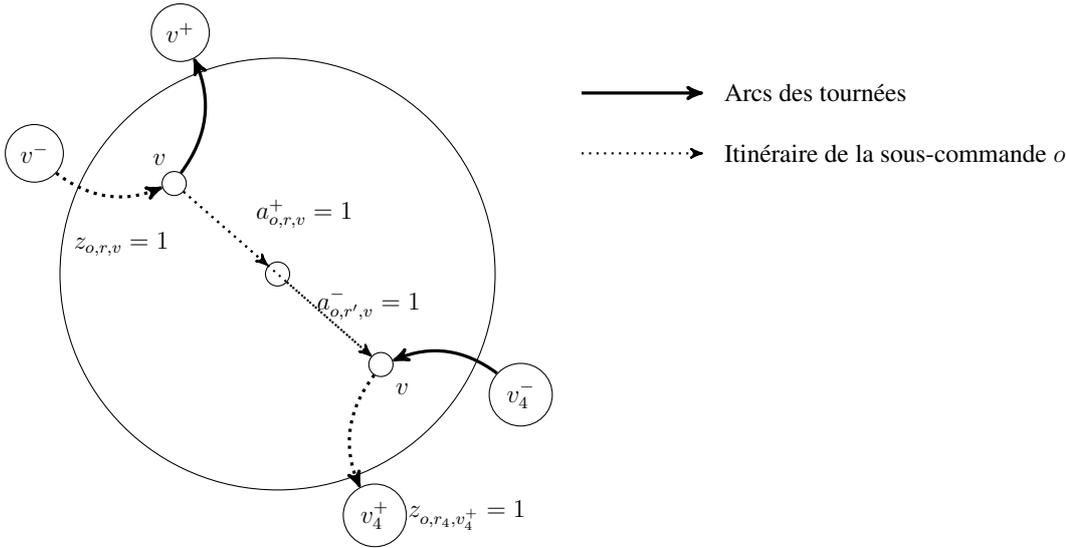


Figure 5.10 – Utilisation de la structure de transfert en étoile

Si la sous-commande o est transférée de la tournée r à la tournée r' sur la plateforme v , alors : $z_{o,r,v} = 1$, $z_{o,r,v^+} = 0$, $z_{o,r',v} = 0$, $z_{o,r',v^+} = 1$, $a_{o,r,v}^+ = 1$, $a_{o,r',v}^- = 0$, $a_{o,r,v}^- = 0$, $a_{o,r',v}^+ = 1$. C'est pourquoi $\sum_{r \in R_v} (a_{o,r,v}^+ + a_{o,r,v}^-) = 2$. De plus $u_{o,v} = 1$.

Les contraintes suivantes connectent entre elles les variables $z_{o,r,v}$, $a_{o,r,v}^+$, $a_{o,r,v}^-$, et $u_{o,v}$.

$$z_{o,r,v} - z_{o,r,v^+} = a_{o,r,v}^+ - a_{o,r,v}^- \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}, \forall r \in R_v \quad (5.6)$$

$$a_{o,r,v}^+ \geq z_{o,r,v} - z_{o,r,v^+} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}, \forall r \in R_v \quad (5.7)$$

$$a_{o,r,v}^- \geq -z_{o,r,v} + z_{o,r,v^+} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}, \forall r \in R_v \quad (5.8)$$

$$a_{o,r,v}^+ + a_{o,r,v}^- \leq 1 \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}, \forall r \in R_v \quad (5.9)$$

$$\sum_{r \in R_v} (a_{o,r,v}^+ + a_{o,r,v}^-) = 2 u_{o,v} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}, \forall r \in R_v \quad (5.10)$$

$$\sum_{r \in R_v} (a_{o,r,v}^+ - a_{o,r,v}^-) = 0 \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}. \quad (5.11)$$

Aucun transfert de la sous-commande o ne peut avoir lieu à l'origine p_o car cela voudrait dire que o est chargée et déchargée sur la même plateforme, d'où les $v \in \mathcal{T}_i \setminus \{p_o\}$ dans l'application des contraintes de (5.6) à (5.11).

Compter les transferts Dans le cas où les transferts doivent uniquement être comptés, comme par exemple pour une sous-commande n'ayant pas besoin de préparation, les équations de (5.6) à (5.11) permettent de définir les variables $u_{o,v}$, $a_{o,r,v}^+$ et $a_{o,r,v}^-$ comme des variables réelles dans l'intervalle $[0, 1]$. En effet, si une sous-commande est transférée de la tournée r à la tournée r' , alors $z_{o,r,v} = 1$, $z_{o,r,v^+} = 0$ et donc $z_{o,r,v} - z_{o,r,v^+} = 1$. On doit donc avoir $a_{o,r,v}^+ - a_{o,r,v}^- = 1$ et donc automatiquement $a_{o,r,v}^+ = 1$. De même pour les contraintes modélisant l'arrivée de la sous-commande o dans la tournée r' : on a $z_{o,r',v} = 0$, $z_{o,r',v^+} = 1$ et donc $z_{o,r,v} - z_{o,r,v^+} = -1$ et $a_{o,r',v}^+ - a_{o,r',v}^- = -1$. Les contraintes (5.10) impliquent que $u_{o,v} = 1$. Dans le cas où la sous-commande o arrive par la tournée r mais n'est pas transférée alors $z_{o,r,v} = 1$, $z_{o,r,v^+} = 1$ et donc $z_{o,r,v} - z_{o,r,v^+} = 0$. Il est alors possible pour les variables $a_{o,r,v}^+$ d'être égales à une valeur comprise entre 0 et 0.5, à condition que leur somme soit inférieure à 1 (contrainte 5.9). Si leur valeur est supérieure à 0, $u_{o,v}$ aurait alors une valeur supérieure à 0, ce qui engendrerait un coût de transfert non nécessaire. Les variables $a_{o,r,v}^+$ sont donc bien égales à 0 dans une solution de coût minimal.

La totalité du flux passant par la plateforme v doit être transmis à ses successeurs. Cette contrainte de conservation des flux est la suivante (5.12) :

$$\sum_{r \in R_v} (z_{o,r,v} - z_{o,r,v^+}) = 0 \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}. \quad (5.12)$$

Forcer les transferts Il s'agit ici de modéliser les transferts obligatoires liés à la préparation de commandes. Forcer les transferts de la sous-commande o revient à forcer la valeur de la somme des variables $u_{o,v}$ à 1 sur les plateformes autorisant la préparation de commandes.

$$\sum_{v \in \mathcal{S}_i} u_{o,v} \geq 1, \quad \forall i \in \mathcal{I}, \forall o \in O_i^P. \quad (5.13)$$

Dans ce cas, le domaine des variables $a_{o,r,v}^+$, $a_{o,r,v}^-$ et $u_{o,v}$ doit être binaire : si la valeur de la somme des variables $u_{o,v}$ est fixée à 1 (contrainte (5.13)) et que le domaine des variables $a_{o,r,v}^+$ et $u_{o,v}$ est réel, on trouve des exemples où la contrainte (5.13) est satisfaite mais où aucun transfert n'a lieu sur la plateforme v au niveau des variables $z_{o,r,v}$.

5.4.2 Itinéraire aval

Dans le réseau aval, les tournées de véhicules sont déjà définies en prétraitement et les transferts ne sont pas autorisés.

Les itinéraires des commandes dans le réseau aval sont modélisés par les variables binaires $x_{j,r}$. $x_{j,r}$ est égale à 1 si la commande j est distribuée par la tournée aval $r \in R^+$, 0 sinon.

On associe à chaque tournée un prix fixe c_k , et une liste de commandes qu'elle transporte. On trouve plusieurs tournées possibles pour une même commande j . On appelle $R_j \subset R^+$ l'ensemble des tournées qui livrent la commande j . C'est pourquoi l'optimisation du choix des tournées de véhicules aval peut être décrit comme un problème d'affectation des commandes dans un ensemble de tournées.

5.4.3 Itinéraires directs

Pour un itinéraire direct, les sous-commandes d'une commande sont distribuées par la même tournée. Toutes les sous-commandes d'une même commande $j \in J$ ont par définition la même destination d_j .

R_{d_j} représente ici l'ensemble des tournées directes pour livrer la sous-commande o au client de sa commande. Ainsi, on a :

$$z_{o,r,d_o} = z_{o',r,d_{o'}}, \quad \forall j \in J, \forall o, o' \in O^j, \forall r \in R_{d_o}. \quad (5.14)$$

5.4.4 Lien entre réseau amont et réseau aval

Si une commande $j \in J$ est transportée par la tournée aval $r \in R_{d_j}$, alors la destination de l'itinéraire amont des sous-commandes O^j et le départ de la tournée aval doivent coïncider au CRC du départ de r . Ceci est modélisé par les contraintes (5.15) :

$$\sum_{r \in R_{d_j} \cap R_v} x_{j,r} = \sum_{r \in R_v \cap R^-} z_{o,r,v}, \quad \forall v \in \mathcal{C}, \forall j \in J, \forall o \in O^j. \quad (5.15)$$

5.4.5 Satisfaction de la demande

Chaque commande est livrée à son client une seule fois, soit par un itinéraire direct, soit par un itinéraire aval depuis un CRC. Cette contrainte est modélisée par les contraintes suivantes (5.16) :

$$\sum_{r \in R_{d_j} \cap R^-} z_{o,r,d_o} + \sum_{r \in R_j} x_{j,r} = 1 \quad \forall j \in J, \forall o \in O^j \quad (5.16)$$

5.4.6 Modélisation des tournées

Le transport étant sous-traité, on considère que la flotte de véhicules à disposition est infinie. De plus, on ne vérifie que de manière globale la capacité des véhicules : on compte seulement combien de véhicules sont nécessaires pour chaque arc selon le nombre de palettes total qui y passent. Cela implique qu'une commande peut-être transportée par plusieurs véhicules réalisant la même tournée.

5.4.6.1 Tournées aval

Les tournées de véhicules aval ont un coût fixe par véhicule. La variable entière y_r représente le nombre de véhicules utilisés sur la tournée aval $r \in R^+$. On définit également l'ensemble J_r des commandes qui peuvent être transportées par la tournée aval $r \in R^+$, et Q la capacité des véhicules.

Les *contraintes de capacité* (5.17) sont utilisées pour déterminer la valeur des variables y_r .

$$\sum_{j \in J_r} q_j x_{j,r} \leq Q y_r \quad \forall r \in R^+ \quad (5.17)$$

Le membre gauche de l'équation représente la charge totale transportée sur la tournée $r \in R^+$. La partie droite représente la capacité totale des véhicules utilisés sur la tournée r .

5.4.6.2 Tournées amont et directes

Comme décrit dans la section 5.2.2.3, les tournées amont et directes sont énumérées dans une phase de pré-traitement. La structure de coût linéaire par morceaux est modélisée par les équations **Multiple Choice Model (MCM)** détaillées en section 3.2.2 du chapitre 3. Chaque tournée est associée à une fonction linéaire par morceaux. On note S_r l'ensemble des segments de la fonction linéaire par morceaux de la tournée $r \in R^-$. Pour chaque segment $s \in S_r$ de la fonction représentant le coût de la tournée r , on note a_s^r est son coefficient directeur, b_s^r son ordonnée à l'origine, et q_r^s sa charge maximale. On reprend ici les variables $\xi_{r,s}$ et $\lambda_{r,s}$ et les contraintes décrites au chapitre 3

$$\sum_{s \in S_r} \xi_{r,s} \leq 1 \quad \forall r \in R^-, \quad (5.18)$$

$$q_{r,s-1} \xi_{r,s} \leq \lambda_{r,s} \leq q_{r,s} \xi_{r,s} \quad \forall r \in R^-, \forall s \in S_r, \quad (5.19)$$

On rappelle que les contraintes (5.18) sélectionne un unique segment de charge de la fonction linéaire par morceaux et que les contraintes (5.19) connectent les variables $\xi_{r,s}$ et $\lambda_{r,s}$.

Si une tournée amont $r \in R^-$ visite plus de deux plateformes, son coût est calculé en prenant la charge maximale transportée durant sa tournée. Cela est exprimé par les contraintes (5.20) :

$$\sum_{s \in S_r} \lambda_{r,s} \geq \sum_{o \in O} q_o z_{o,r,v} \quad \forall r \in R^-, \forall v \in \mathcal{V}_r. \quad (5.20)$$

le coût des tournées amont et directes est :

$$\sum_{r \in R^-} \sum_{s \in S_r} (a_s^r \lambda_{r,s} + b_{s,r} \xi_{r,s}). \quad (5.21)$$

5.4.7 Fonction objectif

La fonction objectif (5.22) minimise la somme des coûts de transport et des coûts de transferts. Comme expliqué dans le chapitre 2, les coûts de production et de préparation de commandes sont constants et donc non considérés dans la fonction objectif.

$$\begin{aligned} \min \sum_{r \in R^+} c_r y_r + \sum_{r \in R^-} \sum_{s \in S_r} (a_s^r \lambda_{r,s} + b_{s,r} \xi_{r,s}) + \sum_{i \in \mathcal{I}} \sum_{o \in O_i} \sum_{v \in \mathcal{T}_i} \rho_v q_o u_{o,v} \\ + \sum_{j \in J} \sum_{r \in R_{d_j} \cap R_v} \sum_{v \in \mathcal{C}} \rho_v q_j x_{j,r}. \end{aligned} \quad (5.22)$$

Le premier terme représente la somme des coûts fixes des véhicules aval. Le second terme correspond au tarif **LTL** des véhicules des tournées amont et directes. $\sum_{i \in \mathcal{I}} \sum_{o \in O_i} \sum_{v \in \mathcal{T}_i} \rho_v q_o u_{o,v}$ est la somme des coûts de transfert sur les sites fournisseurs. Le dernier terme de la somme représente le coût de transfert de marchandises sur les CRC.

5.4.8 Deux modèles mathématiques

5.4.8.1 Modèle 1 - structure de transfert complète

Le modèle 1 est constitué de la fonction objectif (5.22) et des contraintes (5.1) à (5.5) et (5.14) à (5.20). Il utilise la structure de transfert complète.

$$\begin{aligned} \min \sum_{r \in R^+} c_r y_r + \sum_{r \in R^-} \sum_{s \in S_r} (a_{r,s} \lambda_{r,s} + b_{r,s} \xi_{r,s}) + \sum_{i \in \mathcal{I}} \sum_{o \in O_i} \sum_{r \in R^-} \sum_{r' \in R^-} \sum_{v \in \mathcal{V}_r \cup \mathcal{V}'_r} (\rho_v q_o u_{o,r,r',v})/2 \\ + \sum_{o \in J} \sum_{r \in R_{d_j} \cap R_v} \sum_{v \in \mathcal{C}} \rho_v q_j x_{j,r} \end{aligned} \quad (5.23)$$

s.t.

$$\sum_{r \in R_{p_o}} z_{o,r,p_o^+} = 1 \quad \forall o \in O \quad (5.24)$$

$$z_{o,r,d_o} = z_{o',r,d_o} \quad \forall j \in J, \forall o, o' \in O(j), \forall r \in R_{d_o} \quad (5.25)$$

$$z_{o,r,v^+} = z_{o,r,v} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{V} \setminus (\mathcal{T}_i \cup \mathcal{C} \cup \{d_o\}), \forall r \in R_v \quad (5.26)$$

$$\sum_{r \in R_{d_j} \cap R_v} x_{j,r} = \sum_{r \in R_v} z_{o,r,v} \quad \forall v \in \mathcal{C}, \forall j \in J, \forall o \in O(j) \quad (5.27)$$

$$\sum_{r \in R_{d_o}} z_{o,r,d_o} + \sum_{r \in R_{d_j}} x_{j,r} = 1 \quad \forall j \in J, \forall o \in O(j) \quad (5.16)$$

$$z_{o,r,v^+} - z_{o,r,v} = \sum_{\substack{r' \in R_v \\ r \neq r'}} (u_{o,r',r,v} - u_{o,r,r',v}) \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.28)$$

$$\sum_{v \in \mathcal{S}_i} \sum_{r \in R_v} \sum_{\substack{r' \in R_v \\ r' \neq r}} u_{o,r,r',v} \geq 1 \quad \forall i \in \mathcal{I}, \forall o \in O_i^P \quad (5.29)$$

$$z_{o,r,v} \geq \sum_{\substack{r' \in R_v \\ r \neq r'}} u_{o,r,r',v} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.30)$$

$$\sum_{r \in R_v} (z_{v,o,r} - z_{o,r,v^+}) = 0 \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\} \quad (5.31)$$

$$\sum_{j \in J_r} q_j x_{j,r} \leq Q y_r \quad \forall r \in R^+ \quad (5.32)$$

$$\sum_{s \in S_r} \xi_{r,s} \leq 1 \quad \forall r \in R^- \quad (5.33)$$

$$q_{r,s-1} \xi_{r,s} \leq \lambda_{r,s} \leq q_{r,s} \xi_{r,s} \quad \forall r \in R^-, \forall s \in S_r \quad (5.34)$$

$$\sum_{s \in S_r} \lambda_{r,s} \geq \sum_{o \in O} q_o z_{o,r,v} \quad \forall r \in R^-, \forall v \in \mathcal{V}_r \quad (5.35)$$

$$z_{o,r,v} \in \{0; 1\} \quad \forall v \in \mathcal{V}, \forall o \in O, \forall r \in R_v \quad (5.36)$$

$$u_{o,r,r',v} \in \{0; 1\} \quad \forall o \in O, \forall i \in \mathcal{I}, \forall v \in \mathcal{C} \cup \mathcal{T}_i \quad (5.37)$$

$$x_{j,r} \in \{0; 1\} \quad \forall j \in J, \forall r \in R_{d_j} \quad (5.38)$$

$$y_r \in \mathbb{R} \quad \forall r \in R^+ \quad (5.39)$$

$$\xi_{r,s} \in \{0; 1\} \quad \forall r \in R^-, \forall s \in S_r \quad (5.40)$$

$$\lambda_{r,s} \in \mathbb{R}^+ \quad \forall r \in R^-, \forall s \in S_r \quad (5.41)$$

5.4.8.2 Modèle 2 - structure de transfert en étoile

Le modèle 2 est constitué de la fonction objectif (5.22) et des contraintes (5.1) à (5.2) et (5.6) à (5.20). Il utilise la structure de transfert en étoile.

$$\begin{aligned} \min \sum_{r \in R^+} c_r y_r + \sum_{r \in R^-} \sum_{s \in S_r} (a_{r,s} \lambda_{r,s} + b_{r,s} \xi_{r,s}) + \sum_{i \in \mathcal{I}} \sum_{o \in O_i} \sum_{v \in \mathcal{T}_i} \rho_v q_o u_{o,v} \\ + \sum_{o \in J} \sum_{r \in R_{d_j} \cap R_v} \sum_{v \in \mathcal{C}} \rho_v q_j x_{j,r} \end{aligned} \quad (5.42)$$

s.t.

$$\sum_{r \in R_{p_o}} z_{o,r,p_o^+} = 1 \quad \forall o \in O \quad (5.43)$$

$$z_{o,r,d_o} = z_{o',r,d_o} \quad \forall j \in J, \forall o, o' \in O(j), \forall r \in R_{d_o} \quad (5.44)$$

$$z_{o,r,v^+} = z_{o,r,v} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{V} \setminus (\mathcal{T}_i \cup \mathcal{C} \cup \{d_o\}), \forall r \in R_v \quad (5.45)$$

$$\sum_{r \in R_{d_j} \cap R_v} x_{j,r} = \sum_{r \in R_v} z_{o,r,v} \quad \forall v \in \mathcal{C}, \forall j \in J, \forall o \in O(j) \quad (5.46)$$

$$\sum_{r \in R_{d_o}} z_{o,r,d_o} + \sum_{r \in R_{d_j}} x_{j,r} = 1 \quad \forall j \in J, \forall o \in O(j) \quad (5.47)$$

$$z_{o,r,v} - z_{o,r,v^+} = a_{o,r,v}^+ - a_{o,r,v}^- \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.48)$$

$$a_{o,r,v}^+ \geq z_{o,r,v} - z_{o,r,v^+} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.49)$$

$$a_{o,r,v}^- \geq -z_{o,r,v} + z_{o,r,v^+} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.50)$$

$$a_{o,r,v}^+ + a_{o,r,v}^- \leq 1 \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.51)$$

$$\sum_{r \in R_v} (a_{o,r,v}^+ + a_{o,r,v}^-) = 2 u_{o,v} \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i, \forall r \in R_v \quad (5.52)$$

$$\sum_{r \in R_v} (a_{o,r,v}^+ - a_{o,r,v}^-) = 0 \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\}. \quad (5.53)$$

$$\sum_{v \in \mathcal{L}_i} u_{o,v} \geq 1 \quad \forall i \in \mathcal{I}, \forall o \in O_i^P \quad (5.54)$$

$$\sum_{r \in R_v} (z_{v,o,r} - z_{o,r,v^+}) = 0 \quad \forall i \in \mathcal{I}, \forall o \in O_i, \forall v \in \mathcal{T}_i \setminus \{p_o\} \quad (5.55)$$

$$\sum_{j \in J_r} q_j x_{j,r} \leq Q y_r \quad \forall r \in R^+ \quad (5.56)$$

$$\sum_{s \in S_r} \xi_{r,s} \leq 1 \quad \forall r \in R^- \quad (5.57)$$

$$q_{r,s-1} \xi_{r,s} \leq \lambda_{r,s} \leq q_{r,s} \xi_{r,s} \quad \forall r \in R^-, \forall s \in S_r \quad (5.58)$$

$$\sum_{s \in S_r} \lambda_{r,s} \geq \sum_{o \in O} q_o z_{o,r,v} \quad \forall r \in R^-, \forall v \in \mathcal{V}_r \quad (5.59)$$

$$z_{o,r,v} \in \{0; 1\} \quad \forall v \in \mathcal{V}, \forall o \in O, \forall r \in R_v \quad (5.60)$$

$$u_{o,v} \in \{0; 1\} \quad \forall o \in O, \forall i \in I, \forall v \in \mathcal{C} \cup \mathcal{T}_i \quad (5.61)$$

$$x_{j,r} \in \{0; 1\} \quad \forall j \in J, \forall r \in R_{d_j} \quad (5.62)$$

$$y_r \in \mathbb{R} \quad \forall r \in R^+ \quad (5.63)$$

$$\xi_{r,s} \in \{0; 1\} \quad \forall r \in R^-, \forall s \in S_r \quad (5.64)$$

$$\lambda_{r,s} \in \mathbb{R}^+ \quad \forall r \in R^-, \forall s \in S_r \quad (5.65)$$

5.5 Tests numériques

Nous avons conduit des tests numériques afin d'évaluer la pertinence de la mutualisation des transports grâce aux CRC. Une étape intermédiaire était de choisir la meilleure modélisation des transferts (structure complète ou en étoile). La section 5.5.1 décrit tout d'abord la génération des instances utilisées pour ces tests. La section 5.5.2.1 compare les deux modélisations. Dans la section 5.5.2.2, nous analysons les résultats numériques et évaluons la pertinence de la mutualisation.

5.5.1 Instances

Un ensemble d'instances basées sur un réseau logistique réel a été généré.

5.5.1.1 Réseau logistique

Tous les sites logistiques sont basés en France et en Belgique : les sites des fournisseurs sont situés dans les zones denses de l'industrie française et les CRC sont localisés dans les villes où 4S Network souhaite les implanter. Les positions des clients proviennent de la localisation d'entrepôts et/ou magasins de la grande distribution. On crée trois types de réseau :

- type de réseau A : centre de la France avec un CRC ;
- type de réseau B : nord de la France avec trois CRC ;
- type de réseau C : centre et sud de la France avec quatre CRC.

Pour chaque type de réseau, on évalue la performance des modèles et du réseau logistique mutualisé avec 6, 8 et 10 sites de fournisseurs. La figure 5.11 illustre le réseau A avec 10 sites de fournisseurs dont 8 sont éloignés des clients. La figure 5.12 illustre le réseau B avec 10 sites de fournisseurs dont 3 sont éloignés des clients. La figure 5.13 illustre le réseau C avec 10 sites de fournisseurs dont 2 sont proches des clients.

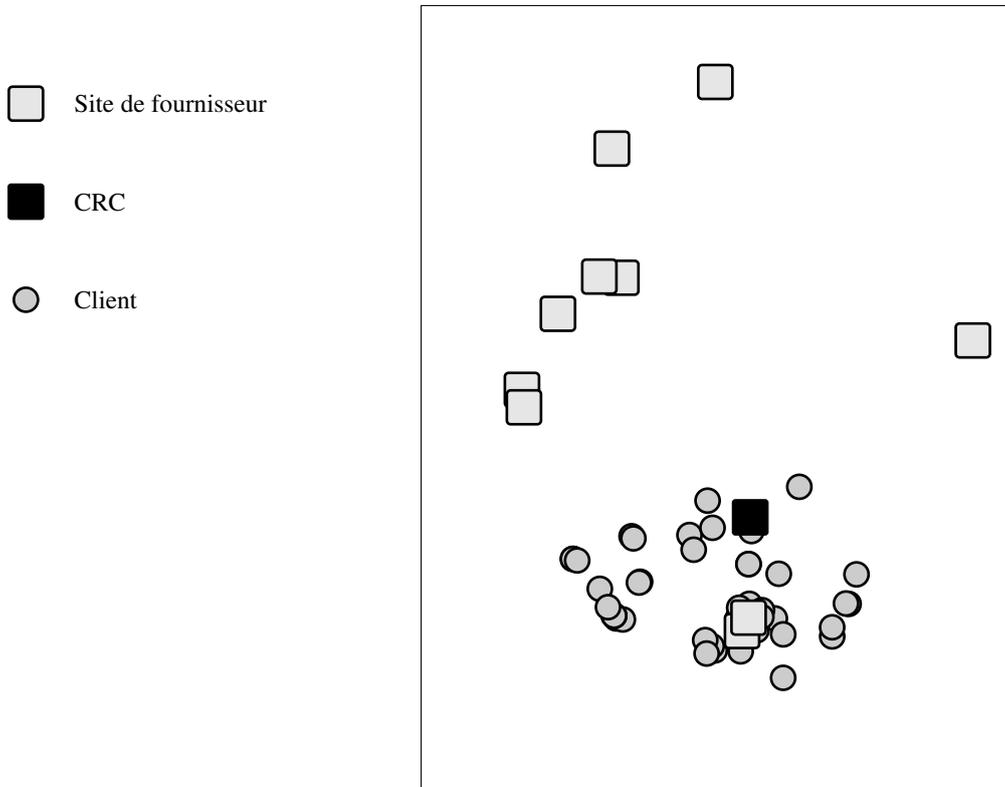


Figure 5.11 – Réseau de type A avec 10 sites de fournisseurs, un CRC et 50 clients

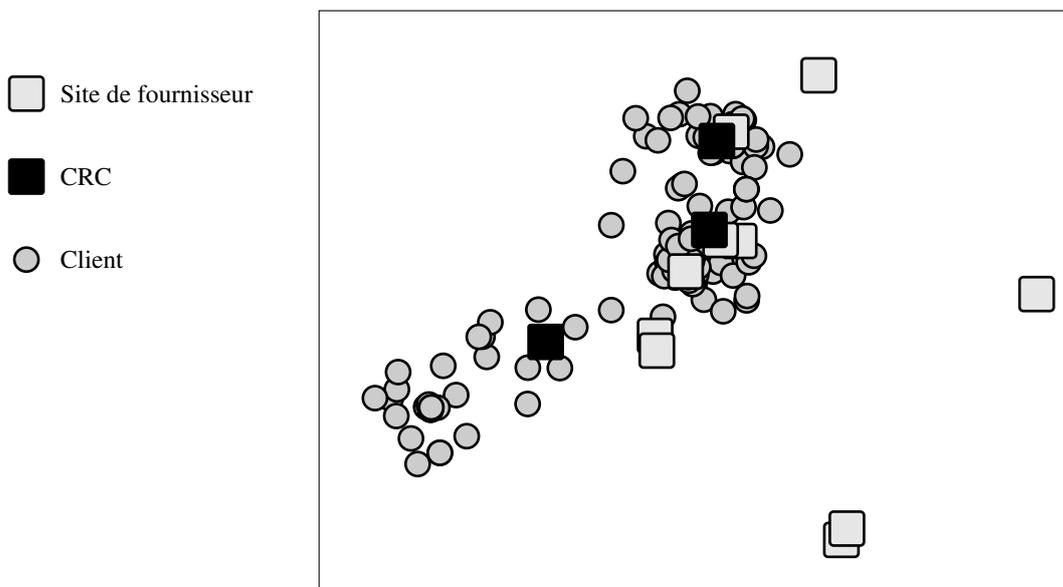


Figure 5.12 – Réseau de type B avec 10 sites de fournisseurs, 3 CRC et 130 clients

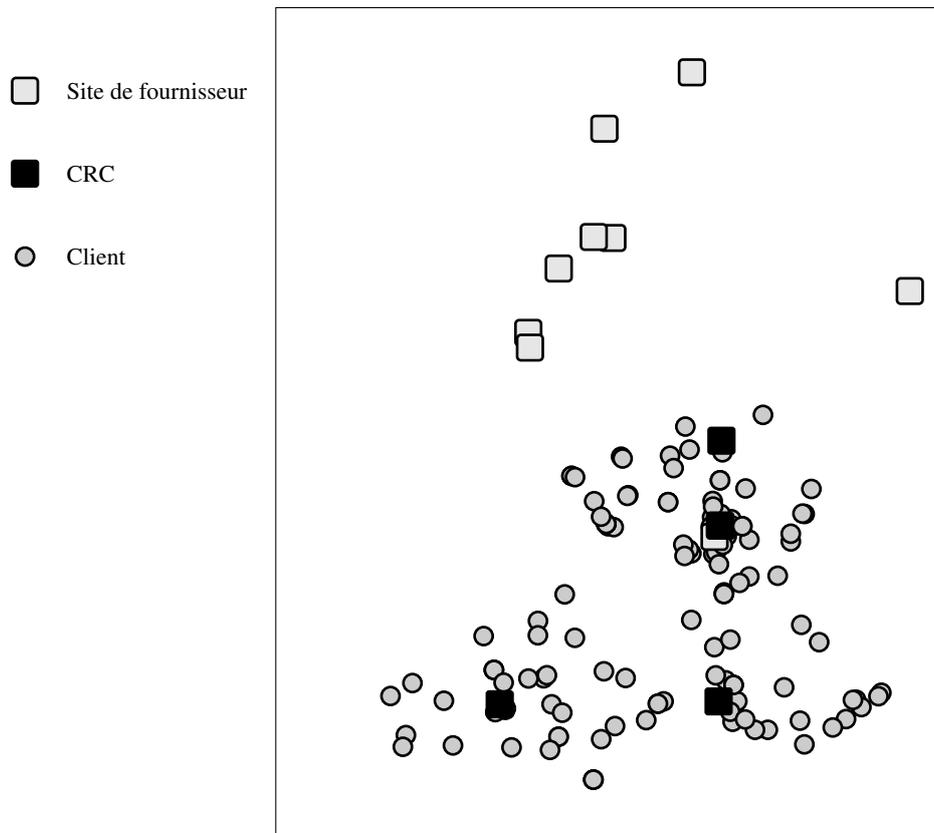


Figure 5.13 – Réseau de type C avec 10 sites de fournisseurs, 4 CRC et 130 clients

5.5.1.2 Facteurs à tester

On souhaite évaluer les performances du réseau mutualisé sur plusieurs plans :

- le nombre de clients présents dans le réseau mutualisé.
- la distance entre les sites fournisseurs et les clients ;
- la distance de couverture δ des CRC ;
- le nombre de palettes par commande.

Le nombre de sous-commandes par commande varie entre 1 et 4 : Les produits d'une commande proviennent d'un ou deux sites de fournisseurs (à probabilité égale), et pour chaque site il y a 50% de probabilité qu'une partie des produits aient besoin de préparation (créant donc deux sous-commandes pour les produits d'un même site d'origine).

5.5.1.3 Instances générées

Afin de pouvoir comparer les facteurs listés précédemment, on génère des instances comprenant un nombre différent de clients : les instances correspondant au réseau de type A comprennent toujours 50 clients, et celles pour le réseau de type B et C comprennent 100 ou 130 clients. Pour les trois types de réseaux, tous les clients peuvent être livrés par au moins un CRC. Les principales différences entre réseaux sont liées aux distances entre les sites fournisseurs et les clients. On compare les résultats des trois configurations de réseau afin de savoir si la distance entre les sites fournisseurs et les clients jouent un rôle important dans les gains apportés par l'utilisation d'un réseau mutualisé. On évalue également chaque instance avec trois couvertures δ de livraison des CRC : $\delta = 150km$, $\delta = 300km$ et $\delta = 400km$. Afin d'analyser ces résultats, le tableau 5.1 montre le nombre de clients qui peuvent être livrés par au moins un, deux, trois, ou quatre CRC respectivement à la couverture choisie. A_50 correspond au réseau A avec 50 clients, B_100 et B_130 correspondent au réseau B avec respectivement 100 et 130 clients, et C_100 et C_130 correspondent au réseau C avec respectivement 100 et 130 clients.

Afin d'observer la performance de la collaboration en fonction de la taille de la commande, trois catégories de fournisseurs sont créées en fonction de leur taille moyenne de commande. On associe une taille de commande à un fournisseur et non à un client car les commandes sont généralement de même taille pour un fournisseur d'un type de produit : par exemple, un fournisseur d'éponges doit livrer des commandes de petite taille, alors qu'un fournisseur de bouteilles d'eau livre des commandes de tailles beaucoup plus importantes. On définit donc un type 1 (petit) de fournisseurs ayant une moyenne de taille de commande de 2 palettes, un type 2 (moyen) de taille de fournisseurs avec une moyenne de 5 palettes par commandes, et un type 3 (grand) avec une moyenne de 9 palettes. Afin de mettre en corrélation la performance de la collaboration et la taille des commandes, on définit 7 types de configurations pour l'association des fournisseurs :

- configuration 1 : petites, moyennes et grande taille de commandes
- configuration 2 : moyennes et grande taille de commandes
- configuration 3 : petites et grande taille de commandes

Livré par # CRC	Couverture δ	A_50	B_100	B_130	C_100	C_130
1 CRC	150km	48	97	127	100	129
	300km	50	100	130	100	130
	450km	50	100	130	100	130
2 CRC	150km	-	71	97	63	78
	300km	-	95	125	100	130
	450km	-	100	130	100	130
3 CRC	150km	-	24	31	5	5
	300km	-	79	109	82	106
	450km	-	100	130	100	130
4 CRC	150km	-	-	-	1	1
	300km	-	-	-	41	55
	450km	-	-	-	99	129

Tableau 5.1 – Nombre de clients qui peuvent être livrés par 1 à 4 CRC

- configuration 4 : petites et moyennes tailles de commandes
- configuration 5 : petite taille de commandes
- configuration 6 : grande taille de commandes
- configuration 7 : moyenne taille de commandes

Pour chaque réseau logistique (3 réseaux pour le type A et 6 réseaux chacun pour les types B et C) et chaque configuration (7) on génère une instance. Ainsi, 7 instances représentant les 7 configurations possibles sont créées pour chaque réseau, soit $(3 + 6 + 6) \times 7 = 105$ au total. Chaque instance est résolue une fois pour chaque couverture δ des CRC. Cela donne donc 315 tests à effectuer, avec 21 configurations différentes (7 types de demandes et 3 couvertures) pour chaque type de réseau. Ces caractéristiques sont résumées dans le tableau 5.2.

5.5.2 Résultats

Chaque modèle est résolu par Cplex avec une limite de temps de 2h. Les modèles sont codés en Java et exécutés sur une machine Xeon X5650 à 2.67 GHz avec 64 GB de RAM. Un commencement à *chaud* est ajouté pour chaque modèle : On résout tout d'abord un premier modèle où les commandes ne sont pas autorisées à passer par le réseau mutualisé. Cette résolution est très rapide (moins d'une seconde pour toutes les instances). Les valeurs des variables sont ensuite injectées dans le modèle autorisant le passage par le réseau mutualisé. Cela donne une première solution au solveur et permet ainsi de gagner un peu de temps pour la résolution du problème.

Type de réseau	# site de fournisseur	# client	instance
A	6	50	1-7
	8	50	36-42
	10	50	71-77
B	6	100	8-14
	6	130	15-21
	8	100	43-49
	8	130	50-56
	10	100	78-84
	10	130	85-91
C	6	100	22-28
	6	130	29-35
	8	100	57-63
	8	130	64-70
	10	100	92-98
	10	130	99-104

Tableau 5.2 – Caractéristiques des instances

5.5.2.1 Comparaison des modélisations des transferts

Avant d'évaluer l'impact de la mutualisation sur le coût du transport, on souhaite observer quel modèle est le plus performant. Le tableau 5.3 résume la comparaison entre ces deux modélisations. Il est très clair que la *structure de transfert en étoile* est beaucoup plus efficace que la *structure de transfert complète*. En effet, la *structure de transfert complète* ne permet pas de trouver de solutions réalisables pour 217 sur 315 instances, car la mémoire est saturée par un trop grand nombre de variables. La structure de transfert en étoile est presque toujours en mesure de fournir une solution réalisable. Cependant, le nombre d'instances résolues à l'optimum décroît lorsque la taille de l'instance augmente (83/315 résolues à l'optimum).

5.5.2.2 Évaluation du réseau collaboratif

Les instances sont lancées avec deux options différentes pour l'itinéraire des commandes :

- Une *première option* permet d'utiliser uniquement le réseau direct. Les commandes ne transitent donc pas sur des CRC.
- Une *seconde option* permet aux commandes de passer par les deux réseaux, direct ou collaboratif.

Le tableau 5.4 montre les économies réalisées par la mise en place d'un réseau collaboratif. Les résultats sont présentés de manière agrégée pour chaque couverture δ de livraison des CRC et pour chaque type de réseau A, B ou C. On observe que la distance de couverture a un fort impact sur le coût. En outre, dans le réseau C, les sites de fournisseurs sont plus lointains des clients que pour les autres réseaux. Dans ce cas là, la collaboration a un impact fort sur l'économie réalisée.

Type de réseau	# site fournisseur	# client	Instances résolues à l'optimum #/21		Instance avec une solution réalisable trouvée #/21 (écart moyen avec la borne inférieure de Cplex)	
			Structure complète	Structure en étoile	Structure complète	Structure en étoile
A	6	50	14	15	21 (0.80%)	21 (0.78%)
	8	50	9	9	21 (3.20%)	21 (3.28%)
	10	50	9	9	21 (3.19%)	21 (3.20%)
B	6	100	8	10	16 (3.82%)	21 (3.12%)
	6	130	0	6	1 (3.10%)	21 (4.59%)
	8	100	0	4	0	21 (4.86%)
	8	130	0	2	0	21 (6.63%)
	10	100	0	4	0	21 (5.76%)
	10	130	0	2	0	19 (5.85%)
C	6	100	6	9	18 (4.11%)	21 (4.36%)
	6	130	0	2	0	21 (8.16%)
	8	100	0	6	0	21 (8.07%)
	8	130	0	1	0	21 (10.58%)
	10	100	0	4	0	21 (8.79%)
	10	130	0	0	0	21 (11.46%)

Tableau 5.3 – Comparaison des structures de transferts

Impact de la couverture δ Le tableau 5.1 montre que presque tous les clients peuvent être livrés par au moins un CRC dans les trois cas de couverture δ utilisés. C'est pourquoi la différence de coûts observée montre qu'un fonctionnement où plusieurs CRC peuvent livrer un même client peut générer de bien meilleures économies que pour un fonctionnement plus rigide. Cependant, attribuer un unique CRC à chaque client est beaucoup plus simple pour la gestion opérationnelle du réseau collaboratif.

Impact du type de réseau Les économies varient beaucoup selon le réseau utilisé. Les différents résultats entre le réseau de type B et C montrent que les économies sont particulièrement réalisées sur le réseau amont au CRC, car un fournisseur peut consolider ses commandes à destination d'une même région et les envoyer vers un unique CRC, au lieu de devoir les envoyer séparément.

δ	Réseau A	Réseau B	Réseau C	Moyenne
150km	-1,7%	-0,5%	-8,9%	-3,7%
300km	-4,1%	-2,4%	-9,6%	-5,4%
450km	-4,3%	-4,4%	-13,4%	-7,4%
Moyenne	-3,4%	-2,4%	-10,6%	

Tableau 5.4 – Economies selon la couverture et le type de réseau

Impact de la taille des commandes Le tableau 5.5 représente les économies réalisées selon la configuration des tailles de commandes de l'instance. G, M et P indique respectivement des grandes, moyennes et petite taille de commandes. On observe généralement que les petite taille de commandes donnent lieu à de meilleurs résultats, avec une moyenne à 7.7% d'économie pour ce type de collaboration. Le type de réseau semble influencer plus fortement sur l'économie réalisée que la taille de commandes.

	Réseau A	Réseau B	Réseau C	Moyenne
G	-0,9%	-3,2%	-10,2%	-4,8%
G M	-3,7%	-3,2%	-9,9%	-5,6%
G P	-1,8%	-2,4%	-10,1%	-4,8%
G M P	-2,8%	-1,8%	-9,0%	-4,5%
M P	-3,9%	-2,5%	-9,0%	-5,1%
M	-5,8%	-2,4%	-11,8%	-6,7%
P	-4,7%	-1,3%	-17,2%	-7,7%

Tableau 5.5 – Économies réalisées par rapport à un fonctionnement sans réseau mutualisé selon la taille moyenne des commandes

5.6 Conclusion

Dans ce chapitre, on propose une première modélisation du problème de plan de chargement dans le réseau logistique mutualisé et direct. Ce modèle permet la comparaison des coûts entre le réseau logistique *direct* et le réseau logistique *mutualisé*. Des instances tirées de données réelles ont été générées afin de pouvoir évaluer les performances des modèles et comparer les coûts des réseaux logistiques. Les économies moyennes réalisées varient entre 3.4% et 10.6% selon le type de réseau logistique et entre 4.5% et 7.7% selon les tailles de commandes en jeu.

Ce chapitre compare également deux modélisations des transferts et montre l'importance d'une modélisation de transferts performante dans la modélisation d'un problème de conception de réseau.

Dans l'entreprise, ce modèle a beaucoup été utilisé au début du projet afin de convaincre les premiers fournisseurs d'utiliser le service CRC que propose 4S Network. Une interface permettant l'utilisation de ce modèle a également été développée et est présentée dans le chapitre 9.

Distribution depuis les CRC : un problème riche de tournées de véhicules

Sommaire

6.1	Description du problème	72
6.1.1	Commandes	72
6.1.2	Modélisation : un sommet pour chaque commande	73
6.1.3	Transport	73
6.1.4	Définition du problème traité	76
6.2	La génération de colonnes pour les problèmes riches de tournées de véhicules	77
6.3	Méthode de résolution	79
6.3.1	Modélisation du problème maître	79
6.3.2	Ensemble de colonnes initiales	81
6.3.3	Résolution des sous-problèmes	81
6.3.4	Tests de réalisabilité d'une insertion	86
6.4	Tests numériques	88
6.4.1	Instances	88
6.4.2	Paramétrage	91
6.4.3	Résultats	92
6.4.4	Analyse des résultats selon le pourcentage d'horaires fixés	94
6.5	Conclusion et perspectives	96

Ce chapitre présente la résolution du problème opérationnel de transport depuis un CRC. L'objectif est de déterminer les tournées de véhicules qui transportent les commandes présentes sur le CRC. Pour réaliser ces tournées de distribution, 4S Network affrète des véhicules à différents transporteurs.

L'algorithme développé a été intégré à un outil d'optimisation utilisé quotidiennement par les personnes assurant la gestion opérationnelle des CRC. Pour être utilisé quotidiennement, cet outil doit répondre à deux exigences : tenir compte de nombreuses contraintes métier des transporteurs et obtenir de bons résultats en temps de calcul limité. La prise en compte des contraintes métier implique de résoudre un problème riche de tournées de véhicules. Pour limiter le temps d'exécution, nous proposons une méthode heuristique basée sur la génération de colonnes dont le sous-problème est résolu grâce à une recherche tabou. L'ensemble des tournées trouvées à chaque itération de la génération de colonnes est ajouté dans un PLNE résolu par un solveur.

6.1 Description du problème

Nous nous positionnons dans le réseau aval de distribution régionale défini par un CRC et l'ensemble de client qu'il peut livrer. Un ensemble de commandes doit être livré aux clients dans des fenêtres horaires spécifiques. Ce problème de tournées de véhicules intègre une flotte hétérogène et des tournées multi-trip définis en section 6.1.3 ainsi que des contraintes de compatibilité entre produits présentées en section 6.1.1. Enfin, chaque tournée est contrainte par une limite sur le nombre d'arrêts, la durée de trajet, le temps d'attente et le détour.

6.1.1 Commandes

Une commande $o \in O$ est définie par une quantité q_o de palettes, un poids w_o , une destination d_o ainsi qu'une fenêtre horaire $[e_o, l_o]$, qui peut être de deux types :

- une heure fixe de livraison imposée ; dans ce cas, $e_o = l_o$;
- une plage horaire correspondant à l'ouverture du site client pour la journée de livraison souhaitée.

Les produits à livrer peuvent appartenir à différentes catégories :

- *Alimentaire sec*, correspondant à des produits alimentaires ne nécessitant pas une conservation à température contrôlée (par exemple des pâtes/riz).
- *Alimentaire 15°C*, correspondant à des produits alimentaires nécessitant une conservation à température contrôlée à 15°C (par exemple des chocolats).
- *Droguerie parfumerie hygiène (DPH)*, correspondant à des produits de droguerie, parfumerie et hygiène qui ne dégagent pas d'odeurs particulières (par exemple des éponges).
- *Droguerie parfumerie hygiène odorant (DPH odorant)*, correspondant à des produits de droguerie, parfumerie et hygiène qui dégagent des odeurs particulières (par exemple des lessives).

Par définition, une commande comporte des produits du même type. Tous les types de produits ne peuvent pas être transportés/chargés simultanément dans la même tournée. Le tableau 6.1 montre les compatibilités de transport entre les types de produits. Par exemple une commande de type *Alimentaire sec* peut-être transportée avec des produits du même type, de type *Alimentaire 15°C* ou *DHP*. Elle ne peut pas être transportée avec des produits de type *DPH odorant*.

	Alimentaire sec	Alimentaire 15°C	DPH	DPH odorant
Alimentaire sec	✓	✓	✓	✗
Alimentaire 15°C	✓	✓	✗	✗
DPH	✓	✗	✓	✓
DPH odorant	✗	✗	✓	✓

Tableau 6.1 – Compatibilités de transport entre types de produits

6.1.2 Modélisation : un sommet pour chaque commande

Il est possible qu'un client reçoive plusieurs commandes par jour, provenant de différents fournisseurs. En revanche, ses commandes ne doivent pas obligatoirement être transportées par le même véhicule. Dans le réseau que l'on modélise, on souhaite que la tournée du véhicule représente le chargement de ce dernier. On considère donc un réseau comportant un sommet CRC et un sommet pour chaque commande. Deux commandes destinées au même client sont liées par une arête de distance et coût nuls, comme illustré par l'exemple 11.

Exemple 11. La figure 6.1 montre un exemple de graphe utilisé dans ce chapitre. Il comporte trois clients, et quatre commandes. On observe 5 sommets dans le réseau modélisant le CRC et les quatre commandes :

- une commande o_1 composée de 16 palettes et pesant 3 tonnes doit être livrée entre 12h et 18h ;
- une commande o_2 composée de 3 palettes, pesant 0.7 tonnes et doit être livrée à 15h ;
- une commande o_3 composée de 4 palettes, pesant 1 tonne et doit être livrée entre 9h et 16h ;
- une commande o_4 composée de 12 palettes, pesant 15 tonnes et doit être livrée entre 9h et 16h.

Notons que o_3 et o_4 sont à destination du même client.

6.1.3 Transport

Le transport est réalisé par les véhicules appartenant à différents transporteurs. Chaque transporteur possède une flotte hétérogène. Nous décrivons en sections 6.1.3.1 et 6.1.3.2 les types de véhicules et les règles d'affrètement. Chaque type de véhicule possède sa propre structure tarifaire (section 6.1.3.3) et ses propres contraintes sur les tournées qu'il réalise (section 6.1.3.4).

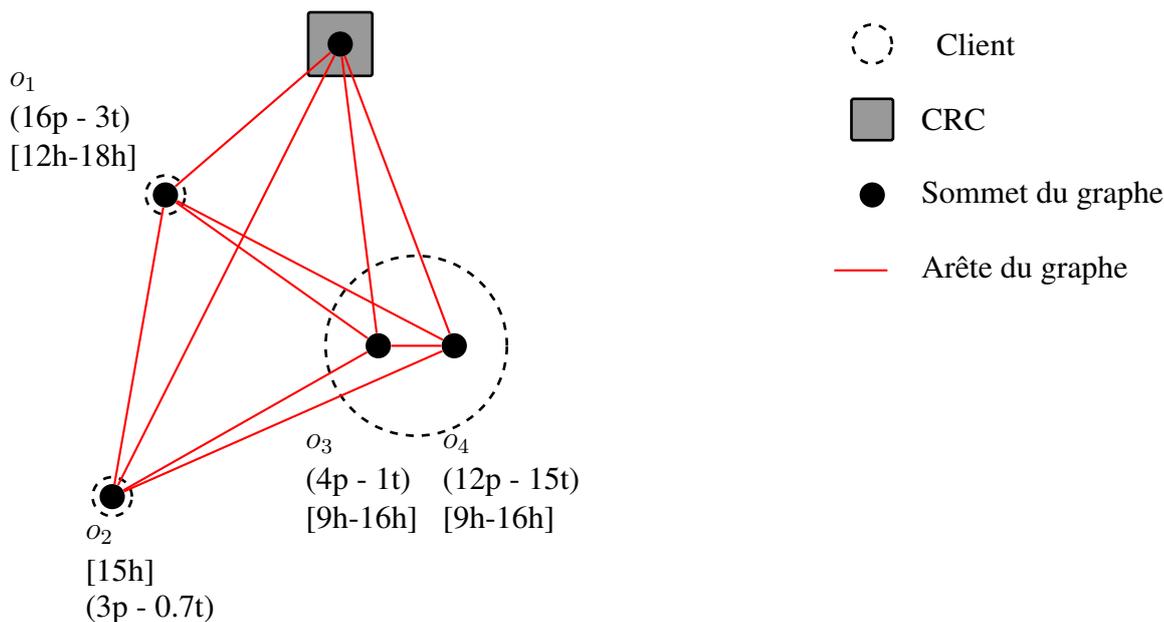


Figure 6.1 – Graphe utilisé dans la résolution du problème

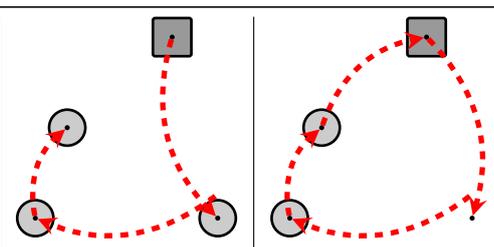
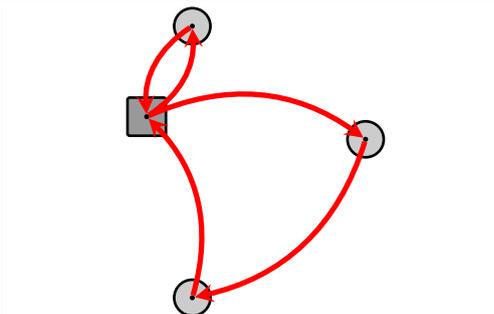
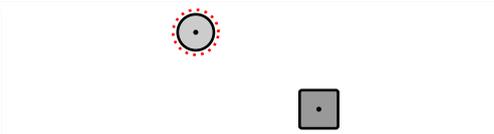
6.1.3.1 Types de véhicules

Soit K l'ensemble des types de véhicules. On différencie tout d'abord les types de véhicules par leur système de contrôle de la température (présent ou non). Si un véhicule n'en possède pas, il ne peut pas transporter de produits de type *alimentaire 15°C*. On définit alors T_k l'ensemble des types de produits que le type de véhicules $k \in K$ peut transporter. La capacité d'un véhicule se mesure par une limitation du nombre de palettes et une limitation de poids. Les notations Q_k et W_k désignent respectivement les capacités en palettes et en poids associées au type de véhicule k . On retrouve généralement deux tailles de véhicules : les *tractions* d'une capacité de 33 palettes et 44 tonnes, et les *porteurs*, limités à 16 palettes ou 19 tonnes. Le nombre de véhicules disponibles de type k est noté M_k .

6.1.3.2 Fonctionnement de la sous-traitance/affrètement

Les transporteurs ne proposent pas tous les mêmes services de transport. Par exemple, certains autorisent la *location* d'un véhicule et d'un chauffeur à la journée. D'autres peuvent proposer de prendre en charge le transport d'une commande sans regard sur la tournée réalisée. On présente ici les trois modes de fonctionnement des transporteurs avec lesquels 4S Network travaille.

- Un transporteur peut proposer l'utilisation d'un véhicule pour une seule tournée dans la journée. Notons qu'une tournée peut être *ouverte* – c'est-à-dire ne pas revenir au CRC à la fin de la tournée – ou *fermée*, ce qui implique de revenir au CRC. Il est important de différencier ces deux types de tournée afin d'en calculer le prix. On appelle ce type de fonctionnement un *affrètement à la tournée*.
- Un transporteur peut facturer l'utilisation de son véhicule pour une journée. On devra alors considérer une structure de tournées *avec départs multiples* (ou *multi-trip*), c'est-à-dire que le véhicule

Type	Exemple	Légende
<i>à la tournée</i>		 Sous-traitance à la tournée
<i>à la journée</i>		 Sous-traitance à la journée
<i>à la commande</i>		 Transport à la commande

 Clients  CRC

Tableau 6.2 – Types de sous-traitance

peut réaliser plusieurs tournées dans la journée. On appelle ce type de fonctionnement un affrètement *à la journée*.

- Une commande peut être confiée directement à un transporteur, sans regard sur la tournée que le véhicule va effectuer. On appelle ce type de fonctionnement une sous-traitance *à la commande*.

Le tableau 6.2 récapitule les types de sous-traitance.

6.1.3.3 Types de coûts

Afin de pouvoir modéliser toutes sortes de contrats de sous-traitance et d'affrètement des véhicules, 7 types de coûts ont été identifiés dans les contrats entre les transporteurs et 4S Network. Plusieurs de ces coûts peuvent être combinés pour définir le tarif d'une tournée.

- un coût fixe à l'utilisation (journée ou tournée) ;
- un coût variable au km où la distance parcourue est multiplié par un coût au kilomètre.
- un coût par zone : le transporteur définit plusieurs zones de livraison de manière concentrique autour des CRC. Le tarif retenu est celui correspondant à la zone la plus chère traversée par la tournée.

- un coût au kilomètre, variable selon la zone. Un coût au kilomètre est associé à chaque zone. Le coût au kilomètre de la zone la plus chère est multiplié par la distance parcourue par la tournée.
- un coût unitaire d'ouverture de portes. Ce coût unitaire peut-être différent selon le nombre d'ouvertures de portes réalisées par la tournée. Le nombre d'ouvertures de portes est multiplié par son coût unitaire afin d'obtenir le coût global d'ouverture de portes.

Dans le cas où la commande est directement confiée à un transporteur sans regard sur la tournée du véhicule, le coût dépend du nombre de palettes confiées et de la destination. Dans ce cas, le transporteur fournit une matrice de coût de transport qui indique le coût à la palette par destination.

Exemple 12. *Un véhicule visite quatre clients et parcourt 100km. Le prix de la tournée comprend un coût fixe de 80 euros et un coût au kilomètre par zone égal à 0.7 euros pour la zone traversée la plus chère. La première ouverture de porte est gratuite, la seconde coûte 30 euros et les suivantes 45 euros.*

Le calcul du prix de la tournée revient à additionner le coût fixe de 80 euros, le coût au kilomètre par zone ($0.7 \times 100 = 70$) et les coûts d'ouverture de porte ($30 + 45 \times 2 = 120$). Le prix de la tournée est donc de $80 + 70 + 120 = 270$ euros.

6.1.3.4 Contraintes sur la structure de la tournée

Chaque transporteur décide des contraintes qu'il souhaite appliquer à chacun de ses véhicules. Chaque véhicule possède alors son propre ensemble de contraintes, déterminé par les paramètres suivants :

- le nombre maximal d'arrêts dans la tournée ;
- la durée maximale d'utilisation du véhicule ;
- le temps maximal d'attente qui correspond à la somme de tous les temps d'attente avant déchargement du véhicule sur chacun de ses points livrés ;
- le détour maximal de la tournée, égal à la différence entre la longueur de la tournée et la longueur du plus court chemin entre le CRC et le client le plus éloigné du CRC.

Dans les calculs de durée, on intègre le temps de chargement au CRC et les temps de déchargement chez chaque client. Pour chaque déchargement, on considère un temps fixe Ξ^f auquel s'additionne un temps de déchargement par palette Ξ^v . Le temps de chargement au CRC est également pris en compte dans la durée totale de la tournée. Il correspond au même temps fixe de déchargement additionné au temps variable de déchargement pour l'ensemble de la charge du véhicule.

6.1.4 Définition du problème traité

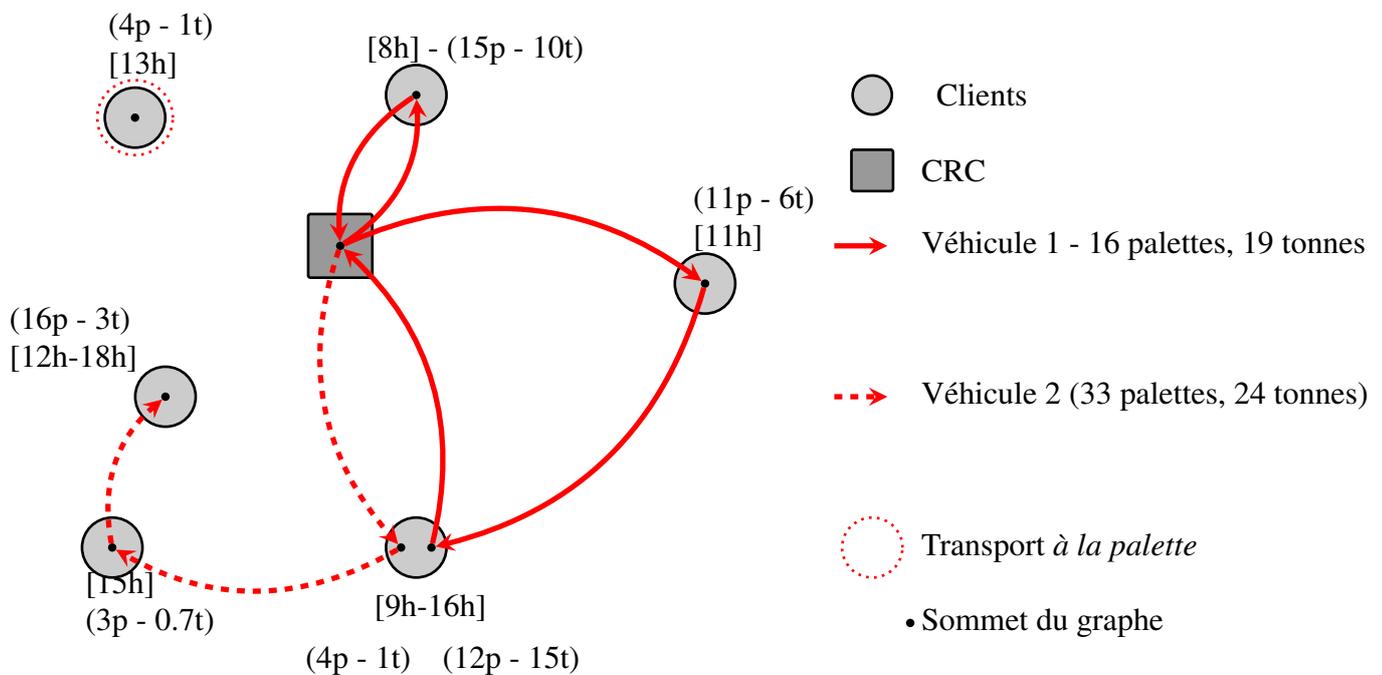
On appelle *VRP-CRC* le problème de distribution depuis un CRC qui consiste à déterminer un ensemble de tournées de coût minimal, prenant en compte les coûts de transport, respectant les contraintes industrielles des tournées, les contraintes de compatibilité entre commandes et permettant la livraison de l'ensemble des commandes dans leurs fenêtres horaires.

6.2. LA GÉNÉRATION DE COLONNES POUR LES PROBLÈMES RICHES DE TOURNÉES DE VÉHICULES

Exemple 13. La figure 6.2 illustre une solution au problème étudié dans ce chapitre. Sept commandes doivent être livrées à 6 clients. Un client doit recevoir deux commandes dont les produits ne sont pas compatibles. Pour chaque sommet du réseau, il est indiqué l'heure ou la fenêtre horaire de livraison, le nombre de palettes et le poids de la commande correspondante.

La solution comporte trois types de tournées :

- Le véhicule 1 est un véhicule de capacité égale à 16 palettes et de poids maximal 19 tonnes. Il est affrété à la journée et réalise deux tournées : il visite tout d'abord un premier client, revient au CRC et repart pour livrer deux autres clients.
- Le véhicule 2 est un véhicule de capacité 33 palettes et de poids maximal 24 tonnes. Il est affrété à la tournée et effectue une tournée ouverte pour livrer les commandes de 3 clients.
- Une commande est distribuée par un transport dédié, dit à la commande.



6.2 La génération de colonnes pour les problèmes riches de tournées de véhicules

L'intérêt des chercheurs pour les problèmes riches de tournées de véhicules est croissant. Caceres-Cruz et al. [14] présentent une revue de littérature sur ce sujet et Lahyani et al. [61] proposent une taxinomie. Sur les 41 articles répertoriés dans [61], seulement deux décrivent des méthodes exactes. Cela peut s'expliquer par trois facteurs.

- Il est difficile d'adapter une méthode exacte aux problèmes de transport riches en raison d'une combinaison importante de contraintes industrielles.
- Les méthodes approchées présentées offrent des solutions de qualité satisfaisante au vu des différentes imprécisions contenues dans les données : les tournées proposées sont parfois modifiées manuellement afin d'intégrer des contraintes supplémentaires non modélisées.
- L'étude des problèmes de transport riches est souvent réalisée en collaboration avec des industriels qui souhaitent un temps de résolution trop court pour l'utilisation d'une méthode exacte.

Parmi les méthodes de résolution approchées, nous nous sommes plus particulièrement intéressés aux heuristiques basées sur la *génération de colonnes*. Les techniques de génération de colonnes (présentées section 3.4) permettent de résoudre des problèmes linéaires de grande taille. Elles sont généralement intégrées dans un *Branch-and-Price* pour résoudre de manière exacte des problèmes linéaires en nombres entiers. De très bons résultats sont également obtenus par des heuristiques utilisant la génération de colonnes. La génération de colonnes est alors utilisée pour générer un ensemble de tournées. Une fois cet ensemble généré, un solveur sélectionne les tournées qui forment la meilleure solution au problème. Le sous-problème de génération de colonnes peut-être résolu de manière exacte ([21],[79]) ou approchée ([17], [45], [53], [85], [99]).

Nous détaillons maintenant les articles utilisant la génération de colonnes pour résoudre des problèmes riches de tournées de véhicules : Chen et Xu [20] considèrent un problème dynamique de tournée de véhicules, où les ordres de transport arrivent au cours du temps. Ils développent une méthode qui résout une série de problèmes statiques au cours du temps. Chaque problème statique est résolu par génération de colonnes. Goel [45] propose une heuristique basée sur la génération de colonnes pour un problème général de tournées de véhicules. Ce problème inclut plusieurs caractéristiques riches, des contraintes de compatibilité et plusieurs dépôts. Le sous-problème de la génération de colonnes, qui génère les tournées, est résolu en retirant et ajoutant des commandes à des tournées existantes. Il compare cette approche avec une méthode LNS. La génération de colonnes se révèle beaucoup plus performante et trouve dans la grande majorité des cas de meilleures solutions que le LNS. Xu et al. [99] considèrent un problème de collectes et livraisons dont les tournées doivent respecter les contraintes de temps de travail des conducteurs. Le coût d'une tournée est déterminé par plusieurs facteurs tels que la distance, le temps d'attente et le temps de travail du conducteur. Un ensemble de tournées est générée par une génération de colonnes dont le sous-problème est résolu grâce à deux heuristiques rapides. Ileri et al. [53] étudient un problème de planification des opérations de camionnage de marchandises. Ils proposent une première méthode basée sur l'énumération de toutes les tournées et une seconde où les tournées sont générées par génération de colonnes. Dans les deux cas, un solveur résout le problème avec l'ensemble des tournées. Le sous-problème de la génération de colonnes est résolu à l'optimalité. Les expérimentations montrent que la méthode de génération de colonnes est beaucoup plus rapide, et fournit les mêmes résultats que la méthode d'énumération. Parragh et al. [79] présentent une heuristique basée sur la génération de colonnes pour résoudre un problème de transport de personnes incluant des contraintes sur le temps de travail des conducteurs. Différentes heuristiques pour la résolution du sous-problème sont présentées et utilisées à chaque itération. De plus, un algorithme exact de programmation dynamique est appelé si aucune des heuristiques ne trouve de colonnes de coût réduit négatif. Ceselli et al [17] résolvent un problème riche de tournées de véhicules en utilisant la génération de colonnes. La méthode est composée de trois phases qui génèrent chacune des colonnes. La différence entre ces trois phases

concerne les hypothèses prises en compte par rapport à la possibilité de transporter séparément les différentes marchandises d'une commande. Les sous-problèmes sont résolus à l'aide d'une recherche tabou ainsi qu'une heuristique basée sur la programmation dynamique. Seixas et al. [85] proposent un algorithme de génération de colonnes pour un problème de tournées de véhicules avec multi-trip intégrant notamment les contraintes sur les temps de conduite. Le sous-problème est résolu par une méthode de recherche tabou ainsi qu'une heuristique basée sur la programmation dynamique.

Tout comme dans les articles cités plus haut, nous proposons une heuristique basée sur la génération de colonnes. Un modèle maître de génération de colonnes est résolu itérativement avec un sous-problème générant de nouvelles colonnes. L'ensemble des colonnes générées sont ensuite ajoutées au PLNE correspondant. L'algorithme résolvant le sous-problème est une recherche tabou. Cette recherche tabou intègre un algorithme d'ordonnancement qui minimise la durée et l'attente totale dans une tournée, et vérifie donc la réalisabilité de chaque tournée générée.

6.3 Méthode de résolution

Dans cette section, un problème riche de tournées de véhicules depuis un CRC est d'abord modélisé comme un PLNE dont les variables représentent les tournées de véhicules. Nous proposons un algorithme heuristique basé sur la génération de colonnes. Le problème riche de tournées de véhicules se réduit polynomialement au problème de tournées de véhicules avec fenêtres de temps ([Vehicle Routing Problem with Time Windows \(VRPTW\)](#)). La résolution du VRPTW étant NP-difficile, le problème traité dans ce chapitre l'est aussi.

La figure 6.3 détaille les étapes de l'algorithme : un premier ensemble de colonnes est tout d'abord généré grâce à une heuristique gloutonne. Ces colonnes constituent le premier Problème Maître Réduit (PMR), c'est-à-dire le problème maître qui ne contient pas toutes les colonnes. La génération de colonnes résout la relaxation linéaire du PMR, appelée PLMR.

La résolution du PLMR permet de récupérer la valeur des variables duales, utilisées dans les sous-problèmes pour trouver de nouvelles colonnes. Chaque sous-problème est résolu heuristiquement par une recherche tabou. Si le critère d'arrêt n'est pas atteint, une nouvelle itération de génération de colonnes est relancée. Sinon, l'ensemble des colonnes générées constitue un PMR de taille acceptable, qui est résolu à l'optimalité grâce à un solveur. Dans notre cas, le critère d'arrêt correspond à un temps limite.

6.3.1 Modélisation du problème maître

On appelle P l'ensemble des tournées et P_k l'ensemble des tournées des véhicules de type $k \in K$. Une tournée $p \in P$ est associée à un coût total c_p et à un ensemble de commandes O_p qu'elle distribue. L'ensemble P_o représente l'ensemble des tournées qui transportent une commande o . On définit une variable binaire x_p égale à 1 si la tournée p est utilisée. Elle représente l'ensemble des tournées d'un véhicule sur une journée, c'est-à-dire un ou plusieurs voyages. Le tableau 6.3 récapitule les notations utilisées dans cette section.

Le problème maître est modélisé par le PLNE suivant :

$$\min \sum_{p \in P} c_p x_p \tag{6.1}$$

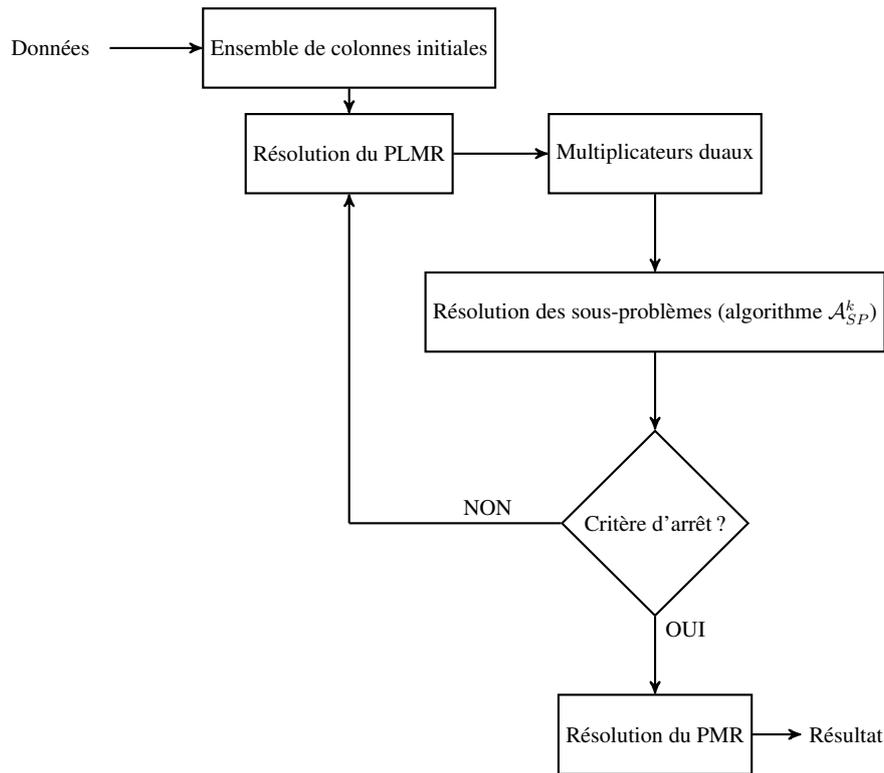


Figure 6.3 – Étapes de l’algorithme

Commandes	O	Ensemble des commandes
	d_o	Client de destination d’une commande
	q_o	Taille en nombre de palettes d’une commande
	w_o	Poids d’une commande en tonne
	t_o	Type de produits d’une commande $o \in O$
	$[e_o, l_o]$	Fenêtre horaire de livraison d’une commande $o \in O$
Types de véhicules	K	Ensemble des types de véhicules
	T_k	Ensemble des types de produits que les véhicules type $k \in K$ peuvent transporter
	Q_k	Capacité du véhicule en nombre de palettes
	W_k	Capacité du véhicule en poids
	M_k	Nombre de véhicules disponibles
Tournées	P	Ensemble des tournées
	P_k	Ensemble des tournées de type $k \in K$
	c_p	Coût de la tournée $p \in P$
	O_p	Commandes transportées par la tournée $p \in P$
	P_o	Ensemble des tournées transportant la commande o
Variable	x_p	=1 si la tournée (ou l’ensemble de tournées d’un véhicule) p est utilisé, 0 sinon

Tableau 6.3 – Notations

$$\begin{aligned} s.c \\ \sum_{p \in P_o} x_p = 1 \quad \forall o \in O \end{aligned} \quad (6.2)$$

$$\sum_{p \in P_k} x_p \leq M_k \quad \forall k \in K \quad (6.3)$$

$$x_p \in \{0; 1\} \quad \forall p \in P. \quad (6.4)$$

La fonction objectif (6.1) minimise le coût des tournées sélectionnées. Les contraintes (6.2) sont des contraintes de type *set partitioning* qui choisissent une tournée unique pour distribuer une commande donnée. Le choix de contraintes de type *set partitioning*, plutôt que des contraintes de *set covering*, est dû à la contrainte de temps maximal d'attente d'une tournée. Supposons que deux tournées puissent servir une même commande. En retirant a posteriori la commande d'une des deux tournées, le véhicule pourrait arriver trop tôt pour livrer la commande suivante, et violer la contrainte de temps maximal d'attente. Enfin, les contraintes 6.3 limitent le nombre de véhicules utilisés.

6.3.2 Ensemble de colonnes initiales

Les colonnes initiales de l'algorithme, utilisées pour pouvoir résoudre le premier PLMR sont générées en deux phases : la génération d'ensembles de commandes, et l'affectation de ces ensembles à des véhicules.

Les ensembles de commandes sont constitués des commandes compatibles entre elles et à destination du même client. De plus, le nombre total de palettes et le poids total ne doivent pas dépasser la capacité maximale du *plus grand* véhicule à disposition. On énumère selon ces règles tous les ensembles de commandes possibles.

Une fois les ensembles de commandes générés, on vérifie la compatibilité de chaque ensemble avec chaque type de véhicule (test de la capacité du véhicule et de la compatibilité véhicules/produit). S'il est possible de transporter un ensemble avec un type de véhicule, une tournée est alors créée. L'ensemble des tournées générées constitue les colonnes du PLMR initial.

6.3.3 Résolution des sous-problèmes

Chaque sous-problème génère de nouvelles colonnes x_p de coût réduit négatif. Le coût réduit se calcule à partir des multiplicateurs duaux obtenus après la résolution du PLMR.

On définit les multiplicateurs duaux λ_o associés à la contrainte (6.2) et $\mu_k \leq 0$ associés à la contrainte (6.3). Le coût réduit \bar{c}_p d'une colonne représentant une tournée se calcule en additionnant le coût réel de la tournée aux coûts associés aux multiplicateurs duaux :

$$\bar{c}_p = c_p - \sum_{o \in K_o} \lambda_o - \mu_k. \quad (6.5)$$

L'objectif du sous-problème est donc de trouver de nouvelles tournées en minimisant le coût réduit tout en respectant les contraintes décrites en section 6.1.3.4.

On souhaite décomposer ces coûts réduits afin d'obtenir les coûts sur chaque arc du réseau. Comme on résout un sous-problème pour chaque type de véhicule $k \in K$, le terme $-\mu_k$ est une constante dans

la fonction objectif du sous-problème. On obtient alors la condition suivante pour ajouter une nouvelle colonne au problème maître :

$$c_p - \sum_{o \in K_o} \lambda_o < \mu_k. \quad (6.6)$$

On rappelle que le réseau aval contient un sommet par commande $o \in O$. On peut donc décomposer une partie du coût réduit sur les arcs : chaque arc, représentant l'arrivée d'un véhicule chez un client afin de livrer la commande o est associé au coût $-\lambda_o$. Néanmoins, le coût réel c_p ne se décompose pas nécessairement sur les arcs, il faut donc dans la plupart des cas le recalculer pour chaque nouvel arc ajouté à la tournée.

Résoudre un sous-problème pour un type de véhicule $k \in K$ revient à résoudre un problème de plus court chemin avec contraintes de ressources (capacité, durée maximale de la tournée, temps d'attente) dans le graphe dont les arcs ont un coût égal à $-\lambda_o$.

6.3.3.1 Méthode de résolution du sous-problème

La résolution d'un sous-problème peut-être exacte ou heuristique. La résolution exacte d'un plus court chemin avec contraintes de ressources utilise souvent des algorithmes de programmation dynamique, comme par exemple l'algorithme de labeling de Feillet et al [41]. Dans ce type d'algorithme, un label est associé à un chemin partiel. Il contient le coût du chemin et la consommation de ses ressources. Un critère de dominance permet de comparer deux chemins partiels passant par le même sommet, et d'éliminer un des deux chemins s'il est dominé par l'autre. Cependant, contrairement à un algorithme de plus court chemin du type *correction de label* comme celui de Bellman-Ford, plusieurs labels doivent être conservés à chaque sommet du graphe, à cause des contraintes de ressources sur le plus court chemin.

L'efficacité de ce type d'algorithme en terme de temps d'exécution dépend de la capacité du critère de dominance à éliminer de nombreux labels. Pour des problèmes comprenant peu de contraintes de ressources sur le plus court chemin, le critère de dominance permet d'écarter assez de labels pour donner une solution rapidement. Cependant, dans notre cas, le nombre de contraintes sur le plus court chemin rend le critère de dominance trop faible pour pouvoir utiliser efficacement un algorithme de programmation dynamique. Nous utilisons donc une heuristique pour résoudre le sous-problème. La recherche tabou est notamment très souvent utilisée pour résoudre ce type de sous-problème, notamment pour ses temps d'exécution très rapides.

6.3.3.2 Algorithme \mathcal{A}_{SP}^k de résolution du sous-problème

Après résolution du PLMR, un algorithme appelé \mathcal{A}_{SP}^k qui génère de nouvelles colonnes est exécuté pour chaque type de véhicules $k \in K$. Comme préconisé par Desaulniers et al. [36], cet algorithme lance à la suite plusieurs recherches tabou (notés \mathcal{A}_{RT} et décrits en section 6.3.3.3), avec des solutions initiales variant à chaque exécution. Ces solutions initiales sont sélectionnées aléatoirement dans l'ensemble P_k qui constitue le PLMR. Le nombre maximal d'exécutions de l'algorithme \mathcal{A}_{RT} dans chaque algorithme \mathcal{A}_{SP}^k est noté $Max - RT$.

Le nombre de colonnes générées par l'algorithme \mathcal{A}_{SB} est limité à une certaine valeur notée $Max - col$. Si cette valeur est atteinte avant la fin de l'exécution de tous les algorithmes \mathcal{A}_{RT} , alors la résolution l'algorithme \mathcal{A}_{SP}^k est stoppée.

6.3.3.3 Algorithme \mathcal{A}_{RT} de recherche tabou

L'algorithme 1 décrit les grandes étapes de l'algorithme \mathcal{A}_{RT} de recherche tabou. Tout d'abord, l'algorithme a besoin d'une colonne initiale. Les lignes de 6 à 11 décrivent une itération de la recherche tabou. Les lignes 7 et 8 représentent les appels aux fonctions *Retrait* et *Insertion* qui créent le voisinage de la solution courante. Toutes les tournées de coût réduit négatif trouvées dans ces voisinages sont ajoutées à la liste de colonnes. On utilise des voisinages différents selon le type de sous-traitance : on distingue les voisinages pour les tournée simples (section 6.3.3.4 de ceux destinés aux tournées multi-trip (section 6.3.3.5). Enfin la meilleure solution trouvée dans les voisinages devient la solution courante (ligne 9), et le mouvement réalisé pour obtenir cette solution est placé dans la liste tabou (ligne 10).

Algorithme 1 : Algorithme \mathcal{A}_{RT} de recherche tabou

Données : Une première colonne

Résultat : Une liste de colonnes de coût réduit négatif

```

1 Liste_de_colonnes =  $\emptyset$  ;
2 Meilleure_solution  $\leftarrow$  première colonne ;
3 Solution_courante  $\leftarrow$  Meilleure_solution ;
4  $\beta \leftarrow$  nombre d'itérations maximal ;
5  $Max - col \leftarrow$  nombre maximal de solutions ;
6 tant que Nombre d'itérations <  $\beta$  ET Nombre de nouvelles colonnes trouvées <  $Max - col$  faire
7   Liste_de_colonnes = Liste_de_colonnes  $\cup$  Retrait (Solution_courante) ;
8   Liste_de_colonnes = Liste_de_colonnes  $\cup$  Insertion (Solution_courante) ;
9   Solution_courante  $\leftarrow$  meilleure solution trouvée dans le voisinage ;
10  Mise à jour de la Liste_tabou ;
11 fin
12 return Liste_de_colonnes ;

```

Concernant la création et mise à jour de la liste tabou, deux tableaux de taille $|O|$ modélisent la liste tabou : un pour l'insertion et un second pour le retrait. Si une commande o est retirée/ajoutée à la solution, cette commande devient tabou. Le numéro de l'itération à laquelle l'ajout/le retrait de cette commande sera de nouveau autorisé est stocké dans le tableau correspondant, à la position de la commande o .

6.3.3.4 Mono-trip

Pour les véhicules qui ne peuvent effectuer qu'une tournée unique, on utilise les voisinages *retrait mono trip* et *insertion mono trip*.

Retrait L'algorithme 2 décrit le fonctionnement de cet opérateur. L'opérateur *retrait* tente de retirer chaque commande de la tournée d'un véhicule de type k . Si le retrait est autorisé (ligne 3) et que la nouvelle solution respecte toutes les contraintes (ligne 5), son coût réduit est calculé (ligne 6). Les fonctions appelées *Réalisabilité1* et *Réalisabilité2* qui vérifient le respect des contraintes sont séparées car elles sont utilisées séparément dans d'autres voisinages. Si le coût réduit est négatif (ligne 7), on précède à un test, dit *test de dominance* entre tournées : l'algorithme vérifie si une autre tournée avec le même type de véhicule et comprenant l'ensemble des commandes existe déjà dans l'ensemble de colonnes (ligne 8). Si oui, et que la nouvelle tournée est moins chère (car les commandes sont positionnées différemment) alors on remplace l'ancienne solution par la nouvelle (lignes 9-10). Sinon on ajoute la

Algorithme 2 : Algorithme de voisinage de retrait Mono-trip

```

Données : solution_courante, meilleure_solution
Résultat : Une liste de colonnes de coût réduit négatif
1 Liste_de_colonnes =  $\emptyset$  ;
2 pour Chaque commande  $o$  de la solution courante faire
3   si Retrait autorisé par la liste tabou alors
4     Nouvelle_solution  $\leftarrow$  solution_courante sans la commande  $o$  ;
5     si Réalisabilité1 (Nouvelle_solution) ET Réalisabilité2 (Nouvelle_solution) alors
6       Coût réduit  $\leftarrow$  Calcul_coût_réduit (Nouvelle_solution) ;
7       si Coût réduit  $\leq 0$  alors
8         si Test_dominance (nouvelle_solution) alors
9           Liste_de_colonnes = Liste_de_colonnes {Ancienne_colonne} ;
10          Liste_de_colonnes = Liste_de_colonnes  $\cup$  {nouvelle_solution} ;
11         sinon
12           Liste_de_colonnes = Liste_de_colonnes  $\cup$  {nouvelle_solution} ;
13         fin
14         si Coût réduit  $\leq$  Coût réduit de la meilleure solution alors
15           meilleure_solution  $\leftarrow$  nouvelle_solution ;
16         fin
17       fin
18     fin
19   fin
20 fin
21 Return Liste_de_colonnes ;

```

nouvelle tournée à la liste des tournées (ligne 12). Si le coût réduit de la nouvelle tournée est inférieur au coût réduit de la meilleure solution déjà trouvée dans le voisinage de la solution courante, alors on remplace la meilleure solution par la nouvelle tournée (ligne 15).

Insertion L'opérateur *insertion mono-trip* tente d'ajouter une commande dans une tournée, à toutes les positions possibles. L'algorithme 3 décrit le voisinage d'insertion. Une commande n'appartenant pas à la solution courante est ajoutée à la solution (ligne 2). En premier lieu, les critères de réalisabilité inhérents à la tournée sont testés (ligne 3) : compatibilité des produits de la nouvelle commande avec les produits déjà présents, capacité, poids. Cette vérification est effectuée par la fonction *réalisabilité1*. Si cette première vérification est validée, et que la liste tabou autorise le mouvement d'insertion de la commande (ligne 4), alors l'insertion de la commande est évaluée à chaque position de la tournée (lignes 5-6). Pour chaque insertion testée, les critères de réalisabilité restants sont évalués (ligne 7) : fenêtres de temps ; durée ; attente ; détour ; nombre maximal d'arrêts d'une tournée. Cette vérification est réalisée par la fonction *réalisabilité2* décrite en section 6.3.4. Si la nouvelle tournée est réalisable, son coût réduit est calculé (ligne 8). S'il est négatif (ligne 9), on regarde si une tournée comprenant les mêmes commandes (placées dans un ordre différent) et le même type de véhicule existe déjà (ligne 10). Si c'est le cas, et si la nouvelle tournée est moins chère, alors on remplace l'ancienne tournée par la nouvelle (lignes 11-12). Sinon, on ajoute la tournée à l'ensemble des colonnes (ligne 14). Si le coût réduit de la nouvelle tournée est inférieur au coût réduit de la meilleure solution déjà trouvée dans le voisinage de la solution courante, alors on remplace la meilleure solution par la nouvelle solution trouvée (lignes 17-18). Ce voisinage est également valable et utilisé pour les tournées à trajets multiples.

Algorithme 3 : Insertion mono-trip**Données :** solution_courante, meilleure_solution, Liste_tabou**Résultat :** Une liste de colonnes de coûts réduits négatifs

```

1 Liste_de_colonnes =  $\emptyset$  ;
2 pour Chaque commande  $o$  non présente dans la solution faire
3   si Réalisabilité1 (solution_courante ,  $o$ ) alors
4     si Insertion autorisée par la liste tabou alors
5       pour Chaque position  $j$  de la solution_courante faire
6         nouvelle_solution  $\leftarrow$  solution_courante avec insertion de la commande  $o$  à la position  $j$  ;
7         si Réalisabilité2 (nouvelle_solution) alors
8           Coût réduit  $\leftarrow$  Calcul_coût_réduit (nouvelle_solution) ;
9           si Coût réduit  $\leq 0$  alors
10            si Test_dominance (nouvelle_solution) alors
11              Liste_de_colonnes= Liste_de_colonnes {Ancienne_colonne} ;
12              Liste_de_colonnes= Liste_de_colonnes  $\cup$  {nouvelle_solution} ;
13            sinon
14              Liste_de_colonnes= Liste_de_colonnes  $\cup$  {nouvelle_solution} ;
15            fin
16          fin
17          si Coût réduit  $\leq$  Coût réduit de la meilleure solution alors
18            meilleure_solution  $\leftarrow$  nouvelle_solution ;
19          fin
20        fin
21      fin
22    fin
23  fin
24 fin
25 Return Liste_de_colonnes ;

```

6.3.3.5 Cas des routes multiples (multi-trip)

Dans le cas où le véhicule peut effectuer plusieurs tournées en une journée, le plus court chemin correspond à la journée entière du véhicule et comprend donc éventuellement des retours au dépôt pendant son parcours. Deux voisinages d'insertion mono-trip sont alors utilisés : le voisinage d'insertion décrit précédemment est tout d'abord utilisé. Un second opérateur d'insertion appelé *division et insertion* tente d'insérer en même temps une commande ainsi qu'un retour au dépôt à chaque position possible. Plus spécifiquement, pour chaque trajet, on évalue chaque position d'insertion possible de la commande dans le trajet et chaque position d'insertion possible d'un retour au dépôt dans le trajet modifié. Si une tournée comporte n positions d'insertion à tester, il y a $o(n^2)$ configurations à tester. Un opérateur de retrait essaie de retirer une commande et/ou un retour au dépôt. Le pseudo-code associé étant très proche du pseudo-code des opérateurs décrits précédemment, nous ne les détaillons pas.

6.3.4 Tests de réalisabilité d'une insertion

Pour chaque voisinage, la réalisabilité de l'insertion d'une commande dans une tournée est testée en deux temps. Le tableau 6.4 résume les notations utilisées.

Dans un premier temps, la fonction appelée *Réalisabilité1* vérifie la compatibilité des produits de la commande avec la tournée et le respect des contraintes de capacité en terme de nombre de palettes et de poids total. Ce test est dissocié des autres tests de réalisabilité, car il peut être effectué avant de connaître la position d'insertion de la commande dans la tournée.

Dans un second temps, pour chaque position d'insertion de la commande, le test appelé *Réalisabilité2* vérifie les contraintes suivantes :

- les fenêtres horaires $[e_o, l_o]$ pour une commande $o \in O$ de l'ensemble des commandes de la tournée obtenue après insertion sont respectées ;
- la somme des temps d'attente sur la tournée est inférieure à Δ ;
- la durée de la tournée est inférieure à Ω ;
- le nombre d'arrêts du véhicule est inférieur à Υ ;
- le détour est inférieur à Θ .

Le nombre maximal d'arrêts ne peut pas être testé dans le premier test de réalisabilité car il peut dépendre de l'ordre des commandes de la tournée : si deux commandes du même client sont servies successivement dans une tournée, ces commandes n'occasionnent qu'un seul arrêt.

L'algorithme 4 décrit le fonctionnement de l'algorithme *Réalisabilité2*. Il est basé sur les travaux de Lehuédé et al. [64] qui s'inspire des travaux de Cordeau et Laporte [26]. Ces travaux utilisent notamment la notion de *forward time slack* définie par Savelsbergh [84].

L'algorithme d'ordonnancement repose sur les variables suivantes pour chaque commande $o \in O_k$ de la tournée k :

- b_o représente l'heure d'arrivée du véhicule chez le client destinataire de la commande o ;
- δ_o représente l'heure de début de déchargement ;

Type	Notation	Description
Commande o	b_o	Heure d'arrivée du véhicule chez le client de la commande o
	δ_o	Heure de déchargement de la commande o
	e_o	Heure de déchargement au plus tôt
	l_o	Heure de déchargement au plus tard
Tournée	f	Marge de décalage temporel (<i>forward time slack</i>)
	w	Temps d'attente total
	τ	Temps de déchargement du dernier client visité
	nd	Nombre d'arrêts de la tournée
Contraintes de la tournée	Δ	Temps d'attente maximal
	Ω	Durée maximale
	Υ	Nombre maximal d'arrêts
	Θ	Détour maximal
	Ξ^f	Temps de déchargement fixe par client
	Ξ^v	Temps de déchargement variable par nombre de palettes

Tableau 6.4 – Résumé des notations pour l'algorithme de réalisabilité

- f représente le *forward time slack* du départ de la tournée, que l'on appelle marge temporelle ;
- w est le temps d'attente dans la tournée ;
- nd est le nombre d'arrêts du véhicule dans la tournée ;
- τ est le temps de déchargement chez le dernier client visité ;

L'algorithme 4 réalise un premier ordonnancement de la tournée *au plus tôt* (lignes 9-19). Cet ordonnancement appelle itérativement l'algorithme 5 appelé algorithme de propagation qui fixe les horaires d'arrivée et de déchargement de chaque commande en propageant ces horaires d'une commande à la commande suivante de la tournée. Cette première phase d'ordonnancement permet de vérifier le respect des fenêtres horaires des commandes, le nombre maximal d'arrêts du véhicule et le détour maximal. Le résultat de cet ordonnancement donne le décalage possible, noté f de l'heure de départ du véhicule afin de minimiser les temps d'attente. La tournée est ensuite ordonnancée en appliquant ce décalage de départ (lignes 29-46). Cette partie de l'algorithme réalise un nouvel ordonnancement et calcule le temps d'attente total à partir des valeurs b_o et f . Enfin, l'algorithme est capable de vérifier les contraintes de durée de la tournée (lignes 47-49) ainsi que les contraintes de temps d'attente (lignes 50-52).

L'algorithme de propagation 5 fixe les horaires d'arrivée et de déchargement de chaque commande de la tournée en propageant les heures d'arrivées tout en tenant compte du temps de déchargement et des fenêtres horaires. Il distingue les cas où les clients des deux commandes successives sont différents (lignes 1-11) ou identiques (lignes 12-26). S'ils sont différents, l'algorithme commence par mettre à jour le nombre de clients dans la tournée (ligne 2) et l'heure d'arrivée chez le client (ligne 3). Si l'heure d'arrivée b_o est supérieure à l'heure d'arrivée au plus tard l_o , la tournée n'est pas réalisable (ligne 5). Ensuite, l'algorithme réduit le décalage dans le temps f de la tournée en fonction de l'heure d'arrivée et de la fenêtre horaire de la commande o (ligne 7). Il met à jour l'heure de déchargement de la commande

correspondant à l'horaire le plus tard entre l'heure d'arrivée b_o du véhicule et l'heure d'arrivée au plus tôt e_o de la commande (ligne 8). Enfin, il rajoute le temps attendu avant déchargement au temps d'attente total (ligne 9) et calcule le temps de déchargement de la commande (ligne 10).

Dans le cas où les deux commandes o_1 et o_2 appartiennent au même client (lignes 12-26), l'algorithme vérifie tout d'abord (lignes 12-14) que l'heure de déchargement de la commande o_1 n'est pas supérieure à l'heure de déchargement au plus tard de la commande o_2 (auquel cas la tournée n'est pas réalisable). Ensuite, deux cas de figures sont possibles :

- Si l'heure de fin de déchargement de o_1 est plus grande que l'heure de déchargement au plus tôt de o_2 (lignes 16-19), alors le temps variable de déchargement de o_2 est additionné au temps de déchargement de o_1 (ligne 16). On agrège alors la livraison de ces deux commandes, et on fixe donc l'heure d'arrivée et l'heure de déchargement de o_2 aux mêmes horaires que ceux de la commande o_1 (lignes 17-18). Enfin le décalage temporel f est mis à jour (ligne 19).
- Si l'heure de fin de déchargement de o_1 est plus petite que l'heure de déchargement au plus tôt de o_2 (lignes 21-25), alors le véhicule doit attendre avant de pouvoir décharger o_2 . Ce cas est très rarement vu en pratique, mais il était nécessaire de traiter tous les cas théoriques dans l'algorithme *réalisabilité*². Une heure *fictive* d'arrivée est programmée après la fin du déchargement de o_1 (ligne 21). Le temps d'attente, le décalage temporel, l'heure de début de déchargement et le temps de déchargement sont mis à jour (lignes 22-25). Dans ce cas, on considère (arbitrairement) que le temps de déchargement est égal uniquement à la durée de déchargement variable à la palette.

6.4 Tests numériques

La méthode de résolution proposée est évaluée sur un ensemble d'instances générées en s'appuyant sur les données de 4S Network.

6.4.1 Instances

Le réseau logistique utilisé pour les instances est celui du premier CRC mis en place par 4S Network, à Saint-Martin-De-Crau (13). Plus de 150 clients sont livrés depuis ce CRC. 18 véhicules, de 6 types différents peuvent être utilisés.

Les données fournies par 4S Network ont été légèrement modifiées afin de faire varier la difficulté des instances et de pouvoir identifier différents facteurs qui rendent l'instance plus ou moins difficile à résoudre. Nous utilisons donc les données fournies par 4S Network en conservant :

- le nombre de palettes, le poids ;
- les coordonnées du CRC et des clients ;
- toutes les données des transporteurs : tarifs, types et nombre de véhicules.

Nous souhaitons faire varier les paramètres suivants.

- Le nombre de commandes : de 10 à 50 commandes, par pas de 10.

Algorithme 4 : Algorithme de réalisabilité2

Données : tournée p avec N commandes, temps d'attente maximal Δ , durée maximale Ω , nombre maximal de clients Υ , temps de déchargement fixe Ξ^f , temps de déchargement variable Ξ^v

Résultat : Un booléen indiquant si la tournée est réalisable ou non

```

1 Commande  $o_1 \leftarrow 1^{re}$  commande de la tournée ;
2  $b_{o_1} \leftarrow e_{o_1}$  ; /* heure d'arrivée de la commande  $o_1$  */
3  $f \leftarrow l_o - b_{o_1}$  ; /* décalage maximal de l'ordonnancement */
4  $w_o \leftarrow 0$  ; /* temps d'attente dans la tournée */
5  $nd \leftarrow 1$  ; /* nombre de clients dans la tournée */
6  $\delta_{o_1} \leftarrow e_{o_1}$  ; /* heure de déchargement de la commande  $o_1$  et  $e_{o_1}$  */
7  $\tau \leftarrow \Xi^f + \Xi^v \times q_{o_1}$  ; /* temps de déchargement de  $o_1$  */
8 pour Chaque position  $i$  de la tournée à partir de la 2eme faire
9    $o_1 \leftarrow (i - 1)$ ème commande de la tournée ;
10   $o_2 \leftarrow (i)$ ème commande de la tournée ;
11  si Planification( $o_1, o_2, b_{o_1}, b_{o_2}, f, w_o, \tau, nd, \Xi^f, \Xi^v, \delta_{o_2}$ ) = false ; /* Algorithme 5 */
12  alors
13  |   return false ;
14  fin
15  si  $nd > \Upsilon$  alors
16  |   return false ; /* nombre trop grand de clients */
17  fin
18 fin
19 si  $f = 0$  ET  $w_o > \Delta$  alors
20 |   return false ; /* Temps d'attente trop grand */
21 fin
22 si  $(\delta_{c_N} + \tau) - (b_{o_1} + f) > \Omega$  ; /* Différence entre l'heure de fin de déchargement du
    dernier client et départ de la tournée */
23 alors
24 |   return false ; /* Durée de la tournée trop grande */
25 fin
26 si CalculDetour(tournée)  $> \Theta$  alors
27 |   return false ; /* Détour trop grand */
28 fin
29 si  $f \geq 0$  alors
30 |    $b_{o_1} \leftarrow b_{o_1} - f$  ;
31 |   pour chaque commande  $o$  dans la tournée faire
32 |   |   si  $b_o \geq e_o$  alors
33 |   |   |    $b_o \leftarrow b_o + f$  ;
34 |   |   sinon
35 |   |   |   si  $f + b_o \leq e_o$  alors
36 |   |   |   |    $w \leftarrow w - f$  ;
37 |   |   |   |    $b_o \leftarrow b_o + f$  ;
38 |   |   |   |    $f \leftarrow 0$  ;
39 |   |   |   sinon
40 |   |   |   |    $f \leftarrow f + b_o - e_o$  ;
41 |   |   |   |    $b_o \leftarrow b_o + f$  ;
42 |   |   |   |    $w \leftarrow w - \delta_o$  ;
43 |   |   |   fin
44 |   |   fin
45 |   fin
46 fin
47 si  $(b_{c_N} + \tau) - (b_{o_1} - (\text{temps de transport entre } o_1 \text{ et le CRC} + \text{temps de chargement})) > \Omega$  alors
48 |   return false ; /* Durée de la tournée trop grande */
49 fin
50 si  $w_o > \Delta$  alors
51 |   return false ; /* Temps d'attente trop grand */
52 fin
53 return true ;

```

Algorithme 5 : Algorithme de propagation

Données : $o_1, o_2, b_{o_1}, b_{o_2}, f, w, \Xi^f, \Xi^v, \tau$ **Résultat** : $b_{o_2}, f, w, \tau, \delta_{o_2}$

```

1  si  $o_1$  et  $o_2$  ne sont pas livrés au même client alors
2       $n_d \leftarrow n_d + 1$  ;
3       $b_{o_2} \leftarrow b_{o_1} + \tau$  + temps de transport entre  $d_{o_1}$  et  $d_{o_2}$  ;
4      si  $b_{o_2} \geq l_{o_2}$  alors
5          | return false ;
6      fin
7       $f \leftarrow \min(f; l_{o_2} - b_{o_2})$  ;
8       $\delta_{o_2} \leftarrow \max(b_{o_2}; e_{o_2})$  ;
9       $w \leftarrow \max(w; w + e_{o_2} - b_{o_2})$  ;
10      $\tau \leftarrow \Xi^f + \Xi^v \times q_{o_2}$  ;
11 sinon
12     si  $b_{o_1} > l_{o_2}$  alors
13         | return false;
14     fin
15     si  $b_{o_1} + \tau > e_{o_2}$  alors
16         |  $\tau \leftarrow \tau + \Xi^v \times q_{o_2}$  ;
17         |  $b_{o_2} \leftarrow b_{o_1}$  ;
18         |  $\delta_{o_2} \leftarrow \delta_{o_1}$  ;
19         |  $f \leftarrow \min(f; l_{o_2} - b_{o_2})$  ;
20     sinon
21         |  $b_{o_2} \leftarrow b_{o_1} + \tau$  ;
22         |  $w \leftarrow w + e_{o_2} - (b_{o_1} + \tau)$  ;
23         |  $f \leftarrow \min(f; l_{o_2} - (b_{o_1} + \tau))$  ;
24         |  $\delta_{o_2} \leftarrow e_{o_2}$  ;
25         |  $\tau \leftarrow \Xi^v \times q_{o_2}$  ;
26     fin
27 fin
28 return  $b_{o_2}, f, w, \tau, \delta_{o_2}$  ;

```

- Le pourcentage de commande avec un horaire fixe : de 25% à 100%, par pas de 25%. Lorsque les commandes ont un horaire fixe, il est généralement plus facile de résoudre l'instance, car l'espace des solutions est réduit.

Cela nécessite quelques aménagements des données :

- le nombre de commandes par jour : une instance correspond à une journée précise de livraison. Afin d'obtenir les tailles d'instances souhaitées, nous avons basculé quelques commandes d'une journée sur une autre.
- Les horaires de livraison : pour pouvoir obtenir le nombre souhaité de commandes à horaire fixe, nous avons modifié certaines heures ou fenêtres horaires de livraison.

Nous avons ainsi généré 20 instances caractérisées par leur nom i_A_B , où A est le nombre de commandes et B est le pourcentage de commandes avec un horaire fixe.

6.4.2 Paramétrage

Les principaux paramètres de la méthode de résolution concerne la résolution du sous-problème, caractérisé par l'algorithme \mathcal{A}_{SP}^k :

- L_{tabou} la longueur de la liste tabou ;
- $Max - RT$ le nombre de recherches tabou exécutées par itération et par sous-problème ;
- $Max - Col$ le nombre maximal de solutions qu'un sous-problème peut trouver ;
- β le nombre maximal d'itérations de la recherche tabou.

Un *paramétrage* de l'algorithme correspond à quatre valeurs $(L_{tabou}; Max - RT; Max - Col; \beta)$.

Nous avons effectué une première phase manuelle de paramétrage afin d'observer un ordre de sensibilité aux paramètres et un intervalle de valeurs raisonnable à tester pour chaque paramètre :

- longueur de la liste tabou (L_{tabou}) de 6 à 14 par pas de 2 ;
- le nombre de recherches tabou lancées par itération et par sous-problème ($Max - RT$) : de 100 à 200 par pas de 25 ;
- le nombre maximum de solutions trouvées pour chaque sous-problème ($Max - Col$) : de 20 à 50 par pas de 10 ;
- nombre maximum d'itérations de la recherche tabou (β) : de 50 à 110 par pas de 20.

Chaque paramétrage est exécuté 5 fois sur un ordinateur Xeon X5650 à 2.67 GHz avec 64 GB de RAM. Nous limitons à 5 minutes la génération de colonnes pour les instances de 10 à 20 commandes, 10 minutes pour les instances de 30 commandes et 15 minutes pour les instances de 40 à 50 commandes.

Afin d'évaluer les performances de l'algorithme selon ces critères, nous enregistrons les résultats suivants :

- La différence entre le coût minimum trouvé pour une instance et le coût minimum trouvé avec le paramétrage à évaluer. Elle est appelée ΔMin dans les tableaux de résultats 6.5 à 6.8.
- La différence entre le coût minimum trouvé pour une instance et la moyenne des coûts obtenus avec le paramétrage évalué. Elle est appelée ΔMoy dans les tableaux de résultats 6.5 à 6.8.

6.4.3 Résultats

6.4.3.1 Nombre d'itérations d'une recherche tabou

Le nombre d'itérations d'une recherche tabou fait partie du critère d'arrêt de la résolution du sous-problème. On teste ce critère avec les valeurs 50, 70, 90 et 110. Les résultats sont présentés dans le tableau 6.5. On remarque que plus le nombre d'itérations de la recherche augmente, et plus le ΔMin est bon : 1% pour 90 itérations et 0.8% pour 110 itérations contre 1.3% pour 50 et 70 itérations. On retrouve les mêmes résultats pour le ΔMoy , même si la valeur 110 donne de moins bons résultats en moyenne : 2.5% et 2.6% pour 50 et 70 itérations et 1.7% et 1.8% pour 90 et 110 itérations. On sélectionne donc la valeur 90 pour le paramètre β de la résolution du sous-problème.

Nombre d'itérations	ΔMin				ΔMoy			
	50	70	90	110	50	70	90	110
i_10_100	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_75	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_50	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_25	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_20_100	0,1%	0,0%	0,0%	0,0%	3,7%	4,0%	0,7%	0,8%
i_20_75	0,0%	0,0%	0,0%	0,0%	3,0%	1,8%	0,0%	0,0%
i_20_50	0,2%	0,0%	0,0%	0,0%	0,8%	1,1%	0,3%	1,1%
i_20_25	2,1%	2,2%	2,0%	1,3%	2,3%	5,5%	3,3%	2,9%
i_30_100	0,7%	4,5%	0,0%	1,6%	3,3%	4,9%	3,3%	4,4%
i_30_75	5,0%	5,0%	3,4%	1,5%	5,9%	5,7%	3,4%	2,4%
i_30_50	0,0%	0,0%	2,3%	0,6%	0,7%	0,6%	2,5%	1,5%
i_30_25	1,0%	2,4%	1,2%	0,3%	2,5%	3,1%	1,7%	0,6%
i_40_100	0,3%	0,3%	0,3%	0,0%	1,3%	1,3%	1,0%	0,4%
i_40_75	3,4%	0,0%	0,9%	0,4%	4,4%	3,6%	1,6%	2,2%
i_40_50	3,2%	0,2%	1,0%	4,0%	4,0%	3,5%	2,6%	6,3%
i_40_25	0,1%	0,5%	0,2%	1,2%	1,9%	3,3%	1,7%	2,3%
i_50_100	0,0%	1,4%	1,4%	1,4%	0,5%	1,8%	2,3%	2,1%
i_50_75	0,3%	1,5%	0,3%	0,5%	2,1%	2,5%	3,0%	2,5%
i_50_50	4,7%	4,5%	3,1%	0,8%	7,4%	4,7%	3,4%	2,1%
i_50_25	5,0%	4,0%	3,5%	2,7%	6,6%	5,0%	3,7%	4,6%
Moyenne	1,3%	1,3%	1,0%	0,8%	2,5%	2,6%	1,7%	1,8%

Tableau 6.5 – Résultat selon le nombre d'itérations de la recherche tabou

6.4.3.2 Nombre de recherches tabou ($Max - RT$)

On observe les différences de résultat selon le nombre de recherches tabou par sous-problème. Les résultats sont présentés dans le tableau 6.6, avec les valeurs 100, 125, 150, 175, 200. La valeur assurant la plus grande stabilité est 150 avec 1.7% pour le ΔMoy en moyenne. On sélectionne donc cette valeur 150 comme valeur du paramètre $Max - RT$.

Max-RT	Δ Min					Δ Moy				
	100	125	150	175	200	100	125	150	175	200
i_10_100	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_75	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_50	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_25	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_20_100	0,0%	0,0%	0,0%	0,0%	0,0%	5,9%	0,9%	0,7%	1,0%	3,2%
i_20_75	0,0%	0,0%	0,0%	0,0%	0,0%	2,3%	3,6%	0,0%	0,0%	3,4%
i_20_50	0,0%	0,2%	0,0%	0,2%	0,0%	1,4%	1,5%	0,3%	0,5%	1,2%
i_20_25	1,5%	2,2%	2,0%	0,0%	2,1%	2,9%	3,7%	3,3%	2,4%	3,7%
i_30_100	5,0%	0,2%	0,0%	5,0%	2,4%	5,1%	2,1%	3,3%	5,2%	3,9%
i_30_75	0,0%	0,0%	3,4%	2,6%	5,0%	1,7%	2,4%	3,4%	4,2%	1,5%
i_30_50	0,1%	0,0%	2,3%	0,0%	0,0%	1,5%	0,3%	2,5%	2,1%	0,6%
i_30_25	0,2%	0,4%	1,2%	1,2%	0,5%	1,4%	1,3%	1,7%	1,7%	5,3%
i_40_100	0,3%	0,0%	0,3%	0,3%	2,0%	1,1%	1,4%	1,0%	1,9%	2,2%
i_40_75	0,0%	0,0%	0,9%	0,0%	1,1%	0,7%	1,8%	1,6%	0,0%	2,4%
i_40_50	1,3%	2,5%	1,0%	0,7%	3,0%	3,8%	4,3%	2,6%	2,5%	5,3%
i_40_25	2,2%	0,2%	0,2%	2,6%	1,6%	4,8%	3,0%	1,7%	3,4%	2,7%
i_50_100	1,4%	2,3%	1,4%	1,4%	0,4%	2,3%	3,1%	2,3%	2,2%	1,6%
i_50_75	4,0%	1,7%	0,3%	0,8%	1,0%	4,8%	4,6%	3,0%	5,3%	5,1%
i_50_50	0,8%	0,0%	3,1%	4,7%	4,0%	3,5%	2,2%	3,4%	5,5%	6,9%
i_50_25	0,9%	3,2%	3,5%	1,9%	3,8%	5,0%	3,9%	3,7%	2,3%	2,3%
Moyenne	0,9%	0,6%	1,0%	1,1%	1,3%	2,4%	2,0%	1,7%	2,0%	2,6%

Tableau 6.6 – Résultats selon le paramètre *Max - RT*

6.4.3.3 Longueur de la liste tabou (L_{tabou})

Nous avons testé 5 valeurs différentes de longueur de liste tabou : 6, 8, 10, 12, 14. Le tableau 6.7 montre les ΔMin et ΔMax pour chaque instance en fonction de la longueur de liste tabou. Même si les moyennes sont très proches, on sélectionne la longueur 10 car elle semble être celle assurant la plus grande stabilité, particulièrement pour les tailles d'instances les plus grandes. Les moins bons résultats des plus petites et plus grandes longueurs de liste tabou s'expliquent facilement. Les plus petites longueurs risquent de faire cycler la recherche tabou autour d'un minimum local. Une liste tabou trop grande interdit plus de mouvements, ce qui risque de trop *diversifier* la solution et empêcher de trouver certains minima locaux.

L _{tabou}	ΔMin					ΔMoy				
	6	8	10	12	14	6	8	10	12	14
i_10_100	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_75	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_50	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_25	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_20_100	0,4%	0,0%	0,0%	1,7%	2,2%	1,5%	0,9%	0,7%	2,1%	6,3%
i_20_75	0,0%	0,0%	0,0%	0,0%	0,0%	1,1%	3,0%	0,8%	2,0%	1,8%
i_20_50	0,2%	0,2%	0,0%	0,0%	0,2%	1,6%	0,5%	0,3%	2,0%	1,3%
i_20_25	2,7%	2,1%	2,0%	2,2%	2,1%	3,8%	3,1%	3,3%	3,6%	2,7%
i_30_100	3,5%	1,3%	0,0%	4,0%	5,0%	4,6%	3,6%	3,3%	6,5%	5,1%
i_30_75	1,9%	0,0%	3,4%	1,0%	2,7%	3,5%	2,2%	3,4%	3,5%	5,1%
i_30_50	1,7%	0,5%	2,3%	0,7%	0,0%	4,0%	2,1%	2,5%	2,0%	1,2%
i_30_25	0,1%	1,5%	1,2%	2,5%	0,4%	2,2%	2,6%	1,7%	4,1%	1,5%
i_40_100	0,3%	0,8%	0,3%	0,0%	0,3%	1,7%	1,8%	1,0%	1,2%	1,5%
i_40_75	0,3%	0,4%	0,9%	0,5%	0,3%	5,4%	1,7%	1,6%	3,3%	5,4%
i_40_50	4,3%	1,2%	1,0%	0,0%	2,8%	5,3%	3,5%	2,6%	3,3%	3,6%
i_40_25	1,4%	1,1%	0,2%	0,0%	1,1%	4,0%	3,8%	1,7%	4,1%	3,4%
i_50_100	1,4%	1,4%	1,4%	0,0%	0,0%	3,2%	3,5%	2,3%	0,6%	3,0%
i_50_75	1,7%	0,0%	0,3%	5,0%	3,9%	4,7%	2,5%	3,0%	6,3%	5,9%
i_50_50	6,4%	0,8%	3,1%	1,5%	5,8%	9,3%	4,6%	3,4%	5,1%	8,6%
i_50_25	4,0%	2,7%	3,5%	2,0%	3,1%	5,9%	4,4%	3,7%	6,2%	7,2%
Moyenne	1,5%	0,7%	1,0%	1,1%	1,5%	3,1%	2,2%	1,8%	2,8%	3,2%

Tableau 6.7 – Écarts de coûts selon la longueur de la liste tabou

6.4.3.4 Nombre maximal de solutions trouvées $Max - col$

Le tableau 6.8 donne les résultats obtenus avec les valeurs 20, 30, 40 et 50 pour le paramètre $Max - Col$. On observe une différence faible des résultats selon les variations de ce paramètre. Nous avons sélectionné la valeur 40 dont le ΔMoy moyen est de 1.8% et le ΔMin moyen égal à 0.8%.

6.4.4 Analyse des résultats selon le pourcentage d'horaires fixés

Une fois les valeurs des paramètres fixés, on analyse les résultats en fonction du pourcentage d'horaires fixés.

Nombre de solution trouvées	Δ Min				Δ Moy			
	20	30	40	50	20	30	40	50
i_10_100	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_75	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_50	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_10_25	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
i_20_100	1,0%	0,0%	0,1%	0,4%	1,5%	0,7%	1,4%	1,1%
i_20_75	0,0%	0,0%	0,0%	0,0%	4,5%	0,0%	1,5%	0,0%
i_20_50	0,0%	0,0%	0,0%	0,0%	0,6%	1,3%	0,6%	0,7%
i_20_25	2,0%	2,0%	2,0%	1,6%	4,2%	3,3%	2,2%	2,6%
i_30_100	1,2%	0,0%	1,2%	4,5%	3,6%	3,3%	3,2%	5,0%
i_30_75	1,9%	3,4%	1,6%	1,4%	2,8%	3,4%	2,6%	3,1%
i_30_50	0,0%	2,3%	0,3%	0,3%	1,0%	2,5%	0,7%	1,0%
i_30_25	0,4%	1,2%	0,2%	0,5%	1,6%	1,7%	0,6%	0,7%
i_40_100	0,3%	0,3%	0,1%	0,1%	1,8%	1,0%	1,3%	0,9%
i_40_75	0,3%	0,9%	0,0%	0,0%	2,9%	1,6%	3,1%	2,6%
i_40_50	3,5%	1,0%	1,3%	1,1%	5,2%	2,6%	2,6%	3,5%
i_40_25	1,3%	0,2%	1,4%	0,7%	3,2%	1,7%	3,3%	2,2%
i_50_100	1,1%	1,4%	1,1%	1,3%	1,6%	2,3%	2,4%	1,6%
i_50_75	0,5%	0,3%	0,8%	0,5%	1,0%	3,0%	2,8%	2,8%
i_50_50	0,8%	3,1%	2,8%	2,4%	3,1%	3,4%	4,2%	3,5%
i_50_25	0,0%	3,5%	3,2%	0,8%	1,9%	3,7%	4,1%	3,5%
Moyenne	0,7%	1,0%	0,8%	0,8%	2,0%	1,8%	1,8%	1,7%

Tableau 6.8 – Résultats selon le nombre maximal de solutions trouvées dans une recherche tabou

Nombre de commandes	Pourcentage d'horaires fixés			
	25	50	75	100
10	178	178	178	178
20	384	394	396	429
30	764	821	826	1 004
40	1 265	1 265	1 344	1 479
50	1 616	1 646	1 753	1 869

Tableau 6.9 – Coût du transport en fonction du nombre de commandes et du pourcentage d'horaires de livraison fixés

Le tableau 6.9 présente le coût des tournées selon le nombre de commandes et le pourcentage d'horaires fixés. On remarque que lorsque 100% des horaires sont fixés, le coût est beaucoup plus élevé que pour les instances où 25% à 75% des horaires sont fixés. Il est donc préférable de ne pas fixer trop d'horaires de livraison à l'avance : 25% d'horaires non fixés suffisent à diminuer le coût de manière significative.

Nombre de commandes	Pourcentage d'horaires fixés			
	25	50	75	100
10	4	4	4	4
20	5	5	5	5
30	6	6	6	7
40	8	8	8	9
50	9	9	10	10

Tableau 6.10 – Nombre de véhicules en fonction du nombre de commandes et du pourcentage d'horaires de livraison fixés

Le tableau 6.10 indique le nombre de véhicules nécessaires en fonction du nombre de commandes et du pourcentage d'horaires fixés. Nous observons que réduire le pourcentage d'horaires fixés permet parfois d'éliminer un véhicule dans la solution.

Enfin, le tableau 6.11 indique le nombre de colonnes générées en fonction du nombre de commandes et du pourcentage d'horaires de livraison fixés. Sans surprise, le nombre de colonnes augmente fortement lorsque le nombre d'horaires fixés diminue, et lorsque le nombre de commandes augmente.

Nombre de commandes	Pourcentage d'horaires fixés			
	25	50	75	100
10	218	219	207	204
20	875	785	705	609
30	1 363	1 235	1 093	964
40	1 807	1 639	1 486	1 296
50	2 253	2 076	1 888	1 681

Tableau 6.11 – Nombre de colonnes générées en fonction du nombre de commandes et du pourcentage d'horaires de livraison fixés

6.5 Conclusion et perspectives

Dans ce chapitre, nous avons présenté une méthode permettant de résoudre le problème riche de tournées de véhicules depuis un CRC. Le transport des commandes peut être effectué par différents transporteurs qui proposent trois contrats de sous-traitance (à la journée, à la tournée, à la commande) avec une flotte

hétérogène. Des contraintes limitant la durée de la tournée et le temps maximal d'attente visent à générer des tournées qui respectent les contraintes légales pour la durée de travail des conducteurs.

L'algorithme développé est basé sur une génération de colonnes, dont le sous-problème est résolu avec une recherche tabou. La réalisabilité d'une nouvelle colonne générée est testée grâce à un algorithme d'ordonnement qui minimise le temps d'attente du véhicule. Les expérimentations montrent les réglages des différents paramètres de cette méthode. Les tests montrent que la méthode obtient de bons résultats pour des temps de résolution courts (de 5 à 15 minutes).

Ce travail a été intégré dans un outil d'optimisation utilisé quotidiennement par 4S Network (voir chapitre 9 section 9.3). Cet outil aide à la planification quotidienne des tournées pour chaque CRC. Des données réelles provenant de cet outil ont été utilisées pour générer des instances de test de l'algorithme.

En perspective, il serait utile de développer un algorithme de résolution exacte pour résoudre le sous-problème, afin de mieux résoudre le problème ou évaluer la performance de la recherche tabou. En outre, paralléliser l'exécution des algorithmes \mathcal{A}_{SP}^k et \mathcal{A}_{RT} permettrait de résoudre plus rapidement les sous-problèmes de la génération de colonnes.



Optimisation conjointe des réseaux amont et aval

Sommaire

7.1	Introduction	100
7.2	Modélisation du réseau amont	100
7.2.1	Description et modélisation du réseau	100
7.2.2	Coût des tournées dans le réseau amont	102
7.3	Modélisation du réseau aval	102
7.3.1	Description et modélisation du réseau	102
7.3.2	Coûts des tournées dans le réseau aval	103
7.4	Commandes et sous-commandes	104
7.5	Modélisation du problème $\mathcal{F}_{SND RP-CRC}$	105
7.5.1	Modèle $\mathcal{F}_{SND RP-CRC}$	106
7.5.2	Modélisation pour la génération de colonnes $\mathcal{F}^{\mathcal{L}}_{SND RP-CRC}$	107
7.6	Matheuristique pour la résolution du SND RP-CRC	110
7.6.1	Solution initiale	110
7.6.2	SP1 – Génération des itinéraires amont	112
7.6.3	SP2 – Génération des tournées de véhicules aval	114
7.7	Tests numériques	116
7.7.1	Génération des instances	116
7.7.2	Paramétrage de la recherche tabou	116
7.7.3	Écart entre la relaxation linéaire et la solution	119

7.7.4	Influence de la discrétisation	119
7.7.5	Influence du coût de stockage	121
7.8	Conclusion et perspectives	123

7.1 Introduction

Ce chapitre présente la résolution du problème de routage des marchandises à travers les réseaux amont et aval. L'objectif est de trouver l'itinéraire des sous-commandes depuis leur origine dans le réseau amont jusqu'à leur destination dans le réseau aval. Le problème de transport dans le réseau amont est équivalent à un [Service Network Design Problem \(SNDP\)](#), où l'objectif est de déterminer l'itinéraire des sous-commandes depuis leur origine jusqu'aux CRC. Les tournées dans ce réseau ont un tarif *LTL* approximé par une fonction linéaire par morceaux. Le problème à résoudre dans le réseau aval revient à résoudre un problème de voyageur avec fenêtres horaires ([Vehicle Routing Problem with Time Windows \(VRPTW\)](#)) au départ de chaque CRC, avec un tarif *FTL*. Les deux réseaux sont liés par un stockage de courte durée au CRC (quelques heures). Nous considérons un coût de stockage dépendant du temps d'attente au CRC.

Nous proposons une heuristique basée sur la génération de colonnes, qui génère deux types de variables : une variable représentant l'itinéraire des sous-commandes dans le réseau amont et une variable modélisant les tournées de véhicules dans le réseau aval. Afin de générer ces deux types de variables, deux sous-problèmes sont résolus à chaque itération. Nous testons les performances de la méthode sur des instances générées à partir du réseau de 4S Network.

Ce chapitre détaille tout d'abord le fonctionnement et la modélisation du réseau amont (section 7.2) et du réseau aval (section 7.3). L'itinéraire des commandes et sous-commandes est décrit en section 7.4. La modélisation du problème est présentée en section 7.5 et la méthode de résolution développée est décrite en section 7.6. Enfin, les résultats numériques sont détaillés en section 7.7.

7.2 Modélisation du réseau amont

Cette section détaille le fonctionnement du transport et la modélisation du réseau amont ainsi que sa structure tarifaire.

7.2.1 Description et modélisation du réseau

Le réseau amont est composé de sites de fournisseurs et des CRC. L'objectif du problème de transport dans le réseau amont est de déterminer les itinéraires des commandes et des sous-commandes dans ce réseau, depuis leur origine jusqu'à un CRC. Des véhicules réalisant des tournées effectuent le transport des sous-commandes, qui peuvent être transférées plusieurs fois sur les sites des fournisseurs et donc être transportées par plusieurs véhicules. Ces véhicules visitent un ou plusieurs sites de fournisseurs, et éventuellement un CRC à la fin de leur tournée. Grâce à la limite sur le nombre total d'arrêts du véhicule, il est possible d'énumérer toutes les tournées amont de manière exhaustive (voir chapitre 5 section 5.2.2.3).

Le réseau amont est modélisé par un réseau espace-temps afin de faciliter la représentation de la synchronisation spatiale et temporelle entre les tournées. On note $D = (N, A)$ le réseau physique du réseau amont où N est l'ensemble des sites fournisseurs et des CRC, et A est l'ensemble des arcs entre ces sommets. On note N_{crc} l'ensemble des sommets représentant un CRC dans le réseau plat. On appelle R_{amont} l'ensemble des tournées dans le réseau amont. Une tournée $r \in R_{amont}$ est constituée d'un ensemble d'arcs du réseau D .

A partir de cet ensemble R_{amont} de tournées, on définit un *réseau plat* $D_{\mathcal{P}} = (N_{\mathcal{P}}, A_{\mathcal{P}})$.

Chaque tournée est copiée dans l'ensemble $N_{\mathcal{P}}$ du réseau plat. Par conséquent, si deux tournées du réseau physique partagent un sommet, on a alors un dédoublement de ce sommet. Une *structure de transfert complète* modélise les transferts entre les tournées qui passent par le même sommet $n \in N$ du réseau physique. On appelle $A^{\mathcal{R}} \subset A_{\mathcal{P}}$ l'ensemble des arcs modélisant le transport d'un sommet physique $i \in N$ vers un sommet $j \in N$, et $A^{\mathcal{M}} \subset A_{\mathcal{P}}$ l'ensemble des arcs (qu'on appelle par la suite *arcs de transfert*) modélisant un transfert d'une tournée à une autre. On note que $A^{\mathcal{R}} \cup A^{\mathcal{M}} = A_{\mathcal{P}}$.

A partir de ce *réseau plat*, nous définissons un horizon de temps T et un réseau espace-temps $D_{\mathcal{T}} = (N_{\mathcal{T}}, H_{\mathcal{T}} \cup A_{\mathcal{T}}^{\mathcal{R}} \cup A_{\mathcal{T}}^{\mathcal{M}})$. Un sommet du réseau plat $i \in N_{\mathcal{P}}$ est répliqué pour chaque une période de temps $t \in T_i$ où il est accessible. Un sommet $(i, t) \in N_{\mathcal{T}}$ représente la visite du sommet physique $i \in N$ à la période $t \in T$. Les ensembles $A_{\mathcal{T}}^{\mathcal{R}}$ et $A_{\mathcal{T}}^{\mathcal{M}}$ contiennent les arcs de la forme $((i, t), (j, \bar{t}))$, avec $(i, j) \in A^{\mathcal{R}} \cup A^{\mathcal{M}}, t \in T_i, \bar{t} \in T_j$. Ces arcs représentent un départ du sommet i à la période $t \in T_i$ et une arrivée en j à la période $\bar{t} \in T_j$. L'ensemble $H_{\mathcal{T}}$ est l'ensemble des arcs d'attente sur un sommet. Ils sont de la forme $((i, t), (i, \bar{t}))$, avec $t, \bar{t} \in T_i$. Enfin, on note R l'ensemble des tournées dans le réseau espace temps (différent de R_{amont}) et A_r l'ensemble des arcs d'une tournée $r \in R$.

Exemple 14. La figure 7.1 illustre un exemple de réseau amont plat avec trois tournées. Les durées de transport sont indiquées sur les arcs. Deux tournées passent par le même site de fournisseur F_2 . Deux sommets (F_2^1 et F_2^2) sont donc créés pour représenter le transfert selon la structure de transfert complète. Un arc de transfert relie ces deux sommets.

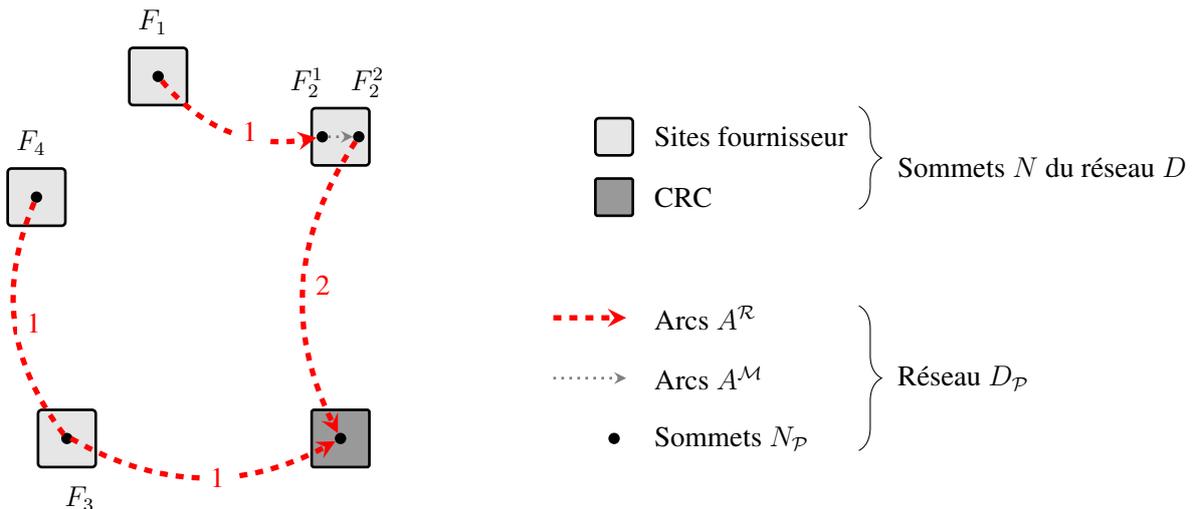


Figure 7.1 – Tournées dans le réseau amont plat

La figure 7.2 représente le réseau espace-temps associé à ce réseau plat. Les trois tournées sont

reproduites à chaque période possible de départ ainsi que les arcs de transferts possibles sur les 6 périodes temporelles du réseau. Enfin, trois exemples d'arcs d'attente sont représentés en trait plein.

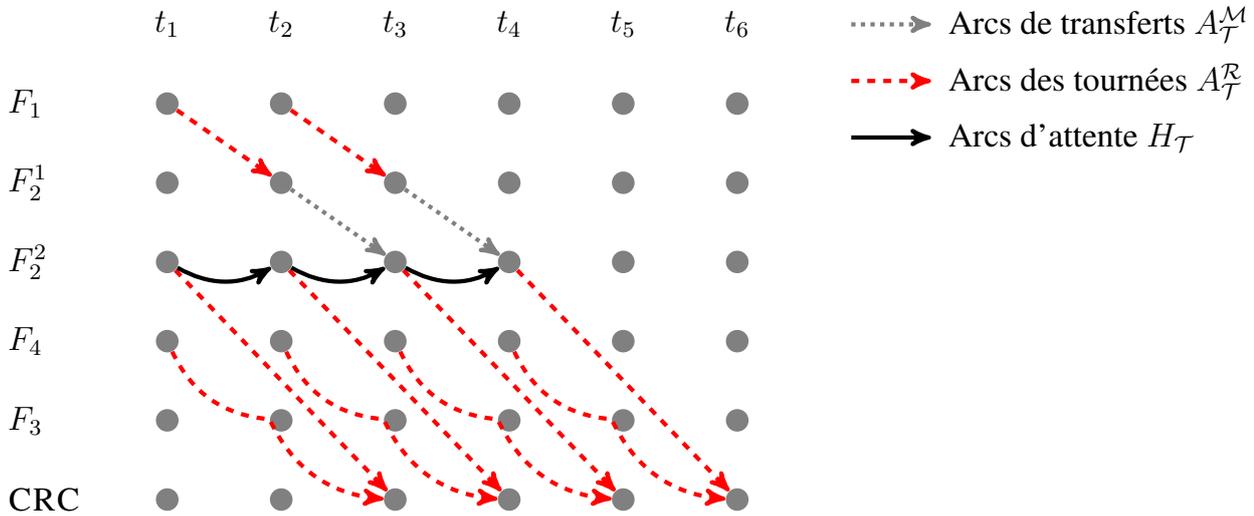


Figure 7.2 – Tournées amont dans le réseau espace-temps

7.2.2 Coût des tournées dans le réseau amont

La tarification des tournées est la même que celle utilisée dans le chapitre 5 (section 5.2.2.1). Le coût d'une tournée dépend de la distance parcourue et de sa charge. On considère une fonction linéaire par morceaux représentant une flotte homogène de véhicules pour chaque tournée (voir plus loin la figure 7.5). La flotte de véhicules est supposée illimitée. En pratique, les modèles mathématiques considèrent un nombre de véhicules suffisant pour répondre à la demande. Contrairement à la grande majorité des approches de la littérature, le coût de la tournée ne se calcule pas en sommant le coût de chacun de ses segments. Il est calculé en considérant la charge maximale transportée par le véhicule.

Chaque fonction linéaire par morceaux représentant le tarif d'une tournée est constituée d'un ensemble S_r de segments. Pour chaque segment $s \in S_r$, α_s^r et f_s^r sont respectivement le coefficient directeur et l'ordonnée à l'origine de chaque segment s de la fonction linéaire par morceaux.

7.3 Modélisation du réseau aval

Cette section détaille le fonctionnement et la modélisation du réseau aval ainsi que sa structure tarifaire.

7.3.1 Description et modélisation du réseau

L'objectif du réseau aval est de transporter un ensemble de commandes depuis chaque CRC jusqu'aux clients, grâce à des tournées de véhicules. Aucun transfert de marchandise entre deux véhicules n'est autorisé dans le réseau aval, il n'y a donc pas besoin de recourir à réseau espace temps. Comme dans le

chapitre 6, nous considérons un réseau plat dans lequel chaque commande est modélisée par un sommet. Il n'est pas possible de discrétiser le temps en aval : les tournées réalisées doivent être planifiées de manière très fine, car tout retard peut entraîner un refus de la commande par le client. Contrairement au réseau amont, il ne peut donc pas être discrétisé à grosses mailles. Nous considérons, comme dans le chapitre 6, un réseau aval dont les sommets représentent les commandes des clients. Deux commandes à destination du même client sont donc représentées par deux sommets liés par une arête de longueur nulle.

Exemple 15. La figure 7.3 représente un exemple de réseau aval avec trois commandes et deux clients. La commande o_1 comporte 16 palettes et doit être livrée entre 12h et 18h. La commande o_2 comporte 4 palettes et doit être livrée à 9h, et la commande o_3 comporte 12 palettes et doit être livrée entre 9h et 16h.

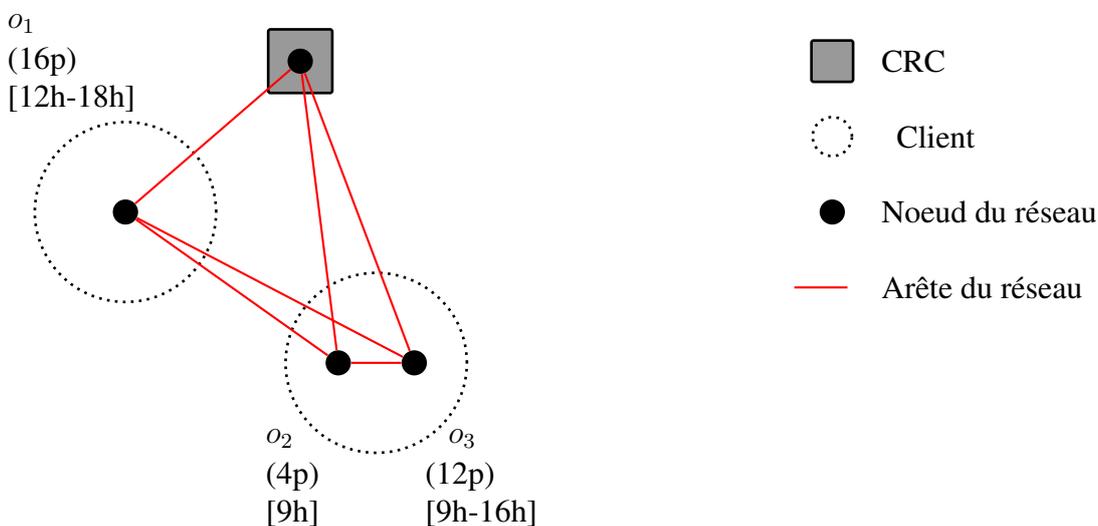


Figure 7.3 – Modélisation du réseau aval

7.3.2 Coûts des tournées dans le réseau aval

Dans le réseau aval, les tournées de véhicules sont composées d'un CRC et d'un ou plusieurs clients. Pour qu'une tournée soit réalisable, il faut qu'elle respecte les contraintes suivantes :

- le respect des fenêtres horaires des commandes transportées ;
- une durée inférieure ou égale à la durée maximale du transport ;
- un temps d'attente total inférieur ou égal au temps d'attente maximal ;
- un détour inférieur ou égal au détour maximal.

Ces contraintes ont été détaillées au chapitre 6, en section 6.1.3.4. Le tarif des tournées dans le réseau aval est la même que dans le chapitre 5, en section 5.2.2.2. Le coût d'une tournée correspond donc au coût de la zone visitée la plus lointaine, additionné à un coût d'ouverture de porte à chaque arrêt.

7.4 Commandes et sous-commandes

Soit O l'ensemble des commandes. Une commande $o \in O$ peut-être composée de plusieurs produits ayant des origines différentes. Une commande o est donc définie par un ensemble de sous-commandes L_o , une destination et une fenêtre horaire de livraison. Cette fenêtre horaire peut correspondre soit à la plage horaire d'ouverture de la plateforme du client, soit à un horaire imposé (rendez-vous). Une sous-commande $l \in L_o$ est caractérisée par une origine (site fournisseur), une fenêtre horaire de mise à disposition, un nombre q_l de palettes, un besoin ou non de préparation et sa commande d'appartenance. Contrairement à ce qui a été présenté dans le chapitre 6, nous supposons ici que tous les produits sont compatibles entre eux et peuvent donc être transportés dans le même véhicule. Nous rappelons que l'ensemble des sous-commandes d'une commande doivent être regroupées sur un CRC afin d'être livrées par le même véhicule au client.

L'itinéraire des marchandises se divise en deux parties.

- L'itinéraire des sous-commandes d'une commande depuis leur origine jusqu'à un CRC. Cet itinéraire comporte des transferts ainsi que des préparations de palettes.
- L'itinéraire de la commande depuis le CRC jusqu'au client.

Nous rappelons que le besoin de préparation d'une sous-commande se traduit par un transfert sur un site de son fournisseur capable de réaliser de la préparation de palettes. Le temps d'attente entre un déchargement et un rechargement d'une sous-commandes sur une plateforme de transfert n'est pas limité. En revanche, un temps minimal de transfert t_{min} est défini pour représenter les temps de chargement/déchargement et rendre les synchronisations plus réalistes et robustes. Un coût de transfert c_{ij} fixe par palette représente les coûts de manutention pour chaque transfert modélisé par l'arc $(i, j) \in A^M$ dans le réseau plat. On considère que le coût d'un transfert reste constant quelque soit la période de temps à laquelle a lieu le transfert. Un coût horaire à la palette h_{stock} est ajouté pour le stockage à court terme des marchandises au CRC. L'exemple 16 illustre le parcours de trois commandes et de leurs sous-commandes dans le réseau amont et aval.

Exemple 16. *On considère 3 commandes :*

- Une commande o_1 à destination du client d_1 entre 12h et 18h et composée de deux sous-commandes : l_1 comportant 6 palettes devant partir de son site fournisseur F_1 entre 13h à 17h le jour précédent la livraison ($J-1$) et ayant besoin de préparation ; et l_2 constituée de 10 palettes au départ de F_4 , mise à disposition de 21h à 1h à $J-1$ et n'ayant pas besoin de préparation.
- Une commande o_2 à destination du client d_2 à 9h, composée d'une sous-commande l_3 de 4 palettes devant partir de son site fournisseur F_4 entre 21h et 1h à $J-1$ et n'ayant pas besoin de préparation.
- Une commande o_3 à destination du client d_2 entre 9h et 16h, composée d'une sous-commande l_4 de 12 palettes qui doit partir du site F_3 entre 13h et 21h à $J-1$ et n'ayant pas besoin de préparation.

Le réseau amont est discrétisé en périodes de 4h sur un horizon de temps allant de 13h à $J-1$ et 9h à J . La sous-commande l_1 doit partir du site F_1 à la période t_1 ou t_2 , les sous-commandes l_2 et l_3 doivent partir de F_4 à la période t_3 ou t_4 et la sous-commande l_4 doit partir de F_3 à la période t_1 , t_2 ou t_3 . Le seul site habilité à préparer des palettes est le site F_2 .

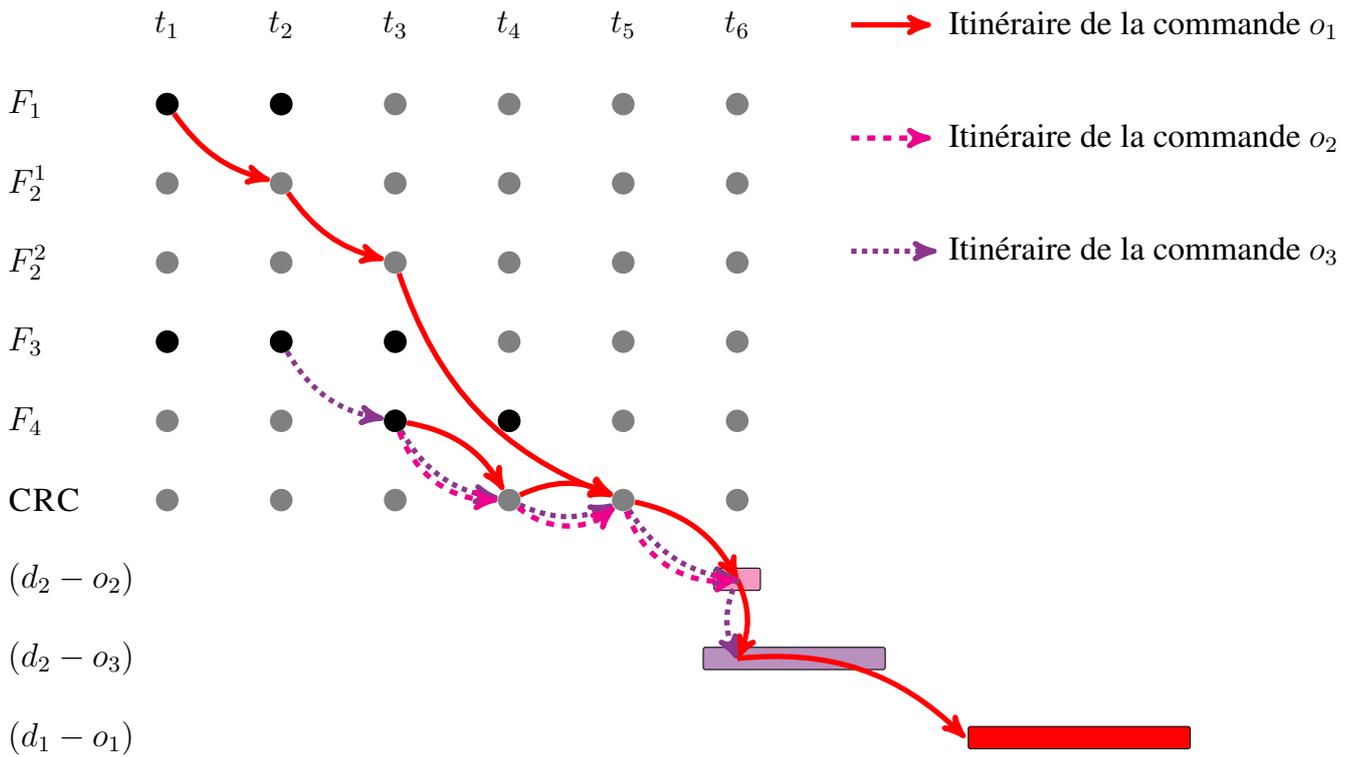


Figure 7.4 – Un exemple d’itinéraire dans le réseau amont

La figure 7.4 montre un exemple d’itinéraire de ces commandes. Les sommets noirs du réseau espace temps représentent les périodes auxquelles les sous-commandes doivent partir de leur site fournisseur. Afin de ne pas surcharger la figure, on ne représente pas dans ce réseau l’attente obligatoire au CRC représentant le chargement/déchargement des commandes au CRC. Chaque commande est représentée par un "sommets" appelé (nom du client - nom de la commande). Le réseau aval n’étant pas discrétisé, des rectangles représentent les plages horaires de livraison possibles pour chaque commande dans le réseau aval.

La sous-commande l_1 de la commande o_1 est tout d’abord acheminée à la période t_1 du site F_1 au site F_2 pour être préparée. La sous-commande est alors transférée et acheminée directement au CRC entre les périodes t_3 et t_5 . Une tournée passant par F_3 , F_4 et le CRC transporte les sous-commandes l_2, l_3 et l_4 jusqu’au CRC. Ensuite, une tournée passant par les deux clients livre les trois commandes dans leur fenêtre horaire respective.

7.5 Modélisation du problème $\mathcal{F}_{SND RP-CRC}$

Soit P l’ensemble des itinéraires dans le réseau amont. Un itinéraire $p \in P$ d’une sous-commande part du sommet physique correspondant à l’origine de la sous-commande et transite dans le réseau amont jusqu’à un CRC. Il est constitué d’un ensemble d’arcs $A_p \in A_{\mathcal{T}}^R \cup A_{\mathcal{T}}^M \cup A_{\mathcal{H}}$ du réseau espace-temps. On note $a_{((i,t),(j,\bar{t}))}^p$ la constante égale à 1 si l’itinéraire p emprunte l’arc du réseau espace-temps $((i,t),(j,\bar{t}))$. Enfin, t_p est égal à l’heure d’arrivée de l’itinéraire p au CRC et e_p^i est égal à 1 si l’itinéraire p arrive au

CRC i . L'ensemble des itinéraires proposés pour une sous-commande l est noté P_l .

On appelle K l'ensemble des tournées dans le réseau aval. Une tournée de véhicule dans le réseau aval $k \in K$ correspond à une liste ordonnée de commandes O_k , ainsi qu'une heure v_k de départ du CRC. c_k correspond au coût de la tournée k et b_k^i est égal à 1 si la tournée k part du CRC i . Enfin, on note K_o l'ensemble des tournées qui peuvent distribuer la commande o .

On définit deux types de variables qui représentent respectivement l'itinéraire d'une sous-commande dans le réseau amont et une tournée de véhicules dans le réseau aval.

- y_p^l est une variable binaire égale à 1 si l'itinéraire amont $p \in P_l$ est choisi pour la sous-commande $l \in L$.
- x_k est une variable binaire égale à 1 si la tournée $k \in K$ est sélectionnée dans le réseau aval.

Deux autres variables sont utilisées dans le modèle, afin de linéariser le coût des tournées dans le réseau amont (voir chapitre 3 section 3.2).

- $w_r^s, r \in R, s \in S_r$ est une variable réelle représentant la charge maximale de la tournée r sur le segment s . Si la charge n'est pas comprise dans l'intervalle de charge du segment s , la variable w_r^s est égale à 0.
- $\xi_r^s, r \in R, s \in S_r$ est une variable binaire égale à 1 si la charge maximale sur la tournée r appartient à l'intervalle de charge du segment $s \in S_r$.

7.5.1 Modèle $\mathcal{F}_{SND RP-CRC}$

Le problème opérationnel de transport dans le réseau mutualisé est modélisé par le PLNE (7.1)-(7.12), noté $\mathcal{F}_{SND RP-CRC}$.

$$\begin{aligned} \min \sum_{r \in R} \sum_{s \in S_r} (\alpha_s^r w_r^s + f_s^r \xi_r^s) & \quad (7.1) \\ + \sum_{l \in L} \sum_{p \in P_l} \sum_{((i,t),(j,\bar{t})) \in A_r^M} a_{((i,t),(j,\bar{t}))}^p c_{ij} q_l y_p^l & \\ + \sum_{k \in K} x_k c_k & \\ + h_{stock} \times \sum_{o \in O} \left(\sum_{k \in K_o} v_k q_o x_k - \sum_{l \in L_o} \sum_{p \in P_l} t_p q_l y_p^l \right) & \end{aligned}$$

s.c.

$$\sum_{p \in P_l} y_p^l = 1 \quad \forall l \in L \quad (7.2)$$

$$\sum_{k \in K_o} x_k = 1 \quad \forall o \in O \quad (7.3)$$

$$\sum_{k \in K_o} b_k^i x_k = \sum_{p \in P_l} e_p^i y_p^l \quad \forall i \in N_{crc}, \forall o \in O, \forall l \in L_o \quad (7.4)$$

$$\sum_{k \in K_o} v_k x_k - \sum_{p \in P_l} t_p y_p^l \geq t_{min} \quad \forall o \in O, \forall l \in L_o \quad (7.5)$$

$$\sum_{s \in S_r} w_r^s \geq \sum_{p \in P_l} a_{((i,t),(j,\bar{t}))}^p q_l y_p^l \quad \forall r \in R, \forall ((i,t),(j,\bar{t})) \in A_r \quad (7.6)$$

$$m_{(s-1)}^r \xi_r^s \leq w_r^s \leq m_s^r \xi_r^s \quad r \in R, \forall s \in S_r \quad (7.7)$$

$$\sum_{s \in S_r} \xi_r^s \leq 1 \quad \forall r \in R \quad (7.8)$$

$$y_p^l \in \{0; 1\} \quad \forall l \in L, \forall p \in P_l \quad (7.9)$$

$$\xi_r^s \in \{0; 1\} \quad \forall r \in R, \forall s \in S_r \quad (7.10)$$

$$w_r^s \in \mathbb{R}^+ \quad \forall r \in R, \forall s \in S_r \quad (7.11)$$

$$x_k \in \{0; 1\} \quad \forall o \in O, \forall r \in R_o. \quad (7.12)$$

Les quatre termes de la fonction objectif (7.1) représentent le coût des tournées amont (défini dans le chapitre 3 section 3.2), le coût des transferts aux sites fournisseurs du réseau amont, le coût des tournées aval et les coûts d'attente aux CRC.

Les contraintes (7.2) associent un unique itinéraire amont p pour chaque sous-commande $l \in L$. Les contraintes (7.3) choisissent une unique tournée $k \in K_o$ pour distribuer chaque commande $o \in O$. Ce sont les deux contraintes de *set partitioning*. Les contraintes (7.4) représentent la synchronisation dans l'espace entre les réseaux amont et aval : il assure que l'itinéraire choisi pour une sous-commande l arrive au même CRC que le CRC de départ de sa commande associée. Les contraintes (7.5) garantissent la synchronisation au niveau temporel : la durée de séjour d'une sous-commande au CRC doit être supérieure ou égale au temps de transfert minimum t_{min} . Enfin, les contraintes (7.6) (7.7) et (7.8) linéarisent le coût des tournées amont en fonction de leur charge (voir chapitre 3 3.2).

7.5.2 Modélisation pour la génération de colonnes $\mathcal{F}^{\mathcal{L}}_{SND RP-CRC}$

Dans cette section, nous présentons le modèle linéaire utilisé dans l'heuristique de génération de colonnes. Nous nous appuyons sur des résultats de la littérature pour proposer un modèle linéaire $\mathcal{F}^{\mathcal{L}}_{SND RP-CRC}$ plus performant pour la génération de colonnes que la relaxation linéaire de $\mathcal{F}_{SND RP-CRC}$.

7.5.2.1 Résultats théoriques concernant l'utilisation d'une fonction linéaire par morceaux

Croxton et al. [32] comparent trois modélisations de la fonction linéaire par morceaux. Ils montrent que la relaxation linéaire de ces trois modélisations est équivalente et correspond à l'enveloppe convexe inférieure de la fonction linéaire par morceaux. En outre, ils observent que si la fonction linéaire par morceaux est concave, alors l'enveloppe convexe inférieure est une fonction linéaire $g(x) = a \times x$ qui passe par le point de charge maximale.

Moccia et al. [75] étendent cette propriété. Ils définissent pour chaque segment une fonction linéaire $g^s(x) = a^s \times x$ qui passe par le point de charge maximale du segment m_s . Si la valeur minimale des a^s , notée \bar{a} , est associée au dernier segment de la fonction, ils montrent que l'enveloppe convexe inférieure est linéaire et de la forme $\bar{a} \times x$. En outre, Moccia et al. [75] étudient un **SNDP** dans un réseau multimodal. L'objectif est de trouver l'itinéraire des commandes dans un réseau, dont les coûts

sur les arcs sont linéaires par morceaux. Ils comparent trois formulations utilisées pour la génération de colonnes :

- une formulation \mathcal{F}_1 qui correspond à la formulation **MCM** de la fonction linéaire par morceaux ;
- une formulation \mathcal{F}_2 qui approxime par une droite l'enveloppe convexe inférieure de la fonction linéaire par morceaux ;
- une formulation \mathcal{F}_3 qui intègre les inégalités valides introduites par Croxton et al. [34].

Un algorithme de génération de colonnes génère un ensemble de variables pour chacune de ces formulations, et une solution entière est obtenue grâce à un solveur. Les résultats montrent que l'heuristique basée la génération de colonne avec la formulation \mathcal{F}_2 est la plus performante, en terme de qualité de la solution comme de temps de résolution. Nous nous basons donc sur ces résultats.

7.5.2.2 Propriétés de la fonction linéaire par morceaux utilisée dans cette thèse

La fonction linéaire par morceaux utilisée pour calculer le prix d'une tournée représente le prix d'une flotte de véhicules (voir chapitre 3 section 3.2). La figure 7.5 illustre le cas d'une tournée nécessitant une flotte de 4 véhicules de capacité Q . La flotte étant homogène, les véhicules ont tous le même tarif, l'enveloppe convexe inférieure est donc la même fonction linéaire pour tous. La droite représentant l'enveloppe convexe inférieure de la fonction linéaire par morceaux passe par tous les points qui représentent un véhicule plein. On appelle δ_r le coefficient directeur de cette droite pour une tournée $r \in R$. Le coût d'une tournée (réalisée avec un ou plusieurs véhicules) est donc calculé en multipliant la charge totale par δ_r .

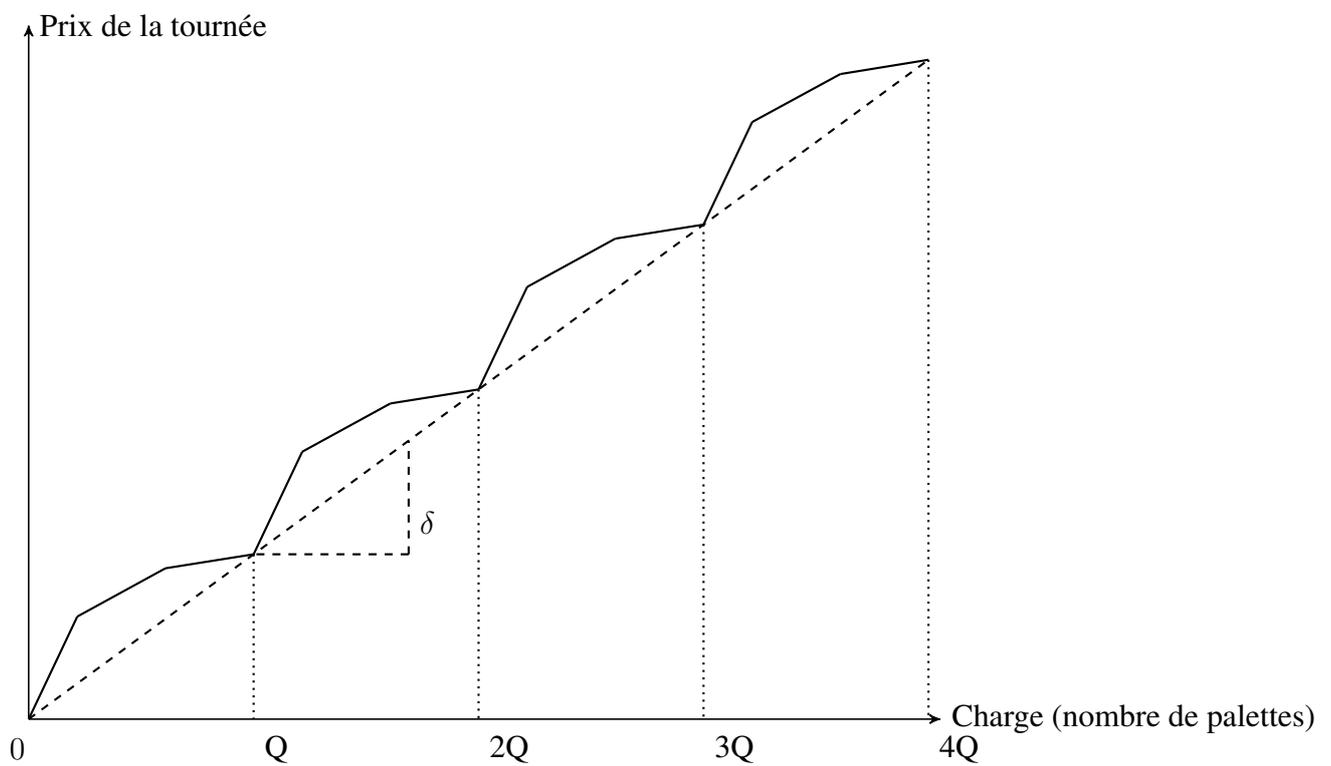
7.5.2.3 Modélisation $\mathcal{F}^{\mathcal{L}}_{SND RP-CRC}$

Nous introduisons une nouvelle variable $\omega_r \in \mathbb{R}^+$ qui représente la charge maximale de la tournée r . Le modèle linéaire $\mathcal{F}^{\mathcal{L}}_{SND RP-CRC}$ utilisé dans la génération de colonnes est le suivant :

$$\begin{aligned}
 \min \quad & \sum_{r \in R} \delta_r \omega_r \\
 & + \sum_{l \in L} \sum_{p \in P_l} \sum_{((i,t),(j,\bar{t})) \in A_{\mathcal{L}}^M} a_{((i,t),(j,\bar{t}))}^p c_{ij} q_l y_p^l \\
 & + \sum_{k \in K} c_k x_k \\
 & + h_{stock} \times \left(\sum_{o \in O} \sum_{k \in K_o} v_k q_o x_k - \sum_{l \in L_o} \sum_{p \in P_l} t_p q_l y_p^l \right)
 \end{aligned} \tag{7.13}$$

s.c.

$$\sum_{p \in P_l} y_p^l = 1 \quad \forall l \in L \tag{7.14} \quad (\sigma_l)$$

Figure 7.5 – Prix d'une tournée avec quatre véhicules de capacité Q

$$\sum_{k \in K_o} x_k = 1 \quad \forall o \in O \quad (\lambda_o) \quad (7.15)$$

$$\sum_{k \in K_o} b_k^i x_k = \sum_{p \in P_l} e_p^i y_p^l \quad \forall i \in N_{crc}, \forall o \in O, \forall l \in L_o \quad (\chi_{l,i}) \quad (7.16)$$

$$\sum_{k \in K_o} v_k x_k - \sum_{p \in P_l} t_p y_p^l \geq t_{min} \quad \forall o \in O, \forall l \in L_o \quad (\tau_l) \quad (7.17)$$

$$\omega_r \geq \sum_{p \in P_l} a_{((i,t),(j,\bar{t}))}^p q_l y_p^l \quad \forall r \in R, \forall ((i,t),(j,\bar{t})) \in A_r \quad (\pi_{((i,t),(j,\bar{t}))}^r) \quad (7.18)$$

$$y_p^l \in [0; 1] \quad \forall l \in L, \forall p \in P_l \quad (7.19)$$

$$\omega_r \in \mathbb{R}^+ \quad \forall r \in R, \forall s \in S_r \quad (7.20)$$

$$x_k \in [0; 1] \quad \forall o \in O, \forall r \in R_o. \quad (7.21)$$

Le premier terme de la fonction objectif représente le coût des tournées amont grâce à la fonction linéaire dont le coefficient directeur est δ_r . Les contraintes (7.14) à (7.17) sont équivalentes aux équations de (7.2) à (7.5). La contrainte (7.18) calcule la charge maximale de la tournée r . Les variables duales $\sigma_l \leq 0 \in \mathbb{R}$, $\lambda_o, \chi_{l,i} \in \mathbb{R}$, $\tau_l \leq 0$ et $\pi_{((i,t),(j,\bar{t}))}^r$ sont associées aux équations respectives (7.2), (7.3), (7.4), (7.5), et (7.18). Notons que la formulation $\mathcal{F}_{SNDRP-CRC}^{\mathcal{L}}$ et la relaxation linéaire de $F_{SNDRP-CRC}$ sont équivalentes.

7.6 Matheuristique pour la résolution du SND RP-CRC

Pour résoudre le problème du SND RP-CRC, nous proposons une méthode de résolution basée sur une double génération de colonnes. Le modèle $\mathcal{F}_{SNDRP-CRC}^{\mathcal{L}}$ et un sous-problème qui génère de nouvelles variables sont résolus itérativement. Dans le réseau amont, la génération de colonnes génère des itinéraires de sous-commandes (variables y_p^l). Un sous-problème est résolu à chaque itération pour chaque sous-commande. Dans le réseau aval, nous générons des tournées de véhicules (variables x_k). Un sous-problème est résolu à chaque itération pour chaque CRC. Les colonnes générées viennent enrichir l'ensemble des colonnes déjà présentes dans le problème maître. Une fois la génération de colonnes terminée (le critère d'arrêt est un temps maximal d'exécution), les colonnes générées sont ajoutées à la formulation $\mathcal{F}_{SNDRP-CRC}$, qui est résolue optimalement à l'aide d'un solveur. La figure 8.6 résume les étapes principales de l'approche proposée.

7.6.1 Solution initiale

Un itinéraire pour chaque commande $o \in O$ est généré de manière heuristique :

1. Le CRC le plus proche du client est identifié.
2. Pour chaque sous-commande $l \in L_o$ n'ayant pas besoin de préparation, un itinéraire correspondant au plus court chemin depuis son origine vers le CRC identifié est créé et ordonnancé au plus tôt.
3. Pour les sous-commandes ayant besoin de préparation, le site de préparation le plus proche de son origine est identifié. On crée ensuite un itinéraire constitué du plus court chemin entre l'origine

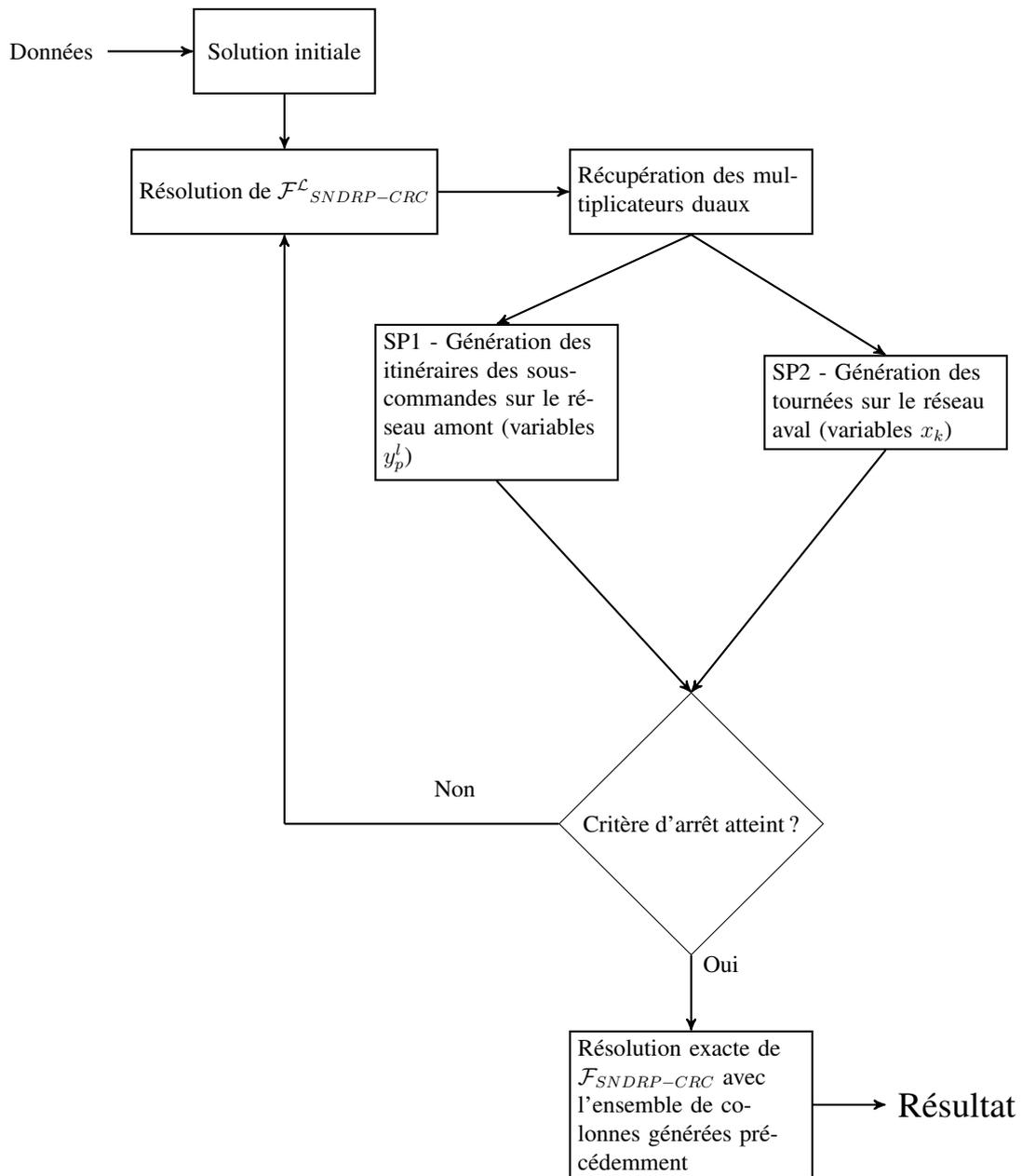


Figure 7.6 – Schéma général de l'algorithme de génération de colonnes

de la sous-commande et le site de préparation identifié, et du plus court chemin entre ce site de préparation et le CRC de destination. Ces itinéraires sont planifiés au plus tôt.

4. Pour le réseau aval, un ensemble de colonnes est généré selon la méthode décrite dans le chapitre 6 section 6.3.2. Toutes les possibilités de tournées comprenant un client unique (qui peut recevoir plusieurs commandes) sont énumérées.

7.6.2 SP1 – Génération des itinéraires amont

7.6.2.1 Formulation

Les variables y_p^l sont présentes dans les contraintes (7.2), (7.4), (7.5) et (7.18). On sélectionne donc les multiplicateurs duaux $\sigma_l, \tau_l, \pi_{((i,t),(j,\bar{t}))}^r$, et $\chi_{l,i}$ associés à ces contraintes afin de calculer le coût réduit. Le coût réduit d'un itinéraire amont est calculé comme suit :

$$\begin{aligned} \bar{c}_p^l = & \sum_{((i,t),(j,\bar{t})) \in A_{\mathcal{T}}^M} (a_{((i,t),(j,\bar{t}))}^p c_{i,j} q_l) - (t_p q_l h_{stock}) \\ & - \sigma_l - \sum_{i \in N_{erc}} a_p^i \chi_{l,i} + t_p \tau_l + \sum_{r \in R} \sum_{((i,t),(j,\bar{t})) \in A_r} a_{((i,t),(j,\bar{t}))}^p q_l \pi_{((i,t),(j,\bar{t}))}^r. \end{aligned} \quad (7.22)$$

La première ligne de cette équation correspond au coût d'une variable y_p^l dans la fonction objectif (7.13). Les quatre termes suivants représentent les coûts associés aux variables duales.

L'équation (7.22) correspond à la fonction objectif à minimiser du sous-problème SP1. Dans cette équation, le terme $-\sigma_l$ dépend uniquement de la sous-commande sélectionnée. On résout un sous-problème par sous-commande donc ce terme est une constante dans la fonction objectif de SP1.

On décompose les coûts de cette équation sur les arcs. Les coûts sont différents selon le type d'arc composant l'itinéraire (arcs de transfert ou arcs de transport) :

Le terme $\sum_{((i,t),(j,\bar{t})) \in A_{\mathcal{T}}^M} (a_{((i,t),(j,\bar{t}))}^p c_{i,j} q_l)$ représente la somme des coûts modifiés des arcs de transfert empruntés par la sous-commande l sur l'itinéraire $p \in P_l$.

Le terme $\sum_{((i,t),(j,\bar{t})) \in A_r} a_{((i,t),(j,\bar{t}))}^p q_l \pi_{((i,t),(j,\bar{t}))}^r$ est la somme des coûts modifiés des arcs de transport de l'itinéraire p .

On peut donc associer un coût modifié $\tilde{c}_l^{((i,t),(j,\bar{t}))}$ pour chacun des arcs de l'itinéraire p pour la sous-commande l , tel que :

$$\tilde{c}_l^{((i,t),(j,\bar{t}))} = \begin{cases} c_{ij} q_l & \text{si } ((i,t), (j,\bar{t})) \in A_{\mathcal{T}}^M \\ \pi_{((i,t),(j,\bar{t}))}^r q_l & \text{si } ((i,t), (j,\bar{t})) \in A_r, r \in R. \end{cases} \quad (7.23)$$

On note qu'aucun coût de la fonction objectif de $\mathcal{F}^{\mathcal{L}}_{SNRT-CRC}$ n'est reporté sur les arcs $a \in A_r$ modélisant le transport par les tournées $r \in R$. En effet, le calcul du coût d'une tournée est basée sur la valeur de la variable ω_r qui représente la charge maximale transportée par la tournée $r \in R$.

Dans la formulation (7.22), nous factorisons t_p . En définissant $\tilde{c}_l^T = (\tau_l - q_l h_{stock})$, il est possible d'associer un coût à l'horaire d'arrivée au CRC. Enfin, le choix du CRC d'arrivée e_p^i est associé uniquement au coût du multiplicateur dual $-\chi_{l,i}$.

On appelle \bar{c}_p^l le coût modifié d'un itinéraire amont, égal à :

$$\tilde{c}_p^j = \sum_{((i,t),(j,\bar{t})) \in A_T^R \cup A_T^M} a_{((i,t),(j,\bar{t}))}^p \tilde{c}_l^{((i,t),(j,\bar{t}))} + c_l^T t_p - \sum_{i \in N_{crc}} e_p^i \chi_{l,i}. \quad (7.24)$$

Pour trouver une colonne avec un coût réduit négatif, on cherche donc un itinéraire amont tel que :

$$\tilde{c}_p^j - \sigma_l < 0 \quad (7.25)$$

L'objectif du sous-problème SP1 est donc de trouver la plus petite valeur de $\tilde{c} - \sigma_l$, c'est-à-dire l'itinéraire minimisant la valeur de \tilde{c}_p^j . Le sous-problème SP1 est donc équivalent à un problème de plus court chemin sur le réseau espace-temps, auquel il faut ajouter les coûts liés à l'heure ainsi qu'au CRC d'arrivée.

7.6.2.2 Algorithme de programmation dynamique

Le sous-problème SP1 est résolu avec un algorithme de programmation dynamique ou algorithme de correction de label. On utilise l'algorithme de Bellman-Ford [9] [42] dans le réseau espace-temps, pour chaque sous-commande. Il permet de trouver le plus court chemin d'un sommet de départ vers tous les sommets du réseau en temps polynomial. Si le problème ne contient pas de cycles de longueur négative, ce qui est toujours le cas pour un réseau espace temps, la complexité de l'algorithme est $O(|N|)$ où $|N|$ est le nombre de sommets du réseau espace-temps. Pour chaque sous-commande, un sommet virtuel de départ est créé ainsi qu'un arc entre ce sommet et chaque sommet de départ possible de la sous-commande.

L'algorithme de Bellman-Ford associe à chaque sommet une valeur (appelée label). Ces labels sont initialement fixées à l'infini et la valeur du sommet de départ à 0. A chaque itération, l'algorithme étend les labels existants, c'est-à-dire qu'il visite les successeurs des sommets dont le label n'est pas $+\infty$. Deux étapes sont nécessaires pour étendre un label noté LAB_1 d'un sommet vers un de ses successeurs dont le label est noté LAB_2 .

1. Le coût du label LAB_1 est additionné au coût de l'arc rejoignant son successeur. Un nouveau label appelé LAB_3 est créé et associé à ce coût.
2. Si le coût de LAB_3 est inférieur (en cas de minimisation) au coût du label LAB_2 , alors le label LAB_3 remplace le label LAB_2 . Sinon, le label LAB_3 est détruit.

Dans notre sous-problème, nous distinguons deux cas :

- Si la sous-commande n'a pas besoin de préparation de commande, alors l'algorithme décrit précédemment est appliqué. A la fin de l'algorithme, la partie du coût réduit correspondant à l'heure d'arrivée et au choix du CRC sont ajoutés au coût de chaque chemin arrivant à un CRC. Tous les chemins ayant un coût réduit négatif sont ajoutés au problème maitre.
- Si la sous-commande a besoin de préparation, alors il faut définir deux labels par sommet. Le premier label correspond aux itinéraires qui ne sont pas passés par un site réalisant de la préparation de commande. Le second est réservé aux itinéraires qui ont déjà visité un site de préparation de commandes. Les relations de dominance ne sont pas uniquement dépendantes du coût du label, mais dépendent également du fait que l'itinéraire ait déjà visité ou non un site de préparation.

- Si deux labels ont le même *état*, c'est-à-dire si tous deux ont déjà visités ou non un site de préparation de commandes, alors il est possible d'éliminer le label de coût supérieur.
- Si deux labels n'ont pas le même état, et que le label de coût inférieur est déjà passé par un site de préparation, alors il est possible d'éliminer le label de coût supérieur. Sinon, il n'est pas possible d'établir de relation de dominance, donc d'éliminer un des deux labels.

À la fin de l'algorithme, la partie des coûts réduits correspondant à l'heure d'arrivée et de choix de CRC sont ajoutés. Les itinéraires de coût réduit négatif et passant par un site de préparation sont ajoutés au problème maître.

On associe un réseau espace-temps différent à chaque sous-commande $l \in L$. En effet, on souhaite générer pour chaque sous-commande un réseau espace temps où on filtre les sommets pour lesquels il est certain que la sous-commande n'arrive pas assez tôt chez son client. En pré-traitement, le plus court chemin entre chaque sommet et le client de la commande o est calculé. Si ce plus court chemin ne permet pas d'arriver au client avant l'heure d'arrivée au plus tard, alors le sommet est écarté du graphe espace temps.

7.6.3 SP2 – Génération des tournées de véhicules aval

7.6.3.1 Formulation

Les variables x_k sont présentes dans les contraintes (7.3), (7.4), (7.5) du problème maître. On retient donc les multiplicateurs duaux λ_o , $\chi_{l,i}$ et τ_l associés à ces contraintes pour calculer le coût réduit d'une nouvelle tournée k . Le coût réduit \bar{c}_k d'une tournée est calculé comme suit :

$$\begin{aligned} \bar{c}_k = & c_k + \sum_{o \in O_k} q_o h_{stock} v_k \\ & - \sum_{o \in O_k} \sum_{l \in L_o} \lambda_l + \sum_{i \in N_{crc}} \sum_{o \in O} \sum_{l \in L_o} b_k^i \chi_{l,i} - \sum_{o \in O_k} \sum_{l \in L_o} \tau_l. \end{aligned} \quad (7.26)$$

La première ligne de cette équation correspond au coût réel d'une tournée k dans la fonction objectif (7.13) et les trois termes suivants représentent les coûts associés aux variables duales. On peut regrouper les différents termes du coût réduit pour obtenir :

$$\bar{c}_k = c_k + \sum_{o \in O_k} (q_o h_{stock} v_k - (\sum_{l \in L_o} (\lambda_l + \tau_l) - \sum_{i \in N_{crc}} \chi_{l,i} b_k^i)). \quad (7.27)$$

Ainsi pour chaque nouvelle commande $o \in O$ d'une tournée k , il faut ajouter le terme $(v_k q_o h_{stock} - (\sum_{l \in L_o} (\lambda_l + \tau_l) - \sum_{i \in N_{crc}} b_k^i \chi_{l,i}))$ au calcul du coût réduit, ainsi que calculer de nouveau le coût réel c_k de la tournée. L'objectif du sous-problème SP2 est de trouver une tournée minimisant le coût réduit (7.27). La résolution de ce sous-problème revient alors à trouver le plus court chemin entre un CRC et les sommets représentant les commandes dans le réseau aval en respectant les capacités des véhicules, les fenêtres horaires, et les contraintes sur le temps total de la tournée ainsi que le temps d'attente maximal.

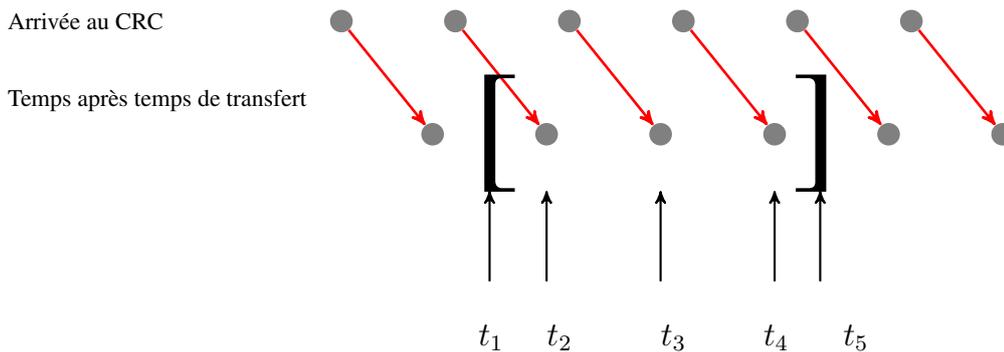


Figure 7.7 – Synchronisation entre réseau amont et réseau aval

7.6.3.2 Résolution

Pour résoudre le sous-problème SP2, nous utilisons une méthode tabou similaire à celle présentée au chapitre 6. Aucune méthode exacte n'a été développée car le critère de dominance est trop faible de par les contraintes riches du problèmes de tournées de véhicules (principalement à causes des contraintes sur les temps d'attente et détour). On utilise un algorithme proche de celui décrit au chapitre 6, section 6.3.3.1. Les différences sont les suivantes :

- Le tarif lié à la sous-traitance est toujours la même dans ce chapitre, et la flotte est homogène. Nous ne considérons donc pas différents types de véhicules.
- Les seuls voisinages utilisés sont le *retrait* et *insertion* décrits section 6.3.3.4 pour des véhicules sans multi-trip.
- La synchronisation avec l'itinéraire amont ainsi que le coût de stockage ne permettent pas de réaliser uniquement un ordonnancement au plus tôt ou au plus tard : le coût réduit dépend de l'horaire de départ du CRC. Cet horaire de départ de la tournée est multiplié par $q_o \times h_{stock} - \sum_{l \in L_o} \tau_l$. Cette valeur peut-être positive ou négative selon la valeur de la somme des multiplicateurs duaux τ_l . On réalise donc un ordonnancement au plus tôt, puis un ordonnancement au plus tard, pour obtenir l'intervalle $[D_{min}; D_{max}]$ des heures de départ possible de la tournée. Ensuite, on calcule dans quel intervalle $[I_{min}; I_{max}] \subset [D_{min}; D_{max}]$ le coût réduit de la colonne est négatif. Afin de réaliser la synchronisation entre le réseau amont et aval, on retient les horaires d'arrivée sur CRC correspondant aux sommets discrétisés du réseau espace temps en leur ajoutant le temps minimum de transfert t_{min} . On obtient ainsi l'ensemble discret $\{d_1, \dots, d_n\}$. On retient parmi les valeurs de l'ensemble $\{d_1, \dots, d_n\}$, celles qui sont situées dans l'intervalle $[I_{min}; I_{max}]$. On retient également les valeurs extrêmes I_{min} et I_{max} . Une tournée par horaire de départ est créée.

Exemple 17. La figure 7.7 représente un exemple de synchronisation temporelle entre le réseau amont et le réseau aval. La première ligne de sommets représente les périodes possibles d'arrivée au CRC dans le réseau amont. La deuxième ligne illustre le décalage dû au temps minimal de transfert. Les deux crochets représentent l'intervalle $[I_{min}; I_{max}]$. Cinq horaires de départ t_1 à t_5 sont sélectionnés, et ainsi cinq colonnes sont ajoutées au problème maître.

7.7 Tests numériques

Cette section détaille les résultats des expérimentations numériques de la matheuristique basée sur la génération de colonnes.

Dans un premier temps, nous décrivons la génération des instances utilisées pour les tests en section 7.7.1 et détaillons le déroulement des expérimentations. Les sections suivantes indiquent les résultats tout d'abord liés au paramétrage de la recherche tabou, puis en fonction du niveau de discrétisation ainsi que du coût de stockage.

7.7.1 Génération des instances

Les réseaux logistiques utilisés sont les mêmes que ceux décrits au chapitre 5. On utilise donc trois réseaux illustrés par les figures 5.11, 5.12 et 5.13. A partir de ces réseaux, nous avons généré des instances avec les paramètres suivants :

- un nombre de sites fournisseurs égal à 6 ou 8 ;
- un nombre de CRC égal à 1, 3 ou 4 ;
- un nombre de commandes qui varie entre 30 et 100 ;
- un nombre de sous-commandes par commande qui varie entre 1 et 4 de manière équiprobable.

Nous avons généré 10 instances, dont le nom résume les caractéristiques. Ainsi, la plus grosse instance, nommée 8_4_100_191, comporte 8 fournisseurs, 4 CRC, 100 commandes et 191 sous-commandes.

L'algorithme est codé en Java, et la librairie Cplex (version 12.51) est appelée pour la résolution des relaxations linéaires ainsi que la résolution du PLNE final. Chaque test est exécuté sur une machine Xeon X5650 à 2.67 GHz avec 64 GB de RAM.

Le critère d'arrêt de la génération de colonnes est une durée maximale, égale à 10 minutes. La durée de résolution du PLNE final est également limitée à 10 minutes.

7.7.2 Paramétrage de la recherche tabou

On souhaite tout d'abord évaluer la performances et la stabilité de la matheuristique, en fonction des paramètres de la recherche tabou qui résout le SP2. Nous faisons varier trois paramètres de la méthodes tabou :

- le nombre de fois qu'une recherche tabou est lancée (avec une solution initiale différente) par itération de la génération de colonnes $Max - RT$;
- le nombre d'itérations de la recherche tabou β ;
- la longueur de la liste tabou L_{tabou} .

On appelle *paramétrage de la recherche tabou* la valeur du triplet : $(Max - RT ; \beta ; L_{tabou})$. Chaque instance est résolue 5 fois avec chaque paramétrage.

Comme dans le chapitre 6, nous évaluons la stabilité de l'algorithme grâce à deux indicateurs :

- La différence, notée ΔMin , entre le coût minimal obtenu avec le paramétrage évalué et le coût minimal obtenu pendant l'ensemble des expérimentations.
- La différence, notée ΔMoy , entre le coût moyen sur les 5 exécutions obtenu avec le paramétrage évalué et le coût minimal obtenu pendant l'ensemble des expérimentations.

7.7.2.1 Nombre de recherche tabou par itération ($Max - RT$)

Le tableau 7.1 présente les résultats obtenus en fonction du nombre de recherches tabou par itération. Il indique les valeurs de ΔMin et ΔMax pour des valeurs de $Max - RT$ variant de 100 à 250. Les résultats montrent des variations assez faibles. La valeur 250 semble récolter les meilleures résultats, avec une moyenne de 0.3% pour le ΔMin et 1.0% pour le ΔMoy .

$Max - RT$	ΔMin				ΔMoy			
	100	150	200	250	100	150	200	250
6_1_30_58	0,0%	0,0%	0,0%	0,0%	0,4%	0,9%	0,7%	0,5%
6_3_47_101	1,4%	0,8%	1,5%	0,0%	2,4%	1,4%	2,0%	0,6%
6_3_63_123	0,0%	2,1%	0,7%	1,1%	2,4%	2,7%	2,2%	2,0%
6_4_47_98	0,7%	0,7%	0,2%	0,0%	1,7%	1,7%	2,2%	0,7%
6_4_63_128	0,3%	0,2%	0,0%	0,0%	0,9%	1,0%	0,8%	1,0%
8_1_50_94	0,2%	0,2%	0,0%	0,2%	0,7%	1,5%	0,3%	1,2%
8_3_75_146	0,0%	0,4%	0,5%	0,1%	0,8%	1,4%	1,6%	1,0%
8_3_100_203	0,0%	0,4%	0,1%	1,1%	1,1%	1,4%	1,4%	1,5%
8_4_75_153	0,0%	0,2%	0,2%	0,2%	0,3%	0,6%	0,5%	0,8%
8_4_100_191	1,0%	1,1%	0,1%	0,0%	1,3%	1,3%	0,8%	0,3%
Moyenne	0,4%	0,6%	0,3%	0,3%	1,2%	1,4%	1,2%	1,0%

Tableau 7.1 – Nombre de recherches tabou par itération

7.7.2.2 Nombre d'itérations de la recherche tabou

Le tableau 7.2 présente les résultats concernant la variation du nombre d'itérations de la recherche tabou. Nous avons testé trois valeurs de β : 70, 90 et 110. Les résultats montrent de très faibles variations : le ΔMin varie entre 0% et 0.3% et le ΔMoy entre 0.9% et 1.3%.

7.7.2.3 Longueur de la liste tabou

Le tableau 7.3 présente les variations du résultat selon la longueur de la liste tabou. Une fois de plus, ce paramètre a peu d'influence sur les résultats. Néanmoins, les valeurs 6 et 8 semblent être légèrement meilleures que les autres (0.2% pour les valeurs de ΔMin , 1.2% et 1.1% pour les valeurs du ΔMoy).

β	Δ Min			Δ Moy		
	70	90	110	70	90	110
6_1_30_58	0,0%	0,0%	0,0%	0,3%	0,7%	0,9%
6_3_47_101	2,4%	0,0%	1,2%	3,1%	1,0%	1,7%
6_3_63_123	0,3%	0,0%	0,2%	2,0%	1,2%	0,8%
6_4_47_98	0,2%	0,0%	0,0%	1,3%	1,2%	0,3%
6_4_63_128	0,4%	0,0%	0,5%	0,9%	1,0%	0,9%
8_1_50_94	0,0%	0,0%	0,0%	1,6%	1,2%	0,8%
8_3_75_146	0,0%	0,0%	0,1%	1,2%	1,1%	0,6%
8_3_100_203	0,0%	0,1%	0,5%	1,0%	1,1%	1,9%
8_4_75_153	0,0%	0,2%	0,2%	0,6%	0,7%	0,4%
8_4_100_191	0,1%	0,0%	0,5%	0,9%	0,8%	0,9%
Moyenne	0,3%	0,0%	0,3%	1,3%	1,0%	0,9%

Tableau 7.2 – Nombre d'itérations d'une recherche tabou

Longueur	Δ Min					Δ Moy				
	6	8	10	12	14	6	8	10	12	14
6_1_30_58	0,0%	0,0%	0,0%	0,0%	0,0%	1,6%	0,9%	0,9%	0,0%	0,5%
6_3_47_101	0,0%	0,3%	0,6%	0,6%	1,2%	1,7%	2,5%	1,2%	1,8%	2,2%
6_3_63_123	1,0%	0,0%	2,1%	2,5%	0,6%	1,7%	1,9%	2,6%	3,0%	1,6%
6_4_47_98	0,0%	0,0%	0,5%	1,3%	0,3%	1,2%	0,6%	1,6%	2,6%	1,5%
6_4_63_128	0,0%	0,5%	0,2%	0,5%	0,5%	0,9%	1,4%	1,0%	1,0%	1,3%
8_1_50_94	0,1%	0,1%	0,4%	0,0%	0,1%	2,1%	0,7%	1,7%	0,9%	0,7%
8_3_75_146	0,4%	0,5%	0,7%	0,0%	0,7%	0,6%	1,0%	1,7%	0,9%	1,9%
8_3_100_203	0,4%	0,3%	0,2%	0,4%	0,0%	1,0%	1,2%	1,1%	0,6%	0,6%
8_4_75_153	0,2%	0,4%	0,6%	0,7%	0,0%	0,6%	0,7%	1,0%	1,0%	0,8%
8_4_100_191	0,0%	0,1%	1,2%	0,6%	0,4%	0,9%	0,5%	1,4%	1,2%	1,2%
Moyenne	0,2%	0,2%	0,6%	0,7%	0,4%	1,2%	1,1%	1,4%	1,3%	1,2%

Tableau 7.3 – Longueur de la liste tabou

7.7.3 Écart entre la relaxation linéaire et la solution

Nous présentons maintenant le détail des résultats obtenus avec le paramétrage retenu : ($max - RT = 250$; $\beta = 110$; $Ltabou = 8$). Le tableau 7.4 indique deux types de résultat.

- l'écart relatif entre la valeur v_l de la fonction objectif de la dernière résolution de $\mathcal{F}_{SNDRP-CRC}^L$, et la valeur v de la fonction objectif obtenue après la résolution de $\mathcal{F}_{SNDRP-CRC}$ avec l'ensemble des colonnes générées. Cet écart est calculé avec la formule $(v - v_l)/v_l$
- L'écart relatif entre la valeur v et la borne inférieure LB_{Cplex} fournie par Cplex à la fin de la résolution de $\mathcal{F}_{SNDRP-CRC}$. On rappelle que la résolution du PLNE est limitée à 10 minutes. On calcule cet écart avec la formule $(v - LB_{Cplex})/LB_{Cplex}$.

Le but de ces tests est d'observer si d'une part, l'écart entre $\mathcal{F}_{SNDRP-CRC}$ et sa relaxation $\mathcal{F}_{SNDRP-CRC}^L$ n'est pas trop grand, et si les 10 minutes de résolution par Cplex permettent d'obtenir un résultat proche de l'optimum. Les valeurs de l'écart relatif entre v et v_l sont satisfaisantes car elles varient entre 2% et 3%, avec une moyenne à 2.9%. Nous trouvons des résultats au plus à 1.1% de l'optimum en moyenne. L'ensemble des 10 instances peut-être résolu en moins de 10 minutes, à 2% de l'optimum.

Instance	Écart relatif PL/PLNE	Écart relatif PLNE
6_1_30_58	3,7%	0,2%
6_3_47_101	2,9%	1,1%
6_3_63_123	2,5%	0,8%
6_4_47_98	2,7%	1,1%
6_4_63_128	3,1%	1,7%
8_1_50_94	2,3%	0,4%
8_3_75_146	3,7%	1,8%
8_3_100_203	2,3%	1,2%
8_4_75_153	2,8%	1,5%
8_4_100_191	2,3%	1,4%
Moyenne	2,9%	1,1%

Tableau 7.4 – Statistiques générales

7.7.4 Influence de la discrétisation

Dans cette section, nous cherchons à évaluer l'impact de la discrétisation du temps dans le réseau espace temps. Nous souhaitons vérifier deux intuitions : lorsque le pas de discrétisation augmente, le nombre d'itinéraires potentiels dans le réseau amont diminue. Cela entraîne une diminution du nombre de tournées amont possibles. Par conséquent, le nombre de colonnes y_p^l générées diminue et le coût de transport augmente. Nous testons quatre pas de discrétisation : 180, 240, 300, 360 (en minutes). Chaque instance est résolue avec un réseau amont discrétisé selon ces valeurs. Le tableau 7.5 indique le nombre de colonnes amont générées, selon plusieurs pas de discrétisation. On remarque très clairement la diminution du nombre de colonnes lorsque le pas de la discrétisation augmente.

Instance	Pas de discrétisation			
	180	240	300	360
6_1_30_58	2867	2245	1943	1628
6_3_47_101	13027	11288	11221	9271
6_3_63_123	18487	16840	14703	12322
6_4_47_98	23676	18656	16056	14879
6_4_63_128	25482	20206	18778	16158
8_1_50_94	4054	3240	2856	2514
8_3_75_146	19015	16536	14010	11720
8_3_100_203	29060	22941	20660	17536
8_4_75_153	18779	15115	10917	9111
8_4_100_191	26359	20503	14855	11964

Tableau 7.5 – Évolution du nombre de colonnes y_p^l générées en fonction du pas de discrétisation

Le tableau 7.6 confirme partiellement l'intuition d'augmentation du coût. En effet, les coûts du réseau amont ont tendance à augmenter légèrement, mais ce n'est pas toujours le cas. Par exemple, pour l'instance 6_3_63_123 les coûts ne suivent pas cette règle. On explique ces résultats par deux facteurs.

Instance	Pas de discrétisation			
	180	240	300	360
6_1_30_58	5 459	5 459	5 459	5 459
6_3_47_101	7 241	7 482	7 547	7 595
6_3_63_123	9 176	9 111	9 602	9 141
6_4_47_98	12 180	12 024	12 332	12 404
6_4_63_128	12 993	13 022	13 172	13 473
8_1_50_94	9 483	9 536	9 741	9 912
8_3_75_146	10 853	10 554	10 963	11 209
8_3_100_203	13 009	13 058	13 310	13 626
8_4_75_153	17 083	17 044	17 442	17 506
8_4_100_191	22 449	22 887	23 084	23 199

Tableau 7.6 – Évolution du coût du transport dans le réseau amont en fonction du pas de discrétisation

- Une discrétisation plus grossière peut permettre, dans certains cas, de favoriser la consolidation des commandes : par exemple si une première commande doit partir entre 7h et 8h de l'usine A, et qu'une deuxième commande doit partir entre 9h et 10h de cette même usine, alors une discrétisation fine empêchera les regroupements de ces commandes, tandis que toute période de temps plus grossière, incluant l'intervalle $[7h, 10h]$ le permet.
- Certaines valeurs de discrétisation peuvent parfois mieux représenter la durée du transport que d'autres valeurs plus petites : on prend l'exemple de 180 et 240 minutes, qui représente une dis-

crétisation en périodes de 3h et 4h. Si un trajet dure 3h50, alors le réseau espace temps discrétisé toutes les 180 minutes compte 2 périodes de temps pour ce trajet, alors que le réseau espace-temps discrétisé toute les 240 minutes ne considère qu'une période.

7.7.5 Influence du coût de stockage

Les coûts de stockage court terme sont une des raisons pour lesquelles il n'est pas possible de traiter le problème de SNDRP-CRC en deux problèmes distincts. Pour évaluer l'impact du coût de stockage, nous résolvons chaque instance avec des coûts de stockage allant de 2 à 8 euros pour 3h d'attente au CRC. Notons que les valeurs 6 et 8 ne sont pas réalistes d'un point de vue logistique, mais elles servent ici à observer les variations de coûts en fonction d'un coût de stockage très élevé.

Instance	Coût de stockage			
	2	4	6	8
6_1_30_58	0%	18%	33%	47%
6_3_47_101	0%	21%	37%	53%
6_3_63_123	0%	22%	36%	52%
6_4_47_98	0%	15%	25%	34%
6_4_63_128	0%	15%	27%	36%
8_1_50_94	0%	22%	40%	55%
8_3_75_146	0%	22%	39%	54%
8_3_100_203	0%	19%	36%	49%
8_4_75_153	0%	20%	38%	54%
8_4_100_191	0%	23%	43%	62%
Moyenne	0%	20%	35%	50%

Tableau 7.7 – Évolution du coût global en fonction du coût de stockage

Le tableau 7.7 montre l'augmentation de la valeur de l'objectif en fonction du coût de stockage (avec comme référence celle obtenue avec un coût de stockage de 2 euros). On remarque un fort impact sur la valeur de la fonction objectif, qui augmente en moyenne de 20% si le coût passe de 2 à 4 euros et de 50% si le coût augmente de 2 à 8 euros.

On identifie dans la valeur de la fonction objectif deux types de coût afin d'observer son évolution en fonction de l'augmentation du coût de stockage unitaire : le coût de stockage total et le coût de transport dans les deux réseaux logistiques.

La tableau 7.8 montre l'évolution du coût total des stocks (le quatrième terme de la fonction objectif (7.1), en fonction des différentes valeurs du coût de stockage.

Ce coût augmente approximativement de 40% à chaque changement de valeur. Si les itinéraires des marchandises (et donc le temps d'attente au CRC) restaient les mêmes, on devrait observer une augmentation respective de 100%, 200% et 300% par rapport à la référence. La résolution du problème détermine donc des solutions différentes en fonction des coûts de stockage, afin de minimiser les coûts globaux de la fonction objectif.

Instance	Coût de stockage			
	2	4	6	8
6_1_30_58	0%	48%	86%	129%
6_3_47_101	0%	49%	93%	109%
6_3_63_123	0%	42%	72%	103%
6_4_47_98	0%	36%	66%	98%
6_4_63_128	0%	47%	82%	113%
8_1_50_94	0%	47%	85%	125%
8_3_75_146	0%	43%	55%	97%
8_3_100_203	0%	40%	64%	88%
8_4_75_153	0%	62%	109%	162%
8_4_100_191	0%	60%	120%	181%
Moyenne	0%	48%	83%	120%

Tableau 7.8 – Évolution du coût total des stocks en fonction du coût de stockage

Le tableau 7.9 présente l'évolution des coûts de transport issus de la fonction objectif (7.1). On remarque que les coûts de transports augmentent en moyenne de 8%, 15% et 20% par rapport à la solution, ce qui confirme cette observation. considérant un coût de stockage de 2 euros. Ces résultats montrent donc un des intérêts de résoudre le **SNDP** du réseau amont conjointement avec le problème de **VRPTW** du réseau aval : la minimisation des coûts comprenant un coût horaire de transfert relatif à la synchronisation entre les deux réseaux.

Instance	Coût de stockage			
	2	4	6	8
6_1_30_58	0%	6%	11%	14%
6_3_47_101	0%	7%	10%	26%
6_3_63_123	0%	11%	18%	26%
6_4_47_98	0%	8%	11%	13%
6_4_63_128	0%	6%	11%	13%
8_1_50_94	0%	10%	17%	20%
8_3_75_146	0%	11%	31%	31%
8_3_100_203	0%	9%	23%	31%
8_4_75_153	0%	3%	11%	13%
8_4_100_191	0%	8%	12%	14%
Moyenne	0%	8%	15%	20%

Tableau 7.9 – Évolution du coût de transport en fonction du coût de stockage

7.8 Conclusion et perspectives

Dans ce chapitre, nous avons étudié le problème opérationnel de routage des commandes dans un réseau comprenant à la fois une partie amont et une partie aval. Le problème résolu dans le réseau amont est équivalent à un **SNDP** modélisé dans un réseau espace temps. Dans le réseau aval, nous résolvons un **VRPTW** pour chacun des CRC.

Nous avons proposé une heuristique basée sur la génération de colonnes. Deux types de variables sont générées : une variable y_p^l représentant l'itinéraire des commandes dans le réseau amont, et une variable x_k représentant une tournée de véhicules dans le réseau aval. La synchronisation du transport amont et aval est réalisée par les contraintes du problème maître. Le sous-problème qui génère les variables y_p^l est résolu grâce à l'algorithme de Bellman-Ford. Le sous-problème qui génère les variables x_k est résolu grâce à une méthode tabou. Les résultats montrent la stabilité de notre algorithme et sa capacité à fournir de bons résultats en un temps compatible avec une utilisation quotidienne.

Ces résultats montrent la nécessité de la résolution conjointe des problèmes de transport liés aux réseaux amont et aval lorsqu'on considère un coût de stockage court terme dépendant du temps passé sur les plateformes. En perspective, il serait utile de développer une méthode de résolution exacte pour le sous-problème concernant le réseau aval, afin d'évaluer plus rigoureusement les performances de la recherche tabou, ainsi que d'obtenir une borne inférieure au problème. De plus, le développement d'un Branch-and-Price permettrait une résolution exacte du problème. Les travaux du chapitre 6 ont été réalisés postérieurement à ceux de ce chapitre. Il serait donc logique de prendre en considération les contraintes riches du réseau aval décrites dans le chapitre 6 pour ce problème. Enfin, l'utilisation d'un algorithme de génération de colonnes qui génèrent des colonnes de différents types pourrait être utile dans d'autres catégories de problèmes. Plus particulièrement, les problèmes nécessitant une synchronisation ou une résolution conjointe de deux problèmes différents pourraient être résolus grâce à un algorithme de ce type.

Conception intégrée de plans de chargement et de tournées de véhicules

Sommaire

8.1	Introduction	125
8.2	Description du problème	127
8.3	Formulations sur les tournées et sur les arcs du SDRP	128
8.3.1	Notations	128
8.3.2	Réseau espace temps	129
8.3.3	Modélisation du SDRP	129
8.3.4	Formulation $\mathcal{F}_{\mathcal{R}}$	130
8.3.5	Formulation $\mathcal{F}_{\mathcal{A}}$	131
8.4	Méthode de résolution	133
8.5	Tests numériques	137
8.5.1	Instances	137
8.5.2	Comparaison des deux formulations	138
8.5.3	Comparaison logistique	138
8.6	Conclusion	142

8.1 Introduction

Ce chapitre présente le travail réalisé dans le cadre d'un séjour de deux mois à la Quinlan School of business de Loyola University à Chicago, en collaboration avec Mike Hewitt. L'objectif est de

contribuer de manière plus fondamentale à la résolution exacte de problèmes intégrés de **SNDP** et de **VRPTW**.

Ce travail se place donc dans un contexte plus général et certaines contraintes riches, évoquées dans les chapitres précédents, ont été abandonnées ou simplifiées. Par exemple, dans ce chapitre, ne sont considérés que des coûts de transport sur les arcs du réseaux, dépendant uniquement de la distance parcourue par un véhicule et pas de sa charge.

Du point de vue applicatif, les hypothèses considérées sont plus proches de celles d'un transporteur en charge de satisfaire un ensemble de requêtes de transport de petites tailles. Le réseau considéré est illustré en figure 8.1.

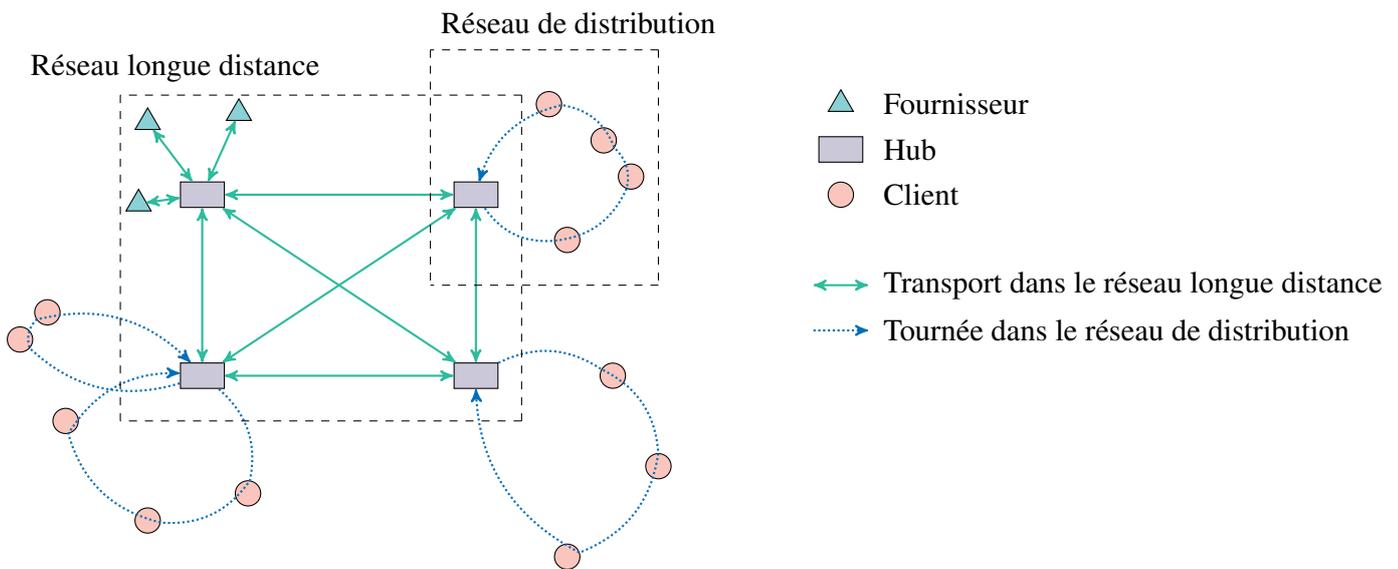


Figure 8.1 – Exemple de réseau logistique considéré

Ce réseau comporte deux sous-réseaux de transport distincts.

- Un réseau *longue distance* constitué de terminaux (ici des fournisseurs) et plusieurs hubs de consolidation. Les commandes provenant de différents fournisseurs peuvent donc être consolidées sur ces hubs.
- Un réseau de distribution local : depuis chaque hub, des tournées de véhicules sont réalisées pour distribuer les commandes aux clients.

Notons que l'on ne considère pas de tournées de collecte régionales chez les fournisseurs jusqu'au premier hub. Néanmoins, les méthodes que nous présentons peuvent très facilement s'adapter à ce type d'organisation. Sur ce réseau, on considère que le coût d'une solution est la somme des coûts des arcs empruntés par les véhicules. Le problème posé consiste à prendre des décisions qui sont généralement prises indépendamment par deux types de problèmes : le **SNDP** et le **VRPTW**. On rappelle que l'objectif du **SNDP** est de trouver l'itinéraire des commandes à travers un réseau de hubs, depuis leur hub d'origine jusqu'à leur hub de destination (appelé ici réseau longue distance), tout en déterminant les capacités de transport à installer sur les arcs. Pour chaque hub, un **VRPTW** est résolu pour constituer un ensemble de tournées de coût minimal qui livrent un ensemble de commandes à leurs clients dans leurs fenêtres de temps respectives.

Si les délais de livraison permettent de considérer un temps de transfert assez long entre ces deux réseaux, ces deux problèmes peuvent être traités séparément. En revanche, lorsque les délais sont courts, il est utile, voire essentiel, de traiter globalement ces deux problèmes. L'objectif est alors de synchroniser efficacement le transport des marchandises entre les véhicules des deux réseaux. Ainsi, ce chapitre se concentre particulièrement sur le transport de commandes de petite tailles dont les délais de livraison sont courts. On appelle le problème traité **SNDRP** (**Service Network Design and Routing Problem**).

Dans ce chapitre, on présente deux formulations adaptées à la méthode de résolution décrite dans l'article Boland et al. [10]. Cette méthode permet de résoudre un problème de **SNDP** en gérant les synchronisations temporelles de façon très fine. Elle est basée sur un algorithme de discrétisation dynamique, appelé **Dynamic Discretization Discovery** (**DDD**), permettant une discrétisation du réseau espace temps en périodes d'une minute. Nous montrons donc comment il est possible de résoudre le **SNDRP** par cette approche : le **SNDP** résolu par Boland et al. [10] est équivalent au problème dans le réseau longue distance défini précédemment. Nous proposons deux formulations adaptées à cette méthode afin d'intégrer des tournées de véhicules depuis les hubs jusqu'aux commandes aux clients. Un résultat important est que la méthode de résolution exacte de Boland et al. [10] appliquée sur les deux formulations proposées permet une résolution exacte du **SNDRP**.

Les deux formulations permettent de résoudre le même problème intégrant les deux niveaux de transport, mais différent par leur capacité d'intégration de contraintes industrielles. Plus particulièrement, la première formulation notée $\mathcal{F}_{\mathcal{R}}$ modélise les tournées du réseau local en créant une variable pour chaque tournée, et peut ainsi représenter toutes les contraintes riches liées aux tournées de véhicules, ce que la seconde formulation, notée $\mathcal{F}_{\mathcal{A}}$, ne permet pas directement. Les deux formulations utilisent l'algorithme présenté dans Boland et al. [10] comme une boîte noire et limitent le champ d'étude aux modèles résolus par cet algorithme. Ainsi, on considère dans la formulation $\mathcal{F}_{\mathcal{R}}$ des tournées énumérées *a priori*, en opposition à des tournées générées dynamiquement, par exemple comme dans une génération de colonnes. Néanmoins, la formulation $\mathcal{F}_{\mathcal{R}}$ devient rapidement inefficace lorsque le nombre de clients augmente : un trop grand nombre de tournées sont créées. Afin de palier ce problème, on propose la seconde formulation $\mathcal{F}_{\mathcal{A}}$ qui offre une résolution plus performante, mais elle ne permet toutefois pas de modéliser autant de contraintes pour les tournées de véhicules que la formulation $\mathcal{F}_{\mathcal{R}}$. Cette dernière représentation se rapproche d'une modélisation sur les arcs d'un problème de VRP. Les expérimentations permettent d'observer les performances de chacune des méthodes proposées, selon la taille de l'instance.

8.2 Description du problème

Cette section décrit en détail le problème étudié dans ce chapitre. On cherche ici à optimiser le transport d'un ensemble de commandes depuis plusieurs fournisseurs jusqu'à un ensemble de clients à travers un réseau logistique similaire à celui présenté en représenté figure 8.1.

A chaque commande est associée un fournisseur, un client, une quantité, une date de mise à disposition dans le réseau par le fournisseurs et une date limite de livraison chez le client. Le site du fournisseur de la commande est affecté à un hub du réseau longue distance, appelé *hub de départ*, vers lequel elle est envoyé directement (sans tournées de véhicules). On considère également que chaque client du réseau de distribution régional est affecté à un unique hub du réseau longue distance, appelé *hub de destination* depuis lequel ses commandes sont livrées. Tous les autres hubs sont appelés *hubs intermédiaires*.

Afin de transporter une commande de son fournisseur jusqu'à son client de destination, une commande doit transiter dans le réseau longue distance, depuis son hub de départ jusqu'à son hub de des-

mination. Depuis le hub de destination, une tournée est constituée pour livrer cette commande conjointement avec d'autres. On considère qu'une tournée part d'un hub et y revient. Lorsqu'une commande arrive sur un hub intermédiaire elle est déchargée du véhicule, puis rechargée dans un autre véhicule en direction d'un autre hub du réseau longue distance. Pour transporter un ensemble de commandes d'un hub à un autre, on considère une flotte illimitée et homogène de véhicules. Les véhicules de cette flotte réalisent uniquement des *trajets directs* entre deux hubs. A trajet pour un véhicule de cette flotte est associé un coût fixe d'utilisation et une capacité. Les coûts ne dépendent pas de la charge des véhicules. On considère que ces coûts fixes (ainsi que la durée de transport de chaque trajet) sont proportionnels à la distance parcourue par les véhicules. Le problème de transport dans le réseau longue distance peut-être modélisé par un [Service Network Design Problem \(SNDP\)](#), qui consiste à trouver l'itinéraire de chaque commande depuis son fournisseur jusqu'à son hub de destination et à déterminer le nombre de véhicules nécessaires pour chaque trajet dans le réseau. Cependant, aucune date d'arrivée au plus tard sur le hub de destination n'est définie, car cette date dépend du départ de la tournée de livraison de la commande.

Pour la distribution depuis les hubs, on considère une flotte de véhicules homogène et infinie. La tournée d'un véhicule part de son hub (considéré comme un dépôt), livre un ensemble de commandes avant leurs dates de livraison au plus tard et retourne à son hub à la fin de la tournée. Le coût d'une tournée dépend uniquement de la distance parcourue par le véhicule. De plus, on considère que le nombre de clients visités est limité (comme dans les autres chapitres de cette thèse) afin de réduire les conséquences en cas de retard. Cette partie du problème peut-être modélisée par un problème de tournées de véhicules avec fenêtres de temps ([Vehicle Routing Problem with Time Windows \(VRPTW\)](#)). Néanmoins, l'heure d'arrivée d'une commande sur le hub de destination (et donc le dépôt du problème de [VRPTW](#)) n'est pas connue : elle dépend de l'itinéraire de la commande dans le réseau longue distance.

Le [SNDRP](#) consiste à déterminer le nombre de véhicules devant circuler sur les arcs du réseau longue distance et le planning de ces transports ; les plans de chargement de ces véhicules ; les tournées de livraison des clients depuis les hubs de telle sorte que : chaque commande soit acheminée de son origine à sa destination dans sa fenêtre horaire ; et la somme des coûts de transport soit minimisée.

On note que dans ce problème, plusieurs commandes peuvent partir d'un même fournisseur, mais qu'un client est associé avec une unique commande.

8.3 Formulations sur les tournées et sur les arcs du SNDRP

En premier lieu on présente un ensemble de notations communes aux deux formulations proposées ainsi que les notations utilisées pour définir un réseau espace-temps. En section [8.3.3](#), on introduit un modèle classique du [SNDP](#), qui est étendu en section [8.3.4](#) par une formulation basée sur une énumération explicite des tournées et en section [8.3.5](#) par une formulation basée sur une modélisation des tournées basée sur les arcs.

8.3.1 Notations

L'ensemble des sommets du réseau de transport peut être décomposé en trois sous-ensembles de sommets :

- N_s qui représente l'ensemble des fournisseurs (*suppliers*) ;
- N_{bb} qui représente l'ensemble des hubs (*breakbulks*) ;

- N_c qui représente l'ensemble des clients.

Chaque client est affecté à un hub qui assure la distribution de sa commande. On note $N_c(i), i \in N_{bb}$, l'ensemble des clients affecté au hub i . Inversement, pour tout $j \in N_c$ on appelle $bb(j) \in N_{bb}$ le hub auquel le client j est affecté. On note $N = N_s \cup N_{bb}$ l'ensemble des sommets du réseau longue distance et A l'ensemble qui contient les arcs qui modélisent les trajets possibles entre les différents types de sommets de ce réseau. Il contient notamment l'arc représentant le transport entre chaque fournisseurs son hub, c'est à dire de la forme $(i, j), i \in N_s, j \in N_{bb}$. L'ensemble A contient également les arcs qui modélisent le transport entre les hubs du réseau longue distance, de la forme $(i, j), i, j \in N_{bb}$. Les ensembles de sommets et d'arcs du réseau de distribution diffèrent selon les deux formulations proposées. Ils sont donc définis plus tard dans cette section. Pour chaque arc $a = (i, j) \in A$ on définit une durée de transport tt_{ij} , un coût fixe f_{ij} , un coût variable à la palette c_{ij} et un nombre maximal de palettes représentant la capacité u_{ij} des véhicules pouvant être affectés sur cet arc. Les durées de chargement et déchargement sur les hubs peuvent avoir un impact critique sur les synchronisations dans le réseau longue distance. On considère que ces durées de chargement et déchargement sont fixes et incluses aux durées de transport associées à chaque arc.

8.3.2 Réseau espace temps

L'un des paramètres principaux de notre problème est le temps : chaque commande a une date de mise à disposition dans le réseau et doit être livrée avant une date donnée. En outre, il est nécessaire de synchroniser l'arrivée d'une commande sur le hub de destination avec le départ du véhicule qui réalise sa livraison au client. On utilise un réseau espace-temps pour modéliser le temps. On rappelle ici les notations utilisées pour représenter un réseau espace-temps. Une explication plus détaillée de la création d'un tel réseau est présentée au chapitre 3 en section 3.3.

A partir d'un réseau plat D , on définit un réseau espace temps $D_{\mathcal{T}} = (N_{\mathcal{T}}, H_{\mathcal{T}} \cup A_{\mathcal{T}})$, où l'ensemble N des sommets physiques est associé à un ensemble de périodes de temps. T_i est l'ensemble des périodes de temps associées au sommet $i \in N$, et $T = \cup_{i \in N} T_i$ est l'ensemble des périodes de temps du problème. Un sommet $(i, t) \in N_{\mathcal{T}}$ représente la visite du sommet physique $i \in N$ au temps $t \in T$. L'ensemble $A_{\mathcal{T}}$ contient les arcs de la forme $((i, t), (j, \bar{t})), (i, j) \in A, t \in T_i, \bar{t} \in T_j$, qui modélisent le transport entre les sommets temporels (i, t) et (j, \bar{t}) . L'ensemble $H_{\mathcal{T}}$ est l'ensemble des arcs d'attente, c'est à dire les arcs de la forme $((i, t), (i, \bar{t})), t, \bar{t} \in T_i$.

8.3.3 Modélisation du SndP

Les deux formulations proposées étendent le modèle classique de SndP utilisé par Boland et al. [10]. Dans cet article, ainsi que dans le présent chapitre, on considère qu'il n'est pas possible de séparer les produits d'une commande.

Soit $D = (N, A)$ le réseau de transport, appelé *réseau plat*, du SndP. Il est constitué de l'ensemble $N = N_s \cup N_{bb}$ et de l'ensemble A des arcs. On note K l'ensemble des commandes. Chaque commande peut-être collectée sur son origine $o_k \in N_s$ à partir de sa date de disponibilité e_k et doit être arrivée à sa destination $d_k \in N_{bb}$ avant sa date de livraison au plus tard, notée l_k . L'origine d'une commande est modélisée dans le réseau espace temps par le sommet (o_k, e_k) et sa destination (d_k, l_k) . La taille d'une commande est notée q_k . Un coût c_{ij} est payé pour chaque unité de produit passant par l'arc $(i, j) \in A$.

On note respectivement f_{ij} et u_{ij} le coût fixe et la capacité d'un véhicule passant par cet arc. Le réseau espace-temps associé au réseau plat D est noté $D_{\mathcal{T}} = (N_{\mathcal{T}}, H_{\mathcal{T}} \cup A_{\mathcal{T}})$.

Soit $x_{ij}^{k\bar{t}\bar{t}}$ la variable binaire égale à 1 si la commande k utilise l'arc $((i, t), (j, \bar{t})) \in A_{\mathcal{T}}$, 0 sinon. La variable $y_{ij}^{\bar{t}\bar{t}}$ représente le nombre de véhicules qui voyagent sur l'arc $((i, t), (j, \bar{t})) \in A_{\mathcal{T}}$.

Le **SNDP** peut être modélisé par le programme linéaire en nombre entier décrit ci-dessous. Cette formulation est notée \mathcal{F} .

$$Z = \min \sum_{((i,t),(j,\bar{t})) \in A_{\mathcal{T}}} f_{ij} y_{ij}^{\bar{t}\bar{t}} + \sum_{k \in K} \sum_{((i,t),(j,\bar{t})) \in A_{\mathcal{T}}} c_{ij} q_k x_{ij}^{k\bar{t}\bar{t}} \quad (8.1)$$

$$\sum_{((i,t),(j,\bar{t})) \in A_{\mathcal{T}} \cup H_{\mathcal{T}}} x_{ij}^{k\bar{t}\bar{t}} - \sum_{((j,\bar{t}),(i,t)) \in A_{\mathcal{T}} \cup H_{\mathcal{T}}} x_{ji}^{k\bar{t}\bar{t}} = \begin{cases} 1 & \text{si } (i, t) = (o_k, e_k) \\ -1 & \text{si } (i, t) = (d_k, l_k) \\ 0 & \text{sinon} \end{cases} \quad \forall k \in K, \forall (i, t) \in N_{\mathcal{T}} \quad (8.2)$$

$$\sum_{k \in K} q_k x_{ij}^{k\bar{t}\bar{t}} \leq u_{ij} y_{ij}^{\bar{t}\bar{t}} \quad \forall ((i, t), (j, \bar{t})) \in A_{\mathcal{T}}, \quad (8.3)$$

$$x_{ij}^{k\bar{t}\bar{t}} \in \{0, 1\} \quad \forall ((i, t), (j, \bar{t})) \in A_{\mathcal{T}} \cup H_{\mathcal{T}}, k \in K, \quad (8.4)$$

$$y_{ij}^{\bar{t}\bar{t}} \in \mathbb{N}, \quad \forall ((i, t), (j, \bar{t})) \in A_{\mathcal{T}}, \quad (8.5)$$

La fonction objectif (8.1) est la somme des coûts fixes et des coûts variables de transport. Les contraintes (8.2) garantissent la conservation des flux pour chaque commande dans le réseau espace-temps. Les contraintes (8.3) garantissent que le nombre de véhicules par arc est suffisant pour transporter toutes les commandes.

Les deux sections suivantes présentent deux formulations qui s'appuient sur la formulation \mathcal{F} du **SNDP** pour modéliser le **SNDRP**. Dans ces sections, on considère qu'une commande est définie sur le réseau espace-temps par un sommet de départ (o_k, e_k) , $o_k \in N_s$ et une destination $(d_k, l_k) \in N_c$.

8.3.4 Formulation $\mathcal{F}_{\mathcal{R}}$

La formulation $\mathcal{F}_{\mathcal{R}}$ repose sur une énumération exhaustive des tournées de distribution. Soit R l'ensemble des tournées de véhicules réalisables dans le réseau de distribution. On note c_r le coût total de chaque tournée $r \in R$ et K_r l'ensemble des commandes servis par une tournée $r \in R$. Pour chaque hub $i \in N_{bb}$, on note R_i l'ensemble des tournées qui partent de ce hub et visitent les clients dans l'ensemble $N_c(i)$.

Pour chaque tournée $r \in R$, on définit un sommet virtuel noté v_r , qui représente le départ de cette tournée. L'ensemble des sommets virtuels est noté $\{v_r | r \in R\} = N_r$. Pour tout hub $i \in N_{bb}$ et tournée $r \in R_i$, on crée les arcs suivants :

- Un arc (i, v_r) de coût fixe égal au coût de la tournée r ($f_{iv_r} = c_r$), de coût variable, capacité et durée nulle ($c_{iv_r} = u_{iv_r} = tt_{iv_r} = 0$).
- Pour toute commande $k \in K_r$, on crée l'arc (v_r, d_k) . Cet arc modélise la durée du trajet entre le hub i et le client d_k sur la tournée r . On lui associe un coût fixe et variable nul $f_{v_r, d_k} = c_{v_r, d_k} = 0$ et une durée de transport égale à la somme du temps de trajet et de service sur la tournée r entre son départ du hub i et son arrivée en d_k .

L'ensemble des arcs modélisant les tournées de l'ensemble R est noté A_R .

On note $N^{\mathcal{F}_R} = N_s \cup N_{bb} \cup N_r \cup N_c$ l'ensemble des sommets et $A^{\mathcal{F}_R} = A \cup A_R$ l'ensemble des arcs du réseau logistique. Le réseau plat de cette formulation est noté $D^{\mathcal{F}_R} = (N^{\mathcal{F}_R}, A^{\mathcal{F}_R})$ et $D_{\mathcal{T}}^{\mathcal{F}_R}$ est le réseau espace temps associé.

Le réseau ainsi créé est illustré par l'exemple suivant :

Exemple 18. Soit 3 clients 1,2 et 3 associés à un hub i . Chaque client doit recevoir une commande. Les clients et les dates de livraison au plus tard de leurs commandes associées sont indiquées dans le tableau 8.2a. De la même manière, les durées de transport sont indiquées dans le tableau 8.2b. Enfin, le réseau est illustré sur la figure 8.2c.

On représente trois tournées réalisables pour distribuer ces commandes à leurs clients. Le tableau 8.3a indique la liste ordonnée des clients visités par chaque tournée et le départ au plus tard du dépôt pour chaque tournée. La figure 8.3b illustre le réseau créé par l'introduction des sommets virtuels.

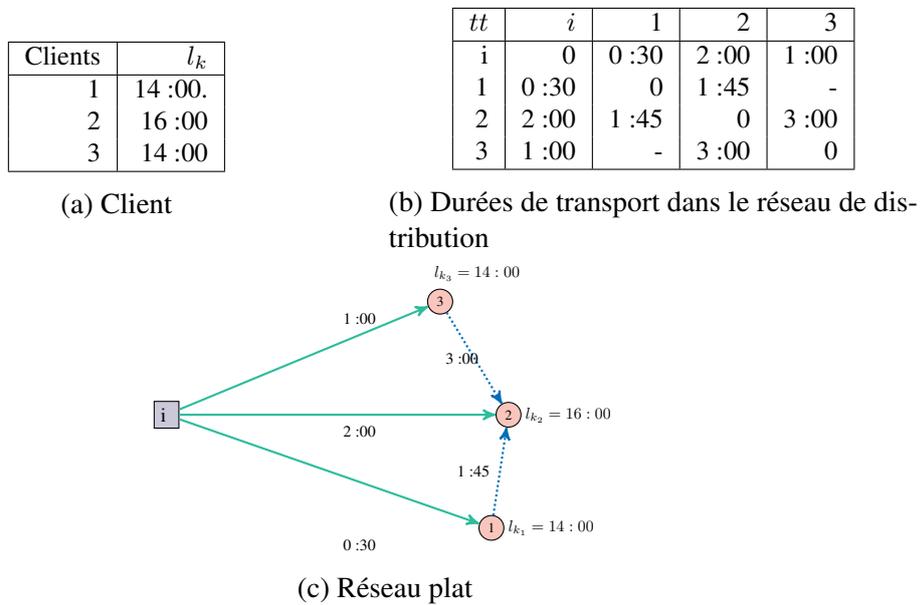


Figure 8.2 – Exemple de création d'un réseau

La formulation \mathcal{F}_R permet de modéliser le Sndrp comme un Sndp et de le résoudre avec la formulation \mathcal{F} .

8.3.5 Formulation \mathcal{F}_A

La formulation \mathcal{F}_A repose sur une modélisation sur les arcs des tournées de distribution. Pour tout hub $i \in N_{bb}$, on crée les arcs suivants :

- pour tout client $j \in N_{bb}(i)$, on crée l'arc (i, j) et l'arc (j, i) ;
- pour toute paire de client (j, j') affectés au hub i , on crée les arcs (j, j') et (j', j) .

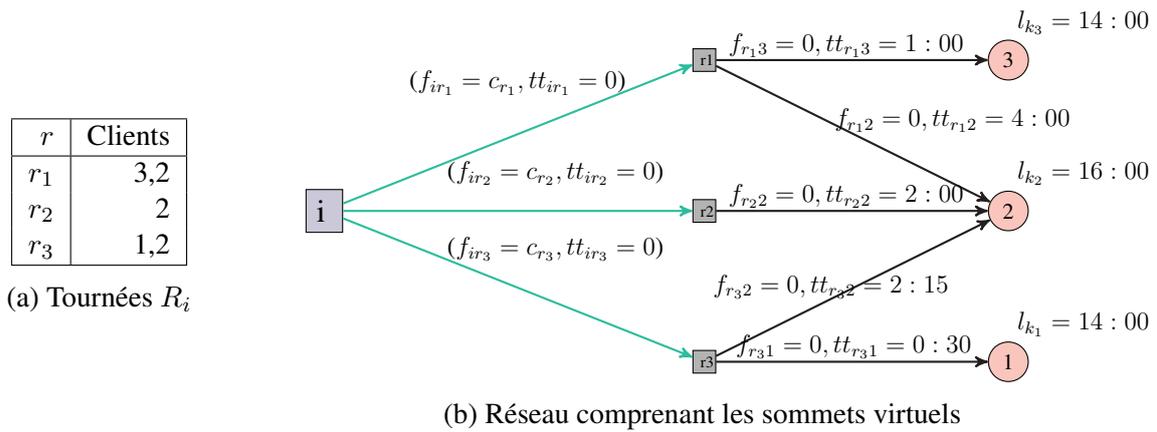


Figure 8.3 – Tournées énumérées et modélisation sur le réseau plat

A chaque arc ainsi créé est associé un coût fixe et une durée correspondant respectivement au coût et à la durée du trajet qu’il modélise (cette dernière durée intégrant le temps de service à l’origine de l’arc). Il lui est également associé un coût variable nul et une capacité égale à la capacité d’un véhicule.

On note A_L l’ensemble qui regroupe l’ensemble de ces arcs pour l’ensemble des hubs. Le réseau plat associé à cette formulation est noté $D^{\mathcal{F}^A} = (N^{\mathcal{F}^A}, A^{\mathcal{F}^A})$ avec $N^{\mathcal{F}^A} = N_s \cup N_{bb} \cup N_c$ et $A^{\mathcal{F}^A} = A \cup A_L$.

Sur le réseau espace-temps, l’ensemble d’arc associé à l’ensemble A_L est noté A_L^T . Le réseau espace-temps est le réseau $D_{\mathcal{T}}^{\mathcal{F}^A} = (N_{\mathcal{T}}^{\mathcal{F}^A}, A_L^T \cup A_{\mathcal{T}} \cup H_{\mathcal{T}})$. On note qu’aucune attente n’est considérée sur la partie distribution car elle ne permet aucune consolidation et on ne considère que des dates de livraison au plus tard chez les clients.

De nouvelles contraintes doivent être ajoutées à la formulation \mathcal{F} du **SNDP** pour que les variables $y_{ij}^{t\bar{t}}$, $((i, t), (j, \bar{t})) \in A_L^T$ modélisent des tournées dans le réseau de distribution. Dans le réseau espace temps, une tournée qui part d’un hub i correspond à un chemin partant d’un sommet (i, t) , servant des clients associés au hub i et terminant à un sommet (i, \bar{t}) . La figure 8.4 illustre un exemple de tournée dans le réseau $D_{\mathcal{T}}^{\mathcal{F}^A}$ qui part du sommet (bb_1, t_1) , visite les clients c_1, c_2 et c_3 et retourne au hub bb_1 pendant la période t_{11} . Les sommets du réseau représentés en gras modélisent les sommets de destination des commandes 1, 2 et 3 à destination des clients c_1, c_2 et c_3 .

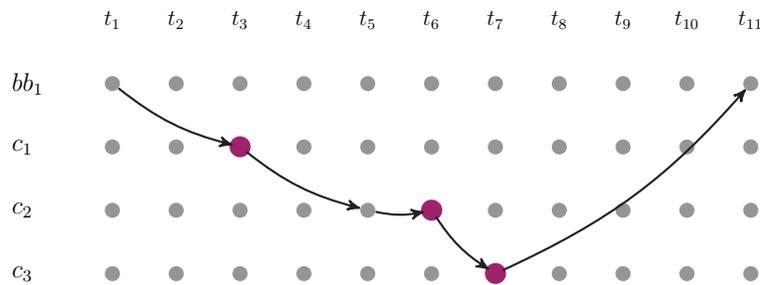


Figure 8.4 – Tournée de véhicule du hub bb_1 dans le réseau espace temps

Afin de garantir que les valeurs des variables $y_{ij}^{t\bar{t}}$, $(i, j) \in A_L$ forment des tournées de véhicules dans le réseau de distribution, on ajoute les contraintes (8.6) et (8.7) à la formulation \mathcal{F} du **SNDP**. La formulation \mathcal{F}_A est la suivante :

(8.1) – (8.5)

$$\sum_{((i,t),(j,\bar{t})) \in A_{Lt} \cup H_t} y_{ij}^{t\bar{t}} - \sum_{((j,\bar{t}),(i,t)) \in A_{Lt} \cup H_t} y_{ji}^{t\bar{t}} = 0 \quad \forall i \in N_c, t \in T_i \quad (8.6)$$

$$\sum_{t \in T_i} \sum_{(j \in N_c(i), \bar{t} \in T_j)} y_{ij}^{t\bar{t}} - \sum_{t \in T_i} \sum_{(j \in N_c(i), \bar{t} \in T_j)} y_{ji}^{t\bar{t}} = 0 \quad \forall i \in N_{bb} \quad (8.7)$$

Les contraintes (8.6) garantissent que chaque arrivée d'un véhicule chez un client est suivie d'un départ vers un autre sommet. On rappelle que le réseau de distribution est constitué uniquement d'arcs entre un hub et les clients qui lui sont affectés ou entre les clients affectés au même hub. C'est pourquoi le départ d'un client peut être réalisé seulement vers un autre client affecté au même hub ou directement vers le hub lui-même. Les contraintes (8.7) assurent que le nombre de véhicules partant d'un hub vers son réseau de distribution soit le même que le nombre de véhicules arrivant depuis son réseau de distribution. La capacité des véhicules réalisant les tournées de distribution est modélisée par les capacités sur les arcs $(bb(j), j), j \in N_c$ et $(i, j) | bb(i) = bb(j), i, j \in N_c$.

On remarque que les contraintes (8.6) et (8.7) n'interdisent pas la création de sous-tours entre les clients d'un même réseau de distribution. Néanmoins, les contraintes (8.2) garantissent l'existence d'un chemin entre l'origine des demandes et leur destination. Un sous-tour ne peut exister dans une solution optimale.

8.4 Méthode de résolution

Jusqu'à très récemment, une modélisation sur un réseau espace temps impliquait un choix *a priori* de l'ensemble des sommets et arcs du réseau temporel. Ce choix relevait de la recherche d'un compromis entre la taille du modèle et le degré de discrétisation du temps. Par exemple, pour modéliser les activités d'une journée, il est possible de créer deux sommets, le premier modélisant les activités de minuit à midi et le second modélisant les activités de midi à minuit. Cette approche a l'avantage de ne générer que deux sommets et un nombre limité d'arcs dans le réseau espace-temps, mais elle ne modélise pas précisément la planification temporelle des activités. Ceci devient particulièrement important dans une configuration où les délais et la synchronisation temporelle sont critiques. Par exemple, si deux commandes doivent être transportées d'un hub i jusqu'à un hub j , la première commande devant impérativement partir du hub i entre 8h et 9h et la seconde impérativement entre 10h et 11h, il n'est pas possible de les transporter ensemble. Néanmoins, avec la modélisation comprenant les deux sommets temporels proposés, cette incompatibilité d'horaires devient impossible à exprimer. Un **SNDP** défini sur ce réseau espace-temps considérerait qu'elles peuvent être transportées ensemble. Un second choix est de modéliser la journée avec 24 sommets temporels, un par heure (par exemple $(i, 0 : 00), (i, 1 : 00), \dots, (i, 24 : 00)$). Une telle approche crée un réseau espace-temps significativement plus grand que le précédent et donc une taille de **SNDP** également plus grande. Cependant, le **SNDP** défini sur ce réseau espace-temps appréhende mieux les opportunités de consolidations.

Les récents travaux de Boland et al. [10] proposent de discrétiser le temps dynamiquement, pendant la résolution du **SNDP**. L'algorithme présenté dans cet article, appelé **Dynamic Discretization Discovery (DDD)** requiert trois types de données d'entrée (illustrés sur la figure 8.5) :

- un réseau plat représentant les arcs de transports ainsi que la durée de transport et coûts associés ;
- un ensemble de commandes à transporter dans le réseau, associées à une date de mise à disposition/distribution au plus tard ;
- le plus petit pas de discrétisation que l'on souhaite : par exemple, si le plus petit pas de discrétisation est de 5 minutes signifie que l'on souhaite résoudre le **SNDP** sur un réseau espace-temps contenant les sommets $(i, 9 : 00)$, $(i, 9 : 05)$, $(i, 9 : 10)$, ... pour chaque sommet du réseau plat.

Un tel problème est appelé par Boland et al. un *problème de service network design en temps continu* ou **Continuous Time Service Network Design Problem (CTSNDP)**. Cet algorithme résout de manière exacte le **CTSNDP** sans énumérer *a priori* tous les sommets et les arcs du réseau espace-temps. Le résultat de l'algorithme (voir figure 8.5) est un ensemble d'itinéraires pour les commandes et un nombre de véhicules sur les arcs non nuls du réseau espace-temps, qui minimise les coûts de transport. On utilise l'algorithme **DDD** comme une boîte noire, pour les deux formulations proposées. Nous ne présentons pas en détail le fonctionnement de cet algorithme. Néanmoins, nous décrivons le fonctionnement de chaque étapes de l'algorithme, illustrées par le diagramme de flux figure 8.6.

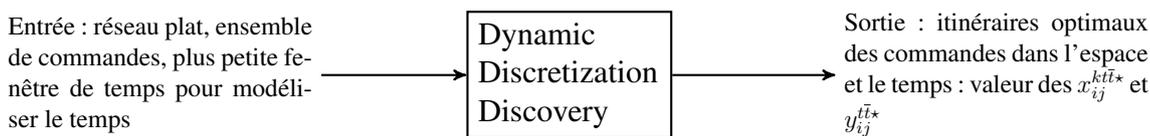


Figure 8.5 – Utilisation de l'algorithme **DDD**

Cet algorithme est basé sur des techniques de relaxation du modèle \mathcal{F} : l'algorithme résout de manière répétitive une relaxation du **CTSNDP** ; vérifie si la solution de la relaxation peut-être convertie en une solution du **CTSNDP** de coût égal et affine le réseau espace-temps si cela n'est pas possible. Si cela est possible, l'algorithme s'arrête et retourne une solution optimale.

La relaxation du **CTSNDP** qui est résolue est une instance du **SNDP** définie sur ce que les auteurs appellent un *réseau espace-temps partiel*. Un réseau espace-temps partiel est un réseau espace-temps où les sommets ne sont pas tous représentés pour chaque période temporelle. Un exemple est donné figure 8.7.

L'algorithme **DDD** utilise initialement un réseau espace-temps qui contient très peu de sommets et d'arcs. Il affine ce réseau à chaque itération en ajoutant de nouveaux sommets et arcs. Cela permet d'éviter de résoudre le problème dans un réseau espace-temps contenant l'ensemble des sommets temporels. Cependant, afin que le **SNDP** associé au réseau espace-temps partiel soit une relaxation sur **CTSNDP**, certaines propriétés doivent être respectées lors de sa création : le réseau espace-temps partiel doit permettre que tous les itinéraires de commandes possibles dans le **CTSNDP** aient un itinéraire correspondant dans le réseau espace-temps partiel, dont la durée est inférieure ou égale. Ce type d'itinéraire est appelé par les auteurs *early arrival path*.

Exemple 19. La figure 8.8 illustre un réseau plat comprenant quatre sommets et trois arcs sur lesquels sont indiqués les durées de transport associées. La figure 8.9 illustre un réseau espace temps partiel et deux *early arrival path*. L'itinéraire $i \rightarrow j \rightarrow k$ devrait durer $2 + 2 = 4$ unités de temps, alors que l'itinéraire dans le réseau espace temps partiel dure seulement 3 unités de temps : l'arc $((j, 3), (k, 4))$ sous-estime la durée réelle du transport sur les arcs de la forme $((j, t), (k, t + 2))$.

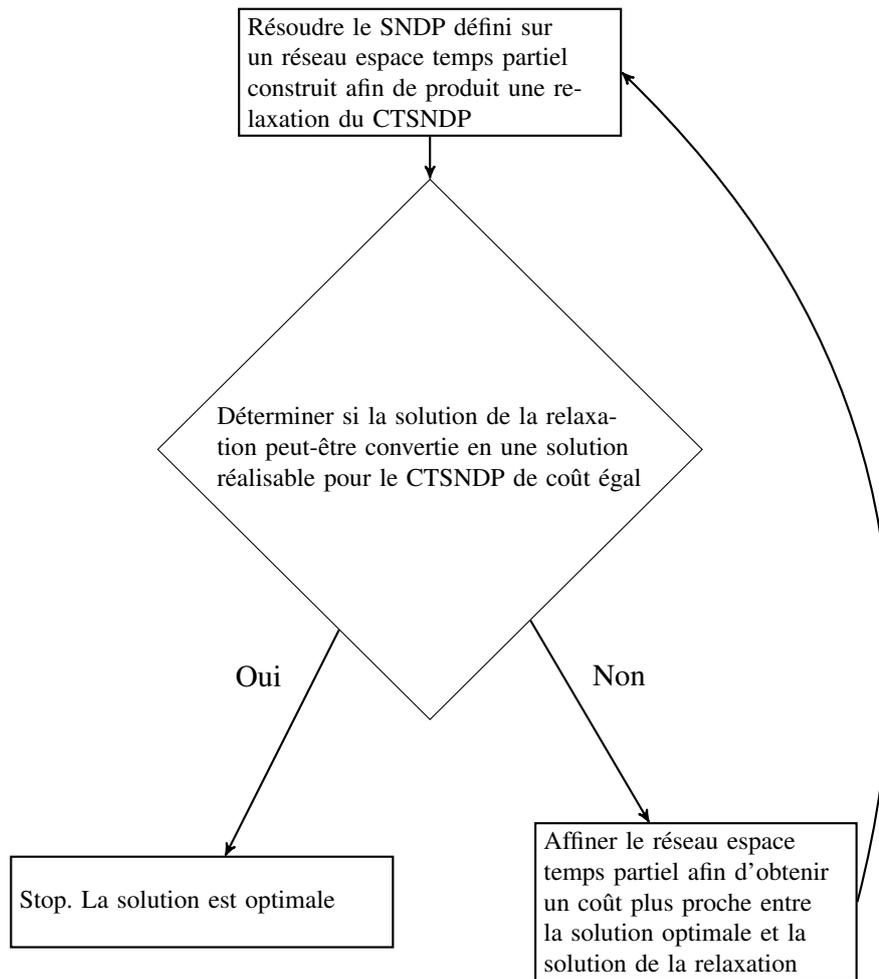


Figure 8.6 – Diagramme de flux

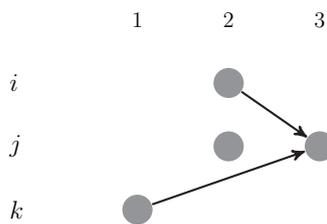


Figure 8.7 – Un réseau espace-temps partiel

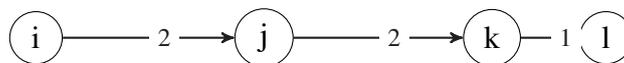
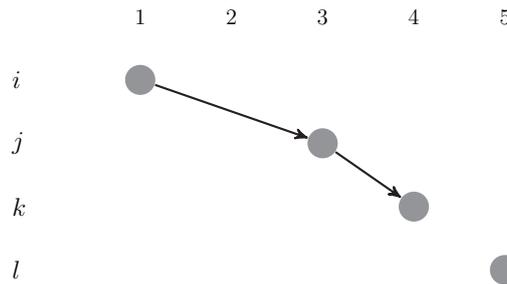


Figure 8.8 – Un réseau plat

Figure 8.9 – Un exemple de *early arrival path*

Boland et al. [10] montrent qu'un réseau espace-temps partiel ainsi construit admet le **SNDP** comme relaxation du **CTSNDP**, si pour chaque sommet $(i, t) \in N_{\mathcal{T}}$ et chaque arc $(i, j) \in A$, on trouve un arc $((i, t), (j, t')) \in A_{\mathcal{T}}$ tel que $t - t' \leq tt_{ij}$.

On décrit maintenant comment un réseau espace-temps est affiné lorsqu'une solution ne peut pas être convertie en une solution pour le **CTSNDP** de même coût avec l'exemple suivant :

Exemple 20. Soit un ensemble de commandes à transporter dans le tableau 8.10a. La figure 8.10b illustre le réseau plat dans lequel ces commandes peuvent transiter. On définit pour chaque arc une capacité égale à 5, un coût fixe égal à 10 et un coût variable nul. Pour une itération donnée, on considère la solution de la relaxation du **CTSNDP** représentée par la figure 8.10c. On observe que dans cette solution de la relaxation, les commandes k_1 et k_2 sont transportées ensemble sur l'arc $((j, 3), (l, 6))$, ce qui signifie que le coût fixe de l'arc (j, l) n'est payé qu'une seule fois. La commande k_1 doit arriver au sommet l avant la période 6. L'arc (j, l) ayant une durée égale à 3, k_1 doit donc partir au plus tard de j à la période 3. La commande k_2 devient disponible à la période 2. L'arc (k, j) ayant une durée de transport égale à 2, k_2 ne peut pas arriver au sommet j avant la période 4. C'est pourquoi le fait que l'arc $((k, 2), (j, 3))$ soit trop court autorise une consolidation sur l'arc $((j, 3), (l, 6))$ alors qu'il n'est pas possible de consolider les deux commandes dans le réseau espace-temps complet. Cette solution ne peut donc pas être convertie en une solution de coût égal du **CTSNDP**.

La raison pour laquelle l'arc $((k, 2), (j, 3))$ est trop court est due à l'absence de sommets dans le réseau espace temps partiel qui permettent de modéliser la durée de transport réelle sur l'arc (k, j) . L'identification de cet arc permet à l'algorithme **DDD** d'affiner le réseau : il découvre un arc qui permet une consolidation fantôme (l'arc $((k, 2), (j, 3))$ dans notre exemple) car il est trop court, ajoute un nouveau sommet et modifie les arcs sur le réseau espace-temps partiel de manière à modéliser la durée réelle des transports. La figure 8.10d montre le réseau espace temps après affinement. On remarque que le sommet $(j, 4)$ a été ajouté, ce qui permet à l'arc $((k, 2), (j, 4))$ de remplacer l'arc $((k, 2), (j, 3))$. De la même manière, l'arc $((j, 4), (l, 7))$ est ajouté au réseau. En utilisant ce nouveau réseau espace-temps, la consolidation de la solution précédente ne peut plus se produire.

On renvoie le lecteur intéressé à l'article [10] pour la description détaillée de la méthode de découverte de consolidations fantômes et des arcs qui leur permettent d'exister. Cet article présente également les méthodes qui convertissent une solution issue de la relaxation en solution du **CTSNDP**.

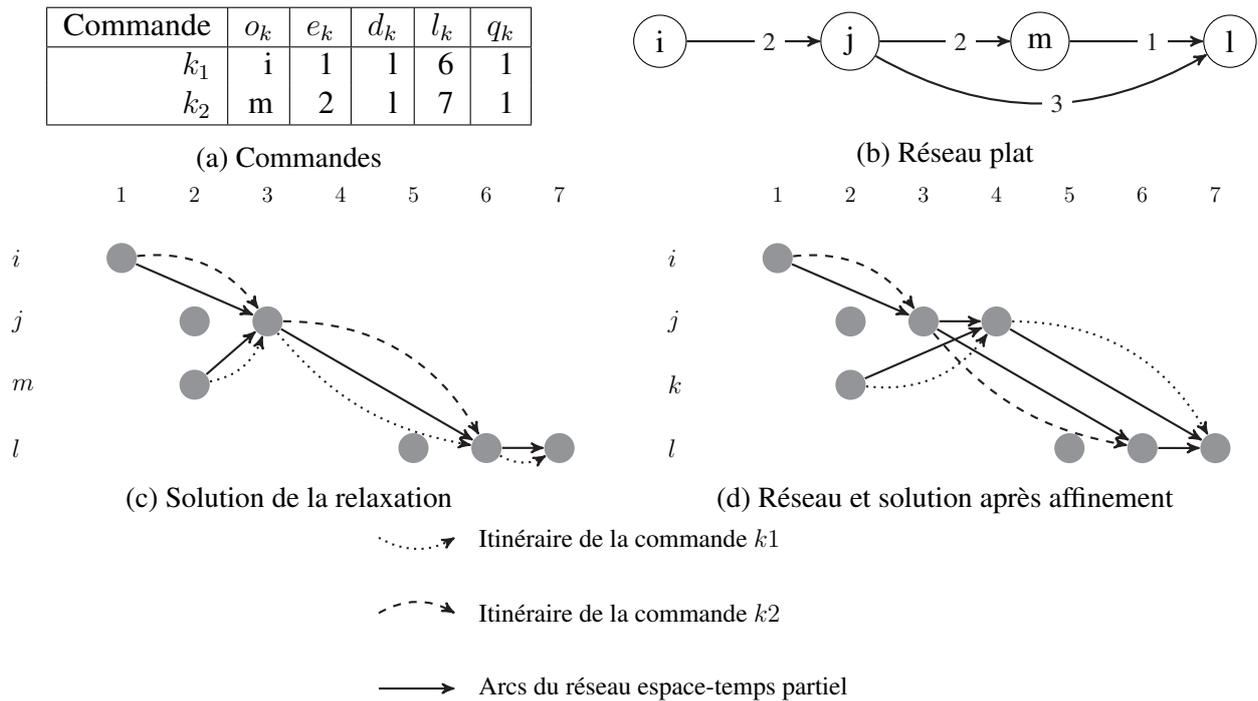


Figure 8.10 – Exemple de l’affinement d’un réseau espace temps-partiel

8.5 Tests numériques

Cette section présente les expérimentations réalisées sur les formulations \mathcal{F}_R et \mathcal{F}_A présentées respectivement dans les sections 8.3.4 et 8.3.5. La section 8.5.1 propose une description des instances de test qui ont été générées. Les performances des deux formulations sont comparées en section 8.5.2. La section 8.5.3 propose une comparaison des coûts en fonction de certains paramètres du réseau logistique.

8.5.1 Instances

Les instances générées dans ce chapitre sont basées sur un réseau réel basé en France : les sites des clients sont des plateformes de la grande distribution ; les hubs sont positionnés dans les lieux où 4S Network souhaite implanter des CRC et les fournisseurs sont placés dans les zones à forte densité industrielle. On considère qu’un client reçoit une unique commande.

Les instances sont générées avec les caractéristiques suivantes :

- Le nombre total de commandes est de 25, 50, 100 ou 200 (un client peut recevoir une seule commande). 10 ensembles de commandes sont créés pour chacune des tailles d’instance ce qui fait un total de $10 \times 4 = 40$ ensembles de commandes différents.
- La largeur des fenêtres de temps $[e_o, l_o]$ des commandes varient entre 24h et 36h selon la distance entre l’origine et la destination de la commande.
- Pour chaque ensemble de commandes généré, quatre instances sont créées, avec respectivement 4,6,8 et 10 hubs. Pour permettre une meilleure comparaison, le réseau de l’ensemble de 4 hubs

est inclus dans l'ensemble de 6 hubs etc. Cela nous permet d'observer les variations de prix en fonction du nombre de hubs dans le réseau. On retrouve donc $4 \times 40 = 160$ instances de test.

Pour la formulation $\mathcal{F}_{\mathcal{R}}$, l'ensemble R des tournées n'a pas été généré intégralement. Pour des raisons d'espace mémoire, cet ensemble a été limité à l'ensemble des tournées comportant au plus quatre clients. Dans nos expérimentations, les deux formulations ne sont donc pas équivalentes et la résolution à l'optimum de la formulation $\mathcal{F}_{\mathcal{R}}$ fournit une borne supérieure de l'optimum du **SNDRP**. La vitesse de résolution de $\mathcal{F}_{\mathcal{R}}$ ne peut en revanche qu'être réduite par cette limitation.

8.5.2 Comparaison des deux formulations

Toutes les instances sont résolues par l'algorithme **DDD** avec une limite de temps d'exécution de 2h. Une solution est considérée optimale si la différence entre la borne supérieure et la borne inférieure calculées par l'algorithme est inférieur à 1%. Nous comparons la performance des deux formulations, on s'intéresse à trois indicateurs : le temps de résolution, le nombre d'instances pour lesquelles l'algorithme trouve une solution optimale et réalisable, et l'écart à la borne inférieure si le lorsque cette solution est trouvée. La méthode trouve une solution réalisable si elle vient à bout de sa première itération. Ce n'est pas toujours le cas en raison d'un dépassement de la mémoire lors de la résolution du premier modèle. On présente donc les temps de résolution uniquement pour les instances résolues à l'optimum. le tableau 8.1 montre le pourcentage d'instances résolues et le pourcentage d'instances pour lesquelles l'algorithme trouve une solution réalisable, en fonction du nombre de clients. La formulation $\mathcal{F}_{\mathcal{A}}$ obtient une solution réalisable pour toutes les instances et une solution optimale pour 100%, 83%, 53% et 45% respectivement pour les instances à 25, 50, 100 et 200 commandes. En revanche, la formulation sur les tournées trouve une solution réalisable pour 100%, 82%, 50%, 25% respectivement pour un nombre de commandes de 25, 50, 100 et 200, et a un pourcentage de solution optimales assez faible (72%, 25%, 4% et 1%). Le tableau 8.2 indique pour chacune des formulations le temps de résolution en fonction du nombre de clients. On remarque clairement que la formulation $\mathcal{F}_{\mathcal{A}}$ est beaucoup plus rapide que $\mathcal{F}_{\mathcal{R}}$, même sur de petites instances. La comparaison entre les deux formulations pour 200 clients n'est pas significative, car très peu d'instances sont résolues pour la formulation $\mathcal{F}_{\mathcal{R}}$:

Enfin, la figure 8.3 montre le saut d'optimalité moyen, en fonction du nombre clients. La formulation $\mathcal{F}_{\mathcal{A}}$ obtient de bons résultats pour les instances de 25 à 50 clients (respectivement 0,7% et 1,7% en moyenne). Les écarts augmentent progressivement avec le nombre de clients, avec 7,1% pour 100 clients et 12,2% pour 200 clients. Pour la formulation $\mathcal{F}_{\mathcal{R}}$, si la moyenne du saut d'optimalité est satisfaisante pour 25 clients (2,6%), elle augmente très rapidement à 10,1%, 12,1% et 11% pour respectivement 50, 100, et 200 clients. Encore une fois, les résultats des instances à 200 clients pour la formulation sur les tournées n'est pas significative au vu du peu d'instances résolues. Ces résultats montrent que la formulation sur les arcs est clairement plus performante que la formulation sur les tournées.

8.5.3 Comparaison logistique

Nous nous intéressons ici aux coûts obtenus avec la formulation $\mathcal{F}_{\mathcal{A}}$ en fonction du nombre de hubs. Le tableau 8.4 indique le coût moyen du transport d'une commande en fonction du nombre de hubs et du nombre de commandes total. En complément d'information, le tableau 8.5 indique les sauts d'optimalité moyens pour ces deux critères. Les valeurs en gras dans ces tableaux sont les configurations (Nombre de commandes Nombre de hub) pour lesquelles les résultats peuvent être interprétés car le saut d'optimalité

# commandes	# hub	\mathcal{F}_R		\mathcal{F}_A	
		# optimales	# réalisables	# optimales	# réalisables
25	4	100%	100%	100%	100%
	6	100%	100%	100%	100%
	8	83%	100%	100%	100%
	10	5%	100%	100%	100%
	Moyenne	72%	100%	100%	100%
50	4	78%	78%	100%	100%
	6	23%	80%	90%	100%
	8	0%	85%	80%	100%
	10	0%	85%	60%	100%
	Moyenne	25%	82%	83%	100%
100	4	18%	50%	80%	100%
	6	0%	50%	40%	100%
	8	0%	50%	45%	100%
	10	0%	50%	45%	100%
	Moyenne	4%	50%	53%	100%
200	4	3%	25%	50%	100%
	6	0%	25%	40%	100%
	8	0%	25%	45%	100%
	10	0%	25%	45%	100%
	Moyenne	1%	25%	45%	100%

Tableau 8.1 – Pourcentage d’instances résolues optimalement et solutions réalisables en fonction du nombre de commandes et du nombre de hubs

# commandes	# hubs	Temps de résolution (en secondes)	
		$\mathcal{F}_{\mathcal{R}}$	$\mathcal{F}_{\mathcal{A}}$
25	4	24	7
	6	205	8
	8	2247	13
	10	5106	20
	Moyenne	813	12
50	4	1084	75
	6	2983	680
	8	-	1119
	10	-	808
	Moyenne	1511	627
100	4	1971	1398
	6	-	56
	8	-	89
	10	-	145
	Moyenne	1971	593
200	4	774	901
	6	-	402
	8	-	670
	10	-	1223
	Moyenne	774	813

Tableau 8.2 – Temps de résolution en fonction du nombre de commandes et du nombre de hubs pour les instances résolues de façon optimale

# commandes	# hubs	Saut d'optimalité moyen	
		\mathcal{F}_R	\mathcal{F}_A
200	4	0,86%	0,44%
	6	0,88%	0,42%
	8	1,54%	0,49%
	10	7,22%	0,37%
	Moyenne	2,62%	0,43%
200	4	0,94%	0,51%
	6	3,44%	2,55%
	8	9,42%	1,07%
	10	25,28%	1,75%
	Moyenne	10,07%	1,47%
200	4	1,77%	0,59%
	6	7,30%	3,15%
	8	14,58%	4,64%
	10	24,91%	7,54%
	Moyenne	12,14%	3,99%
200	4	2,32%	1,55%
	6	6,87%	3,60%
	8	14,76%	8,49%
	10	20,23%	13,59%
	Moyenne	11,05%	6,85%

Tableau 8.3 – Écart à la borne inférieure en fonction du nombre de commandes et du nombre de hubs

est correct. On observe que le nombre de hubs dans le réseau influe très peu sur le coût du transport d'une commande dans nos instances. En revanche, le nombre de commandes de l'instance est un facteur qui joue énormément sur les résultats : par exemple, pour le réseau comportant 4 hub, le coût du transport d'une commande diminue de 41% entre les instances comportant 25 commandes et celles à 200.

# commandes / # hub	4	6	8	10
25	426	406	408	399
50	354	337	339	334
100	293	278	286	287
200	251	236	249	255

Tableau 8.4 – Moyenne du coût moyen de transport par commande en fonction du nombre de clients et du nombre de hubs

# commandes / # hub	4	6	8	10
25	0,4%	0,4%	0,5%	0,4%
50	0,5%	2,6%	1,1%	1,7%
100	0,6%	3,2%	4,6%	7,5%
200	1,5%	3,6%	8,5%	13,6%

Tableau 8.5 – Moyenne de l'écart à la borne inférieur en fonction du nombre de clients et du nombre de hubs

8.6 Conclusion

Ce chapitre présente le travail réalisé en collaboration avec Mike Hewitt à Loyola University de Chicago. Deux formulations sont proposées afin de résoudre un problème de transport intégrant un problème de plan de chargement et de tournées de véhicules. La première méthode permet d'intégrer des contraintes *industrielles* ou *riches* sur les tournées de véhicules, qui sont énumérées en pré-traitement des données. Cependant elle génère un nombre élevé de variables et ne permet pas de résoudre des problèmes de grande taille. La seconde formulation utilise des variables définies sur les arcs pour modéliser les tournées de véhicules. Ces deux formulations sont testées avec l'algorithme développé dans Boland et al. [10]. Les expérimentations permettent de confirmer les différences de performances entre les deux adaptations. Cette contribution est originale sur le plan applicatif et méthodologique car elle constitue une des premières extensions de l'algorithme **DDD**.

Plusieurs perspectives de recherche se dégagent de ces travaux. Concernant la formulation sur les tournées, il serait intéressant de générer dynamiquement les tournées au lieu de les énumérer au préalable, en utilisant par exemple des techniques de génération de colonnes. Cela permettrait notamment de réduire le nombre de variables lorsque la taille du problème augmente. Une seconde perspective est l'extension de la formulation sur les arcs. L'affectation préalable de chaque client à un hub pourrait être évitée en intégrant au modèle des contraintes interdisant qu'un véhicule démarrant sa tournée sur

un hub la termine sur un autre. Une première approche envisagée repose néanmoins sur la formulation d'un nombre exponentiel de contraintes, ce qui implique de mettre en œuvre des techniques de *branch-and-cut*. Une dernière perspective porte sur la résolution de problèmes intégrés de localisation de hubs et de conception de tournées de véhicules.

Outils logiciels réalisés

Sommaire

9.1	Choix technologiques	146
9.2	Outil de simulation	146
9.2.1	Présentation de l’outil	147
9.2.2	Données d’entrées et paramètres	147
9.2.3	Résultats	148
9.3	Outil opérationnel	150
9.3.1	Présentation de l’outil	150
9.3.2	Utilisation de l’outil	151
9.4	Conclusion et perspectives	155

Dans toute thèse en collaboration avec une entreprise, un des points clé est l’application de la recherche pour réaliser une étude ou un outil qui répond aux besoins de l’entreprise. Dans cette thèse, deux outils ont été développés : le premier appelé *outil de simulation* a servi à une pré-étude d’estimation des coûts du réseau logistique mutualisé. Le second est appelé *outil opérationnel* aide à la gestion du réseau aval et est installé et utilisé quotidiennement chez 4S Network.

Le projet CRC a débuté en même temps que cette thèse. Cela explique notamment les différences de niveau de décisions des différents problèmes traités par cette thèse. Dans la première phase du projet début 2014, le concept d’un réseau avec CRC a été validé par un premier outil de simulation. Son développement a été réalisé grâce au modèle décrit au chapitre 5. Par la suite, un outil d’aide à la décision a été développé durant les années 2015 et 2016 pour une utilisation opérationnelle. Il correspond à l’intégration de l’algorithme de tournées de véhicules riches décrit au chapitre 6.

Après une première analyse des besoins permettant d’établir un cahier des charges, le développement et l’évolution des outils se sont déroulés de manière itérative : la précision de certaines contraintes ont

fait évolué les outils pendant les périodes de travail chez 4S Network. Ce fonctionnement a permis de développer des outils en adéquation avec les besoins de l'entreprise.

9.1 Choix technologiques

L'objectif du développement de ces deux logiciels était de réaliser des prototypes utilisables par 4S Network. Néanmoins, il était nécessaire qu'ils soient relativement simples à développer dans le cadre de la thèse. Ces outils devaient pouvoir être installés sur des PC de bureau avec un système d'exploitation Windows récent.

L'interface de l'outil est réalisé via la bibliothèque graphique *Java Swing*. Un fichier Excel regroupe toutes les informations du problème à traiter. Ces données sont extraites grâce à la librairie de l'API POI qui permet notamment de lire et écrire dans un fichier Excel. Ensuite, une matrice regroupant les durées de transport entre tous les sommets du réseau logistique est récupérée en appelant l'API web de Open Source Routing Machine (OSRM). Cette API permet de récupérer les matrices de temps de transport à partir de coordonnées géographiques. Le problème est ensuite résolu dans le module d'optimisation, qui utilise le solveur CBC avec la librairie libre OR Tools. La figure 9.1 représente les interactions entre les modules développés et les librairies utilisées.

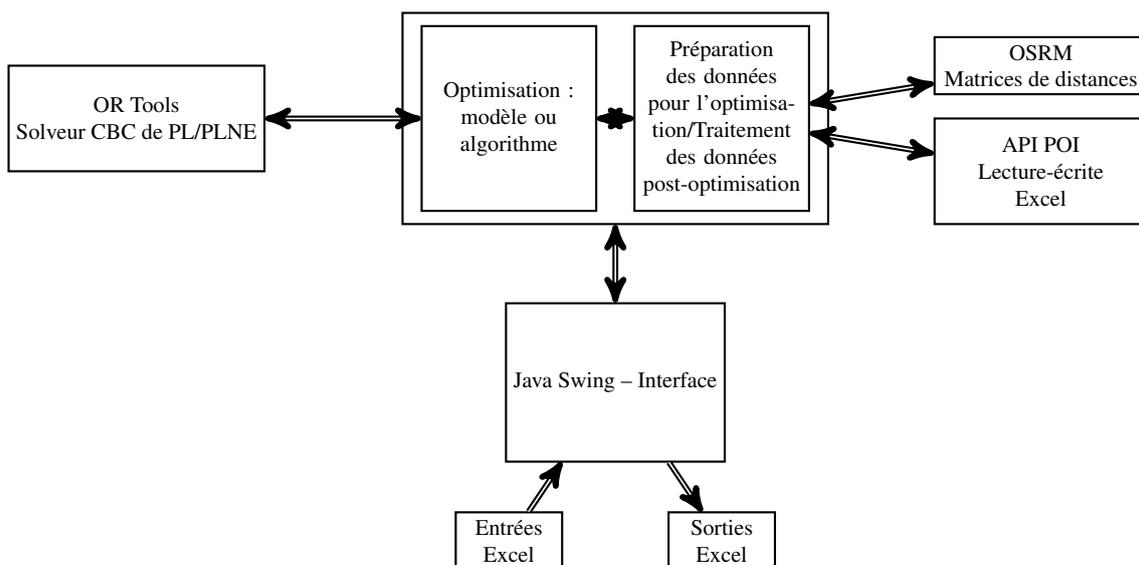


Figure 9.1 – Interactions entre les modules des outils développés

9.2 Outil de simulation

L'outil de simulation réalise une estimation des coûts de transport dans le réseau mutualisé. Les données permettant cette simulation sont les historiques de commandes des fournisseurs sur une année. Dans la phase initiale du projet, 4S Network a utilisé ce module pour proposer aux fournisseurs intéressés par le projet, une étude comparative des coûts de transport entre leur chaîne logistique actuelle et celle pro-

posée par 4S Network. Cette étude comportait plusieurs scénarios/simulations afin de pouvoir observer les meilleures configurations du réseau logistique.

Dans le cadre de cette thèse, j'ai été la première utilisatrice de l'outil pour fournir des résultats initiaux. J'ai par la suite développé une interface simple afin de rendre l'outil utilisable par 4S Network.

9.2.1 Présentation de l'outil

L'outil de simulation permet de résoudre itérativement une série de problèmes d'optimisation présentés dans le chapitre 5 : chaque jour de l'historique est considéré comme un problème à résoudre et modélisé avec le modèle décrit chapitre 5. Le jeu de données est l'historique des commandes d'un fournisseur passées par ses clients, généralement sur un horizon d'un an.

Le module propose deux types de restitution : une première analyse comprenant les coûts de transport détaillés par jour, et une synthèse plus globale sur une période temporelle sélectionnée. Un pré-traitement des données est réalisé afin de modéliser plusieurs scénarios de transports. Il est possible de paramétrer deux éléments :

- La fréquence de livraison : elle correspond au nombre de jours par semaine où un client reçoit les commandes en provenance d'un fournisseur.
- Les jours de livraison : une fois la fréquence définie, il est possible de choisir les jours de livraison pour chaque couple fournisseur-client.

De nouvelles contraintes ont également été rajoutées au modèle présenté dans le chapitre 5 :

- Une structure de coût variable pour les tournées amont et directes : l'utilisateur a le choix entre un tarif linéaire par morceaux (comme définie dans le chapitre 5), un tarif linéaire proportionnel au nombre de kilomètres parcourus, et un tarif fixe.
- Il est possible de définir un seuil exprimé en nombre de palettes pour chaque couple fournisseur/client. Si le nombre de palettes d'une commande dépasse le seuil défini, alors cette commande passe obligatoirement par le réseau direct, et dans le cas inverse la commande passe obligatoirement par le réseau mutualisé.
- De nouvelles contraintes portant sur la génération des tournées ont été introduites. Par exemple, il est possible de paramétrer la valeur maximale du détour. Il est également possible d'interdire une tournée comprenant deux clients qui, pour des raisons de confidentialité, ne souhaitent pas que leurs commandes soit livrées par le même véhicule. Il en est de même pour les fournisseurs, qui peuvent refuser de partager un véhicule avec un de leurs concurrents.

9.2.2 Données d'entrées et paramètres

9.2.2.1 Paramètres du réseau logistique

Un premier fichier Excel regroupe toutes les informations du réseau logistique concernant les sites fournisseurs, les CRC, les clients, les fréquences et jours de livraison par clients. Il regroupe également les caractéristiques de la flotte dans le réseau amont, direct et aval, les matrices de distances et les structures de coût du transport.

9.2.2.2 Commandes

Les différents jeux de données regroupent les informations des commandes et sous-commandes : jour, mois et année de la sous-commande, référence de commande, site fournisseur d'origine, site client d'arrivée, et nombre de palettes.

9.2.2.3 Interface de lancement

L'interface de lancement de la simulation (figure 9.2) permet de spécifier les derniers réglages sans passer par un fichier Excel. Ainsi, il est possible de sélectionner les données d'entrées et le chemin du dossier où les fichiers de sortie sont placés. Plusieurs options permettent également de choisir :

- la modification des fréquences et dates de livraison ;
- si toutes les sous-commandes d'un fournisseur à destination d'un même client doivent être regroupées.

Ensuite, certains paramètres sont réglables pour chaque tournée de chaque réseau :

- Cinq options contraignent la génération des tournées amont : s'il est possible de générer des tournées amont sans CRC, et s'il est possible de générer des tournées amont avec plusieurs CRC. Il est également possible de paramétrer le nombre maximal de sites visités dans une tournée ainsi que le détour maximal d'une tournée. Enfin, on peut déterminer le type de structure de coût pour les tournées dans le réseau amont : fonction linéaire par morceaux, coût au km ou coût fixe.
- Pour les tournées aval, il est possible de paramétrer le nombre de clients visités ainsi que le détour maximal de la tournée.
- Enfin, pour les tournées directes, il est possible comme pour les tournées amont de choisir la structure de coût, le nombre de sites visités et le détour maximal. Ensuite, on détermine si les tournées directes *multipick* – c'est à dire une tournée qui passe par plusieurs sites de fournisseurs avant d'aller chez un client, et/ou *multidrop*, c'est à dire la possibilité de visiter dans la même tournée plusieurs clients – sont autorisées.

9.2.3 Résultats

L'outil de simulation génère deux types de fichiers de restitution : un fichier Excel pour chaque jour de l'historique, détaillant la solution proposée, et un fichier de restitution des résultats globaux sur l'historique de transport.

9.2.3.1 Restitution par jour

Un premier fichier Excel est généré pour chaque jour de la période de résolution concernée. Ce fichier regroupe les itinéraires des sous-commandes et les tournées de véhicules de la journée considérée. Il contient également des statistiques globales sur la journée telles que le nombre de passages sur CRC, le coût total, le nombre de véhicules utilisés, le nombre de tournées utilisées, le coût moyen à la palette, le coût total pour les tournées dans le réseau amont, aval et direct, et le coût par fournisseur.

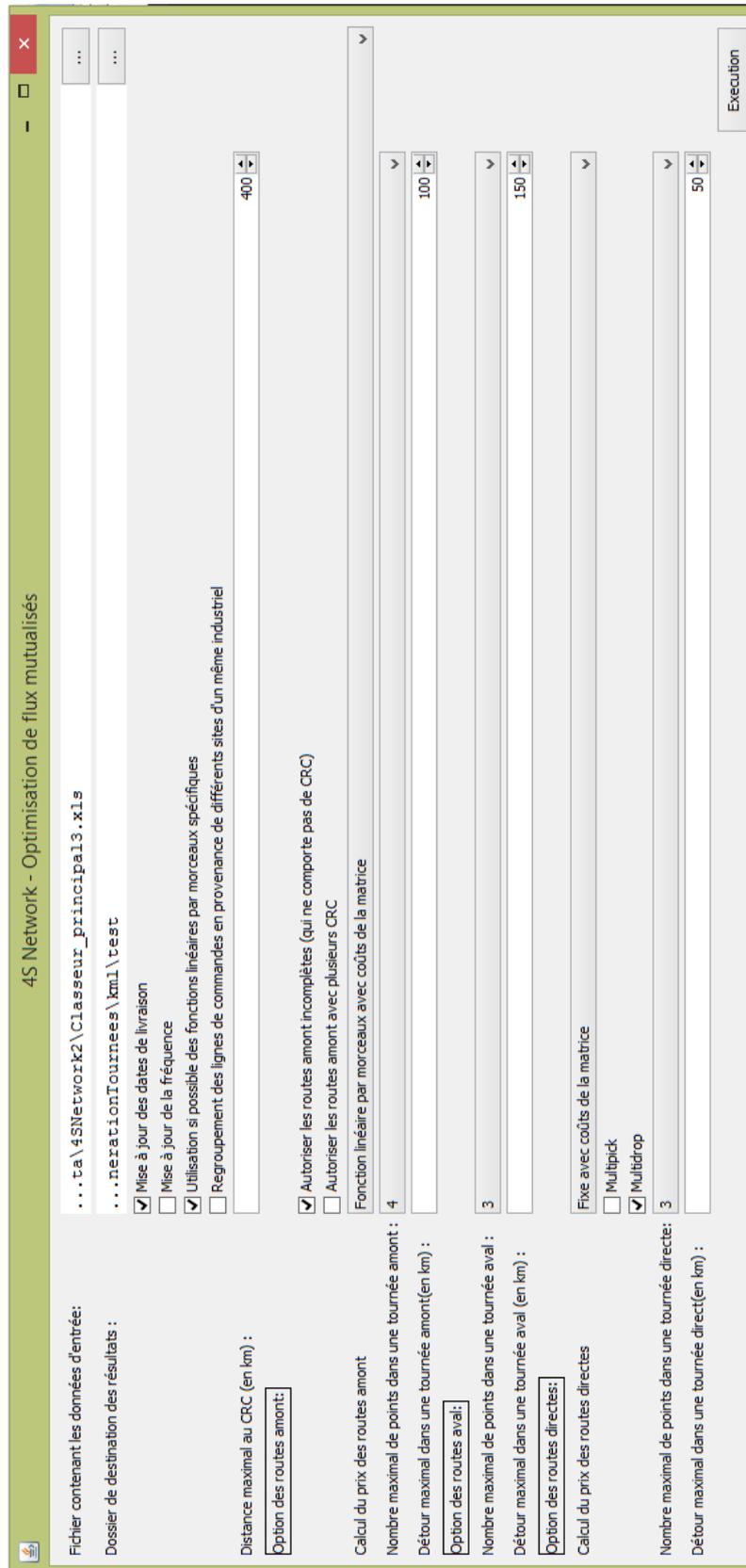


Figure 9.2 – Interface du simulateur

9.2.3.2 Synthèse globale

Un fichier Excel regroupe une synthèse globale des simulations. Chaque ligne de la feuille représente une journée de simulation avec les données suivantes :

- les informations du jeu de données concernant l'année, le mois, le jour, le nombre de palettes total, le nombre de commandes, la moyenne du nombre de palettes par commande ;
- des informations générales sur la solution constitué du nombre et coût de passage sur CRC, du nombre et coût des ouvertures de porte dans les tournées amont et aval, du nombre de véhicules utilisés au total, du coût total et du coût des tournées amont, aval et directes ;
- des informations utiles à la comparaison entre le réseau mutualisé et le réseau direct telles que le nombre de palettes passant par le réseau direct, le coût si toutes les commandes passent par le réseau direct, le coût à la palette, le coût à la palette pour le réseau amont et aval, et le nombre total de livraisons.

9.3 Outil opérationnel

Au début de la deuxième année de thèse, un projet pilote a été initié avec les fournisseurs, transporteurs et clients qui souhaitaient utiliser le service de transport mutualisé proposé par 4S Network. Un premier CRC a été installé à Saint-Martin-De-Crau, dans le sud de la France. Dans un premier temps, afin de simplifier la mise en place du service, 4S Network a décidé de gérer uniquement le transport en aval des CRC. Dans ce but, un outil de gestion opérationnelle des tournées aval au CRC a été développé. Cet outil était initialement dédié à l'optimisation des tournées de véhicules depuis Saint-Martin-De-Crau. Par la suite, deux nouveaux CRC ont ouvert à Arras et Brumath, et l'outil opérationnel a été adapté pour générer des tournées depuis ces plateformes.

9.3.1 Présentation de l'outil

Cet outil a été développé pendant les deuxièmes et troisièmes années de thèse. L'objectif était de réaliser un outil d'aide à la décision pour la détermination des tournées depuis chaque CRC. Il utilise l'algorithme décrit au chapitre 6 qui résout le problème riche de tournées de véhicules depuis un CRC.

9.3.1.1 Données d'entrée et paramètres

9.3.1.1.1 Jeu de données Un fichier Excel est automatiquement généré à partir du TMS (Transportation Management System) de 4S Network. Ce fichier contient toutes les informations des commandes qui doivent être livrées :

- code GLN (Global Location Number, qui est un code d'identification générique du lieu attribué par la société GS1), nom, coordonnées géographiques de l'origine (CRC) et de la destination (client), nom de l'fournisseur d'origine ;
- numéro du bon de livraison, numéro de la commande, nombre de palettes, poids, compatibilité du produit (voir chapitre 6 section 6.1.1) ;
- jour de livraison, horaire fixe ou fenêtre horaire de livraison.

9.3.1.1.2 Paramètres généraux et transporteurs Afin de spécifier chaque caractéristique et structure tarifaire des transporteurs, un fichier Excel regroupe l'ensemble des types de véhicules. Une feuille Excel est utilisée par type de véhicule (voir chapitre 6 partie 6.1.3.1), comme illustré sur la figure 9.3. Le premier tableau en haut à gauche regroupe les caractéristiques du type de véhicule, tel que le nom complet et le type du véhicule, la capacité en nombre de palettes, la capacité en poids, si le véhicule est capable de transporter des produits frais, le nombre de véhicules disponibles et si le véhicule peut réaliser plusieurs tournées de livraison dans la journée. Ensuite, différentes composantes de structures tarifaires sont proposées pour représenter la structure de tarif du transporteur concerné. Pour plus de détails concernant ces structures de coûts, se référer au chapitre 6 en section 6.1.3.3.

Enfin, l'interface qui permet de lancer l'application est illustrée figure 9.4. Elle permet de choisir le jour de livraison pour lequel il faut générer les tournées et quelques options concernant les contraintes des tournées :

- le CRC que l'on sélectionne pour distribuer les commandes ;
- le nombre maximal d'arrêts ;
- le détour maximal ;
- la durée maximale ;
- si les commandes provenant d'un même fournisseur et à destination du même client doivent obligatoirement être transportées ensembles.

9.3.1.2 Données de sorties

Un fichier Excel est généré pour afficher les résultats. Dans ce fichier, deux feuilles Excel représentent deux vues différentes du résultat de l'optimisation.

- Une *vue commande*. La figure 9.5 illustre un exemple de cette vue. Chaque ligne représente une sous-commande.
- Une *vue véhicule*. La figure 9.6 illustre un exemple de la *vue véhicule*. Chaque ligne représente une tournée de véhicule

9.3.2 Utilisation de l'outil

Cet outil est utilisé pour la gestion opérationnelle des CRC. Chaque jour, il est nécessaire de déterminer les tournées et indiquer aux transporteurs partenaires du réseau quelles tournées doivent être réalisées le lendemain. Selon la journée, le nombre et le type de véhicules n'est pas le même. En outre, de nombreux échanges entre transporteurs et 4S Network sont parfois nécessaires à la bonne planification des tournées. C'est pourquoi l'algorithme développé pour cet outil avait pour objectif de fournir de bonnes solutions rapidement. Enfin, les transporteurs indiquent parfois au dernier moment à 4S Network de nouvelles contraintes appliquées à une journée spécifique. Il est difficile d'intégrer ces contraintes très spécifiques et ponctuelles dans un outil qui se veut générique. Le module développé est donc utilisé

	A	B	C	D	E	F	G	H	I	J
	NOM DU VEHICULE		SEMI-REMORQUE	NOM DU CRC ASSOCIE						
1	Nom /TYPE									
2	Capacité max	33								
3	Poid max	24000								
4	Type (0 sec, 1 frais, 3 si tout compatible)	3								
5	Nombre disponible	2								
6	Multi trip (0 ou 1)	0								
7	Distribution à la palette	0								
8										
9	Coût fixe	80								
10	Coût au km	0								
11										
12										
13										
14	Inferieur à	Coût fixe par zone		Coût au km par zone				Coût ouverture de porte		
15		Coût		Inferieur à	Coût			1ère	30	
16					30			2ème	30	
17					65			autres	30	
18					105					
19					500					
20										

Figure 9.3 – Format de la feuille Excel définissant les paramètres pour un transporteur

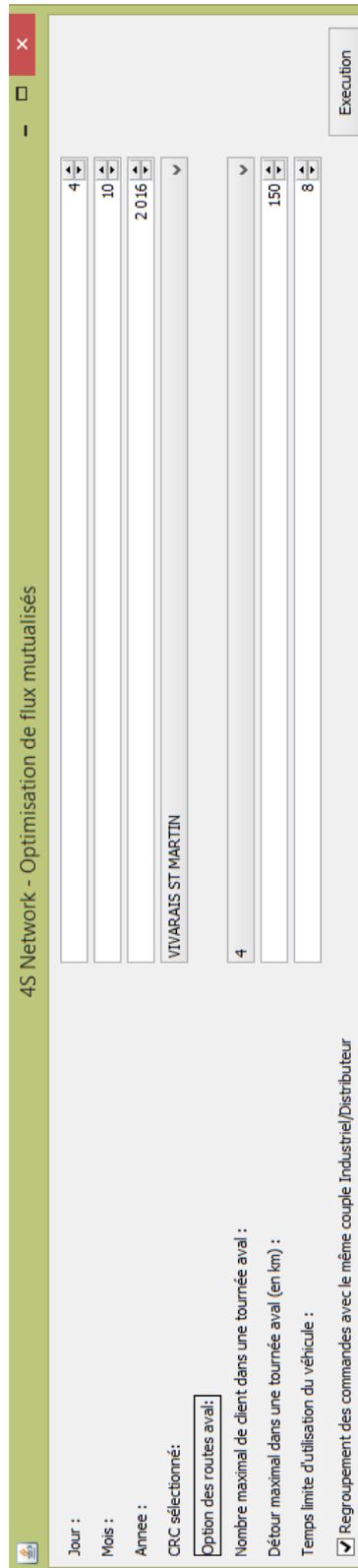


Figure 9.4 – Interface de l’outil opérationnel

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Industriel	Site distributeur	Numero commande	Site distributeur	Depart	Nombre	Poid	Date livraison	Heure de	Heure d	Type de vehicule	Coût total	Distance	Temps total	Compatibilité transport
SC JOHNSON	DISTRIBUTEUR	Num commande	DISTRIBUTEUR				04/10/2016	8h00	7:00	Vehicle PORTEUR	132	52'1h00		DPH ODOORANT
SC JOHNSON	DISTRIBUTEUR	Num commande	DISTRIBUTEUR				04/10/2016	8h00	7:00	Vehicle PORTEUR	132	52'1h00		DPH ODOORANT
1	CRC													
2	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	1	100	04/10/2016	8h00	7:00	Vehicle PORTEUR	132	52'1h00	DPH ODOORANT
3	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	2	200	04/10/2016	8h00	7:00	Vehicle PORTEUR	132	52'1h00	DPH ODOORANT
4														
5	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		83	1	377	04/10/2016	-	6:30	SEMI SEC FRAIS	325	635'5h30	ALIMENTAIRE TEMP 15°
6	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	14	7898	04/10/2016	12h00	6:30	SEMI SEC FRAIS	325	635'5h30	ALIMENTAIRE TEMP 15°
7	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	7	700	04/10/2016	12h00	6:30	SEMI SEC FRAIS	325	635'5h30	ALIMENTAIRE TEMP 15°
8	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	0	10	04/10/2016	12h00	6:30	SEMI SEC FRAIS	325	635'5h30	ALIMENTAIRE TEMP 15°
9	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	5	1472	04/10/2016	8h30	6:30	SEMI SEC FRAIS	325	635'5h30	ALIMENTAIRE TEMP 15°
10														
11	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	2	125	04/10/2016	14h00	7h30	DEDIE SMC SEMI FRAIS	40	94'6h00	DPH ODOORANT
12	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	2	109	04/10/2016	14h00	7h30	DEDIE SMC SEMI FRAIS	40	94'6h00	DPH ODOORANT
13	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	1	48	04/10/2016	14h00	7h30	DEDIE SMC SEMI FRAIS	40	94'6h00	DPH ODOORANT
14	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	1	148	04/10/2016	11h00	7h30	DEDIE SMC SEMI FRAIS	40	94'6h00	DPH ODOORANT
15	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	7	832	04/10/2016	9h00	7h30	DEDIE SMC SEMI FRAIS	40	94'6h00	DPH SEC
16														
17	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	5	500	04/10/2016	14h00	11:00	DEDIE SMC SEMI FRAIS	63	148'2h00	DPH SEC
18	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	1	100	04/10/2016	14h00	11:00	DEDIE SMC SEMI FRAIS	63	148'2h00	DPH SEC
19	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	2	146	04/10/2016	13h00	11:00	DEDIE SMC SEMI FRAIS	63	148'2h00	DPH ODOORANT
20	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	3	197	04/10/2016	13h00	11:00	DEDIE SMC SEMI FRAIS	63	148'2h00	DPH ODOORANT
21	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	2	190	04/10/2016	13h00	11:00	DEDIE SMC SEMI FRAIS	63	148'2h00	DPH ODOORANT
22														
23	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	2	565	04/10/2016	11h00	5:00	DEDIE SMC SEMI FRAIS	93	217'4h00	ALIMENTAIRE TEMP 15°
24	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	3	2188	04/10/2016	9h00	5:00	DEDIE SMC SEMI FRAIS	93	217'4h00	ALIMENTAIRE TEMP 15°
25	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	5	500	04/10/2016	9h00	5:00	DEDIE SMC SEMI FRAIS	93	217'4h00	ALIMENTAIRE TEMP 15°
26	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		34	2	596	04/10/2016	7h00	5:00	DEDIE SMC SEMI FRAIS	93	217'4h00	ALIMENTAIRE TEMP 15°
27														
28	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	14	10629	04/10/2016	8h00	2:30	DEDIE SMC SEMI FRAIS	74	40'5h30	ALIMENTAIRE TEMP 15°
29	RETOUR AU CRC													
30	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	10	5424	04/10/2016	6h00	2:30	DEDIE SMC SEMI FRAIS	74	40'5h30	ALIMENTAIRE TEMP 15°
31	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	12	8084	04/10/2016	4h00	2:30	DEDIE SMC SEMI FRAIS	74	40'5h30	ALIMENTAIRE TEMP 15°
32	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	3	300	04/10/2016	7h00	2:30	DEDIE SMC SEMI FRAIS	74	40'5h30	ALIMENTAIRE TEMP 15°
33	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	0	10	04/10/2016	4h00	2:30	DEDIE SMC SEMI FRAIS	74	40'5h30	ALIMENTAIRE TEMP 15°
34														
35	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	1	106	04/10/2016	14h00	6:00	DEDIE SMC SEMI FRAIS	61	142'8h00	DPH ODOORANT
36	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	5	500	04/10/2016	14h00	6:00	DEDIE SMC SEMI FRAIS	61	142'8h00	DPH ODOORANT
37	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	1	230	04/10/2016	14h00	6:00	DEDIE SMC SEMI FRAIS	61	142'8h00	DPH ODOORANT
38	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		30	2	119	04/10/2016	13h00	6:00	DEDIE SMC SEMI FRAIS	61	142'8h00	DPH ODOORANT
39	VIVARAIS ST MARTIN	Num commande	DISTRIBUTEUR		13	24	2377	04/10/2016	8h00	6:00	DEDIE SMC SEMI FRAIS	61	142'8h00	DPH SEC

Figure 9.5 – Interface de l’outil opérationnel : vue commandes

	A	B	C	D	E	F	G	H	I
1	Numéro de tournée	Coût	Coût fixe du véhicule	Distance parcourue	Durée	Nombre de palettes	Poids	Coût à la	Type
2	1			52	1h00	3	300		
3									
4	1			635	5h30	27	10457		
5									
6	1			94	6h00	13	1262		
7									
8	1			148	2h00	13	1133		
9									
10	1			217	4h00	12	3861		
11									
12	1			133	5h30	25	13818		
13	2			40	5h30	14	10629		
14									
15	1			142	8h00	33	3362		

Figure 9.6 – Interface de l’outil opérationnel : vue véhicules

comme un *outil d’aide à la décision* et les tournées générées sont conservées ou modifiées selon les besoins et contraintes supplémentaires des transporteurs.

Au fur et à mesure de l’avancement du projet CRC et de l’ouverture de nouvelles plateformes, de nouvelles contraintes ont été définies et développées dans le framework de génération de colonnes. En parallèle avec l’arrivée de nouveaux transporteurs dans le projet, de nouvelles structures tarifaires ont été ajoutées aux modèles. Enfin, de nombreuses itérations avec les utilisateurs ont permis de rendre l’interface et les fichiers d’entrée et de sortie plus ergonomiques, assurant une utilisation facile de l’outil.

9.4 Conclusion et perspectives

Le développement d’outils en parallèle de la recherche et de l’évolution du projet fournisseur a été une composante forte de cette thèse. Le fait de pouvoir échanger fréquemment avec les utilisateurs des outils a permis de les faire évoluer à chaque visite chez 4S Network. Le produit final est aujourd’hui utilisé quotidiennement par trois personnes au sein de l’entreprise. Actuellement, CRC services gère le transport de 5000 palettes par mois, avec une progression très rapide qui devrait atteindre les 6000 palettes par mois d’ici la fin de l’année 2016. Les perspectives futures de ce projet seraient de développer une interface de type service web afin de rendre plus ergonomique et modulable l’utilisation de l’algorithme de tournées de véhicules riches. De plus, l’évolution du service CRC ouvre de nouvelles problématiques de recherche pour le transport de marchandises, plus particulièrement dans la synchronisation entre réseau amont et aval. L’objectif de 4S Network est de pouvoir, à terme, gérer le transport dans l’ensemble du réseau logistique (qui comprendrait donc le transport dans le réseau amont). L’algorithme développé

dans le chapitre 7 pourra notamment être modifié afin de répondre à ces nouvelles thématiques. Le travail plus théorique du chapitre 8 peut également contribuer au développement de nouveaux algorithmes intégrant une gestion plus efficace du temps.

Conclusion

Cette thèse était guidée par les problématiques de la société 4S Network, qui a mis en place un réseau logistique permettant la mutualisation du transport multi-fournisseurs et multi-clients adapté à la grande distribution. Notre travail de recherche avait pour but de résoudre conjointement un problème de *Service Network Design* (SNDP) dans un réseau longue distance et un problème riche de tournées de véhicules pour la distribution au niveau régional. Nous appelons ce nouveau problème *Service Network Design and Routing Problem* (SNDRP). Nos contributions scientifiques principales sont regroupées dans les chapitres 5, 6, 7 et 8.

Dans le chapitre 5, nous avons présenté une modélisation du problème tactique de transport dans le réseau logistique proposé par 4S Network. Ce modèle estime les coûts de transport dans le réseau logistique mutualisé, et les compare aux coûts de transport dans un réseau logistique classique. Plus particulièrement, il introduit une nouvelle modélisation des transferts qui permet la résolution de problèmes de taille plus importante qu'en utilisant une modélisation classique. Les résultats montrent que le réseau logistique permet d'économiser de 3% à 10% du coût du transport selon le réseau considéré. Ils montrent également que l'économie réalisée provient principalement des trajets en camions pleins dans le réseau amont qui regroupent plusieurs commandes à destination des clients d'une même région.

Le chapitre 6 concerne le problème de distribution depuis un CRC, modélisé comme un problème riche de tournées de véhicules. La méthode proposée est basée sur une génération de colonnes, dont le sous-problème est résolu par une recherche tabou. Cette dernière intègre un algorithme d'ordonnancement qui permet de minimiser les temps d'attente dans la tournée. Les expérimentations montrent que cette méthode permet d'obtenir rapidement de bons résultats. La méthode développée dans ce chapitre a de plus été intégrée dans un outil d'optimisation installé chez 4S Network et utilisé quotidiennement.

La principale contribution du chapitre 7 est d'aborder la synchronisation temporelle entre le réseau amont et aval. La dimension temporelle du transport est notamment modélisée par un réseau espace-temps pour le réseau amont. Nous proposons une méthode basée sur la génération de colonnes, où deux

types de colonnes sont générées : l'itinéraire d'une sous-commande dans le réseau amont et les tournées de véhicules dans le réseau aval. Ces deux types de colonnes sont synchronisées dans le problème maître de la génération de colonnes. Le sous-problème qui génère l'itinéraire des commandes est résolu de façon optimale, tandis que le sous-problème qui génère des tournées de véhicules est une recherche tabou. Les résultats montrent que la méthode fournit des résultats stables rapidement et que l'optimisation conjointe de ces deux problèmes est particulièrement utile lorsqu'on considère des coûts de stockage court-terme sur les CRC.

Enfin, le chapitre 8 expose les travaux réalisés dans le cadre d'un séjour à Loyola University à Chicago, en collaboration avec Mike Hewitt. L'objectif de ces travaux était de développer des formulations du SNDRP qui puissent être résolues par l'algorithme **Dynamic Discretization Discovery (DDD)** décrit dans l'article de Boland et al. [10]. Cette méthode permet de résoudre une formulation \mathcal{F}_{SND} de **SNDP** classique. L'enjeu était donc d'inclure le réseau de distribution locale dans la formulation \mathcal{F}_{SND} . Deux formulations sont proposées. La première formulation, basée sur l'énumération des tournées de distribution, permet d'incorporer facilement des contraintes riches liées aux tournées. Elle génère cependant un grand nombre de variables qui rend difficile la résolution d'un problème de grande taille. La seconde formulation permet de résoudre des problèmes de taille plus importante, mais n'incorpore pas de contraintes riches. Les expérimentations comparent les deux formulations et montrent que l'algorithme **DDD** permet de résoudre des problèmes du SNDRP jusqu'à 200 clients.

Cette thèse a donc contribué dans sa globalité à résoudre un **SNDP** sur un réseau longue distance synchronisé avec le transport dans les réseaux de distribution locale modélisés par des **VRPTW** tout en y intégrant des contraintes métiers, liées aux problématiques de 4S Network. Plus particulièrement, on présente deux approches globales pour résoudre ce type de problème.

- Une double génération de colonnes (chapitre 7) qui permet de générer séparément les variables relatives à chaque sous-problème et de les synchroniser dans le problème maître.
- Une adaptation à la formulation d'un **SNDP** (chapitre 8) qui permet de réutiliser les méthodes de résolution connues pour ce problème.

De plus, nous avons proposé différentes techniques pour incorporer les contraintes métiers issues des problématiques de 4S Network, notamment dans le chapitre 6. Nous proposons également un algorithme d'ordonnement qui minimise le temps d'attente d'une tournée, utilisé dans les chapitres 6 et 7.

En outre, le modèle développé dans le chapitre 5 et la méthode proposée dans le chapitre 6 ont donné lieu à deux logiciels développés dans le cadre de cette thèse et installés chez 4S Network. Le chapitre 9 décrit les principales fonctionnalités de ces logiciels. Notons que l'outil de résolution du problème riches de tournées de véhicules est utilisé quotidiennement par l'équipe de gestion opérationnelle du réseau de CRC.

Les perspectives issues de ces travaux sont nombreuses :

- Les algorithmes du chapitre 6 ont été développés postérieurement à ceux du chapitre 7, et contiennent des contraintes métier plus riches. Une suite logique de ces travaux serait donc d'étendre la méthode du chapitre 7 en y incorporant les contraintes métier du chapitre 6.

- La recherche tabou proposée dans les chapitres 6 et 7 permet une résolution rapide mais ne garantit pas l'optimalité. Il serait donc utile de développer une méthode de résolution exacte pour ce sous-problème, afin de mieux évaluer les performances de la recherche tabou ainsi que de trouver une borne inférieure du problème à la fin de la génération de colonnes.
- Dans les deux heuristiques de génération de colonnes, paralléliser la résolution des sous-problèmes permettrait de gagner un temps de résolution conséquent.
- Le chapitre 8 propose deux formulations adaptées à l'algorithme DDD. Pour chacune de ces formulations, différentes perspectives de recherche sont identifiées. Concernant la formulation sur les tournées de véhicules, générer dynamiquement les tournées dans l'algorithme DDD permettrait de pallier le problème de passage à l'échelle. Quant à la formulation sur les arcs, l'ajout de nouvelles contraintes dans le modèle permettrait l'affectation dynamique d'un client à une plateforme.

Plus globalement, l'approche de recherche de cette thèse était d'étendre le problème de SNDP au problème de SDRP. Nos modélisations étendent donc celle du SNDP, et nos méthodes de résolution se basent sur des méthodes déjà connues pour le SNDP. Il serait possible de développer la même logique à partir des modélisations et des méthodes de résolution pour des problèmes de tournées de véhicules, en les étendant à des formulations et méthodes dédiées au SDRP. Cela permettrait de réutiliser les travaux sur les tournées de véhicules multi-échelons et avec un ou plusieurs transferts. Enfin, l'algorithme DDD proposé par Boland et al. [10] semble prometteur pour résoudre des problèmes de transport dans des réseaux espace-temps. L'enjeu est désormais d'adapter cette méthode à de nouveaux problèmes d'optimisation, comme proposé dans le chapitre 8, mais également d'y intégrer les méthodes exactes connues de résolution de problème de transport, telles que des techniques de génération de colonnes.

Bibliographie

- [1] Sibel A Alumur, Bahar Y Kara, and M Teresa Melo. Location and logistics. In *Location Science*, pages 419–441. Springer, 2015. [38](#)
- [2] Jardar Andersen, Marielle Christiansen, Teodor Gabriel Crainic, and Roar Grønhaug. Branch and price for service network design with asset management constraints. *Transportation Science*, 45(1) :33–49, 2011. [40](#), [41](#), [42](#)
- [3] Jardar Andersen, Teodor Gabriel Crainic, and Marielle Christiansen. Service network design with asset management : Formulations and comparative analyses. *Transportation Research Part C : Emerging Technologies*, 17(2) :197 – 207, 2009. Selected papers from the Sixth Triennial Symposium on Transportation Analysis (TRISTAN VI). [40](#), [42](#)
- [4] Jardar Andersen, Teodor Gabriel Crainic, and Marielle Christiansen. Service network design with management and coordination of multiple fleets. *European Journal of Operational Research*, 193(2) :377 – 389, 2009. [40](#), [42](#)
- [5] Andrew P. Armacost, Cynthia Barnhart, and Keith A. Ware. Composite variable formulations for express shipment service network design. *Transportation Science*, 36(1) :1–20, 2002. [41](#), [42](#)
- [6] Nabila Azi, Michel Gendreau, and Jean-Yves Potvin. An adaptive large neighborhood search for a vehicle routing problem with multiple routes. *Computers and Operations Research*, 41 :167–173, 2014. [45](#)
- [7] Cynthia Barnhart, Ellis L Johnson, George L Nemhauser, Martin WP Savelsbergh, and Pamela H Vance. Branch-and-price : Column generation for solving huge integer programs. *Operations research*, 46(3) :316–329, 1998. [32](#)
- [8] Huseyin Basligil, Selin Soner Kara, Pelin Alcan, Betul Ozkan, and E. Gozde Caglar. A distribution network optimization problem for third party logistics service providers. *Expert Systems with Applications*, 38(10) :12730 – 12738, 2011. [45](#)
- [9] Richard Bellman. On a Routing Problem. *Quarterly of Applied Mathematics*, 16 :87–90, 1958. [35](#), [113](#)
- [10] Natasha Boland, Mike Hewitt, Luke Marshall, and Martin Savelsbergh. The continuous time service network design problem. Technical report, Technical report, Technical report, Optimization on-line, 2015. [41](#), [42](#), [127](#), [129](#), [133](#), [136](#), [142](#), [158](#), [159](#)

- [11] M.-C. Bolduc, J. Renaud, F. Boctor, and G. Laporte. A perturbation metaheuristic for the vehicle routing problem with private fleet and common carriers. *The Journal of the Operational Research Society*, 59(6) :pp. 776–787, 2008. [46](#)
- [12] Marie-Claude Bolduc, Jacques Renaud, and Fayez Boctor. A heuristic for the routing and carrier selection problem. *European Journal of Operational Research*, 183(2) :926 – 932, 2007. [46](#)
- [13] Berit Dangaard Brouer, David Pisinger, and Simon Spoorendonk. Liner shipping cargo allocation with repositioning of empty containers. *INFOR Journal*, 49(2) :109–124, 2011. [53](#)
- [14] Jose Caceres-Cruz, Pol Arias, Daniel Guimarans, Daniel Riera, and Angel A Juan. Rich vehicle routing problem : Survey. *ACM Computing Surveys (CSUR)*, 47(2) :32, 2015. [44](#), [77](#)
- [15] James F Campbell. Strategic network design for motor carriers. In *Logistics systems : Design and optimization*, pages 245–278. Springer, 2005. [38](#)
- [16] Diego Cattaruzza, Nabil Absi, Dominique Feillet, and Daniele Vigo. An iterated local search for the multi-commodity multi-trip vehicle routing problem with time windows. *Computers and Operations Research*, 51 :257 – 267, 2014. [45](#)
- [17] Alberto Ceselli, Giovanni Righini, and Matteo Salani. A column generation algorithm for a rich vehicle-routing problem. *Transportation Science*, 43(1) :56–69, 2009. [45](#), [78](#)
- [18] Lap Mui Ann Chan, Ana Muriel, Zuo-Jun Max Shen, David Simchi-Levi, and Chung-Piaw Teo. Effective zero-inventory-ordering policies for the single-warehouse multiretailer problem with piecewise linear cost structures. *Management Science*, 48(11) :1446–1460, 2002. [39](#)
- [19] Tsung-Sheng Chang. Best routes selection in international intermodal networks. *Computers and Operations Research*, 35(9) :2877 – 2891, 2008. Part Special Issue : Bio-inspired Methods in Combinatorial Optimization. [39](#), [41](#), [42](#)
- [20] Ping Chen, Yunsong Guo, Andrew Lim, and Brian Rodrigues. Multiple crossdocks with inventory and time windows. *Computers and Operations Research*, 12 :43–63, 2006. [78](#)
- [21] Eunjeong Choi and Dong-Wan Tcha. A column generation approach to the heterogeneous fleet vehicle routing problem. *Computers and Operations Research*, 34(7) :2080 – 2095, 2007. [78](#)
- [22] Sunil Chopra. Designing the distribution network in a supply chain. *Transportation Research Part E : Logistics and Transportation Review*, 39(2) :123 – 140, 2003. [25](#), [39](#)
- [23] Ching-Wu Chu. A heuristic algorithm for the truckload and less-than-truckload problem. *European Journal of Operational Research*, 165(3) :657 – 667, 2005. [46](#)
- [24] M. H. Cole and M Parthasarathy. Optimal design of merge-in-transit distribution networks. Technical Report MBTC FR-1064, Mack-Blackwell National Rural Transportation Study Center, University of Arkansas, Fayetteville, AR, 1998. [39](#)
- [25] Brian Coltin and Manuela Veloso. Scheduling for transfers in pickup and delivery problems with very large neighborhood search. In *Proceedings of AAAI’14, the Twenty-Eighth Conference on Artificial Intelligence*, 2014. [44](#)

- [26] Jean-François Cordeau and Gilbert Laporte. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B : Methodological*, 37(6) :579–594, 2003. 86
- [27] C. E. Cortés, M. Matamala, and C. Contardo. The pickup and delivery problem with transfers : Formulation and a branch-and-cut solution method. *European Journal of Operational Research*, 200(3) :711–724, 2010. 43, 44
- [28] Jean-François Côté and Jean-Yves Potvin. A tabu search heuristic for the vehicle routing problem with private fleet and common carrier. *European Journal of Operational Research*, 198(2) :464 – 469, 2009. 46
- [29] Teodor Gabriel Crainic. Service network design in freight transportation. *European Journal of Operational Research*, 122(2) :272–288, 2000. 40
- [30] Teodor Gabriel Crainic, Mike Hewitt, Michel Toulouse, and Duc Vu. Service network design with resource constraints. *Transportation Science*, 0(0) :null, 2014. 41, 42
- [31] Teodor Gabriel Crainic and Gilbert Laporte. Planning models for freight transportation. *European journal of operational research*, 97(3) :409–438, 1997. 37, 38
- [32] Keely L. Croxton, Bernard Gendron, and Thomas L. Magnanti. A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems. *Management Science*, 49(9) :1268–1273, 2003. 31, 39, 107
- [33] Keely L. Croxton, Bernard Gendron, and Thomas L. Magnanti. Models and methods for merge-in-transit operations. *Transportation Science*, 37(1) :1–22, 2003. 39
- [34] Keely L Croxton, Bernard Gendron, and Thomas L Magnanti. Variable disaggregation in network flow problems with piecewise linear costs. *Operations research*, 55(1) :146–157, 2007. 108
- [35] Guy Desaulniers, Jacques Desrosiers, and Marius M Solomon. *Column generation*, volume 5. Springer Science and Business Media, 2006. 32
- [36] Guy Desaulniers, François Lessard, and Ahmed Hadjar. Tabu search, partial elementarity, and generalized k-path inequalities for the vehicle routing problem with time windows. *Transportation Science*, 42(3) :387–404, 2008. 35, 82
- [37] Rodolfo Dondo, Carlos A. Méndez, and Jaime Cerdá. The multi-echelon vehicle routing problem with cross docking in supply chain management. *Computers and Chemical Engineering*, 35(12) :3002 – 3024, 2011. 43
- [38] Alan Erera, Michael Hewitt, Martin Savelsbergh, and Yang Zhang. Improved load plan design through integer programming based local search. *Transportation Science*, 47(3) :412–427, 2013. 40, 41, 42
- [39] Alan L. Erera, Michael Hewitt, Martin W.P. Savelsbergh, and Yang Zhang. Creating schedules and computing operating costs for {LTL} load plans. *Computers and Operations Research*, 40(3) :691 – 702, 2013. Transport Scheduling. 40

- [40] Dominique Feillet. A tutorial on column generation and branch-and-price for vehicle routing problems. *4or*, 8(4) :407–424, 2010. [32](#), [35](#)
- [41] Dominique Feillet, Pierre Dejax, Michel Gendreau, and Cyrille Gueguen. An exact algorithm for the elementary shortest path problem with resource constraints : application to some vehicle routing problems. *Networks*, 44(3) :216–229, 2004. [35](#), [82](#)
- [42] Lester R. Ford. Network Flow Theory. Report P-923, The Rand Corporation, 1956. [35](#), [113](#)
- [43] Bernard Gendron and Frédéric Semet. Formulations and relaxations for a multi-echelon capacitated location–distribution problem. *Computers and Operations Research*, 36(5) :1335 – 1355, 2009. [38](#)
- [44] Fred Glover and Manuel Laguna. *Tabu Search**. Springer, 2013. [35](#)
- [45] Asvin Goel. *A Column Generation Heuristic for the General Vehicle Routing Problem*, pages 1–9. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. [78](#)
- [46] J Gonzalez-Feliu. The multi-echelon location-routing problem : Concepts and methods for tactical and operational planning. Technical report, Technical Report, Laboratoire d’Economie des Transports, Lyon, France, 2010. [38](#)
- [47] Philippe Grangier, Michel Gendreau, Fabien Lehuédé, and Louis-Martin Rousseau. An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization. *European Journal of Operational Research*, 254(1) :80–91, 2016. [45](#)
- [48] Philippe Grangier, Michel Gendreau, Fabien Lehuédé, and Louis-Martin Rousseau. A matheuristic based on large neighborhood search for the vehicle routing problem with cross-docking. *Les cahiers du CIRRELT*, CIRRELT-2016-09, 2016. [43](#)
- [49] Hans-Otto Gunther and Thorben Seiler. Operative transportation planning in consumer goods supply chains. *Flexible Services and Manufacturing Journal*, 21(1-2) :51–74, 2009. [45](#)
- [50] P. Hansen. The Steepest Ascent Mildest Descent Heuristic for Combinatorial Programming. In *Proceedings of the Congress on Numerical Methods in Combinatorial Optimization*, Capri, Italy, 1986. [35](#)
- [51] James Hill and Michael Galbreth. A heuristic for single-warehouse multiretailer supply chains with all-unit transportation cost discounts. *European Journal of Operational Research*, 187(2) :473–482, 2008. [39](#)
- [52] Peter Hosie, Balan Sundarakani, Albert Wee Kwan Tan, and Aleksandra Koźlak. Determinants of fifth party logistics (5pl) : service providers for supply chain management. *International Journal of Logistics Systems and Management*, 13(3) :287–316, 2012. [16](#)
- [53] Yetkin Ileri, Mokhtar Bazaraa, Ted Gifford, George Nemhauser, Joel Sokol, and Erick Wikum. An optimization approach for planning daily drayage operations. *Central European Journal of Operations Research*, 14(2) :141–156, 2006. [78](#)

- [54] Ahmad I. Jarrah, Ellis Johnson, and Lucas C. Neubert. Large-scale, less-than-truckload service network design. *Operations Research*, 57(3) :609–625, 2009. [40](#), [41](#), [42](#)
- [55] Christian Vad Karsten, David Pisinger, Stefan Ropke, and Berit Dangaard Brouer. The time constrained multi-commodity network flow problem and its application to liner shipping network design. *Transportation Research Part E : Logistics and Transportation Review*, 76 :122–138, 2015. [53](#)
- [56] H. L. M. Kerivin, M. Lacroix, A. R. Mahjoub, and A. Quilliot. The splittable pickup and delivery problem with reloads. *European Journal of Industrial Engineering*, 2(2) :112–133, 2008. [43](#), [44](#)
- [57] Daeki Kim, Cynthia Barnhart, Keith Ware, and Gregory Reinhardt. Multimodal express package delivery : A service network design application. *Transportation Science*, 33(4) :391–407, 1999. [41](#), [42](#)
- [58] Çağrı Koç, Tolga Bektaş, Ola Jabali, and Gilbert Laporte. Thirty years of heterogeneous vehicle routing. *European Journal of Operational Research*, 249(1) :1–21, 2016. [44](#)
- [59] Herbert Kopfer and Xin Wang. Combining vehicle routing with forwarding, extension of the vehicle routing problem by different types of sub-contraction. In *Journal of the Korean Institute of Industrial Engineers (JKIIE)*, pages 1–14, 2009. [45](#)
- [60] Marta Anna Krajewska and Herbert Kopfer. Transportation planning in freight forwarding companies : Tabu search algorithm for the integrated operational transportation planning problem. *European Journal of Operational Research*, 197(2) :741–751, 2009. [45](#)
- [61] Rahma Lahyani, Mahdi Khemakhem, and Frédéric Semet. Rich vehicle routing problems : From a taxonomy to a definition. *European Journal of Operational Research*, 241(1) :1–14, 2015. [44](#), [77](#)
- [62] Sophie D. Lapierre, Angel B. Ruiz, and Patrick Soriano. Designing distribution networks : Formulations and solution heuristic. *Transportation Science*, 38(2) :174–187, 2004. [39](#)
- [63] Young Hae Lee, Jung Woo Jung, and Kyong Min Lee. Vehicle routing scheduling for cross-docking in the supply chain. *Computers and Industrial Engineering*, 51(2) :247 – 256, 2006. [43](#)
- [64] Fabien Lehuédé, Renaud Masson, Sophie N Parragh, Olivier Péton, and Fabien Tricoire. A multi-criteria large neighbourhood search for the transportation of disabled people. *Journal of the Operational Research Society*, 65(7) :983–1000, 2014. [86](#)
- [65] Ching-Jong Liao, Yaoming Lin, and Stephen C. Shih. Vehicle routing with cross-docking in the supply chain. *Expert Systems with Applications*, 37(10) :6868 – 6873, 2010. [43](#)
- [66] Cheng-Chang Lin. The freight routing problem of time-definite freight delivery common carriers. *Transportation Research Part B : Methodological*, 35(6) :525–547, 2001. [41](#), [42](#)

- [67] Cheng-Chang Lin and Sheu-Hua Chen. An integral constrained generalized hub-and-spoke network design problem. *Transportation Research Part E : Logistics and Transportation Review*, 44(6) :986–1003, 2008. [41](#), [42](#), [53](#)
- [68] Kathleen A. Lindsey, Alan L. Erera, and Martin W.P. Savelsbergh. A pickup and delivery problem using crossdocks and truckload lane rates. *EURO Journal on Transportation and Logistics*, 2(1-2) :5–27, 2013. [45](#)
- [69] Marco E Lübbecke and Jacques Desrosiers. Selected topics in column generation. *Operations Research*, 53(6) :1007–1023, 2005. [32](#)
- [70] Renaud Masson, Fabien Lehuédé, and Olivier Péton. An adaptive large neighborhood search for the pickup and delivery problem with transfers. *Transportation Science*, 47 :356–379, 2013. [43](#), [44](#)
- [71] Renaud Masson, Fabien Lehuédé, and Olivier Péton. The dial-a-ride problem with transfers. *Computers and Operations Research*, 41 :12–23, 2014. [43](#), [44](#)
- [72] Renaud Masson, Stefan Ropke, Fabien Lehuédé, and Olivier Péton. A branch-and-cut-and-price approach for the pickup and delivery problem with shuttle routes. *European Journal of Operational Research*, 236(3) :849–862, 2014. [43](#), [44](#)
- [73] M Teresa Melo, Stefan Nickel, and Francisco Saldanha-da Gama. Facility location and supply chain management—a review. *European journal of operational research*, 196(2) :401–412, 2009. [38](#)
- [74] S. Mitrović-Minić and G. Laporte. The pickup and delivery problem with time windows and transshipment. *INFOR*, 44(3) :217–228, 2006. [43](#), [44](#)
- [75] Luigi Moccia, Jean-François Cordeau, Gilbert Laporte, Stefan Ropke, and Maria Pia Valentini. Modeling and solving a multimodal transportation problem with flexible-time and scheduled services. *Networks*, 57 :53–68, 2011. [39](#), [41](#), [42](#), [53](#), [107](#)
- [76] Vinicius W.C. Morais, Geraldo R. Mateus, and Thiago F. Noronha. Iterated local search heuristics for the vehicle routing problem with cross-docking. *Expert Systems with Applications*, 41(16) :7495 – 7506, 2014. [43](#)
- [77] Ana Muriel and Farhad N Munshi. Capacitated multicommodity network flow problems with piecewise linear concave costs. *IIE Transactions*, 36(7) :683–696, 2004. [39](#)
- [78] Phuong Khanh Nguyen, Teodor Gabriel Crainic, and Michel Toulouse. A tabu search for time-dependent multi-zone multi-trip vehicle routing problem with time windows. *European Journal of Operational Research*, 231(1) :43–56, 2013. [44](#)
- [79] Sophie N Parragh, Jean-François Cordeau, Karl F Doerner, and Richard F Hartl. Models and algorithms for the heterogeneous dial-a-ride problem with driver-related constraints. *OR spectrum*, 34(3) :593–633, 2012. [45](#), [78](#)

- [80] Michael Berliner Pedersen, Teodor Gabriel Crainic, and Oli B. G. Madsen. Models and tabu search metaheuristics for service network design with asset-balance requirements. *Transportation Science*, 43(2) :158–177, 2009. [40](#), [42](#)
- [81] A. Rais, F. Alvelos, and M.S. Carvalho. New mixed integer-programming model for the pickup-and-delivery problem with transshipment. *European Journal of Operational Research*, 235(3) :530–539, 2014. [43](#), [44](#), [53](#)
- [82] Fernando Afonso Santos, Geraldo Robson Mateus, and Alexandre Salles da Cunha. A novel column generation algorithm for the vehicle routing problem with cross-docking. In Julia Pahl, Torsten Reiners, and Stefan Voss, editors, *Network Optimization*, volume 6701 of *Lecture Notes in Computer Science*, pages 412–425. Springer Berlin Heidelberg, 2011. [43](#)
- [83] Fernando Afonso Santos, Geraldo Robson Mateus, and Alexandre Salles da Cunha. A branch-and-price algorithm for a vehicle routing problem with cross-docking. *Electronic Notes in Discrete Mathematics*, 37(0) :249 – 254, 2011. [43](#)
- [84] Martin W. P. Savelsbergh. The vehicle routing problem with time windows : Minimizing route duration. *ORSA Journal on Computing*, 4(2) :146–154, 1992. [86](#)
- [85] Michel Povlovitch Seixas and André Bergsten Mendes. Column generation for a multitrip vehicle routing problem with time windows, driver work hours, and heterogeneous fleet. *Mathematical Problems in Engineering*, 2013, 2013. [78](#), [79](#)
- [86] J. S. Shang and C. K. Cuff. Multicriteria pickup and delivery problem with transfer opportunity. *Computers and Industrial Engineering*, 30(4) :631–645, 1996. [43](#), [44](#)
- [87] Haiqing Song, Vernon N. Hsu, and Raymond K. Cheung. Distribution coordination between suppliers and customers with a consolidation center. *Operations Research*, 56(5) :1264–1277, 2008. [39](#)
- [88] Simon Spoorendonk. *Cut and column generation*. PhD thesis, PhD thesis, Technical University of Denmark, 2008. [33](#)
- [89] D.M. Stein. Scheduling dial-a-ride transportation systems. *Transportation Science*, 12 :232–249, 1978. [43](#), [44](#)
- [90] Éric D Taillard, Gilbert Laporte, and Michel Gendreau. Vehicle routing with multiple use of vehicles. *Journal of the Operational research society*, 47(8) :1065–1070, 1996. [44](#)
- [91] Christos D. Tarantilis. Adaptive multi-restart tabu search algorithm for the vehicle routing problem with cross-docking. *Optimization Letters*, 7(7) :1583–1596, 2013. [43](#)
- [92] Nicolas Teypaz, Susann Schrenk, and Van-Dat Cung. A decomposition scheme for large-scale service network design with asset management. *Transportation Research Part E Logistics and Transportation Review*, 46(1) :156–170, 2010. [40](#), [41](#), [42](#)
- [93] Paolo Toth and Daniele Vigo. *Vehicle routing : problems, methods, and applications*, volume 18. SIAM, 2014. [37](#)

- [94] Thibaut Vidal, Teodor Gabriel Crainic, Michel Gendreau, and Christian Prins. Heuristics for multi-attribute vehicle routing problems : A survey and synthesis. *European Journal of Operational Research*, 231(1) :1–21, 2013. [44](#)
- [95] Benedikt Vornhusen, Xin Wang, and Herbert Kopfer. Vehicle routing under consideration of transshipment in horizontal coalitions of freight carriers. *Procedia CIRP*, 19 :117–122, 2014. 2nd CIRP Robust Manufacturing Conference (RoMac 2014). [46](#)
- [96] Xin Wang and Herbert Kopfer. Collaborative transportation planning of less-than-truckload freight. *OR Spectrum*, 36(2) :357–380, 2014. [46](#)
- [97] M Wen, J Larsen, J Clausen, J-F Cordeau, and G Laporte. Vehicle routing with cross-docking. *Journal of the Operational Research Society*, 60(12) :1708–1718, 2008. [43](#)
- [98] Nicole Wieberneit. Service network design for freight transportation : a review. *OR Spectrum*, 30(1) :77–112, 2008. [40](#)
- [99] Hang Xu, Zhi-Long Chen, Srinivas Rajagopal, and Sundar Arunapuram. Solving a practical pickup and delivery problem. *Transportation science*, 37(3) :347–364, 2003. [78](#)

Acronymes

3PL Third-party logistics. [15](#)

ALNS Recherche adaptative à voisinage large. [43](#), [44](#)

CTSNDP Continuous Time Service Network Design Problem. [134](#), [136](#)

DDD Dynamic Discretization Discovery. [127](#), [133](#), [134](#), [136](#), [138](#), [142](#), [158](#), [159](#), [174](#)

FTL Full-truckload. [23](#), [27](#), [28](#), [37](#), [40](#), [45](#), [52](#), [100](#)

LNS Recherche à voisinage large. [78](#)

LTL Less-than-truckload. [23](#), [27](#), [28](#), [29](#), [37](#), [38](#), [41](#), [42](#), [45](#), [50](#), [60](#), [100](#), [173](#)

MCM Multiple Choice Model. [31](#), [59](#), [108](#)

PDPT Pickup and Delivery Problem with Transfer. [42](#), [43](#), [46](#)

PLNE Programme Linéaire en Nombres Entiers. [29](#), [35](#), [45](#), [48](#), [72](#), [79](#), [106](#), [116](#), [119](#)

SNDP Service Network Design Problem. [4](#), [6](#), [11](#), [12](#), [39](#), [40](#), [41](#), [42](#), [100](#), [107](#), [121](#), [122](#), [125](#), [126](#), [127](#), [128](#), [129](#), [130](#), [131](#), [132](#), [133](#), [134](#), [157](#), [158](#), [159](#)

SNDRP Service Network Design and Routing Problem. [126](#), [127](#), [128](#), [130](#), [131](#), [138](#)

VRP Vehicle Routing Problem. [11](#)

VRPCD Tournées de véhicule avec cross-dock. [42](#), [43](#)

VRPTW Vehicle Routing Problem with Time Windows. [79](#), [100](#), [121](#), [122](#), [125](#), [126](#), [128](#), [158](#)

Liste des tableaux

4.1	Caractéristiques générales du réseau et des transferts	42
4.2	Problèmes de collectes et livraison avec transferts	44
5.1	Nombre de clients qui peuvent être livrés par 1 à 4 CRC	67
5.2	Caractéristiques des instances	68
5.3	Comparaison des structures de transferts	69
5.4	Economies selon la couverture et le type de réseau	69
5.5	Économies réalisées par rapport à un fonctionnement sans réseau mutualisé selon la taille moyenne des commandes	70
6.1	Compatibilités de transport entre types de produits	73
6.2	Types de sous-traitance	75
6.3	Notations	80
6.4	Résumé des notations pour l’algorithme de réalisabilité	87
6.5	Résultat selon le nombre d’itérations de la recherche tabou	92
6.6	Résultats selon le paramètre $Max - RT$	93
6.7	Écarts de coûts selon la longueur de la liste tabou	94
6.8	Résultats selon le nombre maximal de solutions trouvées dans une recherche tabou	95
6.9	Coût du transport en fonction du nombre de commandes et du pourcentage d’horaires de livraison fixés	95
6.10	Nombre de véhicules en fonction du nombre de commandes et du pourcentage d’horaires de livraison fixés	96
6.11	Nombre de colonnes générées en fonction du nombre de commandes et du pourcentage d’horaires de livraison fixés	96
7.1	Nombre de recherches tabou par itération	117
7.2	Nombre d’itérations d’une recherche tabou	118
7.3	Longueur de la liste tabou	118
7.4	Statistiques générales	119
7.5	Évolution du nombre de colonnes y_p^l générées en fonction du pas de discrétisation	120
7.6	Évolution du coût du transport dans le réseau amont en fonction du pas de discrétisation	120
7.7	Évolution du coût global en fonction du coût de stockage	121
7.8	Évolution du coût total des stocks en fonction du coût de stockage	122
7.9	Évolution du coût de transport en fonction du coût de stockage	122

8.1	Pourcentage d'instances résolues optimalement et solutions réalisables en fonction du nombre de commandes et du nombre de hubs	139
8.2	Temps de résolution en fonction du nombre de commandes et du nombre de hubs pour les instances résolues de façon optimale	140
8.3	Écart à la borne inférieure en fonction du nombre de commandes et du nombre de hubs	141
8.4	Moyenne du coût moyen de transport par commande en fonction du nombre de clients et du nombre de hubs	142
8.5	Moyenne de l'écart à la borne inférieur en fonction du nombre de clients et du nombre de hubs	142

Table des figures

2.1	Structure globale du réseau	19
2.2	Processus de regroupement de produits d'une palette hétérogène	21
2.3	Itinéraires des sous-commandes	22
2.4	Un exemple de réseau aval	24
2.5	Répartition des clients en zones tarifaires	24
2.6	Coût fixe de la tournée selon les zones	25
2.7	Tournées des véhicules	26
2.8	Itinéraire de la commande	26
3.1	Exemple de tarif LTL	29
3.2	Tarif LTL pour une flotte homogène	30
3.3	Exemple d'un réseau plat	32
3.4	Réseau espace-temps	33
3.5	Principe de la génération de colonnes	33
3.6	Schéma général de la génération de colonnes	34
5.1	Un exemple de réseau amont	49
5.2	Un exemple de réseau aval	49
5.3	Un exemple de réseau direct	50
5.4	Tournées dans un réseau amont	51
5.5	Tournées dans un réseau direct	51
5.6	Tournées dans le réseau aval	52
5.7	Structure de transfert complète	54
5.8	Structure de transfert en étoile	54
5.9	Utilisation de la structure complète de transfert	56
5.10	Utilisation de la structure de transfert en étoile	57
5.11	Réseau de type A avec 10 sites de fournisseurs, un CRC et 50 clients	64
5.12	Réseau de type B avec 10 sites de fournisseurs, 3 CRC et 130 clients	64
5.13	Réseau de type C avec 10 sites de fournisseurs, 4 CRC et 130 clients	65
6.1	Graphe utilisé dans la résolution du problème	74
6.2	Un exemple de solution au problème riche de tournées de véhicules	77
6.3	Étapes de l'algorithme	80
7.1	Tournées dans le réseau amont plat	101

7.2	Tournées amont dans le réseau espace-temps	102
7.3	Modélisation du réseau aval	103
7.4	Un exemple d'itinéraire dans le réseau amont	105
7.5	Prix d'une tournée avec quatre véhicules de capacité Q	109
7.6	Schéma général de l'algorithme de génération de colonnes	111
7.7	Synchronisation entre réseau amont et réseau aval	115
8.1	Exemple de réseau logistique considéré	126
8.2	Exemple de création d'un réseau	131
8.3	Tournées énumérées et modélisation sur le réseau plat	132
8.4	Tournée de véhicule du hub bb_1 dans le réseau espace temps	132
8.5	Utilisation de l'algorithme DDD	134
8.6	Diagramme de flux	135
8.7	Un réseau espace-temps partiel	135
8.8	Un réseau plat	135
8.9	Un exemple de <i>early arrival path</i>	136
8.10	Exemple de l'affinement d'un réseau espace temps-partiel	137
9.1	Interactions entre les modules des outils développés	146
9.2	Interface du simulateur	149
9.3	Format de la feuille Excel définissant les paramètres pour un transporteur	152
9.4	Interface de l'outil opérationnel	153
9.5	Interface de l'outil opérationnel : vue commandes	154
9.6	Interface de l'outil opérationnel : vue véhicules	155

Thèse de Doctorat

Juliette MEDINA

Modèles et méthodes d'optimisation pour la mutualisation des chaînes logistiques

Optimization models and methods for collaborative supply chains

Résumé

Cette thèse a pour but d'apporter des solutions méthodologiques pour la mutualisation des transports entre les fournisseurs et les plateformes de la grande distribution. Cette mutualisation permet en effet de réduire les coûts, les émissions de CO₂, et d'augmenter la qualité de service. Elle est organisée autour d'un réseau de plateformes de cross-docking appelées Centres de Routage Collaboratifs, développé par la société 4S Network. Nos travaux consistent à modéliser et résoudre à l'aide de techniques de recherche opérationnelle plusieurs problèmes d'optimisation du transport dans le réseau mutualisé. Le verrou scientifique majeur est de résoudre conjointement un problème de plan de chargement (Service Network Design Problem) dans un réseau logistique national, et des problèmes de tournées de véhicules à une échelle régionale. Nous prenons en compte des contraintes additionnelles issues du monde industriel et les tarifs réellement pratiqués par les transporteurs, notamment des coûts non linéaires. Les problèmes d'optimisation résultants sont résolus au moyen de méthodes dites matheuristiques, c'est-à-dire combinant des approches exactes telles que la génération de colonnes et des approches (méta)heuristiques telles que la recherche tabou. Les algorithmes développés dans cette thèse ont donné lieu à un outil logiciel aujourd'hui en exploitation chez 4S Network.

Mots clés

Recherche Opérationnelle, conception de plans de chargement, problèmes de tournées de véhicules, génération de colonnes, matheuristique

Abstract

The main purpose of this PhD. thesis is to provide methodological solutions for a collaborative transport between suppliers and retail platforms. The outcomes of this collaboration are numerous: cost reduction, greenhouse gas emission reduction and higher quality of service. The network is structured around cross-docking platforms developed by the company 4S Network. We model and solve several optimization problems in this collaborative network, using operations research techniques. The major scientific challenge is to simultaneously solve a Service Network Design Problem in a national logistics network and several Vehicle Routing Problems at regional level. We consider additional constraints and prevailing pricing arising from the carriers, in particular non-linear costs. The resulting optimization problems are solved by matheuristic methods, that combine exact approaches as column generation and (meta)heuristic approaches as tabu search. The algorithms developed in this thesis are the core functions of a software tool developed for 4S network.

Key Words

operations Research, Service Network Design, Vehicle Routing Problem, collaborative logistics, column generation – matheuristic