



# Méthodes de modélisation et d'optimisation par recherche à voisinages variables pour le problème de collecte et de livraison avec transbordement

Rodrigue Tchapnga Takoudjou

## ► To cite this version:

Rodrigue Tchapnga Takoudjou. Méthodes de modélisation et d'optimisation par recherche à voisinages variables pour le problème de collecte et de livraison avec transbordement. Gestion et management. Université de Bordeaux, 2014. Français. NNT : 2014BORD0052 . tel-01151665

**HAL Id: tel-01151665**

<https://theses.hal.science/tel-01151665>

Submitted on 13 May 2015

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

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

THÈSE PRÉSENTÉE  
POUR OBTENIR LE GRADE DE

**DOCTEUR DE  
L'UNIVERSITÉ DE BORDEAUX**

ÉCOLE DOCTORALE DES SCIENCES PHYSIQUES ET DE  
L'INGÉNIEUR

SPÉCIALITÉ : PRODUCTIQUE

Par Tchapnga Takoudjou Rodrigue

**MÉTHODES DE MODÉLISATION ET  
D'OPTIMISATION PAR RECHERCHE À  
VOISINAGES VARIABLES POUR LE  
PROBLÈME DE COLLECTE ET DE  
LIVRAISON AVEC TRANSBORDEMENT**

Soutenue le 12 Juin 2014 devant le jury composé de :

M. RÉMY DUPAS	Professeur, Université de Bordeaux	Directeur de thèse
M. JEAN-CHRISTOPHE DESCHAMPS	Maître de conférences, Université de Bordeaux	Co-directeur de thèse
M. BERNARD GRABOT	Professeur, ENI de Tarbes	Rapporteur
M. PIERRE LOPEZ	Directeur de Recherche, LAAS CNRS Toulouse	Rapporteur
M. FRANÇOIS CLAUTIAUX	Professeur, Université de Bordeaux	Examinateur
M. TIENTE HSU	Maître de conférences, Université d'Artois	Examinateur



*À mon père,  
Pour tes immenses sacrifices, ton courage, ta patience, ta générosité.  
Grâce à tes efforts, ta bienveillance, ton Amour, ton soutien inlassable, j'ai  
pu terminer mes études. Je t'exprime mon immense gratitude et je te  
manifeste ma reconnaissance sincère et éternelle.*

*À ma mère,  
Pour la tendresse, l'amour et l'affection dont tu m'as toujours entouré,  
pour ta présence, ton dévouement, ton encouragement sans limite que tu  
n'as jamais cessé de manifester. Je te remercie, te dis mon profond respect,  
ainsi que ma profonde reconnaissance.*



# REMERCIEMENTS

**J**e remercie tous les membres du jury qui ont accepté d'évaluer et juger ce travail de recherche. Je remercie sincèrement Pierre Lopez, directeur de recherche au LAAS de Toulouse et Bernard Grabot Professeur à l'ENIT de Tarbes pour leur travail de rapporteur. Je leur en suis très reconnaissant.

Je souhaite remercier Rémy DUPAS mon directeur de thèse et Jean-Christophe DESCHAMPS mon co-directeur de thèse, pour avoir su me conseiller tout au long de celle-ci, pour leur qualité de relecture, mais également pour m'avoir donné des clés qui ont permis ce travail d'écriture, afin de présenter aujourd'hui le fruit de ces échanges.

Je remercie également toutes les personnes qui ont contribué de près ou de loin à la réussite et à l'aboutissement de cette thèse : Le laboratoire IMS de BORDEAUX et mes collègues du Projet PRODIGE pour nos échanges et nos nombreuses réunions.

Je tiens à remercier les personnes qui tout au long de mes études ont cru en moi, en mon travail. Même si il n'est pas le seul, je remercie particulièrement le professeur Jean-Jacques LOISEAU.

Je ne peux terminer les remerciements sans remercier ma famille. Je remercie tous mes frères et sœurs pour leur soutien. Je remercie Mélanie LOCOCHE pour m'avoir soutenu alors que je perdais espoir et pour ses encouragements. Je remercie mes parents sans qui rien n'aurait été possible, pour tout ce qu'ils ont fait pour moi et pour tous les sacrifices qu'ils ont consentis pour que je puisse arriver à ce niveau. Je les remercie d'avoir fait de mon bonheur et celui de mes frères le but de leur vie.

Ce travail de Recherche a été financé par l'ANR, via le projet PRODIGE avec la référence VTTPRODIGE (ANR-09-VTT0901) et labélisé par NOVALOG.

# TABLE DES MATIÈRES

TABLE DES MATIÈRES	vi
LISTE DES ALGORITHMES	xi
LISTE DES FIGURES	xi
LISTE DES TABLEAUX	xiii
INTRODUCTION GÉNÉRALE	1
1 LE TRANSPORT ROUTIER DE MARCHANDISES AU SEIN DE LA CHAÎNE LOGISTIQUE	5
1.1 INTRODUCTION . . . . .	7
1.2 LA CHAÎNE LOGISTIQUE ET LE TRANSPORT . . . . .	7
1.2.1 La chaîne logistique . . . . .	8
1.2.2 La place du transport routier . . . . .	10
1.2.3 Les offres de transport . . . . .	11
1.2.3.1 Affrètement . . . . .	11
1.2.3.2 Le groupage . . . . .	12
1.2.3.3 La messagerie . . . . .	12
1.2.3.4 Le transport express . . . . .	12
1.3 LA GESTION DE LA CHAÎNE LOGISTIQUE . . . . .	13
1.3.1 Les principales fonctions de la chaîne logistique . . . . .	14
1.3.2 La fonction distribution/transport . . . . .	15
1.3.2.1 Le niveau de décision opérationnel . . . . .	16
1.4 LES PRATIQUES ACTUELLES ET LES NOUVELLES TENDANCES DANS LE TRANSPORT . . . . .	16
1.4.1 L'externalisation des chaînes d'approvisionnements . . . . .	16
1.4.2 La pratique de la mutualisation . . . . .	16
1.4.3 Le cross-docking . . . . .	17
1.4.4 Le transbordement . . . . .	17
1.4.5 Le TMS . . . . .	18
1.5 LA PERFORMANCE DANS LE SECTEUR TRANSPORT . . . . .	19
1.5.1 Les contraintes économiques et sociales . . . . .	20
1.5.1.1 Le problème énergétique . . . . .	20
1.5.1.2 Le développement durable . . . . .	20
1.5.2 Leviers et indicateurs de performance . . . . .	21
1.5.3 Vers un référentiel de performances pour le transport routier	22
1.5.3.1 Indicateurs élémentaires . . . . .	23
1.5.3.2 Indicateurs composites de productivité . . . . .	24
1.5.3.3 Indicateurs composites de charge . . . . .	25

1.5.3.4	Indicateurs éco-logistique . . . . .	25
1.5.3.5	Indicateurs de qualité de service . . . . .	26
1.6	CONCLUSION . . . . .	27
<b>2</b>	<b>CLASSIFICATION DES PROBLÈMES DE COLLECTE ET DE LIVRAISON ET ÉTAT DE L'ART</b>	<b>29</b>
2.1	INTRODUCTION . . . . .	31
2.2	TYPOLOGIE DES PROBLÈMES DE ROUTAGE . . . . .	31
2.2.1	Caractéristiques générales d'un problème de tournées de véhicules . . . . .	32
2.2.1.1	Produits . . . . .	32
2.2.1.2	Véhicules . . . . .	32
2.2.1.3	Clients . . . . .	33
2.2.2	Caractéristiques propres au problème de collecte et livraison	34
2.2.2.1	Problème « one-to-one » . . . . .	34
2.2.2.2	Problème « many-to-many » . . . . .	35
2.2.2.3	Problème « one-to-many-to-one » . . . . .	36
2.2.3	Grille d'analyse et positionnement du problème étudié . . . . .	37
2.3	MÉTHODES DE RÉSOLUTION . . . . .	39
2.3.1	Méthodes exactes . . . . .	40
2.3.2	Méthodes approchées . . . . .	41
2.3.2.1	Algorithme glouton . . . . .	41
2.3.2.2	Recherches locales . . . . .	42
2.3.2.3	Recherche à voisinage variable . . . . .	44
2.3.2.4	GRASP . . . . .	46
2.3.2.5	Méthode tabou . . . . .	46
2.3.2.6	Recuit simulé . . . . .	47
2.3.2.7	Algorithmes génétiques . . . . .	47
2.4	PROBLÈME DE COLLECTE ET LIVRAISON AVEC FENÊTRE DE TEMPS	48
2.4.1	Formulation du PDPTW . . . . .	48
2.4.2	Modélisation en variables mixtes du PDPTW . . . . .	49
2.4.3	Présentations et analyses des travaux sur le problème de collecte et livraison . . . . .	51
2.4.3.1	Problème statique de collecte et livraison . . . . .	54
2.4.3.2	Problème de collecte et livraison dynamique . . . . .	56
2.5	SYNTHÈSE DES TRAVAUX SUR LES PROBLÈMES AVEC TRANSBORDEMENT . . . . .	58
2.5.1	Problème de tournées de véhicules avec cross-dock (VRPCD) . . . . .	59
2.5.2	Résolution du problème de collecte et livraison avec transbordement . . . . .	64
2.5.2.1	Problème de collecte et livraison avec transbordement statique . . . . .	66
2.5.2.2	Problème de collecte et livraison avec transbordement dynamique . . . . .	70
2.6	CONCLUSION . . . . .	70
<b>3</b>	<b>MODÉLISATION DU PROBLÈME DE COLLECTE ET LIVRAISON AVEC TRANSBORDEMENT</b>	<b>75</b>
3.1	INTRODUCTION . . . . .	77

3.2	CARACTÉRISATION DU PDPT . . . . .	77
3.2.1	Le concept de transbordement . . . . .	77
3.2.2	Exemple de problème de tournées avec transbordement . . . . .	78
3.3	MODÉLISATION DU PROBLÈME . . . . .	80
3.3.1	Hypothèses du problème étudié . . . . .	80
3.3.2	Formalisation du PDPT . . . . .	80
3.3.3	Du modèle PDP au modèle PDPT . . . . .	81
3.3.3.1	Originalité du modèle . . . . .	87
3.4	TESTS EXPÉRIMENTAUX . . . . .	87
3.4.1	Objectifs de l'expérimentation . . . . .	88
3.4.2	Description de la méthode de génération de problèmes PDPT . . . . .	88
3.4.2.1	Principe général . . . . .	88
3.4.2.2	Génération des 3 premières requêtes . . . . .	90
3.4.2.3	Génération de requêtes restantes . . . . .	91
3.4.3	Analyses expérimentales sur la base d'instances simples . . . . .	92
3.5	CONCLUSION . . . . .	94
4	UNE MÉTHODE DE RÉSOLUTION APPROCHÉE DU PROBLÈME DE RAMASSAGE ET DISTRIBUTION AVEC TRANSBORDEMENT	95
4.1	INTRODUCTION . . . . .	97
4.1.1	Hypothèses et notations . . . . .	97
4.2	VNS POUR LA RÉSOLUTION DU PDPT . . . . .	99
4.2.1	Fonction objectif . . . . .	99
4.2.2	Algorithme général du VNS . . . . .	100
4.2.2.1	Spécificités du VNS proposé . . . . .	100
4.2.2.2	Algorithme du VNS pour le PDPT . . . . .	101
4.2.3	Détermination d'une solution initiale . . . . .	104
4.2.4	Voisinages . . . . .	106
4.2.4.1	Voisinages utilisés pour la perturbation . . . . .	106
4.2.4.2	Voisinages utilisés pour la recherche locale sans transbordement . . . . .	108
4.2.5	Recherche locale . . . . .	109
4.2.5.1	Recherche locale sans transbordement . . . . .	109
4.2.5.2	Recherche locale avec transbordement . . . . .	110
4.3	RÉSULTATS EXPÉRIMENTAUX . . . . .	113
4.3.1	Validation du VNS sur des instances de PDP . . . . .	114
4.3.2	Validation du VNS sur des instances de PDPT . . . . .	116
4.3.2.1	Test du VNS sur les instances générées au chapitre 3 . . . . .	116
4.3.2.2	Validation du VNS sur les instances issues de la littérature . . . . .	117
4.4	CONCLUSION . . . . .	120
5	CADRE D'APPLICATION DES TRAVAUX DE LA THÈSE – LE PROJET PRODIGE	121
5.1	INTRODUCTION . . . . .	123
5.2	PRÉSENTATION GÉNÉRALE DU PROJET PRODIGE . . . . .	123
5.2.1	Produit intelligent . . . . .	124
5.2.1.1	Définition . . . . .	124

5.2.1.2	Adaptation au projet PRODIGE . . . . .	125
5.2.2	Routage orienté produit . . . . .	125
5.2.3	Architecture matérielle et logicielle . . . . .	126
5.2.4	Intégration des modules Routage et Portail Web . . . . .	130
5.3	CONTEXTE DE TRANSPORT ÉTUDIÉ DANS PRODIGE . . . . .	132
5.3.1	Reconfiguration des tournées dans PRODIGE . . . . .	133
5.3.1.1	Exemple de reconfiguration de tournées . .	133
5.3.2	Synthèse des caractéristiques du problème étudié . . . . .	136
5.3.3	Résolution du PDPT – du contexte statique au contexte changeant . . . . .	137
5.3.3.1	Analyse des contraintes liées au contexte changeant . . . . .	137
5.3.3.1.1	Prise en compte de la position quelconque d'un véhicule . . . . .	138
5.3.3.1.2	Prise en compte d'une charge initiale dans les véhicules . . . . .	138
5.4	ÉVOLUTION DE LA MÉTHODE DE RÉSOLUTION . . . . .	139
5.4.1	Prise en compte d'une position quelconque des véhicules	139
5.4.1.1	Mise à jour du distancier . . . . .	140
5.4.2	Prise en compte d'une charge initiale . . . . .	141
5.4.3	Adaptation de la méthode de résolution . . . . .	142
5.5	VALIDATION EXPÉRIMENTALE . . . . .	143
5.5.1	Expérimentation théorique . . . . .	143
5.5.1.1	Validation d'une tournée à origine et destination quelconque . . . . .	144
5.5.1.2	Validation du principe de reconfiguration .	145
5.5.2	Expérimentation en situation réelle . . . . .	149
5.5.2.1	Mise en œuvre réelle de la solution PRODIGE	150
5.5.2.1.1	Éléments de caractérisation des scénarios .	150
5.5.2.1.2	Scénario 1 – Transbordement . . . . .	151
5.5.2.1.3	Scénario 2 – Reconfiguration . . . . .	152
5.6	CONCLUSION . . . . .	154
	<b>CONCLUSION GÉNÉRALE</b>	<b>157</b>
<b>A</b>	<b>ANNEXES</b>	<b>161</b>
A.1	MODÈLE COMPLET DU PDPT . . . . .	163
A.2	EXEMPLES D'INSTANCES DE PDPT . . . . .	165
A.3	FONCTION DE BASES . . . . .	172
A.4	CHOIX TECHNIQUES ET LIMITES DU RFID . . . . .	174
A.5	GÉNÉRATION D'UNE POSITION COURANTE CORRESPONDANT À UN RETARD SIMULÉ . . . . .	175
A.6	EXEMPLE DE TOURNÉES TYPES GÉNÉRÉES DANS PRODIGE . .	176
	<b>BIBLIOGRAPHIE</b>	<b>179</b>



# LISTE DES ALGORITHMES

1	LS . . . . .	43
2	VNSDeBase . . . . .	45
3	VNSPourLePDPT . . . . .	102
4	InsertionParallèle . . . . .	105
5	ParametrerVoisinagePourLaPerturbation . . . . .	107
6	Transbordement . . . . .	111
7	InitialisationRoute . . . . .	140
8	PriseEnCompteChargeInitiale . . . . .	142
9	SolutionInitialeDansLecontexteChangeant . . . . .	143
10	VerifTemps . . . . .	172
11	VerifCapacité . . . . .	173
12	Initialisation . . . . .	173
13	Insertion . . . . .	174

# LISTE DES FIGURES

1.1	Une représentation de la chaîne logistique . . . . .	8
1.2	Évolution du nombre d'emplois en France dans le transport routier . . . . .	11
1.3	Illustration de l'affrètement . . . . .	11
1.4	Illustration du groupage . . . . .	12
1.5	Illustration de la messagerie . . . . .	12
1.6	Illustration du transport express . . . . .	13
1.7	Fonctions principales de la chaîne logistique . . . . .	14
1.8	Le cercle vertueux du transport . . . . .	15
1.9	Le principe du cross-docking . . . . .	17
1.10	Répartition par activité des émissions de CO <sub>2</sub> en France en 2009 . . . . .	20
2.1	Le PDP « one-to-one » . . . . .	35
2.2	Le PDP « many-to-many » . . . . .	36
2.3	Le « one-to-many-to-one » . . . . .	37

2.4	Présentation et classification de quelques méthodes de résolution . . . . .	39
2.5	Illustration de deux types de mouvements . . . . .	42
2.6	Principe de la recherche locale . . . . .	43
2.7	Illustration de la recherche locale . . . . .	44
3.1	Le principe du transbordement . . . . .	78
3.2	Impact du transbordement sur la minimisation des véhicules et de la distance . . . . .	79
3.3	Modélisation d'un point de transbordement . . . . .	81
3.4	Principe de base de construction d'une instance . . . . .	89
3.5	Le principe de construction d'une instance . . . . .	90
4.1	Le voisinage RNR . . . . .	108
4.2	Le voisinage SWAP . . . . .	109
4.3	Le voisinage PRR . . . . .	109
4.4	Gestion du temps dans un point de transbordement . . . . .	113
5.1	Architecture matérielle de PRODIGE . . . . .	127
5.2	Architecture matérielle et logicielle de PRODIGE . . . . .	128
5.3	Données caractérisant un véhicule . . . . .	131
5.4	Données caractérisant l'ordre de transport . . . . .	131
5.5	Exemple de ligne du distancier . . . . .	132
5.6	Données en sortie de l'algorithme . . . . .	132
5.7	Tournée retardée . . . . .	134
5.8	Tournées reconfigurées . . . . .	135
5.9	Synthèse de la méthode de résolution . . . . .	137
5.10	Exemple de tournée avec dépôt de départ et d'arrivée différents pour deux véhicules . . . . .	144
5.11	Représentation des clients . . . . .	145
5.12	Représentation des tournées . . . . .	147
5.13	Distribution spatiale des clients . . . . .	151
A.1	Position des clients dans un repère euclidien orthonormé . . . . .	175
A.2	Position réelle du véhicule en cas de congestion et calcul d'angle . . . . .	176
A.3	Position réelle du véhicule en cas de congestion et calcul d'angle . . . . .	177
A.4	Exemple de feuille de route représentée sous forme de tournées . . . . .	177
A.5	Détail des informations associées à une tournée . . . . .	178

# Liste des tableaux

1.1	Quelques référentiels permettant l'évaluation de la performance . . . . .	21
1.2	Quelques indicateurs élémentaires . . . . .	23
1.3	Indicateurs de productivité . . . . .	24
1.4	Indicateurs de charge . . . . .	25
1.5	Indicateurs éco-logistique . . . . .	25
1.6	Indicateurs de qualité de service . . . . .	26
1.7	Synthèse des indicateurs utilisés dans le cadre de la thèse . . . . .	26
2.1	Caractérisation du problème étudié . . . . .	38
2.2	Critères et méthodes utilisées . . . . .	48
2.3	Quelques travaux sur le problème de collecte et livraison . . . . .	53
2.4	Critères et méthodes utilisées pour résoudre le PDP . . . . .	54
2.5	Quelques travaux sur le problème de routage avec cross-docking . . . . .	62
2.6	Critères et méthodes utilisées pour résoudre le cross-docking . . . . .	63
2.7	Synthèse de quelques travaux sur le problème de collecte et livraison avec transbordement . . . . .	65
2.8	Critères et méthodes utilisées pour résoudre le PDPT . . . . .	66
2.9	Positionnement de la thèse par rapport à la grille d'analyse définie . . . . .	72
2.10	Positionnement de la thèse par rapport aux critères d'optimisation et méthodes de résolutions. . . . .	73
3.1	Tableau des paramètres . . . . .	81
3.2	Tableau des variables de décisions . . . . .	82
3.3	Mise en œuvre de la décision de transbordement . . . . .	83
3.4	Résolution exacte du PDPT . . . . .	93
4.1	Stratégie d'utilisation des voisinages dans LS . . . . .	110
4.2	Résultat du VNS sur des instances de PDP . . . . .	115
4.3	Temps CPU du VNS comparé au temps CPU du solveur . . . . .	117
4.4	Résolution du PDPT par le VNS . . . . .	119
4.5	Mise en évidence de la différence entre le VNS et le GRASP . . . . .	120
5.1	Différences entre les deux problèmes traités . . . . .	136
5.2	Distancier du problème initial . . . . .	141
5.3	Une partie du distancier du problème reconfiguré . . . . .	141
5.4	Tournées de véhicules avec dépôts différents . . . . .	145
5.5	Récapitulatif des tournées prévisionnelles des véhicules . . . . .	146
5.6	Propagation du retard dans la tournée du véhicule 1 . . . . .	147

5.7	Effet de la reconfiguration sur une tournée . . . . .	149
5.8	Définition des clients potentiels . . . . .	150
5.9	Définition des requêtes . . . . .	151
5.10	Définition des fenêtres de temps . . . . .	151
5.11	Tournées générées par le module de routage . . . . .	152
5.12	Définition des requêtes . . . . .	153
5.13	Définition des fenêtres de temps . . . . .	153
5.14	Tournée initiale sans retard . . . . .	153
5.15	Tournée initiale impactée par le retard . . . . .	154
5.16	Reconfiguration de la tournée . . . . .	154

# INTRODUCTION GÉNÉRALE

**L**e transport par voie routière joue un rôle clé dans la performance de la chaîne logistique. En effet, il est le support du déplacement des flux physiques dans cette chaîne depuis l'acquisition des matières premières, jusqu'à la distribution du produit final aux clients ou consommateurs. Aujourd'hui, la maîtrise de l'activité du transport est une nécessité absolue pour des industries sujettes à une concurrence exacerbée et à des problématiques environnementales et économiques. Sur le plan environnemental, le transport doit être pensé dans une logique de développement durable caractérisée par une réduction de l'émission de gaz à effet de serre, et plus particulièrement une meilleure maîtrise de l'empreinte carbone liée au déplacement de tout produit. Sur le plan économique, il est nécessaire de rationaliser la gestion transport notamment dans l'utilisation optimale des ressources humaines et matérielles. Les ressources humaines qualifient par exemple les chauffeurs, les logisticiens, tandis que les ressources matérielles concernent principalement la flotte de véhicules et l'ensemble des moyens de manutention et de stockage. La rationalisation de l'utilisation de la flotte de véhicules et l'optimisation du déplacement des véhicules permettent de réduire les coûts énergétiques liés à la consommation des énergies fossiles qui sont amenés à se raréfier et qui connaissent aujourd'hui une augmentation de prix incessante de plus en plus difficilement supportables pour les transporteurs.

Pour relever ces défis, plusieurs actions ont été engagées par les entreprises telles que la mutualisation de leurs ressources tant humaines que matérielles, l'externalisation de leurs prestations de transport et l'utilisation d'outils d'aide à la décision. La mutualisation est une stratégie dans laquelle les entreprises mettent en commun des ressources telles que les entrepôts, plates-formes, stocks et partagent l'utilisation de véhicules à des fins de réduction de coûts. L'externalisation des prestations de transport, consiste pour les entreprises à recourir à des transporteurs extérieurs pour assurer la distribution de leurs produits. La mutualisation et l'externalisation qui relèvent de décisions dites stratégiques ont pour but de contribuer à la réduction des coûts de gestion du transport. Des outils d'aide à la décision sont généralement utilisés pour optimiser le coût de distribution des produits tels que le nombre de véhicules utilisés ou la distance totale parcourue et s'appuient sur des approches issues de la recherche opérationnelle. Au-delà de la maîtrise du coût du transport, la réduction du nombre de véhicules utilisés et de la distance totale concourent également à la réduction de l'empreinte carbone des produits transportés et de la consommation des énergies fossiles.

L'un des tous premiers travaux dont le but était de fournir un outil d'aide à la décision pour optimiser le transport des produits et des tour-

nées des véhicules associées remonte aux années cinquante (Dantzig 1959). Depuis cette publication, l'augmentation croissante des puissances de calculs des ordinateurs modernes associées aux nombreux développements théoriques s'appuyant sur des théories aussi diverses que la programmation linéaire, la programmation par contraintes, les heuristiques et métahéuristiques ont permis le développement d'outils d'aide à la décision offrant aux décideurs la possibilité de générer des tournées de véhicules pour satisfaire les demandes des clients en respectant un ensemble de contraintes et visant un ensemble d'objectifs précis.

Parallèlement à ces travaux, le développement des nouvelles technologies de l'information et de la communication (NTIC) a engendré une mutation et un remodelage de l'organisation des transports. En effet, les NTIC favorisent la connectivité entre partenaires de la chaîne logistique, ainsi qu'un meilleur suivi des activités afférentes qui sont par nature spatialement distribuées. Ainsi, en transport, les demandes de service sont de plus en plus nombreuses, et transmises de manière quasi instantanée, via l'utilisation de standards d'échange EDI (Echange de Données Informatiques) ou de la vente en ligne accessible 24 heures sur 24. De surcroît, les prestataires sollicités ne sont pas obligatoirement des acteurs à proximité des lieux de chargement et livraison, et les volumes de marchandises transportées peuvent amener à utiliser les services de multiples transporteurs. Dans un tel contexte, les problèmes posés ne concernent plus simplement l'optimisation des coûts, mais aussi l'amélioration de la traçabilité des produits et la prise en compte d'objectifs liés au développement durable. La gestion optimale de l'activité du transport pour répondre à ces défis nécessite des systèmes réactifs, flexibles et reconfigurables dont la mise en œuvre passe par une instrumentation du transport. Cette instrumentation est souvent étendue au niveau des opérations de préparation du transport tels que par exemple le chargement, le déchargement des palettes, les opérations d'éclatement et groupage des produits sur les plateformes logistiques, ainsi que toutes les opérations de manutention associées.

Dans cette thèse, nous nous intéressons au problème de routage de véhicules et plus particulièrement aux problèmes de collectes et livraisons pour lesquels nous proposons d'utiliser le transbordement comme un moyen d'optimiser le transport. De manière générale, l'objectif vise à minimiser le coût d'exploitation des véhicules qui dépend du nombre de véhicules et des distances totales parcourues pour satisfaire les requêtes des clients. Une requête de transport consiste à charger un produit chez un fournisseur afin de le livrer chez un client. L'exploitation du transbordement dans un tel problème consiste à utiliser plus d'un véhicule dans l'acheminement d'un produit de son fournisseur vers son client. Ainsi, le véhicule ayant collecté un produit donné n'est plus nécessairement celui qui le livre chez le client si cela permet de réduire le coût du transport. Dans un tel scénario, le véhicule ayant collecté le produit le dépose dans un lieu spécifique appelé point ou centre de transbordement et continue sa tournée. Un second véhicule intègre le point de transbordement dans sa tournée afin d'y collecter le produit ayant été déposé précédemment pour le transporter jusqu'à sa destination finale. Le caractère « opportuniste » du transbordement tel qu'il est étudié dans le présent travail propose alors une rupture de pratique avec le classique « cross-docking » et

qualifie une exécution de transbordement en réponse à la détection de dérives de temps ou d'aléas menant à une reconfiguration possible des tournées dans une recherche systématique d'une meilleure performance. Cette caractéristique se définit par ailleurs par la possibilité d'exécuter une telle activité en un ensemble de lieux, qu'il s'agisse de clients visités dans les tournées, ou de plateformes ad-hoc, dont le choix relève de logiques d'optimisation face au contexte de transport observé. Le transbordement peut ainsi être vu comme un levier d'optimisation permettant une réduction des coûts d'exploitation d'une flotte de véhicules et offrant une meilleure flexibilité à tout système de transport.

Pour présenter nos travaux, la présente thèse est divisée en 5 chapitres. Le premier chapitre porte sur les pratiques usuelles dans le transport routier de marchandises afin de fixer les bases et le contexte général de la thèse. Cette analyse se structure en plusieurs étapes. Dans un premier temps, les activités de transport sont positionnées au sein de la chaîne logistique, en mettant en avant les caractéristiques propres à ces activités. Dans un deuxième temps, un regard particulier sera porté sur les décisions d'ordre opérationnel inhérentes au pilotage de chaîne logistique et du transport. Les pratiques de terrain utilisées dans les schémas de transport actuels sont ensuite analysées. Enfin les principaux indicateurs permettant la mesure de la performance de l'activité de transport routier sont identifiés.

Le deuxième chapitre est consacré à un état de l'art sur les problèmes de collectes et livraisons, en s'intéressant plus particulièrement aux problèmes avec transbordement. Les principaux travaux scientifiques proches de la présente étude sont analysés et classés selon une typologie de problèmes largement inspirée de la littérature. Les méthodes de résolution des problèmes de routage sont également étudiées en se focalisant plus particulièrement sur les méthodes de résolution dites approchées, compte tenu de la complexité des problèmes à résoudre. Un tel état de l'art nous permet alors de situer plus précisément la singularité du problème étudié dans la présente thèse, et d'identifier un ensemble de méthodes « efficaces » susceptibles d'être adaptées à la résolution de ce problème.

Le troisième chapitre propose un modèle linéaire en variables mixtes du problème considéré. L'intérêt d'une telle modélisation réside dans la volonté de spécifier le problème étudié et de connaître les limites des méthodes exactes pour le problème en question. Dans une première phase, les hypothèses qui sont prises en compte dans notre modélisation sont énoncées et quelques exemples sont également proposés pour illustrer le principe du transbordement dans le cadre du transport de produits par une flotte de véhicules homogènes. Dans une seconde phase, nous caractérisons le problème par un modèle exact en variables mixtes, avant de procéder à des tests expérimentaux à partir d'instances de petite taille générées par une méthode ad hoc. Les performances du modèle sont ainsi observées à travers un ensemble d'indicateurs tels que le nombre de véhicules utilisés, les distances parcourues ou encore le temps de calcul nécessaire pour trouver une solution. L'observation de l'évolution du temps de calcul en fonction de la taille du problème à résoudre confirme alors l'inadéquation d'une telle méthode de résolution pour des problèmes de

taille réaliste et nous conduit au développement de méthodes approchées, telles que présentées au chapitre suivant.

Le quatrième chapitre est ainsi axé sur le développement des heuristiques et mét-heuristiques comme alternative au modèle exact proposé dans le chapitre précédent. La mét-heuristique que nous avons choisi de développer est une adaptation de la recherche à voisinages variables (VNS). La spécificité de la mét-heuristique que nous proposons par rapport au VNS classique s'appuie sur deux caractéristiques : la notion de solution prometteuse et une recherche locale utilisant ou non le principe de transbordement. Les spécificités introduites ont pour but de maîtriser le temps de calcul tout en permettant une recherche efficace conduisant à de « bonnes » solutions. Cette approche de VNS est validée sur la base des instances utilisées au chapitre 3 afin de montrer la capacité de la mét-heuristique à résoudre en temps raisonnable et de manière efficace des instances du problème de collectes et livraisons avec transbordement. Ces tests sont complétés par de nouvelles expérimentations basées sur des instances issues de la littérature scientifique en comparant le résultat de calculs obtenus aux meilleurs résultats publiés.

Le cinquième chapitre permet de montrer l'application des développements effectués durant la thèse et leur exploitation en situation réelle. En effet les solutions obtenues tout au long de la thèse ont vocation à être intégrées dans l'outil d'aide à la décision développé au sein du projet ANR PRODIGE. Ce dernier a pour objectif de développer une instrumentation de l'activité de transport routier à des fins d'optimisation de flux, permettant entre autres fonctions de localiser de manière quasi permanente les véhicules et les produits. Aussi, dans ce chapitre, le projet PRODIGE est plus précisément présenté ainsi que la problématique du transport qui y est étudiée. L'analyse de cette problématique fait ressortir la notion de re-configuration des tournées dans un contexte changeant, caractérisée par le fait que les données initiales et hypothèses utilisées pour planifier les tournées de véhicules peuvent évoluer au cours du temps, du fait de l'occurrence d'événements perturbateurs. Afin de prendre en considération cette nouvelle problématique, une nouvelle approche de résolution reposant sur une adaptation de la mét-heuristique développée au chapitre 4 est présentée et analysée, avant d'être interfacée avec la solution informatique développée dans le cadre du projet PRODIGE. La validation de cette méthode de résolution ainsi modifiée est réalisée en s'appuyant sur un jeu d'instances théoriques et son exploitation est testée en condition réelle au travers d'une expérimentation de terrain, basée sur quelques scénarios de tournées.

Enfin, nous proposons une conclusion, rappelant les différentes contributions de la thèse et les résultats les plus saillants, avant de proposer de nouvelles pistes de travail futur.

# LE TRANSPORT ROUTIER DE MARCHANDISES AU SEIN DE LA CHAÎNE LOGISTIQUE

1

## SOMMAIRE

1.1	INTRODUCTION . . . . .	7
1.2	LA CHAÎNE LOGISTIQUE ET LE TRANSPORT . . . . .	7
1.2.1	La chaîne logistique . . . . .	8
1.2.2	La place du transport routier . . . . .	10
1.2.3	Les offres de transport . . . . .	11
1.3	LA GESTION DE LA CHAÎNE LOGISTIQUE . . . . .	13
1.3.1	Les principales fonctions de la chaîne logistique . . . . .	14
1.3.2	La fonction distribution/transport . . . . .	15
1.4	LES PRATIQUES ACTUELLES ET LES NOUVELLES TENDANCES DANS LE TRANSPORT . . . . .	16
1.4.1	L'externalisation des chaînes d'approvisionnements . . . . .	16
1.4.2	La pratique de la mutualisation . . . . .	16
1.4.3	Le cross-docking . . . . .	17
1.4.4	Le transbordement . . . . .	17
1.4.5	Le TMS . . . . .	18
1.5	LA PERFORMANCE DANS LE SECTEUR TRANSPORT . . . . .	19
1.5.1	Les contraintes économiques et sociales . . . . .	20
1.5.2	Leviers et indicateurs de performance . . . . .	21
1.5.3	Vers un référentiel de performances pour le transport routier	22
1.6	CONCLUSION . . . . .	27



## 1.1 INTRODUCTION

**C**es dernières années, la mondialisation<sup>1</sup> plus qu'un concept est devenue une réalité tangible. L'une des manifestations les plus visibles est la disparition des barrières douanières entre les états qui partagent des intérêts économiques réciproques. C'est ainsi que l'on trouve des entreprises qui possèdent des sources d'approvisionnement et des sites de production répartis aux quatre coins du globe. D'autre part, le récent développement des NTIC (Nouvelles Technologies de l'Information et de la Communication) a permis aux entreprises de renforcer l'efficience et la qualité de leurs activités spatialement dispersées. Ces évolutions majeures engendrent une complexité accrue des échanges commerciaux. Dans ce contexte, le transport devient une fonction essentielle qui est en charge d'assurer la liaison et la circulation du flux physique entre les différents acteurs, tout en garantissant un ensemble d'objectifs qui peuvent être parfois antagonistes et en respectant un ensemble de contraintes axées sur la structure du réseau de distributions, les spécificités des moyens de transport, les législations des pays ou l'intégrité des marchandises transportées.

Dans ce chapitre, le transport routier de marchandises est analysé pour fixer les bases et le contexte général de ce travail de thèse. Pour dégager des éléments permettant d'atteindre ces objectifs, le présent chapitre est divisé en 5 sections. La première section présente notamment la chaîne logistique et le transport, en insistant notamment sur la place et les différentes offres de transport routier qui existent. La deuxième section est dédiée à la gestion de la chaîne logistique en analysant de façon plus fine la fonction du transport en particulier. Cette analyse porte notamment sur le niveau de décision opérationnelle relative au transport. La troisième section présente les pratiques et les nouvelles tendances du transport. Cette présentation couvre les aspects organisationnels et technologiques du transport. La quatrième section discute de la performance dans le secteur du transport. Cette performance est évaluée suivant plusieurs axes, en tenant compte des contraintes économiques et sociales. Un référentiel de la performance pour le transport routier y est aussi proposé. La cinquième et dernière section permet de conclure le chapitre.

## 1.2 LA CHAÎNE LOGISTIQUE ET LE TRANSPORT

La satisfaction d'une commande d'un produit par un client nécessite au préalable l'exécution d'un ensemble d'opérations. Ces opérations concernent : la recherche et la mise à disposition des matières premières, la transformation de celles-ci en produits finis, le stockage et la distribution de ces produits aux clients finaux. Chacune de ces opérations a une complexité et une performance qui lui est propre. La complexité de chaque opération dépend des processus qui doivent être exécutés pour la réaliser, tandis que la performance de chacune d'elle peut influencer le bon déroulement des autres opérations. Par exemple, un retard dans le processus

1. D'après le PETIT LAROUSSE, en économie, la mondialisation est la tendance des entreprises multinationales à concevoir des stratégies à l'échelle planétaire, conduisant à la mise en place d'un marché mondial unifié.

de distribution peut entraîner le non-respect de la livraison d'un produit à un client donné. L'ensemble des opérations précédemment présentées fait partie intégrante d'une chaîne appelée chaîne logistique dont l'étude a pour objectif d'aboutir à une optimisation et à une performance globale de manière à satisfaire à terme le client dans un environnement toujours plus concurrentiel et incertain.

### 1.2.1 La chaîne logistique

Qu'est-ce que la chaîne logistique ? Plusieurs auteurs ont tenté depuis de nombreuses années de répondre à cette question. Les nombreuses définitions proposées dans la littérature, sont là pour en témoigner. Chacune d'elles, est généralement orientée par la sensibilité de l'auteur, et le domaine d'étude concerné. Pour cette raison, il n'existe pas une définition unique de la chaîne logistique, mais des définitions de la chaîne logistique. Nous en proposons quelques-unes :

(Lee1993) proposent la définition suivante : « *Un réseau d'installation qui assure les fonctions d'approvisionnement en matières premières, de transformation de ces matières premières en composants puis en produits finis, et de distribution du produit fini vers le client* » .

(New1995) présentent la chaîne logistique comme étant : « *Une succession d'activités de création de valeur. Cette création repose sur un ensemble d'entreprises qui assurent l'extraction des matières premières, et la production, le grossiste, les détaillants et enfin le client final* » .

La figure 1.1 adaptée de (New1995) avec la considération de la logistique inverse (recyclage) permet d'illustrer cette définition.

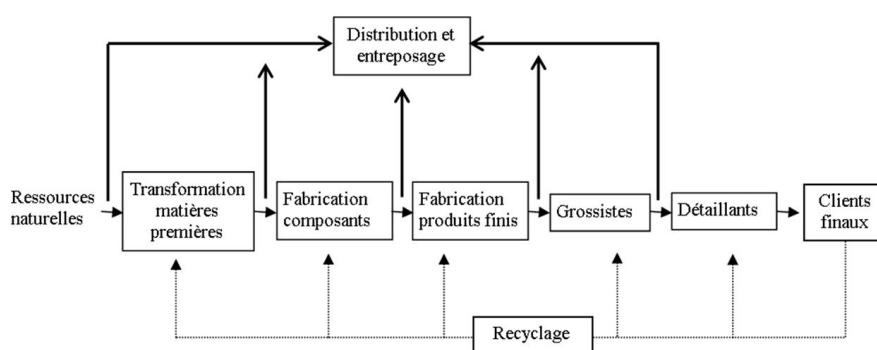


FIGURE 1.1 – Une représentation de la chaîne logistique

(Tayur1999) définissent la chaîne logistique comme « *Un système de sous-traitants, de producteurs, de distributeurs, de détaillants et de clients entre lesquels s'échangent les flux matériels dans le sens des fournisseurs vers les clients et des flux d'information dans les deux sens* » .

(Chopra2007) donnent dans leur livre la présente définition :« *La chaîne logistique consiste en toutes les étapes impliquées directement ou indirectement dans la satisfaction de la requête d'un client. La chaîne logistique inclut non seulement le fabriquant et ses fournisseurs, mais aussi les transporteurs, les centres d'entreposage, les détaillants et les clients eux-mêmes* ».

Cet échantillon de définitions permet de rendre compte des définitions possibles de la chaîne logistique. Leurs analyses permettent de caractériser la chaîne logistique comme :

- Une chaîne liant successivement des clients et des fournisseurs (Tayur1999).
- Un ensemble ordonné d'activités successives qui permet de créer de la valeur (New1995).
- Un ensemble de fonctions ou processus qui sont l'approvisionnement, la transformation, et la distribution (Lee1993).

En effectuant une analyse des définitions proposées, il est possible de remarquer qu'elles reposent toutes sur quelques idées communes :

- Un même type de produits ou plusieurs types de produits différents sont considérés.
- Plusieurs entreprises interviennent tout au long de la chaîne.
- Trois types de flux permettent de lier les entreprises entre elles : le flux d'informations, par exemple : une demande de transport ; le flux physique qui concerne le transfert des produits ; le flux financier.
- Chaque entreprise concernée joue un ou plusieurs des rôles suivants : approvisionnement, production, distribution et vente.
- Une entreprise peut être intégrée dans plusieurs chaînes logistiques.

Au regard de ces derniers éléments, et en faisant abstraction du flux financier qui est hors de la portée de cette thèse, nous proposons notre propre définition de la chaîne logistique.

*La chaîne logistique est constituée d'un ensemble de maillons interconnectés, dont la finalité est de satisfaire une demande de client. Cette demande peut porter sur un seul type de produit, ou des types de produits différents. Deux types de flux traversent les maillons considérés : le flux informationnel et le flux physique.*

Sur la base de la précédente définition, il apparaît que l'interconnexion entre les différents maillons est essentielle au bon fonctionnement de la chaîne logistique. En effet, la sortie d'un maillon ou étage de la chaîne est généralement l'entrée d'un autre maillon qui apportera sa compétence à la réalisation du produit final. Pour rendre cela possible, il est nécessaire de transporter la matière d'un maillon à un autre. Le transport apparaît comme un processus essentiel de la chaîne logistique. Le transport dans le contexte de la chaîne logistique consiste à planifier, piloter et optimiser l'acheminement de la matière entre les lieux d'approvisionnement et de distribution avec des échanges possibles entre les différents sites de production.

Comme il peut être observé dans la figure 1.1 issue de (New1995)

Le transport permet d'acheminer :

- Les matières premières aux différents sites de transformation ;
- Les matières transformées entre sites de production ;
- Les produits à l'issue de leurs fabrications vers des sites d'entreposage ;
- Les produits des sites d'entreposage pour les distribuer au client final.

Plusieurs moyens de transport peuvent être utilisés pour déplacer le flux. Il peut s'agir de moyens aériens, routiers, ferroviaires, fluviaux par mer ou intermodal. Lorsque le transport est intermodal, une combinaison des moyens de transport peut être utilisée pour déplacer les produits. Dans les cas d'un transport bimodal par exemple, cette combinaison peut coupler les moyens de transport utilisant la mer et la route, ou la route et le rail.

Ce chapitre porte sur le transport routier de marchandises dans le

contexte de la chaîne logistique. Le choix du transport routier est motivé par le fait qu'il est l'objet d'étude principal de la thèse. En effet, les différentes contributions qui seront proposées dans les chapitres suivants ont vocation à être intégrées dans des outils d'aide à la décision pour améliorer la gestion du transport.

### **1.2.2 La place du transport routier**

La contribution du transport routier de marchandises au fonctionnement de chaîne logistique est présentée dans cette partie suivant deux axes : sa domination par rapport aux autres modes de transport et son rôle majeur au niveau économique.

Le transport routier de marchandises domine (Bernadet 1997) les autres modes de transport qui sont : le fluvial, le maritime, le rail, et l'aérien. Cela s'explique par le fait, que les autres modes de transport de par leur nature ne sont pertinents que pour le transport de longues distances massifiées. Tandis que le transport routier est généralement le seul de par sa nature à pouvoir desservir tout un territoire lorsque les infrastructures de ce réseau ont été préalablement développées. L'évolution technologique a aussi permis de développer des camions moins énergivores, de tailles adaptées pour les différents espaces urbains. Ainsi, dans une situation de flux tendu, de réduction de stock, de transport à la demande, et de transport porte à porte, le transport routier s'avère être mieux adapté que les autres modes de transport. Le transport routier de marchandises est dominant et dans le cas de la France par exemple, 80 % des biens industriels sont transportés par route. Il en est de même pour 99 % des biens de consommations.

Sur le plan social, en France (source : Insee<sup>2</sup> et Soes<sup>3</sup>), le transport routier représente plus de 37000 entreprises, 380000 personnes, 300000 véhicules pour un chiffre d'affaire total de 44 milliards d'euros en 2010. Le transport routier contribue au produit intérieur brut (PIB) à hauteur de 2,3% et engendre des emplois dans plusieurs domaines transversaux : logistiques (manutention), scientifiques (recherche opérationnelle), sécuritaire etc. Dans la figure 1.2 réalisée à partir des données issues du site web (Insee 2008), l'axe des abscisses représente les années et l'axe des ordonnées correspond aux emplois en millier dans le domaine du transport correspondant aux années.

---

2. Institut national de la statistique et des études économiques

3. Service de l'observation et des statistiques du ministère du développement durable

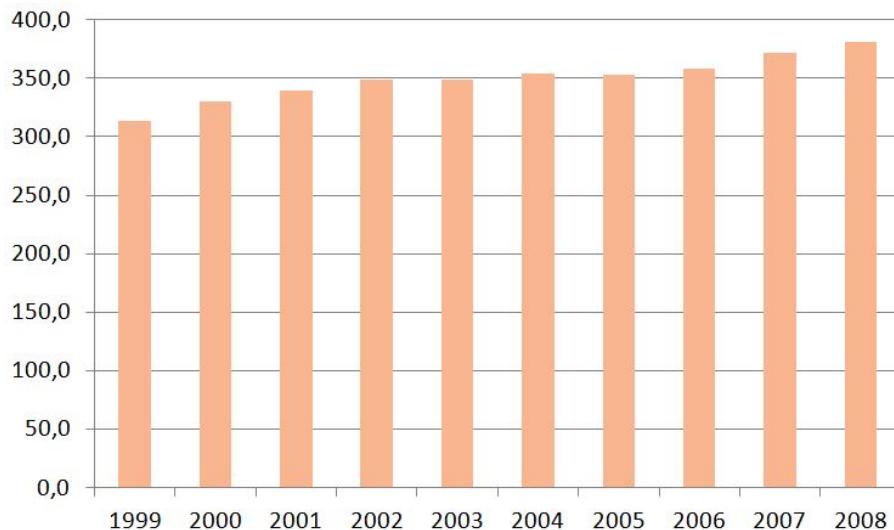


FIGURE 1.2 – Évolution du nombre d’emplois en France dans le transport routier

Après avoir analysé le rôle important du transport routier dans l’économie, une typologie des différentes offres de transport est présentée.

### 1.2.3 Les offres de transport

Plusieurs critères permettent de catégoriser les différentes offres de transport routier de marchandises. Ces critères portent notamment sur : la taille des colis transportés, la complexité des prestations de transport offertes, la rapidité du transport, le mode de calcul du prix de transport et la pratique de rupture de charge<sup>4</sup>. Quelques offres de transport sont présentées et analysées ci-dessous.

#### 1.2.3.1 Affrètement

Dans le cas de l'affrètement ou transport de lot complet, l'intégralité de la capacité d'un véhicule est utilisée pour satisfaire la demande de transport d'un client. Il n'y a pas de rupture de charge durant l'acheminement et le transport s'effectue directement d'un point à un autre.



FIGURE 1.3 – Illustration de l'affrètement

4. Rupture de charge : transfert de marchandises en cours du transport d'un véhicule à un autre avec ou sans stockage

### 1.2.3.2 Le groupage

Le transport avec groupage a la spécificité d'exiger du transporteur qu'il ait recours à des plateformes pour effectuer le groupage et le dégroupage des marchandises. Le groupage consiste à ramasser des colis provenant de plusieurs expéditeurs pour plusieurs destinataires différents. Après cette phase, un transport par lot complet est effectué pour transporter les produits collectés de la plateforme de groupage à la plateforme de dégroupage. Dans la plateforme de dégroupage, l'organisation de la livraison des marchandises aux destinataires finaux est effectuée.

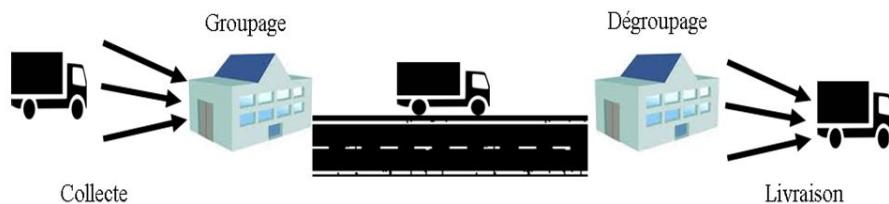


FIGURE 1.4 – Illustration du groupage

### 1.2.3.3 La messagerie

La messagerie est axée sur le transport des colis de moins de trois tonnes. Les produits sont enlevés, groupés, dégroupés et distribués à leurs destinataires respectifs. La messagerie diffère du groupage par le fait que les enlèvements ne sont pas effectués à la demande, mais à heure fixe. D'autre part, le traitement des marchandises au sein des plateformes est effectué sans stockage.

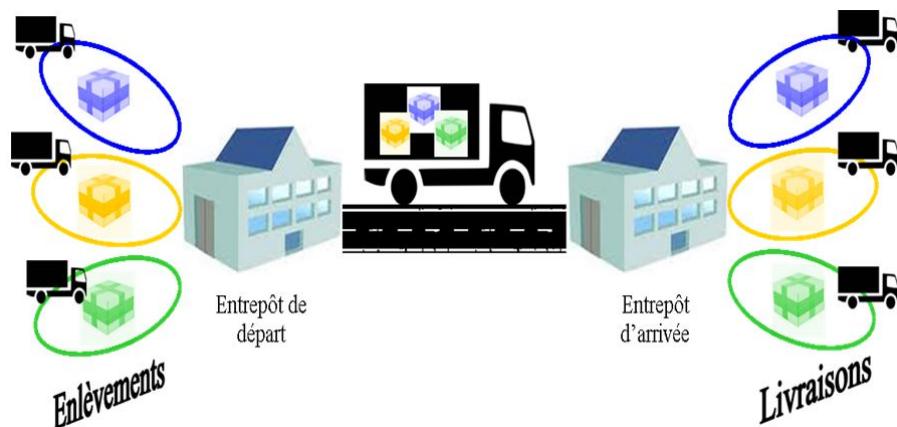


FIGURE 1.5 – Illustration de la messagerie

### 1.2.3.4 Le transport express

Le transport express (Branche2014) assure le transport des colis d'un poids inférieur à trois tonnes. Dans son organisation, il est très proche de la messagerie. Il en diffère principalement par la garantie d'exécution du transport dans un délai contractuel exprimable en heure ou en demi-journées.



FIGURE 1.6 – Illustration du transport express

Cette section a permis de présenter les différentes offres de transport routier qui existent. Le caractère très hétérogène de ces offres permet au transport routier de marchandises de disposer d'une grande flexibilité comparée à d'autres moyens de transport. Cette hétérogénéité associée à la complexité relative du pilotage du transport nécessitent de disposer d'un système de gestion efficace afin d'assurer un transport de qualité optimale pour garantir l'échange de flux physiques entre les différents partenaires de la chaîne logistique. Le paradigme appelé gestion de la chaîne logistique permet de disposer d'un ensemble d'approches de pilotage de la chaîne logistique à travers les différentes fonctions qui la constituent. La gestion de la chaîne logistique est présentée et analysée en insistant plus particulièrement sur la fonction transport – distribution et en mettant l'accent sur les différents types de décisions qui s'y appliquent.

### 1.3 LA GESTION DE LA CHAÎNE LOGISTIQUE

L'objectif de la maximisation des performances d'une chaîne logistique engendre la nécessité d'un pilotage adapté de celle-ci. Ce pilotage aussi appelé gestion de la chaîne logistique est désigné dans la littérature anglaise par le terme SCM pour « Supply Chain Management ». De nombreuses et diverses définitions existent pour la gestion de la chaîne logistique. Elles dépendent du domaine d'étude auquel elles se rapportent (Mentzer2001). Nous présentons quelques définitions issues de la littérature.

(Stadtler2005) définit la gestion de la chaîne logistique comme « *La démarche permettant l'intégration d'unités organisationnelles le long de la chaîne logistique et la coordination des flux physiques, informationnels et financiers dans le but de satisfaire le consommateur final et d'améliorer la compétitivité de la chaîne dans son ensemble* ».

(Simchi2003) définissent la gestion de la chaîne logistique comme : « *Un ensemble d'approches utilisées pour intégrer efficacement les fournisseurs, les producteurs, les distributeurs et les détaillants de façon à garantir la production et la distribution des produits finis au bon moment, au bon endroit, en bonne quantité, en respectant les exigences des clients finaux, et ce à moindre coût* ».

(Dominguez2004) observent que « *L'intérêt du Supply Chain Management (SCM) est de faciliter les ventes en positionnant correctement les produits en bonne quantité, au bon endroit, au moment où le client en a besoin et enfin à moindre coût. Le principal objectif du SCM est d'allouer efficacement les ressources de production, distribution, transport et d'information, en présence d'objectifs conflictuels, dans le but d'atteindre le niveau de service demandé par les clients au plus bas prix* ».

Ces trois définitions permettent de mettre en exergue une caractéris-

tique principale de la gestion de la chaîne logistique en tant que moyen d'intégration de l'ensemble des partenaires que sont : les fournisseurs, les producteurs, les distributeurs et le client final. Cette intégration vise à l'optimisation des différentes fonctions de la chaîne logistique dans le but d'améliorer la compétitivité des entreprises.

### 1.3.1 Les principales fonctions de la chaîne logistique

Les principales fonctions de la chaîne logistique sont : l'approvisionnement, la production, la distribution et la vente. Une classification de ces fonctions et des tâches qui composent chacune d'elles est présentée sous forme matricielle dans la figure 1.7 issue de (Stadtler2010).

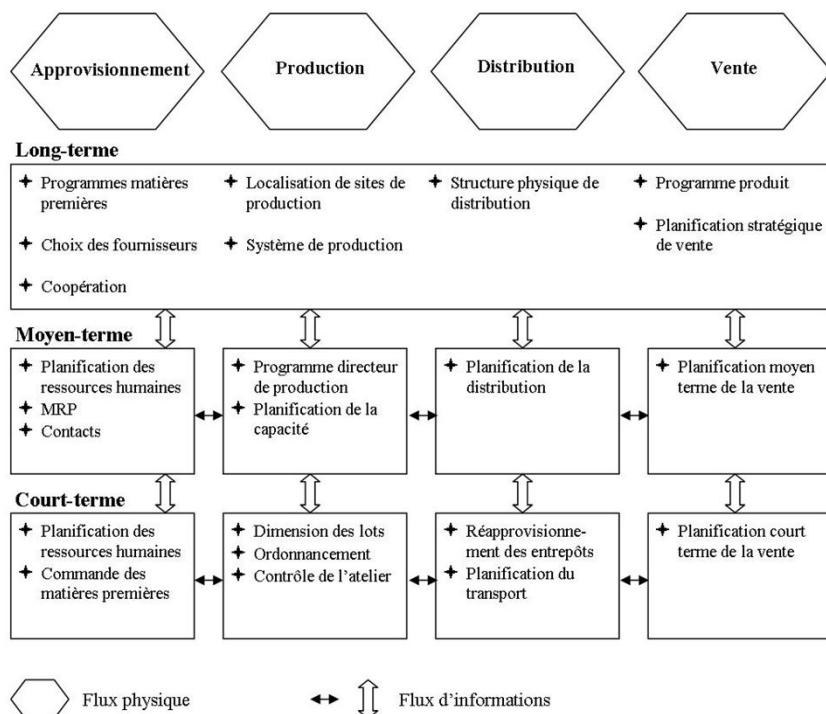


FIGURE 1.7 – Fonctions principales de la chaîne logistique

Dans cette figure, chaque tâche qui constitue une des principales fonctions de la chaîne logistique est représentée avec le niveau décisionnel auquel il appartient. La gestion de la chaîne logistique fait appel à plusieurs niveaux de décisions généralement utilisés en gestion de production (Hax1984). Les décisions sur le long terme sont stratégiques, celles à moyen terme sont tactiques. Les décisions sur le court terme sont quant à elles opérationnelles. L'aperçu des tâches proposées dans la matrice permet de donner une idée des différents types de décisions à prendre pour exécuter chacune d'elles.

La présente thèse concerne plus particulièrement les problèmes de transport et les contributions qui seront développées tout au long des prochains chapitres ont vocation à être intégrées dans des outils d'aides à la décision pour le niveau tactique et le niveau opérationnel. Au niveau tactique, les décisions peuvent notamment porter sur l'achat de nouveaux véhicules, le nombre de visites autorisées chez un client durant une tournée, etc. Au niveau opérationnel, les décisions concernent la planification

des tournées, le suivi d'exécution et leur reconfiguration éventuelle en cas d'aléa. La fonction transport est particulièrement décrite dans la section suivante ainsi que le niveau de décision opérationnelle qui s'y rapporte.

### 1.3.2 La fonction distribution/transport

La partie distribution dans la fonction transport/distribution consiste à écouler les produits finis vers les clients finaux. Elle nécessite l'exécution d'un ensemble d'activités et la gestion des réseaux de distributions telles que les plateformes de groupage/dégroupage. Les activités concernent respectivement la réception des commandes, la définition des délais de livraison, la planification des livraisons et l'expédition.

La partie transport dans la fonction distribution/transport se décompose en trois principales activités : le dimensionnement du réseau de transport, la planification des tournées, la gestion de la flotte de véhicules.

- Le dimensionnement du réseau du transport consiste par exemple à déterminer les nœuds ou sommets successifs du réseau de transport à visiter.
- La planification des tournées a pour objet la détermination des tournées d'une flotte de véhicules homogène ou hétérogène afin de satisfaire la demande d'un ensemble de clients. La demande peut consister en un transport point à point d'un objet à un autre, ou d'une demande d'intervention (maintenance, réparation), etc. L'objectif est de déterminer pour chaque véhicule de la flotte une feuille de route qui permet, tout en satisfaisant les demandes des clients, de réduire les coûts de transport. La planification permet ainsi de faire de manière anticipatoire des arbitrages et équilibrages nécessaires en fonction de la capacité de transport disponible et du taux effectif du remplissage des véhicules.

Pour faire du transport un véritable vecteur de compétitivité et de performance les décisionnaires doivent prendre plusieurs décisions qui sont d'ordres stratégiques, tactiques, opérationnelles et temps réel. La figure 1.8 qui représente le cercle vertueux du transport est issue de (Dode2010) et propose une représentation stratifiée de ces différents niveaux de décision ainsi que les questions afférentes à ces niveaux. La thèse étant principalement orientée sur les décisions de type opérationnel, une discussion portant notamment sur ce type de décisions est proposée.

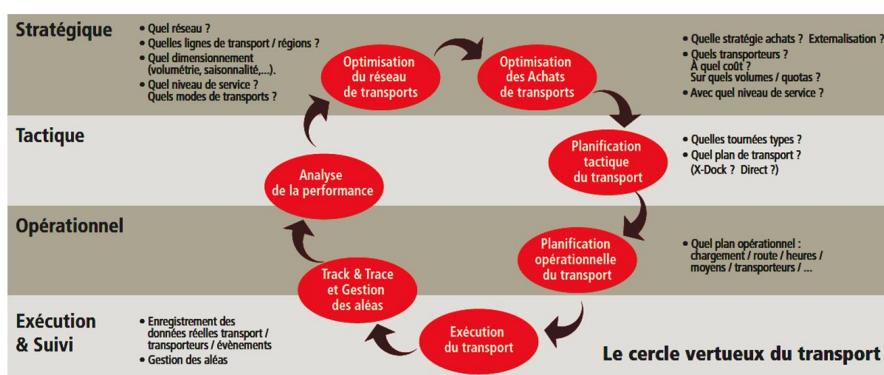


FIGURE 1.8 – Le cercle vertueux du transport

### 1.3.2.1 Le niveau de décision opérationnel

Les décisions d'ordre opérationnelles relatives au pilotage des différentes fonctions de la chaîne logistique en général et du transport en particulier couvrent un horizon de quelques jours à environ une semaine. Il s'agit ici des décisions qui portent sur les tournées qui seront effectivement générées pour satisfaire les demandes de transport. Suite à l'occurrence d'un aléa ou à un événement perturbateur qui empêche l'exécution normale d'une tournée initialement prévue, il peut être nécessaire de reconfigurer une tournée de transport pour garantir une certaine performance. La décision portant sur la nécessité d'une reconfiguration est dite « temps réel » et couvre un horizon de temps allant de la minute à l'heure en fonction de la politique choisie.

En complément de la prise de décision, ci-dessus présentée, pour affiner, améliorer et gérer de manière efficace le transport, de nouvelles approches ont été proposées ou sont en cours de mise en place. Pour s'en convaincre, nous nous proposons de présenter quelques unes.

## 1.4 LES PRATIQUES ACTUELLES ET LES NOUVELLES TENDANCES DANS LE TRANSPORT

Les nouvelles tendances observées dans l'organisation des transports sont d'ordre organisationnel ou technologique.

### 1.4.1 L'externalisation des chaînes d'approvisionnements

La tendance actuelle des entreprises est à l'externalisation de leurs chaînes d'approvisionnements (Lanneau2006). Par ce processus, elles se recentrent sur leur cœur de métier et sous-traitent les activités connexes dont le transport. L'objectif est de réaliser des économies d'échelles sur le transport.

### 1.4.2 La pratique de la mutualisation

D'après le dictionnaire Petit Robert, la mutualisation consiste à « *répartir à égalité entre les membres d'un groupe* ». Décliné au sens de la chaîne logistique, cette définition peut être perçue comme la mise en commun par plusieurs entreprises de leurs ressources afin d'optimiser une fonction particulière de leur chaîne logistique réciproque. Cette association peut être nécessaire en temps de crise ou dans un contexte de mondialisation ou de concurrence exacerbée pour accéder à un service ou atteindre un objectif inaccessible de façon individuelle.

Au niveau du transport, cette tendance peut s'observer par une logique de mutualisation dans laquelle le transporteur, les expéditeurs et les destinataires mettent en commun leurs ressources mobiles (véhicules) et leurs infrastructures logistiques (entrepôts, plates-formes). Dans un tel contexte, les infrastructures logistiques ne sont plus uniquement des sites d'expédition et de réception mais deviennent des points de transbordement. Dans ces points, le groupage/ dégroupage ainsi que le transfert de produits entre véhicules différents sont autorisés si cela permet d'apporter de la flexibilité et une réduction des distances parcourues.

### 1.4.3 Le cross-docking

Le cross-docking (CD) (Kreng2008) est une stratégie logistique dans laquelle les produits enlevés chez un ensemble de fournisseurs sont "consolidés" lors d'un passage par une plateforme logistique (ou entrepôt) avant d'être livrés en aval chez un ensemble de destinataires. La consolidation consiste à réceptionner à l'entrepôt les produits provenant des différents véhicules, les trier et les classer en fonction de leurs destinations finales. Dans certains cas, les produits peuvent être momentanément stockés. Très souvent, l'entrepôt est aussi utilisé comme dépôt de véhicules. À cet effet, les véhicules chargés d'effectuer la collecte et la distribution des produits partent et rentrent à l'entrepôt.

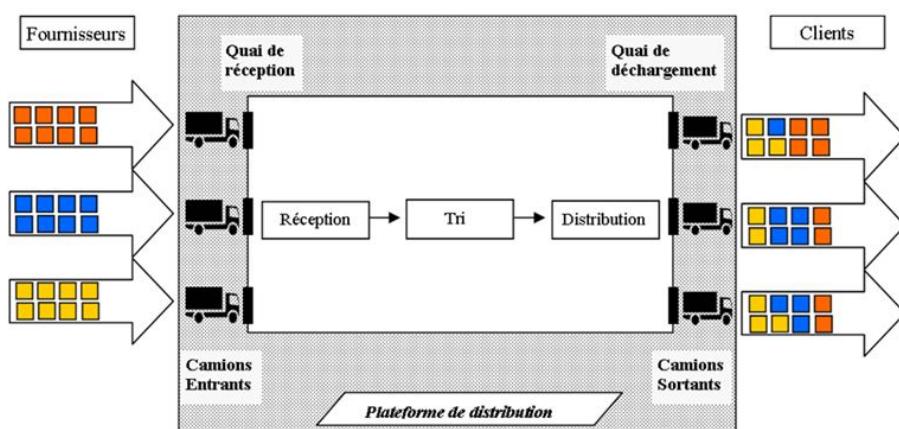


FIGURE 1.9 – Le principe du cross-docking

Un certain nombre d'auteurs (Galbreth2008)(Boysen2010) ont travaillé sur l'intérêt de pratiquer le cross-docking. Les avantages d'une telle pratique sont : la réduction du niveau des stocks, des coûts de production, et du coût de distribution ; économie de temps grâce à la réduction des délais d'approvisionnement ; l'augmentation de la durée des produits dans le rayon, l'augmentation en linéaire de la durée de vie des produits notamment lorsqu'il s'agit de produit frais. Le cross-docking présente néanmoins un ensemble d'inconvénients : une préparation minutieuse des palettes doit être effectuée en amont par le fournisseur pour faciliter la consolidation de celles-ci au cross-dock. Cela induit une charge supplémentaire de gestion.

### 1.4.4 Le transbordement

Le transbordement<sup>5</sup> représente l'action de transférer des marchandises ou des voyageurs d'un bateau, d'un train, d'un véhicule à un autre. Dans le transport routier de marchandises, le transbordement est effectué entre véhicules dans un but de flexibilité (Nakao2008) et de réduction de coût (Shang1996). La flexibilité découle du principe même du transbordement. En effet, en autorisant le transfert de marchandises entre véhicules, il est possible qu'une marchandise soit livrée par un véhicule qui ne l'a pas chargé en amont. Au moins deux raisons importantes peuvent justifier la

5. Cette définition est issue du PETIT LAROUSSE

pratique du transbordement : une panne du véhicule ayant collecté le produit peut nécessiter le transfert des marchandises qu'il transporte vers un autre véhicule. Une autre justification peut être d'ordre économique voire écologique. En effet, dans une logique de planification des tournées de véhicules, le transbordement n'est autorisé que si cela permet de réduire le nombre de véhicules utilisés ou la distance parcourue. Dans ce cas, le transbordement est effectué de manière opportuniste. Le transbordement peut aussi être intéressant dans la réduction des kilomètres à vide.

Bien que pratiqué de manière plus importante par des compagnies de transport de courriers ou de colis, le transbordement peine à se développer. À notre niveau de connaissance de la littérature, quelques raisons peuvent justifier ce fait : le coût de manutention lié au transfert des produits et la nécessité d'un outil d'aide à la décision pour aider les décisionnaires. Le coût de manutention est élevé dans la mesure où contrairement au cross-docking, les produits ne sont pas obligatoirement transférés en totalité ou en masse au point de transbordement. Il faut, localiser, décharger et recharger le bon produit, il faut ensuite aligner ou mettre à jour le système d'informations, etc. Sur le plan décisionnel cela peut poser des décisions de gestions aux décisionnaires qui doivent choisir parmi toutes les marchandises, lesquelles transborder, à quelle date, et avec quels véhicules.

#### 1.4.5 Le TMS

Un type de logiciel de gestion de transport (Brewer2001) de plus en plus utilisé par les entreprises et qui repose sur l'utilisation des NTIC est le TMS (Transport Management System).

Les TMS ou systèmes de gestion de transport sont des logiciels d'aide à la décision qui permettent au décisionnaire, de gérer le transport. Dans le détail, les TMS supportent de nombreux aspects liés à la gestion du transport : activité d'expéditions, document de travail, étiquetage des produits à transporter, facturation, etc. Les champs d'activités couverts par un TMS sont très variables et dépendent de la compagnie qui l'acquiert. Certains champs peuvent être optionnels et certaines compagnies peuvent acheter plusieurs TMS pour assurer la gestion d'un ensemble d'activités.

Le TMS apporte une réponse à la plupart des questions posées par la gestion du transport suivant le niveau décisionnel considéré : stratégique, tactique, opérationnel, et temps réel. L'ensemble de ces questions apparaît dans la figure 1.8 du 1.3.2. Ainsi, sur le plan stratégique, le TMS aidera le décisionnaire à faire des choix dans la conception du schéma optimal de transport, en mettant notamment l'accent sur le type et l'emplacement des plateformes logistiques. Sur le plan tactique le TMS fournit au décisionnaire des outils de simulation pour tester par avance différents plans de transport, le choix de flux : direct ou par cross-docking. Sur le plan opérationnel, il s'agit de l'élaboration des tournées sur le court terme généralement quelques heures ou une journée. À chaque niveau décisionnel, l'ensemble des contraintes portant sur les véhicules et les clients sont prises en compte de manière précise. Sur le plan de l'exécution et du suivi, l'évolution des NTIC permet au TMS de suivre en temps réel l'exécution des tournées et de capturer en quasi temps réel l'occurrence de tous les

événements. Cela permet ainsi de garantir une meilleure traçabilité au niveau du transport.

Un autre avantage non négligeable du TMS est son interaction possible avec le WMC (Warehouse Management System). Le WMC est à l'entrepôt ce que le TMS est au transport. Le WMC présente de nombreuses fonctionnalités qui sont : la gestion générale de l'entrepôt, la gestion des stocks, la réception des articles dans le site d'entreposage, la préparation et les expéditions de commandes. En intégrant le TMS avec le WMC on peut optimiser le taux de remplissage des véhicules. Au niveau du cross-docking, le TMC communique au WMC les produits en cours d'arrivée dans l'entrepôt ce qui permet ainsi d'anticiper certaines actions telles que la préparation des emplacements de stockage ; le WMC peut également informer le TMS du temps nécessaire pour traiter des commandes de manière à ce que celui-ci les intègre dans son plan de routage. L'analyse de l'historique des préparations de commande du WMC peut permettre au TMS, d'anticiper certains besoins de transport à moyen terme.

Toutefois, le TMS du fait de son large spectre d'applications peut être complexe à utiliser. Son coût élevé de l'ordre de plusieurs milliers d'euros à des dizaines de millions d'euros empêche son déploiement et son utilisation dans la plupart des entreprises de transport.

Les analyses précédemment effectuées tant sur la structuration des niveaux de décisions, les pratiques actuelles et les nouvelles tendances dans le transport concourent toutes à améliorer la performance du transport. Nous nous proposons d'affiner cette analyse par la mise en œuvre d'indicateurs permettant la mesure de performance.

## 1.5 LA PERFORMANCE DANS LE SECTEUR TRANSPORT

L'évaluation de la performance du transport est une tâche nécessaire pour améliorer son efficacité et par voie de conséquence celle de la chaîne logistique dans son ensemble. (Shaw 2003) donnent une définition de la mesure de la performance : « *La mesure des performances est un processus consistant à évaluer les progrès accomplis dans des buts définis à l'avance* ». Cette évaluation nécessite dans le domaine du transport de disposer des informations sur : l'efficacité avec laquelle les matières premières sont transformées en service ou produits ; la façon dont les produits sont fournis aux clients et leur degré de satisfaction.

Dans le cadre du transport routier, la mesure de la performance peut être déclinée en plusieurs axes. Sans avoir la prétention de l'exhaustivité, nous en citons quelques uns : économique, environnemental, sécurité, qualité, flux, satisfaction des clients. Ces notions seront finement analysées en section 1.5.2.

Dans la sous section suivante, la performance du transport est présentée dans un contexte économique et social, en insistant plus particulièrement sur les problèmes énergiques et le développement durable. Ce choix est motivé par la tendance actuelle qui au-delà de l'aspect économique classiquement privilégié tend à promouvoir un transport écologique et respectueux de l'environnement, afin de préserver la planète pour les générations futures.

### 1.5.1 Les contraintes économiques et sociales

Au-delà de l'acheminement des produits qui lui incombe, le transport routier pose des problématiques importantes notamment en matière énergétique et environnementale. Il est donc nécessaire de disposer d'un transport performant qui assure une consommation d'énergie moindre et qui garantit le développement durable notamment.

#### 1.5.1.1 Le problème énergétique

Sur le plan énergétique, la diminution des ressources renouvelables et l'instabilité dans les pays producteurs du fait des guerres qui s'y déroulent ne cessent de causer une croissance toujours soutenue du prix du pétrole. Cela entraîne ainsi, des coûts de fonctionnement très élevés pour des véhicules. Cette situation peut devenir insupportable pour certaines entreprises ou de simples usagers. Pour réduire ces coûts, il est nécessaire de disposer d'un transport performant sur le plan énergétique. Cela passe par divers moyens parmi lesquels nous pouvons citer de manière non exhaustive le développement des véhicules moins énergivores, l'apprentissage de l'éco-conduite, l'exploitation des biocarburants.

#### 1.5.1.2 Le développement durable

L'un des effets négatifs du transport est la pollution environnementale. Le souci majeur consiste aujourd'hui en la préservation de l'environnement et des différents écosystèmes. Cela passe par une maîtrise du réchauffement climatique et une limitation des émissions de gaz à effet de serre (GES). Le transport tel qu'il existe aujourd'hui contribue pour une large part aux émissions de GES. En 2009, le transport routier représentait même une part prépondérante au bilan des émissions de GES. La figure 1.10 (source CITEPA<sup>6</sup>, 2010) permet de s'en rendre compte.

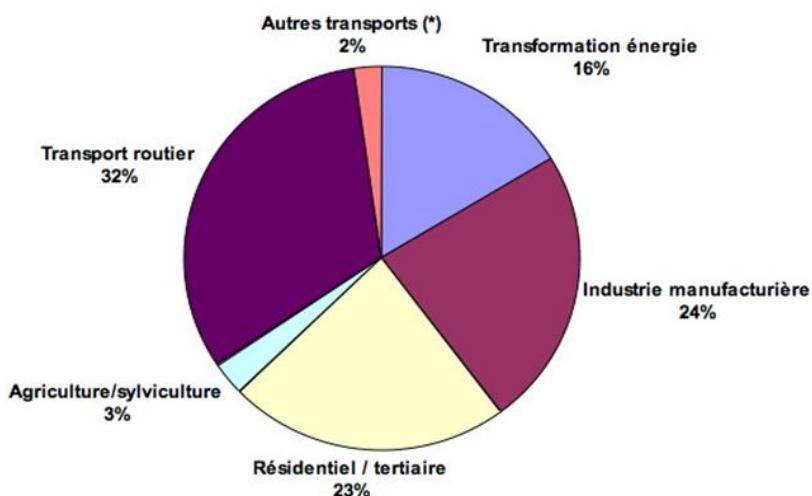


FIGURE 1.10 – Répartition par activité des émissions de CO<sub>2</sub> en France en 2009

6. CITEPA : Centre interprofessionnel technique d'étude et de pollution atmosphériques ([www.citepa.org](http://www.citepa.org))

Dans le cas de la France, comparé à d'autres modes de transport, et aux autres secteurs industriels, le seul secteur dont les émissions de GES progressent est le transport. Toutefois, la contraction de l'activité économique depuis 2008 a causé le ralentissement de la consommation des ménages et par conséquent de la demande du transport. Cela a eu pour conséquence une légère stabilisation des émissions de GES.

Pour mesurer précisément la performance du transport et le piloter de manière optimale afin d'atteindre un ensemble d'objectifs en matière de consommation énergétique, de développement durable et de gains économiques, il est nécessaire de disposer d'un ensemble d'indicateurs de performance. D'après OCDE<sup>7</sup> (OCDE2001), un indicateur de performance permet de :

- Mesurer l'efficacité d'une opération ou d'une organisation ;
- Juger ou d'évaluer un résultat obtenu par rapport à un objectif fixé.

Les indicateurs de performance sont généralement définis, structurés et catégorisés de manière finie sur la base d'un ensemble de propriétés souvent génériques appelés leviers de performance. Les leviers de performance offrent la possibilité de lier au sein de la chaîne logistique les performances individuelles et collectives des différents acteurs dans l'optique d'atteindre une performance globale.

### 1.5.2 Leviers et indicateurs de performance

Les leviers de performance sont des moyens d'action qui permettent d'atteindre une performance et sont généralement basés sur les notions suivantes : coût, efficience, compétitivité, fiabilité, flexibilité, productivité, qualité de service, rentabilité, profitabilité. Il n'existe pas de définition précise et communément acceptée de ces notions. Chaque référentiel d'évaluation de la performance existant essaye d'apporter une définition plus ou moins précise des leviers qu'il préconise. Il existe plusieurs référentiels d'évaluation de la performance dans le domaine de la gestion de la chaîne logistique dont : SCOR (Supply Chain Operation Reference)(Cazier2007), TBP (Tableau de bord prospectif – Balanced Score Card)(Kaplan2003), APICS (Association for Operation Management). Le tableau 1.1 permet de présenter les leviers de performance considérés par chacun des référentiels.

Référentiels	Leviers de performance
SCOR	Fiabilité, flexibilité, réactivité, coût, actif
TBP	Niveau de service client, coût, organisation et processus interne, innovation, croissance
APICS	Délai, coût, niveau de service, agilité, maîtrise des risques, développement durable

TABLE 1.1 – Quelques référentiels permettant l'évaluation de la performance

Dans le cadre du projet PRODIGE (Programme ANR - VTT 2009), l'analyse des différents référentiels et des discussions avec des acteurs métiers du transport (GT-SA, TLF) ont permis de conclure que les axes de performance préconisés par chacun de ses référentiels n'étaient pas à

7. Organisation de coopération et de développement économiques

euX seuls suffisants pour complètement évaluer la performance du transport dans la chaîne logistique. Pour cette raison, en vue de développer un référentiel plus complet et proposer un ensemble d'indicateurs précis permettant de mesurer la performance du transport les leviers suivants ont été préconisés :

- Coût / Efficience : c'est le rapport entre l'effort et les moyens totaux déployés dans une activité, d'une part, et l'utilité réelle que les utilisateurs en tirent sous forme de valeur d'usages d'autre part. C'est le degré d'atteinte des objectifs fixés à moindre coût.
- Fiabilité et taux de service client : c'est la capacité de l'entreprise à livrer des commandes parfaites, conformes à l'attente du client.
- Flexibilité : c'est la capacité de l'entreprise à s'adapter à son environnement. Elle constitue le mode de réponse à l'instabilité et à l'imprévisibilité de l'environnement.
- Réactivité et agilité : la réactivité concerne la capacité de l'entreprise à réagir de manière prompte aux aléas qu'ils soient d'origines internes ou externes. Les pannes internes concernent les machines, les accidents. Les pannes externes concernent par exemple : une rupture d'approvisionnement, la modification ou l'apparition d'une nouvelle commande.
- Maîtrise des risques et résilience : la gestion du risque est devenue une préoccupation majeure pour toutes les entreprises compte-tenu de l'instabilité de l'environnement. Le risque est au cœur même de toutes les décisions que l'entreprise doit entreprendre qu'elles soient stratégiques, opérationnelles ou fonctionnelles. La gestion du risque constitue un facteur déterminant de la performance de toute entreprise. Celle-ci est confrontée à différents risques, nous pouvons citer : aléa de la demande, vol, changement de température et interruption de la chaîne de froid. La résilience peut être définie par la capacité de l'entreprise à affronter avec succès les risques et revenir à son état initial après une situation inopinée.
- Démarche éco-logistique et développement durable : le développement durable s'appuie sur trois dimensions : environnementale, sociale et économique. Dans notre contexte, nous parlons de chaîne logistique verte qui est définie comme la démarche visant à minimiser l'empreinte écologique d'un produit tout au long de son cycle de vie, c'est-à-dire à travers la conception, les achats, la production, la logistique et le recyclage. La chaîne logistique durable regroupe les flux des matières premières et des services allant des fournisseurs aux clients en passant par les industriels ainsi que les flux de leur retour en améliorant les impacts sociaux, environnementaux et économiques.

Sur la base des leviers précédents, une tentative est effectuée pour proposer un référentiel de performances pour le transport routier.

### 1.5.3 Vers un référentiel de performances pour le transport routier

Afin de mesurer la satisfaction de l'ensemble des objectifs fixés dans le transport routier, un ensemble d'indicateurs de performance est proposé. Ces indicateurs peuvent être intégrés dans les outils d'aides à la décision

Notations	Définitions	Notations	Définitions
$T_v^{panne}$	Temps durant lequel le véhicule v est en panne	$Q^{delayed}$	Nombre de livraisons en retard
$T_v^{congestion}$	Temps durant lequel le véhicule v est arrêté pour cause de congestion (trafic, bouchon)	$Q^{total}$	Nombre total de livraisons
$K_v^c$	Nombre de kilomètres parcourus par un véhicule v alors qu'il transporte une charge c	$S_c$	Volume du colis c
$N_{v,i}^p$	Nombre de palettes p transportées dans un véhicule v durant l'activité i	$S_v$	Volume maximum en charge du véhicule v

TABLE 1.2 – Quelques indicateurs élémentaires

dans le but d'aider les décisionnaires à piloter de manière efficace le système de transport et nous nous en servirons dans les chapitres suivants de la thèse pour définir les objectifs des modèles d'optimisation. Les indicateurs qui sont présentés dans cette partie sont quantitatifs c'est-à-dire qu'ils représentent une mesure de grandeur physique. Il existe aussi des indicateurs qualitatifs qui représentent une appréciation sur une échelle de valeur subjective.

Les indicateurs quantitatifs peuvent être élémentaires ou composites. Ils sont élémentaires lorsqu'ils sont directement mesurables par le biais d'un capteur par exemple. Les indicateurs composites sont issus d'opérations arithmétiques simples qui permettent d'agrégner les indicateurs élémentaires en indicateurs plus complexes. Cette agrégation peut être nécessaire afin de suivre l'évolution de l'organisation du système de transport, pour être en mesure de toujours disposer d'une information simple et compréhensible.

Les différents indicateurs proposés sont issus du livrable « Livrable T1.3 » du projet PRODIGE<sup>8</sup>. Ils sont présentés sous forme de tableau et structuré en deux groupes qui sont les indicateurs élémentaires et composites. Le nombre d'indicateurs étant relativement élevé, nous présentons ci-dessous une sélection des indicateurs en rapport avec les objectifs des modèles décisionnels présentés dans la suite de la thèse.

### 1.5.3.1 Indicateurs élémentaires

L'ensemble des indicateurs élémentaires retenus est présenté dans le tableau 1.2. Pour décrire les indicateurs, les indices suivants sont utilisés :

- « i » – activité ;
- « v » – véhicule,
- « c » – charge ;
- « ch » – chauffeur ;
- « e » – employeur ;
- « p » – palette ;
- « f » – flotte

Dans ce tableau, les indicateurs élémentaires présentés permettent de couvrir et suivre une large palette d'activités et d'acteurs durant l'exécu-

8. Projet PRODIGE : [www.anr-prodigé.com](http://www.anr-prodigé.com)

tion du processus de transport. Leur observation permet ainsi de juger de la qualité de service chez un client, la gestion du personnel notamment des chauffeurs en tenant compte des temps de repos légaux de ceux-ci.

Pour assurer une gestion fine, et globale du transport, les indicateurs précédents peuvent être combinés afin de rendre compte de certaines réalités observables. La combinaison des indicateurs est effectuée sur la base des différents leviers de performance présentés plus haut. Les indicateurs composites issus de ces différentes combinaisons sont présentés dans les tableaux suivants :

#### 1.5.3.2 Indicateurs composites de productivité

Une sélection de quelques indicateurs composites de productivité est présentée dans le tableau 1.3.

Dénominations de l'indicateur	Notations	Définitions
Temps d'occupation d'un véhicule	$T_v^c$	Durée de l'ensemble des activités pendant lesquelles un véhicule v supporte une charge partielle ou complète c, pour une période de temps donnée (ex : 24 heures)
Temps d'utilisation d'un véhicule	$D_v^c$	Durée cumulée de l'ensemble des activités correspondant à un déplacement effectif de charges transportées c par un véhicule v d'un chargeur vers un client, exclusion faite de tout arrêt technique, légal ou autres, pour une période donnée (ex : 24 heures)
Taux(moyen) de rendement horaire d'un véhicule	$\bar{R}_v^{productivity}$	Ratio entre temps d'utilisation d'un véhicule v et temps d'occupation du véhicule pour une période donné (ex : 24 heures)
Taux de rendement horaire de la flotte de véhicule	$\bar{R}_f^{productivity}$	Pourcentage d'utilisation d'un ensemble de véhicules appartenant à la même flotte f rapporté sur une période de temps donnée (ex : 24 heures)
Taux de kilométrage à vide	$\bar{R}_v^{empty}$	Nombre de kilomètres parcourus par un véhicule v sans aucune marchandise à bord
Taux de kilométrage à vide	$\bar{R}_f^{empty}$	Nombre (moyen) de kilomètres parcourus par une flotte de véhicules f sans aucune marchandise à bord
Temps de chargement – déchargement rapporté à la flotte de transport	$T_f^{(un)load}$	Durée cumulée de l'ensemble des activités de chargement et décharge à l'échelle d'une flotte f de véhicules de transport, pour une période de temps donnée (ex : 24 heures)
Nombre de véhicules utilisés pour exécuter une activité de transport	$ K $	Nombre de véhicules de la flotte f effectivement utilisés pour satisfaire l'ensemble des demandes de transport R.

TABLE 1.3 – Indicateurs de productivité

Les indicateurs de productivité présentés dans le tableau ci-dessus ont vocation à aider le décisionnaire à évaluer le rendement de l'activité du transport. Pour mesurer le rendement d'une activité, il suffit de faire le rapport entre le résultat obtenu pour cette activité et les intrants utilisés

pour réaliser la dite activité. Par exemple, dans un problème de transport de personnes, la qualité de service associée à un client peut être mesurée comme le rapport entre le temps de trajet effectif qui représente la durée passée par l'usager dans le véhicule et le temps de trajet pour aller directement de son point de départ à son point d'arrivée.

#### 1.5.3.3 Indicateurs composites de charge

Les indicateurs de charge sont présentés dans le tableau 1.4.

Dénominations de l'indicateur	Notations	Définitions
Nombre (moyen) de palettes transportées par un véhicule	$\bar{N}^p$	Nombre de palettes p transportées dans un véhicule calculé sous la forme : – D'une valeur moyenne – D'une moyenne pondérée par le nombre de kilomètres parcourus au cours de la tournée
Nombre (moyen) de charges transportés par palette	$\bar{N}^c$	Moyenne du nombre de charges c par palette transportée
Taux de remplissage pondéral d'un véhicule	$L_v^w$	Taux de remplissage d'un véhicule v en raisonnant sur une logique de masse w – Moyenne pondérée par le nombre de kilomètres parcourus au cours de la tournée

TABLE 1.4 – *Indicateurs de charge*

Le coût élevé de l'énergie oblige les entreprises de transport à faire des économies. Cela passe par l'utilisation rationnelle et optimale des véhicules disponibles, notamment en ce qui concerne leurs capacités. Les indicateurs de charge offrent en cela un outil précieux pour aiguiller le décisionnaire dans ses choix.

#### 1.5.3.4 Indicateurs éco-logistique

Les indicateurs éco-logistique sont présentés dans le tableau 1.5.

Dénominations de l'indicateur	Notations	Définition
Tonnes de CO <sub>2</sub> rejeté	$E^{GES}$	Quantité de CO <sub>2</sub> émit par SKU (unité de gestion de stock) pendant le transport/ flotte de véhicules
Taux de formation des chauffeurs à l'éco-conduite	$R^{GES}$	Nombre de chauffeurs formés à l'éco-conduite sur une flotte donnée par rapport au nombre total de chauffeurs
Taux de congestion	$R^{Traffic}$	Temps d'immobilisation d'un véhicule à cause de la congestion (bouchon) par rapport au temps d'utilisation du véhicule

TABLE 1.5 – *Indicateurs éco-logistique*

Les indicateurs présentés ci-dessus permettent de suivre l'évolution de la situation du transport dans une démarche écologique.

### 1.5.3.5 Indicateurs de qualité de service

Les indicateurs de qualité de service sont présentés dans le tableau 1.6.

Dénominations de l'indicateur	Notations	Définition
Taux de respect des horaires de rendez-vous	$R^{temps}$	Mesure le nombre de visites effectuées en respectant la date de rendez-vous initialement fixé par rapport au nombre totale de visite
Taux de pannes	$R_v^{panne}$	Mesure le taux d'immobilisation d'un véhicule à cause de pannes (nombre et durée) par rapport au temps d'utilisation du véhicule
Taux de pannes d'une flotte	$R^{panne}$	Mesure le taux d'immobilisation d'une flotte de véhicules à cause de pannes (nombre et durée) par rapport au temps d'utilisation de cette flotte
Taux d'avaries	$R^{accident}$	Taux d'accidents au niveau du chargement (ex : renversement de palettes) par rapport au nombre de produits transportés
Taux de démarques	$R^{perde}$	Nombre de produits perdus par rapport au nombre de produits transportés : <ul style="list-style-type: none"> <li>– Par véhicule</li> <li>– Pour la flotte</li> </ul>
Taux de litiges	$R^{dispute}$	Nombre de litiges rapportés au nombre total de livraisons <ul style="list-style-type: none"> <li>– Par véhicule</li> <li>– Pour la flotte</li> </ul>
Ratio de maintenance des véhicules	$R^{maintenance}$	Nombre d'heures dédiées à la maintenance des véhicules de la flotte par rapport au temps de disponibilité des véhicules

TABLE 1.6 – *Indicateurs de qualité de service*

Dans un environnement fort concurrentiel, la qualité de service doit être garantie pour tous clients du transport. Les indicateurs cités dans le tableau 1.6 permettent d'affiner les décisions prises dans l'objectif de garantir des niveaux de services de qualité chez les clients.

Les travaux menés dans cette thèse contribueront à développer des méthodes permettant d'améliorer un certain nombre d'indicateurs parmi ceux cités dans les tableaux précédents. La liste des indicateurs spécifiquement visés par nos travaux est récapitulée dans le tableau 1.7.

Dénominations de l'indicateur	Notations
Nombre de véhicules utilisés pour exécuter une activité de transport	$ K $
Nombre de kilomètres parcourus par un véhicule v alors qu'il transporte une charge c	$K_c^v$
Taux de respect des horaires de rendez-vous	$R^{temps}$

TABLE 1.7 – *Synthèse des indicateurs utilisés dans le cadre de la thèse*

## 1.6 CONCLUSION

Dans ce chapitre, le transport routier de marchandises a été présenté à travers un ensemble d'éléments qui le caractérise. Les axes qui ont été identifiés et pris en compte pour son analyse sont : le poids économique du transport routier de marchandises, les différents segments de métiers qui le constituent, quelques problématiques et son évolution actuelle. De cette analyse, il apparaît que l'organisation et la gestion du transport est particulièrement complexe.

Dans un tel contexte, nos travaux ont pour objectif de contribuer à l'amélioration du processus décisionnel dans la gestion du transport. Cette contribution passe par la proposition d'un ensemble d'algorithmes destinés à être intégrés dans un outil d'aide à la décision. Cela permettra d'aider les décisionnaires dans les choix difficiles et souvent antagonistes qu'ils ont à faire.

Afin de situer clairement notre contribution, nous la positionnons par rapport à plusieurs axes en répondant aux questions suivantes : Quelle est l'offre de transport ciblée ? Quel est le niveau décisionnel concerné ? Quelle tendance récente est intégrée dans cette contribution ?

- Au sens de la typologie de l'offre de transport annoncée plus haut, la présente thèse se situe au niveau de la messagerie couplée avec des logiques de groupage/dégroupage à échelle régionale, plus particulièrement dans le transport concernant les derniers kilomètres en ville. De façon plus précise, il sera question d'étudier un problème de collecte et livraison avec fenêtre de temps et transbordement. L'analyse de ce problème sera effectuée d'un point de vue uniquement déterministe avec une variante statique et une autre dite à tournées reconfigurables. À notre niveau de connaissance de la littérature, et le chapitre 2 axé sur l'état de l'art permettra de conforter davantage cette position, il existe très peu de travaux scientifiques qui traitent du transbordement et nous ne connaissons pas de logiciel d'aide à la décision qui intègre cette problématique avec les hypothèses qui seront formalisées dès le chapitre 3.
- Les travaux développés dans la thèse concernent deux niveaux de décision : opérationnel et suivi d'exécution. Sur le plan opérationnel, il s'agit de l'élaboration des tournées sur le court terme généralement quelques heures ou une journée avec prise en compte des contraintes portant sur les véhicules, des clients mais aussi liées à l'utilisation du transbordement comme levier d'optimisation. Sur le plan du « suivi d'exécution », il s'agira de suivre les tournées planifiées et en cours d'exécution pour les reconfigurer en cas d'alea conduisant à la détérioration d'une performance attendue.
- La nouvelle approche exploitée dans la thèse consiste à tirer partie du développement des NTIC pour instrumenter le transport afin d'améliorer la traçabilité et indirectement le développement durable via le calcul de l'empreinte carbone de chaque produit. Aujourd'hui, une grande majorité de logiciels de transport dont le TMS garantissent de façon efficace un suivi et une traçabilité dans le transport. Cette traçabilité est généralement axée sur la ressource de transport c'est-à-dire les camions. La faiblesse de cette forme de traçabilité ré-

side dans le fait que les suivis des produits transportés sont effectués de manière discrète en des points de ruptures de charge tels que les entrepôts, les péages ou les clients. Pour combler cette lacune, tirant parti de l'évolution des NTIC, nous faisons le postulat que l'on dispose d'un système de traçabilité reposant sur l'instrumentation du transport en général et du produit en particulier, afin de garantir pour ce dernier une traçabilité permanente, et continue.

# CLASSIFICATION DES PROBLÈMES DE COLLECTE ET DE LIVRAISON ET ÉTAT DE L'ART

2

## SOMMAIRE

2.1	INTRODUCTION . . . . .	31
2.2	TYPOLOGIE DES PROBLÈMES DE ROUTAGE . . . . .	31
2.2.1	Caractéristiques générales d'un problème de tournées de véhicules . . . . .	32
2.2.2	Caractéristiques propres au problème de collecte et livraison	34
2.2.3	Grille d'analyse et positionnement du problème étudié . . . . .	37
2.3	MÉTHODES DE RÉSOLUTION . . . . .	39
2.3.1	Méthodes exactes . . . . .	40
2.3.2	Méthodes approchées . . . . .	41
2.4	PROBLÈME DE COLLECTE ET LIVRAISON AVEC FENÊTRE DE TEMPS	48
2.4.1	Formulation du PDPTW . . . . .	48
2.4.2	Modélisation en variables mixtes du PDPTW . . . . .	49
2.4.3	Présentations et analyses des travaux sur le problème de collecte et livraison . . . . .	51
2.5	SYNTHÈSE DES TRAVAUX SUR LES PROBLÈMES AVEC TRANSBORDEMENT . . . . .	58
2.5.1	Problème de tournées de véhicules avec cross-dock (VRPCD)	59
2.5.2	Résolution du problème de collecte et livraison avec transbordement . . . . .	64
2.6	CONCLUSION . . . . .	70



## 2.1 INTRODUCTION

Le transport routier de marchandises a été analysé au chapitre précédent de manière générique en mettant l'accent sur son rôle, ses fonctionnalités, son organisation et son positionnement au sein de la chaîne logistique. Dans ce chapitre, le discours se veut plus précis et un état de l'art sur les problèmes de collecte et livraison en insistant plus particulièrement sur les problèmes avec transbordement est effectué. Pour réaliser cet objectif, une typologie des problèmes de routage est présentée en deuxième partie. Les principales classes de méthodes de résolution sont analysées en partie trois en mettant un accent particulier sur les méthodes de résolution approchées. Cela permettra de fournir les bases nécessaires à la compréhension de la recherche à voisinages variables utilisée pour résoudre le problème de transport abordé dans les chapitres suivants de cette thèse. En partie quatre, le problème de collecte et livraison avec fenêtre de temps est présenté. L'intérêt de ce problème vient du fait qu'il est un cas particulier du problème étudié dans la présente thèse. La cinquième partie présente les problèmes de routage de véhicules dans lesquels le transfert de charge entre véhicules est autorisé. Une analyse plus fine de différentes contributions de la littérature est effectuée pour les problèmes de collecte et livraison avec fenêtre de temps et transbordement. Le chapitre se termine par une conclusion dans la sixième partie.

## 2.2 TYPOLOGIE DES PROBLÈMES DE ROUTAGE

Les problèmes de tournées de véhicules, ou problèmes de routage, sont des problèmes traités depuis longtemps en recherche opérationnelle (Dantzig 1959). Dans le cas général, il s'agit de déterminer la tournée optimale d'un ensemble de véhicules afin qu'ils puissent satisfaire des ordres de transport. La tournée d'un véhicule est une liste ordonnée de clients à visiter en définissant pour chaque client l'horaire de passage.

La détermination de l'ordre de visite des clients vise la plupart du temps à minimiser les coûts d'exploitation de la flotte de véhicules considérée. La qualité d'une tournée se mesure classiquement :

- Par une distance parcourue : le coût d'exploitation d'une ressource de transport est souvent jugé proportionnel à la distance parcourue ; cette dernière peut également contribuer à estimer la consommation énergétique, ou encore l'usure du véhicule.
- Par le nombre de véhicules permettant de couvrir l'ensemble des ordres de transport. Réduire le nombre de véhicules exploités, c'est réduire le nombre d'actifs pour une entreprise de transport et donc diminuer les coûts d'exploitation.

L'optimisation peut également se faire au sens de la maximisation d'un profit, se déclinant par exemple en :

- Une maximisation du nombre de clients visités par véhicule – ainsi la rentabilité du véhicule est améliorée et sa durée d'amortissement réduite.
- Une maximisation du nombre de clients servis. Cela revient à minimiser le nombre de clients non servis sur une période de temps

donnée, qui peut donner lieu à des pénalités financières proportionnelles aux retards de livraison constatés.

De multiples contraintes viennent également limiter l'espace de décision dans lequel la solution d'un problème de tournées doit être trouvée. Citons par exemple :

- Celles inhérentes à la nature de l'activité chez le client, impliquant la nécessaire précédence du chargement dans un véhicule avant son déchargement chez le destinataire.
- Les contraintes relatives aux nombres de visites autorisées chez un client donné. En effet, des clients peuvent par exemple autoriser des visites multiples ou uniques.

De manière générale, de nombreux paramètres influent sur la complexité du problème de routage, et donc sur sa modélisation et sa résolution. Dans la suite de ce paragraphe, nous mettons en exergue, sans exhaustivité, les principaux éléments pouvant induire un nombre plus ou moins important de contraintes à intégrer dans la représentation du problème à traiter.

### **2.2.1 Caractéristiques générales d'un problème de tournées de véhicules**

Classiquement, la résolution d'un problème de tournées de véhicules nécessite de prendre en compte des informations liées à la nature des véhicules, des clients, ainsi que la législation en vigueur.

#### **2.2.1.1 Produits**

La diversité des produits transportés peut impacter la combinatoire de certains problèmes d'élaboration de tournées. Si cette diversité n'a pas dans le cas général d'incidence sur les problèmes classiques de tournées (Vehicle Routing Problems – VRP), elle peut influencer la combinatoire des problèmes de collecte et livraison (Pickup and Delivery Problems – PDP). En effet, soit un problème de transport dans lequel seulement un ensemble de fournisseurs spécifiques peut livrer un produit donné. La satisfaction d'une demande de transport liée à ce produit nécessite de choisir parmi tous les fournisseurs celui ou ceux à même de permettre la livraison du produit à moindre coût.

#### **2.2.1.2 Véhicules**

Un problème de tournée de véhicules peut être appréhendé sous l'angle de la gestion de flux, dont une dimension importante concerne les ressources en charge d'exécuter les activités, ne serait-ce que par la notion de capacité qui caractérise l'aptitude des véhicules à effectuer une ou plusieurs opérations en parallèle. En effet de manière générale, tout flux est ralenti par les activités mobilisant les ressources et consommant du temps. Dimensionner les moyens ou exploiter au mieux le potentiel capacitaire de ceux-ci permet donc de tendre vers le meilleur niveau de performances. Le problème de transport ne déroge pas à cette règle, et la complexité des problèmes à traiter est fortement impactée par les caracté-

ristiques propres à la flotte de véhicules support à l'activité de transport. Ainsi, ces principales caractéristiques sont :

- Le nombre de véhicules peut être limité ou illimité : lorsque le transport est effectué pour compte propre, l'entreprise possède généralement un parc de véhicules dont la taille est limitée au strict minimum pour réduire les actifs. Avec l'émergence de l'externalisation, les entreprises sous-traitent le transport de leurs produits à des sociétés spécialisées ou louent occasionnellement des véhicules si besoin. La flexibilité qu'offre le recours à un nombre adapté de ressources au besoin de transport peut être assimilée à la gestion d'une flotte illimitée de véhicules.
- Les véhicules sont considérés comme étant à capacité limitée ou illimitée : chaque véhicule a une capacité définissant la quantité maximale transportable exprimée en poids et en volume. Certains travaux de recherche portant notamment sur le transport du courrier postal considèrent cependant la contrainte de capacité comme négligeable au regard du volume transporté, et assimilent un véhicule à une ressource de capacité illimitée.
- L'ensemble des véhicules peut être homogène ou hétérogène : une flotte est déclarée homogène lorsque tous les véhicules constituant celle-ci ont exactement les mêmes caractéristiques. Dans le cas contraire, la flotte est dite hétérogène. L'élément principal considéré dans la caractérisation de l'hétérogénéité d'une flotte est la capacité de chaque véhicule. La capacité correspond au poids et au volume maximum transportable par un véhicule.
- Les véhicules peuvent être compartimentés : un véhicule compartimenté est un véhicule dont l'espace de stockage des produits est divisé en plusieurs compartiments. Chaque compartiment permet de recevoir des produits dont les contraintes de stockages durant un transport sont identiques. Ces types de véhicules sont souvent exploités pour le transport à température dirigée. Les différents compartiments permettent de déplacer à l'aide d'un seul véhicule des produits secs, des produits frais, et des produits congelés ou surgelés.

Du point de vue de la modélisation, le véhicule peut être assimilé à une ressource dont certaines caractéristiques sont variables au cours du temps. Par exemple, les caractéristiques telles que la disponibilité du véhicule et sa localisation sont variables, tandis que sa capacité maximale de stockage est figée.

#### 2.2.1.3 Clients

Les entreprises clientes d'un service de transport ont leurs propres contraintes liées à la livraison de leur production chez leurs clients ou aux besoins de réapprovisionnement de leurs stocks de matières premières ; ces contraintes peuvent se caractériser par un ensemble de données influençant l'ordre dans lequel ils seront visités. Parmi les caractéristiques les plus influentes sur la difficulté du problème, citons :

- La définition de fenêtres de temps : la visite d'un véhicule chez un client peut être soumise ou non au respect d'un rendez-vous. Ce

dernier est modélisé par une fenêtre de temps sous la forme d'un intervalle de temps au sein duquel la livraison doit s'effectuer. Le respect de la fenêtre de temps associée à un client peut être strict et représenter l'impossibilité de visiter le client en dehors d'un intervalle de temps bien défini, ou souple lorsqu'un retard est accepté mais pénalisé de manière proportionnelle à la dérive de temps observée. L'existence de fenêtres de temps constraint considérablement la résolution du problème en excluant certaines séquences de visite ne respectant pas les contraintes de temps.

- La nature du service demandé : l'objet même d'un problème de tournées de véhicules consiste à effectuer à coût optimal un service de transport pour un client. Ce service peut consister en une collecte et/ou en une livraison de produits. La nature du service permet souvent de différencier ou hiérarchiser les problèmes de tournées de véhicules. En fonction des méthodes et moyens mis en œuvre par une entreprise pour organiser et gérer le flux de ses produits, des situations plus complexes peuvent survenir. Ainsi, la prise en compte de stocks géographiquement distribués peut nécessiter une collecte chez de multiples fournisseurs avant de livrer la totalité de la charge au destinataire. C'est par exemple le cas pour le transport en vrac. Dans d'autres problèmes, des clients sont divisés en deux catégories. La première catégorie concerne les clients dont le seul service possible est la collecte, et la seconde concerne les clients pour lesquels le seul service autorisé est la livraison. Dans la suite de ce travail, des caractéristiques plus détaillées et propres à des problèmes plus généraux de collecte et livraison sont présentées.

### **2.2.2 Caractéristiques propres au problème de collecte et livraison**

Plusieurs auteurs se sont intéressés à définir un cadre général pour catégoriser les différents problèmes de tournées de véhicules. Une synthèse est proposée par (Berbeglia2007). Au-delà des éléments précédemment discutés, les auteurs proposent de classer les problèmes de routage et leurs variantes en trois catégories selon les caractéristiques du transport à organiser. Ces caractéristiques sont discutées dans ce qui suit, dans un contexte d'élaboration de tournées sans contraintes de fenêtres de temps, couplant collecte et livraison de marchandises.

#### **2.2.2.1 Problème « one-to-one »**

Les problèmes de type « one-to-one » sont des problèmes de transport bien connus dans lesquels les clients sont structurés en deux ensembles disjoints : le premier ensemble regroupe les clients chez qui des produits doivent être collectés. Le deuxième ensemble regroupe les clients auxquels les produits collectés dans le premier ensemble sont destinés. Les produits collectés chez un client du premier ensemble, sont destinés à un et un seul client du deuxième ensemble. Il existe donc une unicité du couple de clients associé à une requête de transport.

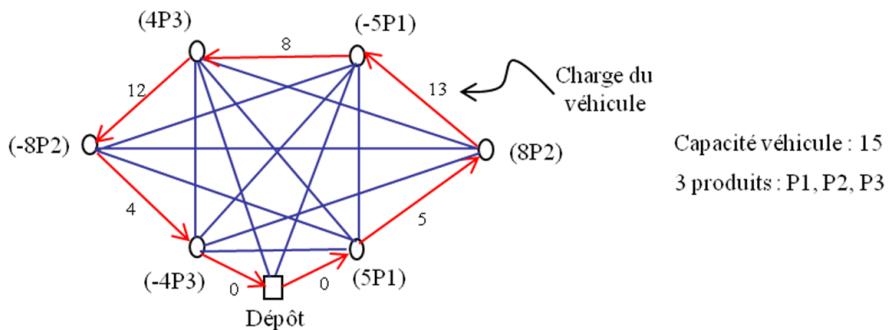
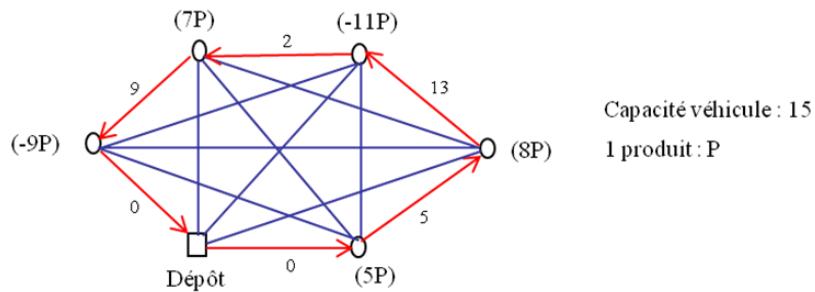


FIGURE 2.1 – Le PDP « one-to-one »

La figure 2.1 illustre le fonctionnement des problèmes de transport de type « one-to-one ». Dans cette figure, la notation  $(xPy)$  représente le label du noeud visité et signifie qu'une quantité de  $x$  unités de produits de type  $P_y$  doit être collectée ou livrée en ce noeud. Ainsi, si  $x$  est négatif il s'agit d'une livraison et dans le cas contraire, il s'agit d'une collecte. Par exemple,  $-5P1$  signifie que 5 unités de produits de type  $P1$  seront livrées au noeud de label  $-5P1$ , tandis que  $4P3$  signifie que 4 unités de produits devront être collectées au noeud de label  $4P3$ . Bien que la vision « multiples produits » ne soit en général pas considérée dans la formulation d'un problème de cette catégorie, l'exemple proposé différencie la nature des charges transportées afin d'en favoriser la compréhension. Le graphe ainsi représenté modélise les clients caractérisés par les sommets, les alternatives de chemins possibles représentées par les successions d'arcs, ainsi que les caractéristiques des demandes de chaque client. La variation de la charge du véhicule induite par un chargement ou déchargement de produit est modélisée par une quantité de produits respectivement positive ou négative associée à chaque sommet du graphe. Ainsi, le véhicule de capacité égale à 15 unités de charge, et dont le trajet est représenté par les arcs en rouge, part du dépôt à vide et visite tous les sommets pour y collecter ou livrer des produits en parcourant une distance qui se veut minimale, si la séquence de visites ou trajets tend à optimiser cette distance parcourue tout en prenant en compte les différentes contraintes du problème. Ces contraintes concernent le respect du couplage entre point de collecte et point de livraison définies par les différentes requêtes de transport à servir ; le respect des capacités. La charge du véhicule représentée par la valeur numérique mentionnée sur les arcs du graphe ne doit jamais excéder la capacité du véhicule.

### 2.2.2.2 Problème « many-to-many »

Les problèmes de type « many-to-many » sont structurés comme les problèmes précédemment présentés. Cependant, la livraison d'une quantité de produit chez un client destinataire peut nécessiter une collecte chez un ensemble de clients fournisseurs ; de manière dual, la charge de produits collectés chez un client fournisseur peut permettre de servir la demande d'un ensemble de clients destinataires.



**FIGURE 2.2 – Le PDP « many-to-many »**

Dans la figure 2.2, le problème « many-to-many » est caractérisé d'une manière comparable à celle utilisée pour la famille de problèmes précédemment présentée en se limitant à considérer un seul type de produit. Cela n'est pas représentatif du problème en soit – un problème « many-to-many » peut être également caractérisé par des requêtes de transport concernant différentes familles de produits. La spécificité principale d'un problème « many-to-many » est la non existence de contraintes d'appariage entre clients représentées par le couple (fournisseur, destinataire). Néanmoins, la charge d'un véhicule doit être suffisante pour servir un client destinataire au moment de sa visite, ce qui nécessite d'avoir collecté préalablement la quantité nécessaire auprès de fournisseurs, tout en respectant la capacité du véhicule. Ainsi, la résolution du problème consiste à définir la trajectoire du véhicule depuis son départ du dépôt jusqu'à son retour, de manière à déterminer la séquence de visites des clients qui respectent l'ensemble des contraintes décrites plus tôt : les arcs du graphe en rouge représentent une telle solution supposant que le trajet ainsi décrit est celui de distance minimale si tel est le choix du critère d'optimisation.

### 2.2.2.3 Problème « one-to-many-to-one »

Les problèmes de type « one-to-many-to-one » sont des problèmes de transport qui se distinguent des deux précédentes familles par les caractéristiques suivantes : les véhicules au départ du dépôt ne partent pas à vide, et la décomposition de l'ensemble des clients en deux sous-ensembles distincts n'est plus considérée : chaque client est en soi un point de collecte et un point de livraison, si bien qu'un véhicule le visitant peut lui livrer une partie de son chargement et accepter de nouvelles charges. Un problème de ce type a notamment été étudié dans (Prive2004) dans le cadre d'un problème de distribution d'eau en bouteille dans la ville de Québec. Des bouteilles vides sont collectées auprès de chaque client visité pour être recyclées au dépôt, avec pour objectif de minimiser le coût du transport et maximiser les bénéfices issus du recyclage des bouteilles vides collectées.

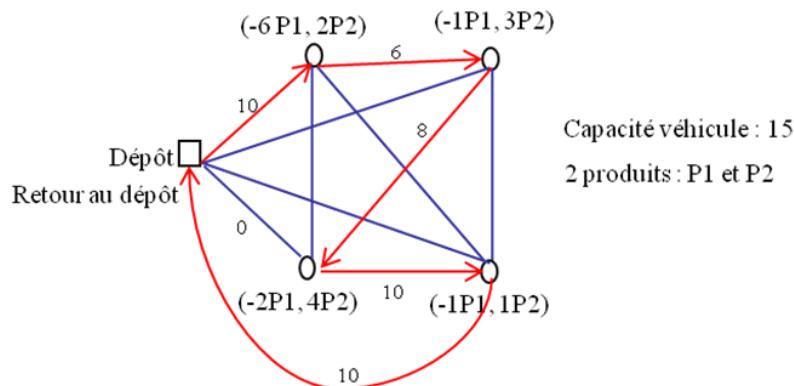


FIGURE 2.3 – Le « one-to-many-to-one »

Dans la figure 2.3, le problème « one-to-many-to-one » est caractérisé d'une manière semblable à celle utilisée pour la famille de problèmes précédemment présentée. Un véhicule de capacité quinze part initialement du dépôt avec dix produits de type  $P_1$ . À chaque sommet visité, le véhicule livre des produits de type  $P_1$ , et collecte une ou plusieurs unités d'un produit de type  $P_2$ . Les produits de type  $P_2$  collectés en chaque sommet traversé sont par la suite retournés au dépôt, si bien que le véhicule rentre au dépôt avec une charge de 10 produits de type  $P_2$ . Il convient de noter que les valeurs numériques associées aux différents arcs définissent la charge du véhicule induite par le transport des différentes familles de produits ; ces valeurs agrégées représentant les quantités de  $P_1$  et  $P_2$  présentes dans le véhicule.

### 2.2.3 Grille d'analyse et positionnement du problème étudié

Plusieurs grilles de référence ont été proposées par différents auteurs pour classer les problèmes de routage (Berbeglia2007)(Parragh1)(Parragh2). Une grille de références est définie dans ce qui suit, afin de catégoriser différents travaux représentatifs des études menées sur le routage de véhicules ; cette grille se veut reprendre des éléments des synthèses existantes dans la littérature et s'appuie sur les différents éléments de classement préalablement présentés.

Travaux étudiés	Produits	Dépôts	Véhicules			Fenêtres de temps	Nature du service	Nombre de visites	Synchronisation	Consolidation	Facultatif	Obligatoire
			Nombre	Capacité	Flotte							
	Uniques	Multiples	Uniques	Multiples	Limitée	Hétérogène	Homogène	Limitee	Unique	Multiples	Sans	Avec

TABLE 2.1 – Caractérisation du problème étudié

Le tableau 2.1 synthétise l'approche de classement retenue et l'état de l'art réalisé dans la suite de ce chapitre s'appuiera sur cette grille d'analyse. Notons que les travaux de cette thèse se plaçant exclusivement dans un contexte de problèmes de type « one-to-one », l'analyse de la littérature se focalisera donc sur cette classe de problèmes.

## 2.3 MÉTHODES DE RÉSOLUTION

La résolution d'un problème général d'optimisation combinatoire P et d'un problème de transport en particulier est souvent décomposée en trois étapes. La première consiste à identifier et à modéliser le problème. En fonction du modèle établi, la seconde consiste à choisir une méthode de résolution et la dernière étape consiste à intégrer ou à mettre en œuvre la solution obtenue dans le processus de décision.

Plusieurs approches de modélisation sont possibles. Dans le cadre de la programmation mathématique, la modélisation est basée sur l'utilisation de variables de décision qui sont soumises à un ensemble de contraintes exprimées sous forme d'équations ou d'inéquations. Les contraintes définissent l'ensemble des solutions possibles de P et, par conséquent l'ensemble des valeurs possibles de chaque variable de décision.

Le problème une fois modélisé peut être résolu de manière exacte ou approchée. La figure 2.4 inspirée de (Nemhauser1988)(Gendreau2010) présente une classification de ces méthodes dont quelques unes seront présentées de manière synthétique.

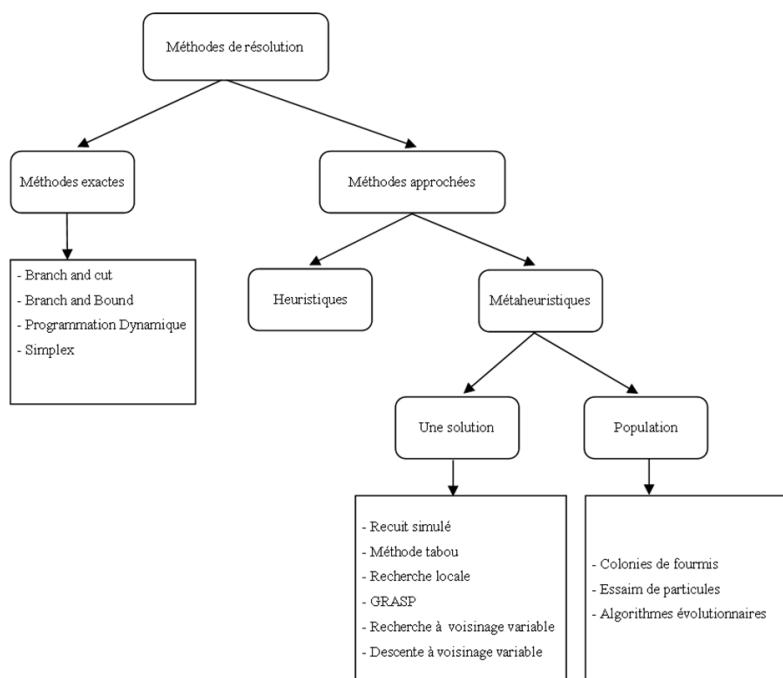


FIGURE 2.4 – Présentation et classification de quelques méthodes de résolution

La présentation des méthodes de résolution débute par celles dites exactes. Ces méthodes sont analysées afin de faciliter la compréhension de la bibliographie et dans la mesure où elles sont souvent implémentées dans la plupart des solveurs qui permettent de résoudre des problèmes combinatoires de manière exacte. Cette dernière approche sera exploitée dès le chapitre 3.

### 2.3.1 Méthodes exactes

Une méthode de résolution exacte est un algorithme qui trouve une ou des solutions optimales pour un problème P donné. Pour trouver une solution exacte à P, il faut agir sur les variables de décision qui caractérisent P afin de leur faire prendre les bonnes valeurs. Dans le cas de contraintes linéaires, si les variables sont continues, P peut être résolu de manière optimale avec l'algorithme bien connu du simplexe ; si les variables sont entières il s'agit de programmation linéaire en nombres entiers. La formalisation proposée au chapitre trois s'appuyant sur la programmation linéaire en nombres entiers, il convient de présenter plus en détail les fondements de cette approche.

La bibliographie dans le domaine de la programmation en nombres entiers est très riche et fait ressortir la difficulté de résoudre de tels problèmes. Deux approches de résolution très utilisées sont ici présentées : les algorithmes de séparation et évaluation et les méthodes de coupes. Rapelons que ces deux méthodes sont utilisées dans la plupart des solveurs de programmes linéaires.

Les algorithmes de séparation et évaluation appelés aussi Branch and Bound (B & B) permettent de résoudre de manière exacte un problème d'optimisation quelconque. Ils ont été utilisés avec succès pour la résolution de problèmes linéaires en nombres entiers. Ces méthodes sont dues principalement à plusieurs auteurs (Dakin1965)(Beale1979). L'idée générale qui sous-tend un algorithme de B & B consiste à éliminer précoce-ment des sous-ensembles de solutions partielles dont on pense qu'elles ne conduisent pas à la solution optimale recherchée. Par ce mécanisme, on arrive souvent à obtenir la solution recherchée en temps raisonnable. Toutefois, dans le pire des cas, le processus d'élimination précoce des solu-tions non optimales peut ne pas être efficace et conduire l'algorithme à fonctionner comme une méthode d'énumération. Un algorithme de sé-pARATION et évaluation est constitué de trois principales composantes : un mécanisme de séparation des solutions, une fonction d'évaluations des solu-tions et une stratégie d'exploration. Une bonne synthèse sur la théorie et sur les applications des algorithmes de séparation et d'évaluation peut être trouvée dans (Lawler1966)(Minoux1983).

Une autre approche pour résoudre les problèmes dont toutes les va-riables sont astreintes à prendre que des valeurs entières sont les méthodes dites de coupes (Still2009)(Stillb2009). Le principe d'une méthode de coupe est le suivant : la relaxation continue du problème est résolue avec un solveur ou par toute autre méthode appropriée. Cette relaxation auto-rise que certaines variables soient continues. Si après résolution, toutes les variables de la solution obtenue sont entières, le problème est résolu. Si dans la solution obtenue, toutes les variables ne sont pas entières, et c'est généralement le cas, l'espace de recherche des solutions est réduit de fa-çon à éliminer la dite solution sans qu'aucune solution à variables entières ne soient exclues. Pour réduire le domaine de solutions, des contraintes supplémentaires sont rajoutées au problème. De telles contraintes sont appelées coupes ou inégalités valides. Il peut être nécessaire de rajou-ter plusieurs inégalités valides pour éliminer une solution non entière ou fractionnaire. Après avoir rajouté une ou plusieurs coupe(s), la relaxation

continue du programme est à nouveau résolue. Si la solution optimale obtenue est à valeurs entières, le problème est résolu, sinon le processus précédent est répété en recherchant de nouvelles coupes à rajouter au problème, dont la relaxation continue sera de nouveau optimisée.

L'association d'un algorithme de coupes et d'une méthode de séparation et évaluation permet d'obtenir un algorithme dit de branchements et coupes (Branch and Cut). Dans un tel algorithme, les nœuds de l'arbre sont évalués grâce à un algorithme de coupes. Pour avoir un aperçu récent des différentes techniques portant sur les algorithmes de branchements et coupes, le lecteur intéressé pourra se référer à (Feillet2010).

Une autre méthode utilisée pour résoudre des problèmes d'optimisation linéaire en nombres entiers est la méthode de Branch and Price. Cette méthode combine l'algorithme de Branch and Bound classique avec une méthode de génération de colonnes (Desaulniers2005).

Les méthodes exposées précédemment ne peuvent pas être appliquées concrètement à des problèmes industriels de grandes tailles, notamment lorsque le problème est modélisé avec des variables en nombres entiers. En effet, le temps de calcul requis « exploserait » du fait de l'exploration de l'ensemble des solutions. Il est donc nécessaire de faire appel aux méthodes approchées dont la présentation fait l'objet de la prochaine sous-section.

### 2.3.2 Méthodes approchées

Une alternative aux méthodes exactes est donc connue sous le paradigme de méthodes approchées. Il s'agit de méthodes permettant d'obtenir rapidement de bonnes solutions - i.e. les plus proches possibles de la solution optimale dans des temps de calculs maîtrisés. Les méthodes approchées sont généralement des heuristiques ou métahéuristiques. Une heuristique est un mécanisme qui permet d'obtenir rapidement des solutions réalisables à un problème donné. Elle est très spécifique au problème pour lequel elle a été construite et ne peut donc être généralisée. Les métahéuristiques sont de puissantes méthodes génériques qui permettent de résoudre plusieurs problèmes d'optimisation. Elles peuvent être classées en fonction du nombre de solutions qu'elles manipulent simultanément.

#### 2.3.2.1 Algorithme glouton

Un algorithme glouton est un algorithme dans lequel à chaque itération la valeur d'une ou plusieurs variables décrivant le problème à résoudre sont fixées. Une variable dont la valeur est fixée l'est de façon définitive et cette valeur ne sera pas remise en cause. Par exemple, le problème du rendu de monnaie consiste à donner une somme avec le moins de pièces possibles. Pour résoudre ce problème, l'algorithme consistant à répéter le choix de la pièce de plus grande valeur qui ne dépasse pas la somme restante est un algorithme glouton. La faiblesse de tels algorithmes réside dans le fait qu'ils peuvent fournir une solution très éloignée de la solution optimale.

### 2.3.2.2 Recherches locales

La recherche locale est une technique utilisée en optimisation combinatoire pour résoudre des problèmes d'optimisation difficiles. Sans nuire à la généralité, la recherche locale est ici présentée dans un contexte de minimisation. De ce fait, il s'agit de trouver une solution qui minimise un critère fixé parmi un ensemble de solutions candidates. Un algorithme de recherche locale débute par une solution initiale. À partir de celle-ci, l'espace de recherche est exploré de manière à trouver une meilleure solution en se déplaçant dans son voisinage par application de mouvements ou transformations.

Un voisinage  $N$  est défini par l'utilisation d'un mouvement  $\mu$ . Le voisinage d'une solution  $S$  noté  $N(S)$  est l'ensemble des solutions qui sont atteignables à partir de  $S$  par application du mouvement  $\mu$  sur  $S$ . Par exemple, dans le cas d'un problème de tournées de véhicules tel que le VRP, une solution est constituée d'une ou plusieurs routes qui peuvent être modélisées par un ensemble de séquences de clients à visiter. Une nouvelle solution peut être obtenue par application de plusieurs mouvements, dont deux possibles sont respectivement illustrés dans la figure 2.5-1 et la figure 2.5-2. Le premier mouvement consiste à permutez deux clients appartenant à deux routes différentes entre ces routes. Le second mouvement consiste à permutez deux clients appartenant à une même route.

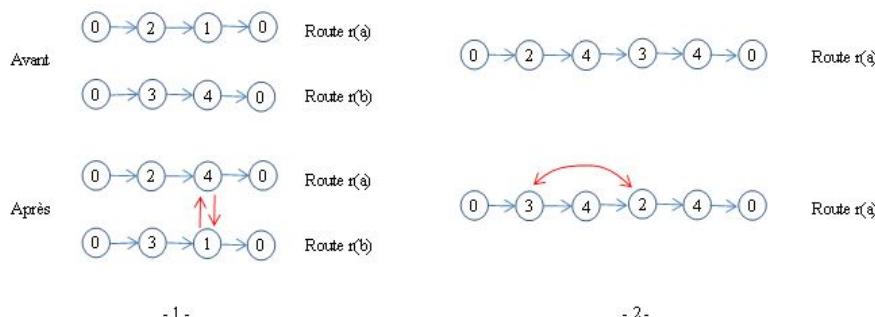


FIGURE 2.5 – Illustration de deux types de mouvements

Ainsi la recherche locale repose sur deux notions principales qui sont :

- La définition d'une solution initiale ;
- L'amélioration de cette solution en explorant dans son voisinage.

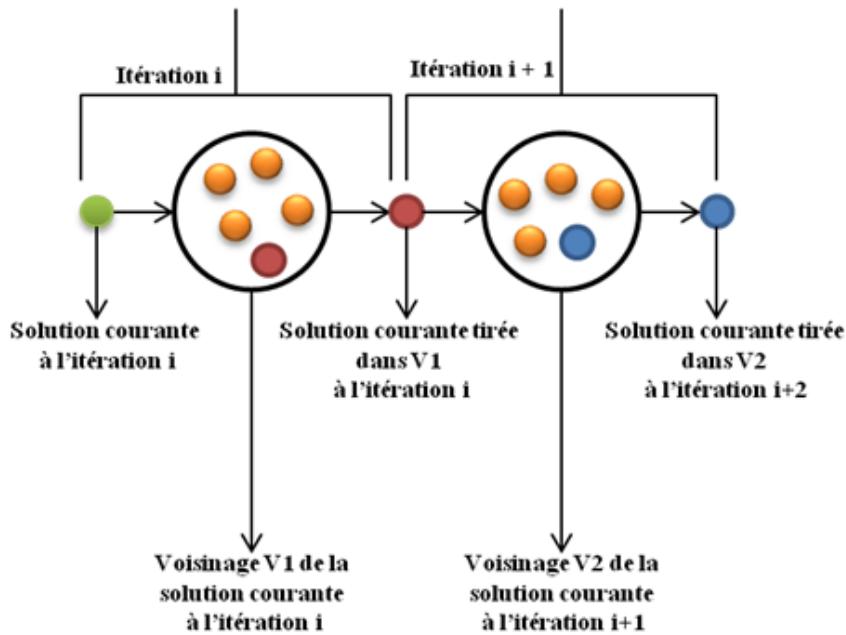


FIGURE 2.6 – Principe de la recherche locale

Dans la figure 2.6, à chaque itération, un ensemble de solutions est générée dans le voisinage de la solution courante par application d'un mouvement donné. Le processus est itérativement répété jusqu'à ce qu'une condition d'arrêt soit vérifiée. L'algorithme 1 permet d'illustrer les principales étapes d'une recherche locale .

---

**Algorithme 1 : LS**


---

**Entrées :** Solution initiale –  $S$

**Sorties :** Optimum local –  $S$

- 1 **tant que** la condition d'arrêt n'est pas vérifiée **faire**
  - 2     $N(S) \leftarrow$  générer des solutions voisines de  $S$ ;
  - 3    choisir une solution  $S_1$  dans  $N(S)$ ;
  - 4     $S \leftarrow S_1$ ;
  - 5 Retourne  $S$ ;
- 

L'observation de cet algorithme permet de soulever les questions qui suivent :

- Quel critère d'arrêt choisir ?
- Comment générer la solution initiale ?
- Comment choisir une solution  $S_1$  dans  $N(S)$  et mettre à jour  $S$  avec  $S_1$  ?

Le critère d'arrêt dépend du problème à résoudre. Il peut s'agir d'un nombre maximal d'itérations, d'un temps de calcul, du nombre maximal d'itérations entre deux améliorations de la solution courante, etc.

La solution initiale est généralement construite en utilisant essentiellement une stratégie aléatoire ou gloutonne. Notons qu'une solution initiale « de bonne qualité » ne conduit pas nécessairement à la solution optimale à l'issue du processus d'optimisation.

Le choix d'une solution  $S_1$  dans  $N(S)$  peut se faire de manière aléa-

toire, ou par l'utilisation de la fonction d'évaluation  $f$ . La vocation de la fonction d'évaluation est de guider la recherche dans l'espace de recherche. Lorsqu'une fonction d'évaluation est utilisée, la solution  $S_1$  choisie dans  $N(S)$  est celle qui vérifie la condition  $f(S_1) < f(S)$ . Plusieurs solutions appartenant à  $N(S)$  peuvent vérifier ce critère. Dans ce cas, deux stratégies permettent de décider du choix de  $S_1$ . Dans la première stratégie appelée stratégie de meilleure amélioration (Best Improvement – BI),  $S_1$  est la meilleure solution trouvée au sens de la fonction d'évaluation en parcourant tous les voisins de  $S$ . La seconde stratégie est dite de première amélioration (First Improvement – FI). Dans FI, les voisins sont parcourus dans un ordre fixé.  $S_1$  est le premier voisin tel que  $f(S_1) < f(S)$ .

La principale faiblesse d'une méthode de recherche locale est que si la solution courante se situe dans une « vallée » (voir figure 2.7) dans laquelle le minimum global ne se trouve pas, l'évolution systématique de cette solution vers une solution voisine par la minimisation de la valeur du critère considéré exclura toute possibilité de converger vers la solution optimale globale. Le terme « vallée » est utilisé du fait de la forme de la courbe qui est assimilée aux contours d'une montagne.

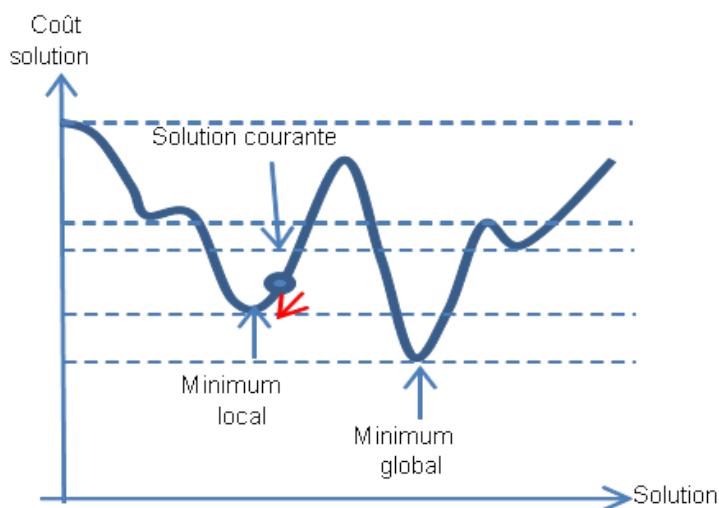


FIGURE 2.7 – Illustration de la recherche locale

Pour pallier le défaut présenté par la recherche locale, de nombreuses stratégies ont été développées. Chacune des stratégies utilisées pour s'extirper d'une vallée est souvent intégrée dans une métaheuristique et fait la spécificité de celle-ci. Dans ce qui suit, quelques métaheuristiques reposant sur ce principe sont présentées.

### 2.3.2.3 Recherche à voisinage variable

La recherche à voisinage variable (Variable Neighborhood Search - VNS) est une métaheuristique qui a été récemment proposée par (Mladenovic1997) pour résoudre de manière efficace des problèmes combinatoires. L'idée novatrice du VNS pour éviter d'être piégé dans un optimum local est d'utiliser alternativement un ensemble de voisinages qui vont permettre à chaque itération de disposer d'une nouvelle solution voi-

sine de la précédente. Le fait d'utiliser à chaque nouvelle itération une nouvelle solution initiale permet de réduire le risque d'être piégé dans un optimum local. Le pseudo de code du VNS en version de base est présenté dans l'algorithme 2.

---

**Algorithme 2 : VNSDeBase**


---

**Entrées :** Solution initiale – S  
**Sorties :** Optimum local – S

```

1 //Phase d'initialisation;
2 Sélection de l'ensemble des voisinages  $N_k$  ( $k = 1, \dots, k_{max}$ );
3 //Phase itérative;
4 répéter
5    $k \leftarrow 1;$ 
6   répéter
7     Perturbation – générer aléatoirement une solution  $S_1$ 
      dans le voisinage  $N_k (S)$  ;
8     Recherche locale – appliquer une méthode de recherche
      locale avec  $S_1$  comme solution initiale. Soit  $S_2$  la solution
      obtenue ;
9     si  $S_2$  est meilleure que  $S$  alors
10    |    $S \leftarrow S_2;$ 
11    |    $k \leftarrow 1$  // La recherche est recentrée dans le premier
      voisinage  $N_1$ ;
12    sinon
13    |    $k \leftarrow k + 1$  ;
14  jusqu'à  $k = k_{max}$ ;
15 jusqu'à Condition d'arrêt;
16 Retourne  $S$ ;
```

---

Le VNS est composé de deux phases principales. Une phase d'initialisation et une phase d'itération.

L'étape d'initialisation décrite dans le VNS présenté ci-dessus consiste à sélectionner l'ensemble des voisinages et déterminer un critère d'arrêt. La sélection de l'ensemble des voisinages  $N_k$  ( $k = 1, \dots, k_{max}$ ) dépend du problème étudié. Par exemple, considérons une tournée  $T$  dans laquelle un véhicule visite une séquence de points. Considérons le mouvement élémentaire  $\mu$  qui consiste à supprimer et réinsérer de manière aléatoire un client dans  $T$ . Soit  $N_k (S)$  le voisinage ainsi construit. Pour générer  $k$  voisinage ( $k \geq 1$ ), il est par exemple possible d'appliquer  $\mu$ ,  $k$  fois successivement à  $S$ . Ainsi, pour  $k = 2$ ,  $N_2 (S)$  sera le voisinage obtenu en supprimant et réinsérant de manière aléatoire 2 clients dans  $T$ .

Après l'étape d'initialisation du VNS, plusieurs itérations sont opérées, tant que la condition d'arrêt n'est pas vérifiée. Chaque itération est constituée de trois phases qui sont la perturbation (ligne 7), la recherche locale (ligne 8) et la décision (lignes 9 – 13). La phase de perturbation consiste à tirer de façon aléatoire une solution  $S_1$  dans le voisinage  $N_k (S)$ . Cette solution est fournie comme entrée d'une méthode de recherche locale dont l'objet est de l'améliorer. Une méthode quelconque peut être utilisée pour améliorer  $S_1$ . Il peut s'agir de la méthode tabou, du recuit simulé, de la re-

cherche locale de base  $S$ , etc. La phase de décision a pour but de décider si oui ou non la meilleure solution connue doit être mise à jour. Si la solution  $S_2$  obtenue après amélioration de  $S_1$  par la recherche locale est plus intéressante que la meilleure solution connue  $S$ , la solution  $S_2$  remplace la solution  $S$  ( $S \leftarrow S_2$ ) et la recherche est effectuée autour de cette nouvelle solution en utilisant à nouveau le premier voisinage ( $k \leftarrow 1$ ). Dans le cas où  $S_2$  est moins intéressante que  $S$ , le voisinage courant utilisé pour la prochaine perturbation sera le voisinage suivant ( $k \leftarrow k + 1$ ) appliqué à la même solution  $S$ .

#### 2.3.2.4 GRASP

Le GRASP est une métaheuristique itérative dont chacune des itérations est indépendante des autres et est constituée de deux phases. La première phase consiste en la construction d'une solution réalisable. La deuxième phase est une recherche locale effectuée pour trouver un optimum dans le voisinage de la solution obtenue dans la première phase. La phase de construction est un algorithme glouton qui possède plusieurs itérations. Pour déterminer l'élément à insérer, l'ensemble des éléments candidats ou éléments constitutifs de  $S$  et non encore insérés sont triés de manière croissante suivant le coût de leur insertion dans  $S$ . Ensuite, certains des premiers éléments sont sélectionnés et placés dans une liste appelée liste des candidats restreints (RCL – Restricted Candidate List). Le paramétrage principal du GRASP réside dans le choix du nombre d'éléments à insérer dans la RLC. Une fois la RCL construite, un seul élément y est sélectionné de façon aléatoire et rajouté dans  $S$ . Le processus qui vient d'être décrit est répété jusqu'à ce que la solution  $S$  soit complètement constituée. Il est à noter que lorsqu'un élément est ajouté à  $S$ , avant de reconstituer la RCL, la liste d'éléments candidats est de nouveau triée pour refléter le fait que le coût de leur insertion ait potentiellement changé. Le recalcul et le tri de chaque élément candidat à la fin de chaque itération représente l'aspect adaptatif du GRASP, alors que la sélection de l'élément à rajouter dans la RLC caractérise l'aspect aléatoire du GRASP.

#### 2.3.2.5 Méthode tabou

La méthode tabou, développée par (Glover1989), combine une procédure de recherche locale associée à des mécanismes permettant de surmonter l'obstacle des extrema locaux (Glover2001)(Glover2006). Le principe est de générer un ensemble de solutions voisines et de tendre vers celles qui améliorent le coût de solution, ou celles qui le dégradent le moins possible. Ce mécanisme permet donc d'éviter le piège du minimum local mais présente l'inconvénient d'un retour possible en arrière ou de cycles. Les cycles caractérisent une propriété de réversibilité qui se traduit comme suit : si une solution  $S_1$  a permis une évolution vers une solution voisine  $S_2$ ,  $S_1$  reste potentiellement une solution voisine de  $S_2$  et peut représenter la meilleure solution du nouveau calcul de voisinage de  $S_2$ . Pour éviter de tels cycles, une liste tabou de solutions est construite et mise à jour progressivement ; elle correspond aux évolutions antérieures dans la recherche de solutions qui sont à exclure dans les évolutions fu-

tures. La méthode tabou a été appliquée avec succès à la résolution de plusieurs problèmes de transport (Braysy2002)(Cordeau2003)(Ho2004).

#### 2.3.2.6 Recuit simulé

Le recuit simulé a été développé par Kirkpatrick en 1983 (Kirkpatrick1983) et s'inspire du principe de recuit des métaux visant à améliorer leur structure moléculaire. À partir d'une solution initiale ou courante, une nouvelle solution voisine est évaluée. Si cette solution est meilleure que la solution courante, elle remplace cette dernière. Dans le cas d'une solution dégradant le coût, la solution peut être retenue selon une probabilité dépendante de l'écart de coût entre la solution courante et la solution voisine, ainsi que d'un paramètre appelé « température ». Cette température décroît au fur et à mesure des itérations, réduisant la probabilité d'évoluer vers une solution dégradant le coût jusqu'à l'annuler. De nombreux auteurs ont utilisé le recuit simulé pour optimiser les tournées de véhicules (Chiang1996). D'autres problèmes d'optimisation tels que par exemple le design des circuits électroniques, la restauration d'images ont été résolus de façon satisfaisante avec le recuit simulé (Kirkpatrick1984).

#### 2.3.2.7 Algorithmes génétiques

Les algorithmes génétiques (Holland1992)(Goldberg1989) sont des algorithmes d'optimisation s'inspirant de l'évolution naturelle et de la génétique en se basant sur les concepts de croisements, mutations et sélections. La base d'un algorithme génétique repose sur l'évolution d'une population de solutions. La population de solutions peut être choisie a priori au hasard ou par un autre mécanisme. Pour faire évoluer une population, les meilleures solutions sont sélectionnées à chaque étape pour constituer une nouvelle génération. Après avoir été sélectionnées, les solutions sont croisées entre elles afin d'obtenir de nouvelles solutions pour former une nouvelle génération marquant ainsi le processus d'évolution. Le croisement de deux individus ou parents consiste à effectuer une combinaison de leurs caractéristiques pour obtenir un nouvel individu. Les solutions ainsi obtenues peuvent éventuellement subir des perturbations aléatoires aussi appelées mutations. Le principe d'évolution qui vient d'être décrit laisse apparaître une grande possibilité de mises en œuvre notamment en ce qui concerne : le codage des solutions c'est-à-dire leur représentation, la taille de la population à sélectionner ainsi que les opérateurs de sélection, croisements et mutations. L'évolution récente de la puissance de calculs des ordinateurs a permis la mise en œuvre avec succès des algorithmes génétiques sur les problèmes de tournées de véhicules (Ombuki2006)(Peng2008) et bien d'autres problèmes combinatoires (Man1999).

Les travaux de la littérature dans le domaine du routage de véhicules font la plupart du temps état de méthodes de résolution adaptées à un contexte de résolution en particulier. C'est pourquoi, dans la même logique que celle que nous avons adoptée pour proposer une grille d'analyse des familles de problèmes de routage étudiés (Tableau 2.1), nous nous proposons également d'identifier et classer les différents travaux analysés

selon les critères d'optimisation étudiés ainsi que les méthodes de résolution développées. Lors de la réalisation de l'état de l'art, ces informations seront synthétisées sous la forme d'un tableau (Tableau 2.2), tel que décrit ci-dessous.

Travaux étudiés	Critères d'optimisation étudiés	Méthodes de résolution développées
	<ul style="list-style-type: none"> <li>– Réduction du nombre de véhicules</li> <li>– Réduction de la distance, etc.</li> </ul>	<ul style="list-style-type: none"> <li>– Programmation linéaire (méthode exacte)</li> <li>– Métaheuristiques (recherche tabou, etc.)</li> <li>– Heuristiques</li> </ul>

TABLE 2.2 – Critères et méthodes utilisées

## 2.4 PROBLÈME DE COLLECTE ET LIVRAISON AVEC FENÊTRE DE TEMPS

Le problème de transport étudié dans cette thèse, est un problème connexe du problème de base de collecte et de livraison avec fenêtres de temps. C'est pourquoi, afin d'être en mesure de résoudre efficacement le problème ciblé, il est important de présenter la structure, et les approches de résolution du problème de base dont il est issu.

Le problème de collecte et livraison avec fenêtre de temps noté PDPTW (Pickup and Delivery Problem with Time Windows) est un problème de transport dans lequel des produits sont collectés et distribués dans une même tournée. Il s'agit d'un problème NP – difficile (Savelsbergh 1990) principalement caractérisé par deux contraintes qui sont : la précédence et le couplage. La contrainte de précédence impose que le client chargeur (expéditeur) soit visité avant le client à qui se destine la charge. La contrainte de couplage exige que le client auprès duquel la marchandise est chargée et le client vers qui cette marchandise est acheminée soient visités par un même véhicule. Une définition plus formelle du PDPTW reposant sur la théorie des graphes est maintenant proposée.

### 2.4.1 Formulation du PDPTW

Soit  $n$  le nombre de requêtes de transport qui correspond au nombre de demandes d'acheminement de produits d'un expéditeur vers un destinataire et soit  $N = P \cup D$  un ensemble de  $2n$  éléments partitionnés en deux sous ensembles :  $P = \{1, \dots, n\}$  est l'ensemble des points de ramassages et  $D = \{1 + n, \dots, 2n\}$  est l'ensemble des points de distributions. Chaque demande ou requête de transport est caractérisée par un couple de points noté  $R_i = (i, i + n)$ . Par convention et afin de faciliter la modélisation du problème, si le point de ramassage ou origine de l'expédition est indexé par  $i \in P$ , la destination ou le point de livraison de la demande sera indexé par  $i + n \in D$ . La variation de charge en ces points est modélisée par une quantité  $l_i$  positive représentant la collecte au point  $i \in P$  et  $l_{i+n}$  négative de produits déposée en  $i + n \in D$  telle que  $l_{i+n} = -l_i$ . Nous notons  $K$

l'ensemble homogène des véhicules de capacité  $Q$  chacun. Les sommets  $o_1$  et  $o_2$  sont respectivement les dépôts de départ et d'arrivée des véhicules. Dans la plupart des problèmes  $o_1$  et  $o_2$  représentent en réalité un seul et même dépôt qui a été dupliqué pour faciliter la modélisation du problème. L'ensemble de tous les sommets est désigné par  $V = N \cup o_1 \cup o_2$ . Chaque sommet  $j \in V$  possède une fenêtre de temps  $[a_j, b_j]$  et une durée de service  $s_j \geq 0$ . Les valeurs  $a_j$  et  $b_j$  sont respectivement la borne inférieure et supérieure de la fenêtre de temps. Le problème de ramassage et distribution est formellement caractérisé par un graphe  $G = (V, E)$  tel que  $E = V \times V$  est l'ensemble des arcs de  $G$ . Le temps de parcours, entre deux sommets  $i \in V$  et  $j \in V$  est représenté par  $t_{ij} \geq 0$ . Chaque arc du graphe  $G$  reliant deux sommets  $i \in V$  et  $j \in V$  est associé à une pondération  $c_{ij} \geq 0$ , qui représente le coût d'utilisation de l'arc.

#### 2.4.2 Modélisation en variables mixtes du PDPTW

Un modèle de programmation linéaire en variables mixtes, dans lequel un certain nombre de variables sont contraintes à être entières, est présenté ici. La modélisation utilise la variable de décision à trois indexées  $x_{ij}^k$  pour exprimer le fait que le véhicule  $k \in K$  parcourt l'arc reliant les sommets  $i \in V$  et  $j \in V$  ou non ; ainsi que trois variables à deux indices. La première  $dt_j^k$  permet de dater l'arrivée du véhicule  $k \in K$  à un sommet  $j \in V$ . La seconde  $L_j^k$  représente la charge du véhicule  $k \in K$  au moment de quitter  $j \in V$ . Cette formulation très utilisée dans la littérature est inspirée de (Toth2002)(Laporte1992). La troisième variable  $y_i^k$  est une variable qui vaut 1 si le véhicule  $k \in K$  visite le nœud  $i \in V$  et 0 sinon. Afin de mettre sous forme linéaire certaines équations du modèle, le paramètre  $M$  qui est supposé être un très grand nombre positif, sera utilisé par la suite.

Préalablement à la présentation du modèle, le récapitulatif de l'ensemble des hypothèses nécessaires à sa description est nécessaire :

- Chaque sommet de collecte ou de livraison est visité une et une seule fois.
- Toutes les requêtes doivent être satisfaites.
- Tous les véhicules partent à l'instant initial du dépôt de départ et retournent au dépôt d'arrivée.
- Tous les véhicules partent du dépôt avec une charge initiale nulle.
- Tout véhicule qui arrive à un sommet avant l'ouverture de sa fenêtre de temps doit attendre.
- Tout véhicule qui arrive à un sommet de collecte ou de livraison doit repartir de ce sommet

Sur la base de ces hypothèses, le modèle linéaire en variables mixtes du PDPTW est le suivant :

$$\text{Min} \sum_{k \in K} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij}^k \quad (2.1)$$

$$\sum_{j \in P \cup \{o_2\}} x_{o_1 j}^k = 1, \forall k \in K \quad (2.2)$$

$$\sum_{j \in D \cup \{o_1\}} x_{jo_2}^k = 1, \forall k \in K \quad (2.3)$$

$$\sum_{j \in V} x_{ij}^k = y_i^k, \forall k \in K, \forall i \in N \quad (2.4)$$

$$\sum_{i \in V} x_{ij}^k = y_j^k, \forall k \in K, \forall j \in N \quad (2.5)$$

$$\sum_{k \in K} y_i^k = 1, \forall i \in N \quad (2.6)$$

$$\sum_{j \in V} x_{ji}^k - \sum_{i \in V} x_{ij}^k = 0, \forall k \in K, \forall i \in N \quad (2.7)$$

$$\sum_{j \in V} x_{ji}^k - \sum_{j \in V} x_{i+n,j}^k = 0, \forall k \in K, \forall i \in P \quad (2.8)$$

$$x_{ij}^k = 1 \Rightarrow dt_j^k \geq dt_i^k + s_i + t_{ij}, \forall k \in K, \forall (i, j) \in V \times V \quad (2.9)$$

$$dt_j^k \geq dt_i^k + s_i + t_{ij} - M(1 - x_{ij}^k), \forall k \in K, \forall (i, j) \in V \times V \quad (2.10)$$

$$a_i \cdot y_i^k \leq dt_i^k \leq b_i \cdot y_i^k, \forall i \in V, \forall k \in K \quad (2.11)$$

$$dt_i^k \leq dt_{i+n}^k, \forall k \in K, \forall i \in P \quad (2.12)$$

$$dt_{o_1}^k = 0, \forall k \in K \quad (2.13)$$

$$x_{ij}^k = 1 \Rightarrow L_j^k = L_i^k + l_j, \forall k \in K, \forall (i, j) \in V \times V \quad (2.14)$$

$$L_j^k \leq L_i^k + l_j + M(1 - x_{ij}^k), \forall k \in K, \forall (i, j) \in V \times V \quad (2.15)$$

$$L_j^k \geq L_i^k + l_j - M(1 - x_{ij}^k), \forall k \in K, \forall (i, j) \in V \times V \quad (2.16)$$

$$0 \leq L_i^k \leq Q, \forall i \in V, \forall k \in K \quad (2.17)$$

$$L_i^k = 0, \forall k \in K, \forall i \in o_1 \cup o_2 \quad (2.18)$$

$$x_{ij}^k, y_i^k \in \{0, 1\}, \forall k \in K, \forall (i, j) \in V \times V \quad (2.19)$$

L'expression 2.1 définit la fonction objectif du modèle. Elle porte sur la minimisation du coût du transport nécessaire pour satisfaire toutes les demandes de transport. L'équation 2.2 garantit que tous les véhicules quittent le dépôt de départ tandis que l'équation 2.3 garantit qu'ils terminent leurs tournées au dépôt de retour. Les équations 2.4, 2.5 et 2.6 assurent que chaque sommet de collecte ou de livraison est visité une et une seule fois. L'équation 2.7 assure que tout véhicule qui arrive chez un client repart de celui-ci. L'équation 2.8 garantit que le point d'enlèvement d'une requête et le point de livraison correspondant sont visités par un même véhicule. Cette équation est aussi appelée contrainte de couplage. L'équation 2.9 permet d'assurer la cohérence de l'évolution du temps de parcours d'un véhicule en fonction de la durée du trajet et de la durée de service. L'équation 2.10 est la forme linéaire de l'équation 2.9. L'équation 2.11 impose le respect de la fenêtre de temps en chaque nœud. L'utilisation de la variable  $y_i^k$  dans l'équation 2.11 permet d'assurer que si un nœud n'est pas visité par un véhicule, la date d'arrivée du véhicule en ce nœud est fixée à 0. L'association de l'équation 2.9 et de l'équation 2.11 assure qu'un véhicule attend l'ouverture de la fenêtre de temps d'un client, lorsqu'il anticipe son arrivée chez ce dernier. L'équation 2.12 assure que la visite par un véhicule du nœud correspondant au point d'enlèvement d'une requête est effectuée avant la visite du nœud correspondant au point de livraison de cette requête. Cette équation est appelée aussi contrainte de précédence. L'équation 2.13 impose que tous les véhicules partent du dépôt de départ à l'instant initial. L'équation 2.14 permet à un véhicule de mettre à jour sa charge après la visite de chaque client de sa tournée. Les équations 2.15 et 2.16 permettent de mettre sous forme linéaire l'équation 2.14. L'équation 2.17 impose le respect de la capacité des véhicules. Cette équation est appelée aussi contrainte de capacité. L'équation 2.18 impose que tous les véhicules partent du dépôt avec une charge initiale nulle. L'équation 2.19 assure l'intégrité des variables.

Le PDPTW vient d'être formalisé et modélisé au moyen d'un ensemble d'équations linéaires. Sa résolution fait appel à plusieurs méthodes dont certaines sont exactes et d'autres approchées. Son étude et sa résolution ont fait l'objet de très nombreux travaux dont quelques-uns sont analysés afin de s'en inspirer pour résoudre le problème proposé dans la présente thèse.

#### 2.4.3 Présentations et analyses des travaux sur le problème de collecte et livraison

Quelques travaux portant sur les modèles et méthodes de résolution du PDPTW statique et dynamique sont analysés ici afin de montrer la

diversité des travaux existants. Le PDPTW statique consiste en la génération d'un ensemble de tournées prévisionnelles en réponse à un ensemble de requêtes de transport. Le PDPTW dynamique consiste à reconfigurer les tournées en fonction de l'observation de situations imprévues. Plus généralement un problème de routage est dynamique (Dupas2013) si l'ensemble des informations n'est pas connu lorsque les tournées prévisionnelles sont élaborées ou si certaines informations peuvent changer après que les tournées aient été élaborées, voire débutées. Les situations imprévues sont caractérisées par l'occurrence d'aléas tels qu'un retard lié à une congestion de trafic, ou encore l'arrivée de nouveaux clients, alors que les tournées sont en cours d'exécution.

Avant de détailler les différents travaux, une synthèse des lectures effectuées est proposée sous la forme de deux tableaux, conformément aux grilles de références présentées plus tôt dans ce chapitre.

Travaux étudiés	Produits	Dépôts	Véhicules			Fenêtres de temps	Nature du service	Nombre de visites	Synchronisation	Consolidation
			Nombre	Capacité	Flotte					
(Kalantari1985)	✓	✓	✓	✓	✓	✓✓	✓	✓	✓	✓
(Dumas1991)	✓		✓	✓		✓	✓✓	✓	✓	✓
(Vander1993)	✓	✓	✓	✓		✓	✓✓	✓	✓	✓
(Savelsbergh1998)	✓	✓	✓	✓			✓✓	✓	✓	✓
(Nanry2001)	✓	✓	✓	✓		✓✓	✓✓	✓	✓	✓
(Li2001)	✓	✓	✓	✓		✓✓	✓✓	✓	✓	✓
(Mitrovic2004a)	✓		✓✓	✓		✓	✓✓	✓	✓	✓
(Mitrovic2004)	✓		✓✓	✓		✓	✓✓	✓	✓	✓
(Bent2006)	✓	✓	✓	✓		✓	✓✓	✓	✓	✓
(Ropke2006)	✓		✓✓	✓		✓	✓✓	✓	✓	
(Ropke2007)	✓	✓	✓	✓		✓✓	✓✓	✓	✓	

TABLE 2.3 – Quelques travaux sur le problème de collecte et livraison

Travaux étudiés	Critères d'optimisation étudiés	Méthodes de résolution développées
(Kalantari1985)	Réduction du coût du transport	Algorithme de séparation et d'évaluation
(Dumas1991)	Réduction du coût du transport	Génération de colonnes
(Vander1993)	Réduction du coût du transport	Heuristique en deux phases
(Savelsbergh1998)	Réduction de la masse salariale des chauffeurs	Stratégie Basique et Branch-and-Price
(Nanry2001)	Somme pondérée de la distance totale parcourue, de la violation des fenêtres de temps et des capacités des véhicules	Recherche tabou réactive
(Li2001)	Minimisation du nombre de véhicules, de la distance parcourue et du coût du transport	Recherche tabou couplée au recuit simulé
(Mitrovic2004a)	Réduction de la distance totale parcourue	Heuristique d'insertion et recherche tabou
(Mitrovic2004)	Réduction de la distance totale parcourue	Heuristique d'insertion et recherche tabou
(Bent2006)	Minimisation du nombre de véhicules et de la distance parcourue	Méthode en deux phases : recuite simulé et recherche à voisinage large
(Bent2006)	Minimisation du nombre de véhicules et de la distance parcourue	Recherche adaptative à voisinage large
(Ropke2007)	Réduction du coût du transport	Branch and cut and price

TABLE 2.4 – Critères et méthodes utilisées pour résoudre le PDP

En première partie de ce chapitre, un ensemble d'éléments permettant de caractériser les problèmes de routage a été présenté. Certains de ces éléments en rapport direct avec le problème de transport étudié dans cette thèse sont repris dans les colonnes du tableau 2.3 pour discriminer et affiner la présentation des problèmes qui sont analysés dans le prochain paragraphe. Le tableau 2.4 permet de présenter les méthodes de résolutions employées pour résoudre les dits problèmes.

#### 2.4.3.1 Problème statique de collecte et livraison

Cette partie fait état de l'analyse de travaux effectués dans un contexte de PDPTW statique. Un certain nombre d'études se sont focalisées sur le développement de méthodes exactes de résolution.

Un algorithme exact basé sur le Branch-and-Bound a été proposé dans (Kalantari1985) pour résoudre un PDPTW avec un seul véhicule contenant jusqu'à 15 requêtes ce qui correspond à 30 clients. L'algorithme utilisé est une extension de celui proposé dans (Little1963) pour résoudre un problème de voyageur de commerce et permet d'éliminer en chaque nœud de l'arbre de recherche des arcs qui pourraient produire des solutions dans lesquelles la contrainte de précédence n'est pas respectée.

(Dumas1991) ont construit un algorithme basé sur la génération de colonnes pour résoudre de manière exacte le problème de collecte et livraison. Des tests ont été effectués sur des instances contenant jusqu'à 55 requêtes.

Les travaux réalisés autour du développement de méthodes de résolu-

lution exactes montrent clairement les limites de ces méthodes lors de la considération des problèmes de grandes tailles, et la nécessité de développer des méthodes de résolution approchées plus propices à traiter de tels problèmes. Parmi les travaux réalisés dans cette optique, nous citons ici les plus représentatifs vis-à-vis de notre problème.

Dans (Nanry2001), une recherche tabou réactive est utilisée pour résoudre le PDPTW. La méthode est dite réactive car la longueur de la liste tabou est auto ajustable. Une première solution initiale est construite en utilisant un algorithme glouton. Cette solution est ensuite améliorée par la recherche tabou. La recherche tabou est basée sur une recherche locale qui utilise successivement trois voisinages. Le premier voisinage consiste à supprimer une requête dans une route pour l'insérer dans une autre, le second consiste à échanger deux requêtes appartenant à deux routes différentes, la troisième consiste à retarder ou anticiper la position d'un nœud de collecte ou de livraison dans une tournée. Pour valider l'algorithme, des expériences ont été effectuées sur deux catégories d'instances d'une taille allant jusqu'à 100 clients : un premier ensemble d'instances a été générée de manière totalement aléatoire ; la deuxième catégorie a consisté à transformer des instances de VRPTW accessibles dans la littérature en des instances de PDPTW tout en veillant à ne pas modifier la solution optimale initiale. L'observation des résultats a permis aux auteurs de conclure que l'algorithme était capable en moins de 5 minutes de trouver des solutions très proches de l'optimum.

L'hybridation de la recherche tabou avec le recuit simulé a été proposée dans (Li2001) pour résoudre le PDPTW. La spécificité de la recherche tabou proposée dans ces travaux réside dans la construction de la liste tabou. En effet le codage d'une solution est basé sur des éléments tels que : le nombre de véhicules utilisés, le temps d'attente, la distance totale parcourue, la durée des tournées. Le recuit simulé est utilisé à chaque itération de la recherche tabou pour améliorer la solution courante. Pour valider la méthode proposée, les auteurs ont construit 100 instances à partir d'instances de VRPTW issues de la littérature avec un objectif consistant prioritairement à réduire le nombre de véhicules utilisés et ensuite la distance totale parcourue.

Toujours dans le cadre de méthodes approchées, une métaheuristique en deux phases a été proposée dans (Bent2006) pour résoudre un PDPTW dont l'objectif est identique à celui considéré dans (Li2001). La technique de recuit simulé est appliquée dans une première phase pour réduire le nombre de véhicules utilisés et une heuristique est utilisée dans une seconde phase pour réduire la distance totale parcourue. Pour mettre en œuvre cette seconde phase, un ensemble de requêtes est supprimé de la solution courante puis réinséré. Pour limiter le temps de calcul, la réinsertion des requêtes supprimées est effectuée avec un Branch-and-Bound dans lequel une partie seulement de l'arbre de recherche est parcourue. La technique utilisée pour ne parcourir qu'une partie de l'arbre de recherche est appelée « limited discrepancy search (LDS) » (Harvey1995). La LDS est une technique de recherche arborescente partielle. Elle permet d'obtenir des solutions de bonne qualité en temps de calcul jugée raisonnable pour des problèmes dans lesquels les espaces de recherche sont si grands qu'ils ne peuvent être parcourus par une méthode arborescente exhaustive. Avec

un temps de calcul n'excédant pas quatre-vingt-dix minutes, l'heuristique proposée a été capable de trouver de nouvelles solutions sur des instances connues dont la taille variait entre 100 et 600 clients.

(Ropke2006) ont proposé une méthode en deux phases basée sur la recherche adaptative à voisinage variable pour résoudre le PDPTW. Il s'agit d'une méthode de destruction et reconstruction qui partage de nombreuses similitudes avec le VNS notamment dans la variété des voisinages utilisés durant le processus d'optimisation. La destruction d'une solution consiste à supprimer une ou plusieurs requêtes au moyen d'une ou plusieurs heuristiques, tandis que la reconstruction consiste à réinsérer dans la solution les requêtes supprimées par le biais d'une ou plusieurs heuristiques. Le choix de l'heuristique de destruction ou de reconstruction à employer est effectué à chaque itération de manière adaptative en fonction de sa performance passée. Ainsi, à chaque itération, un score est attribué à chaque heuristique de destruction ou de reconstruction. Ce score correspond au nombre de nouvelles solutions qui ont été identifiées grâce à son utilisation. Une fois les scores attribués, le choix de l'heuristique à utiliser est effectué par un tirage aléatoire sur une roulette dans laquelle la surface occupée par chaque heuristique est proportionnelle à son score. Pour valider la méthode proposée, des tests expérimentaux ont été effectués sur des instances proposées dans (Li2001) et les meilleures solutions connues pour un certain nombre d'instances ont pu être améliorées.

#### **2.4.3.2 Problème de collecte et livraison dynamique**

Un problème de collecte et livraison dynamique est un problème de transport dans lequel un ensemble d'événements perturbateurs peuvent apparaître alors même que les tournées sont déjà en cours d'exécution. La variation du contexte de transport nécessitant d'avoir recours à des méthodes de résolution dynamique est due à de multiples facteurs : apparition de nouveaux clients, retards sur chargement/ déchargement, congestion de trafic, météorologie, etc. Dans la littérature, l'événement le plus étudié dans le cadre d'un PDPTW dynamique concerne l'apparition de nouvelles requêtes (Berbeglia2010), c'est pourquoi notre étude bibliographique se centrera plus particulièrement sur les travaux traitant de ce problème.

Afin de garder le système de transport dans un état de fonctionnement nominal, une reconfiguration partielle ou totale de celui-ci peut être nécessaire suite à l'apparition de ces perturbations. Pour effectuer une reconfiguration de tournées, plusieurs stratégies peuvent être adaptées. La littérature scientifique en propose quelques-unes : stratégie basique, stratégie de mise à jour, d'anticipation et stratégie d'attente.

La stratégie basique (Berbeglia2010) est une stratégie utilisée pour résoudre un problème de tournées de véhicules dans lequel peut surveiller de nouvelles requêtes alors que les véhicules sont déjà en route. Le principe de cette stratégie est de recalculer la tournée de chaque véhicule en considérant les nouvelles requêtes et en supprimant celles déjà satisfaites. Il s'agit ainsi de résoudre une succession de ré-optimisation de problèmes statiques sur la base des informations disponibles à une date donnée. Ces informations définissent une photographie de la situation

courante de transport. Le principal handicap de cette stratégie réside dans le temps de calcul nécessaire pour permettre une ré-optimisation globale des tournées. Cela peut rendre la stratégie inadaptée pour des problèmes de grandes tailles.

Pour pallier le défaut de la stratégie basique, une autre stratégie (Berbeglia2010) consistant à reconfigurer seulement une partie des tournées et non toutes les tournées est souvent utilisée. Les tournées à reconfigurer sont celles qui sont situées le plus près du lieu d'apparition de la requête dynamique. Dans cette logique, un ensemble d'opérateurs de suppression, de réinsertion et de permutation de requêtes dans une même route ou entre routes différentes peut être utilisé pour modifier la solution courante en y insérant à moindre coût la ou les nouvelle(s) requête(s). Il est à noter que cette stratégie de mise à jour ne garantit pas l'obtention d'une bonne solution, dans la mesure où la reconfiguration ne porte pas sur l'ensemble des routes, mais peut être suffisamment rapide pour être utilisée dans un problème de grande taille.

La stratégie dite d'attente, est une stratégie utilisée dans les problèmes de tournées dynamiques dont l'objectif est de minimiser la déviation des véhicules de leur route courante. Ainsi, lors de l'apparition d'une nouvelle requête, une analyse des tournées existantes sera effectuée afin de déterminer celle(s) pouvant la servir sans causer une déviation des véhicules ou conduisant à une déviation minimale. Pour y arriver, des routes sont construites de manière à garantir un degré de liberté maximum pour l'insertion de futures requêtes. Ce degré de liberté porte sur les temps d'attente liés au respect des fenêtres de temps. La répartition efficace de ce temps d'attente aux différents nœuds constituants la séquence de visites d'un véhicule peut permettre d'insérer dans certains cas de nouvelles requêtes sans aucune déviation. L'intérêt de la stratégie d'attente a été démontrée de manière analytique dans (Branke2005).

L'exploitation de ces différentes stratégies à des fins de reconfigurations de tournées dans des problèmes de collecte et livraison avec fenêtre de temps a fait l'objet de travaux dont un petit nombre est maintenant analysé.

Pour optimiser l'activité de transport d'une compagnie de distribution, l'algorithme proposé dans (Dumas1991) a été étendu dans (Savelsbergh1998). La mise à jour de l'algorithme a eu pour objet de permettre l'insertion dynamique des requêtes dans les tournées en cours d'exécution au fur et à mesure de leur apparition.

(Savelsbergh1998) ont proposé un algorithme pour résoudre un PDPTW dynamique. Dans le PDPTW considéré, les contraintes relatives à la législation telle que le temps de repos obligatoire des chauffeurs sont prises en compte et l'horizon d'optimisation est divisé en périodes couvrant plusieurs jours chacune. La fonction objectif du problème consiste en une minimisation de la masse salariale des chauffeurs. Le salaire d'un chauffeur est une fonction de sa distance totale parcourue et du nombre de nuits passées à l'extérieur de sa zone de résidence. Afin d'employer la stratégie basique pour résoudre ce problème, les auteurs l'ont partitionné en plusieurs sous problèmes statiques couvrant tout l'horizon de planification. Lorsqu'une nouvelle requête arrive, celle-ci est introduite dans un sous-problème à l'aide d'une heuristique basée sur la méthode

de Branch-and-Price. L'algorithme a été testé sur un horizon de 10 jours, sur des instances réelles contenant jusqu'à 200 requêtes. Après une phase d'analyse expérimentale, les auteurs ont conclu que le coût de la solution trouvée par leur algorithme était meilleur que celui trouvé manuellement par le planificateur de tournées.

Dans la même optique que les travaux précédents, (Mitrovic2004a) ont proposé un algorithme pour résoudre un PDPTW dynamique rencontré par une compagnie de transport de courriers. La stratégie de résolution employée pour satisfaire l'apparition de nouvelles requêtes est un couplage de la stratégie basique et de la stratégie d'attente. L'arrivée d'une nouvelle requête entraîne une ré-optimisation du problème par une heuristique en deux phases. La première phase est une heuristique d'insertion à moindre coût qui permet d'insérer la requête dans une route constituant la solution courante. Une fois la requête insérée, la deuxième phase consiste en l'utilisation de la recherche tabou pour optimiser la distance totale parcourue. Les auteurs ont aussi montré de manière expérimentale que par rapport à une stratégie dans laquelle le temps d'attente n'est pas exploité il est possible de réduire la distance totale parcourue par les véhicules en distribuant correctement les temps d'attentes des véhicules aux différents nœuds de leur tournée respective.

L'algorithme développé dans le papier précédent a été étendu dans (Mitrovic2004) en divisant l'horizon de planification en deux périodes distinctes ou double-horizons. Le premier horizon consiste à optimiser les routes sur le court terme tandis que le deuxième horizon consiste à les optimiser sur le long terme. Optimiser une route sur le court terme signifie ne prendre en compte dans l'optimisation que les premiers clients de la tournée correspondante, tandis que optimiser sur le long terme porte sur l'amélioration du coût de la tournée en la considérant dans sa globalité. Les auteurs ont montré que dans un problème dynamique une ré-optimisation des tournées dans une approche à double-horizons est plus efficace qu'une ré-optimisation dans une approche à un seul horizon. La démonstration a été effectuée de manière expérimentale sur des instances construites à partir de données recueillies chez deux compagnies de transport de courriers de la ville de Vancouver au Canada.

D'autres travaux sur le PDP dynamique ainsi que de nombreuses variantes de ce problème de base sont décrits dans (Berbeglia2010).

## 2.5 SYNTHÈSE DES TRAVAUX SUR LES PROBLÈMES AVEC TRANSBORDEMENT

Bien que couvrant une large gamme de pratiques rencontrées dans le transport, l'ensemble des travaux précédemment présentés tant en statique qu'en dynamique ne permettent pas de pratiquer le transbordement ou une rupture de charge. Or la pratique du transbordement ou de rupture de charge telle que le cross-docking peut induire des gains importants dans le transport. L'objet de cette partie est d'analyser les singularités liées à la pratique du transbordement dans les problèmes de collecte et livraison et présenter quelques travaux scientifiques couvrant cet aspect du transport de marchandises par véhicules.

La pratique du transbordement consiste à utiliser plus d'un véhicule pour livrer un produit. Ainsi, lorsqu'un produit doit être transbordé, il est collecté à son point de chargement pour ensuite être déposé à un point de transbordement. De ce point, il est chargé par un autre véhicule pour être directement livré au destinataire ou être transporté à un autre point de transbordement. Le PDPTW étant régi par deux contraintes fortes qui sont le couplage et la précédence, l'application du transbordement au PDPTW consiste à relâcher la contrainte de couplage qui exige que le sommet de collecte d'une requête et le sommet de distribution correspondant soient visités par un même véhicule. De façon formelle, les singularités apportées par la prise en compte du transbordement sont les suivantes :

- En plus de l'ensemble des points de collecte et de livraison, lorsque le transbordement est autorisé, un ensemble supplémentaire de points doit être considéré à savoir l'ensemble des points de transbordement  $T$ .
- La contrainte 2.8  $\sum_{j \in V} x_{ji}^k - \sum_{j \in V} x_{i+n,j}^k = 0, \forall k \in K, \forall i \in P$  qui caractérise le couplage dans le modèle du PDPTW est relâchée.
- La contrainte de précédence 2.12  $dt_i^k \leq dt_{i+n}^k, \forall k \in K, \forall i \in P$  ne permet plus d'assurer l'antériorité entre l'enlèvement et la distribution d'un produit. En effet, le point de livraison et le point de collecte peuvent être visités par deux véhicules différents.
- Une autre singularité liée à l'introduction du transbordement dans le PDPTW est le phénomène dit d'interdépendance « temporelle » de routes. Dans le PDPTW de base, les tournées sont indépendantes les unes des autres. Lorsqu'un produit est transbordé entre deux véhicules  $k$  et  $v$  au point de transbordement  $t$ , la tournée de  $v$  est dépendante de la tournée de  $k$ . En effet,  $v$  ne peut arriver au point de transbordement après  $k$  ( $dt_t^k \leq dt_t^v$ ). De ce fait, tout retard contracté par  $k$  sera propagé à  $v$ .

Dans certains problèmes de routage en contexte de cross-docking, le véhicule ayant collecté une marchandise peut ne pas assurer la livraison finale au destinataire. La différence avec un problème de collecte et livraison et transbordement dont les singularités viennent d'être présentées est que dans le cross-docking, tous les véhicules sont obligés de passer par la plateforme de consolidation. Le cross-docking possède donc quelques similarités avec le problème de collecte et livraison avec transbordement (PDPT) et à ce titre, son étude peut permettre d'affiner certaines méthodes de résolutions dédiées au PDPT.

### 2.5.1 Problème de tournées de véhicules avec cross-dock (VRPCD)

Le cross-docking est une stratégie de transport dans laquelle des produits provenant de fournisseurs différents sont dégroupés, puis regroupés dans une plateforme de tri ou cross-dock (CD) en fonction de leur destination. Dans le cross-docking, l'activité du transport est donc structurée en trois phases distinctes : collectes, consolidation et livraisons. Comme cela vient d'être dit, la similitude entre le VRPCD et le PDPT tient au fait que le véhicule qui collecte une marchandise peut ne pas être celui qui le livre au destinataire final. La différence entre le PDPT et le VRPCD est liée à la subdivision en trois phases énoncées plus haut. En effet, dans le VRPCD le

passage au quai de l'ensemble des véhicules ayant collectés des produits chez les fournisseurs est systématique, alors que dans le PDPT un véhicule ayant collecté un produit peut directement le livrer à son destinataire sans passer par la plateforme.

De façon générale, la pratique du cross-docking pose de nombreuses problématiques qui concernent la localisation des plateformes, l'affectation des camions en entrée/ sortie la plateforme, la planification des tournées de véhicules, l'ordonnancement de camions, l'ordonnancement des ressources à l'intérieur de la plateforme, les problèmes de chargement et déchargement des produits dans les camions, etc. Quelques travaux illustrant certaines des problématiques citées plus haut sont ici présentés.

L'affectation des camions aux différentes portes de la plateforme est un problème important dans la gestion de celle-ci. En effet, si les camions sont bien affectés, cela peut permettre de réduire le coût de déplacement des produits à l'intérieur de la plateforme. Des économies d'échelle sont ainsi réalisées par la réduction du coût de manutention. C'est dans cette optique que (Tsui1990) ont proposé une approche dans laquelle aucun stockage temporaire n'est autorisé au CD. Par conséquent, il est nécessaire que les véhicules qui collectent les produits arrivent de manière synchrone à la plateforme. L'objectif de l'étude qui est effectuée dans cet article est d'affecter les camions aux portes d'entrées/sorties de la plateforme de manière à réduire la distance parcourue par les chariots élévateurs utilisés à l'intérieur de la plateforme pour déplacer les produits. Pour solutionner le problème décrit dans ce paragraphe, les auteurs ont développé une heuristique de recherche locale, mais aucune analyse expérimentale n'a été réellement effectuée.

La prise en compte du problème d'affectation des camions aux différentes portes peut être intéressante dans la volonté de mieux prendre en considération les contraintes de synchronisation des ressources de transport avec les ressources de manutention et la gestion des flux dans les entrepôts.

Pour réduire les coûts logistiques liés aux opérations de consolidation dans le CD, il est nécessaire de déterminer l'ordonnancement optimal des tâches de manutention. Cet ordonnancement porte sur le transfert des produits déchargés en entrée de la plateforme vers sa sortie. Une contribution à la résolution de cette problématique est proposée dans (Yu2008). Cet aspect logistique est intéressant car dans les problèmes de collecte et livraison avec transbordement, les véhicules sont déchargés au quai d'entrée du point de transbordement, puis déplacés au quai de chargement pour être chargés dans un autre véhicule. Du fait de la considération des fenêtres de temps, tout retard dans cette phase peut impacter la faisabilité d'une potentielle solution. Le précédent problème à été résolu à l'aide de trois approches avec pour objectif de réduire le coût et le temps de transferts des produits entre les portes. La première est un programme mathématique en variables mixtes. La seconde consiste à énumérer de façon explicite toutes les combinaisons possibles. La troisième est une méthode heuristique. La deuxième méthode de résolution est utilisée pour valider les résultats trouvés par les deux autres méthodes. Toutefois, elle est limitée par la taille des instances. Après une phase expérimentale sur des problèmes tests générés de façon aléatoire, les auteurs ont conclu que l'heuris-

tique permettait de trouver en temps raisonnable des solutions quasiment proche de l'optimum.

Un aperçu plus général des travaux portant sur la pratique et la gestion du cross-docking est fourni dans le récent état de l'art de (Van2012).

Contrairement à d'autres problèmes d'optimisation rencontrés dans la gestion du cross-docking, il existe peu d'articles (Boysen2010) centrés sur le routage de véhicules avec cross-docking. En se basant sur les grilles d'analyse définies précédemment, les éléments discriminants des papiers présentés sont synthétisés dans les tableaux 2.5 et 2.6 qui suivent.

Travaux étudiés	Produits	Dépôts	Véhicules			Fenêtres de temps	Nature du service	Nombre de visites	Synchronisation	Consolidation
			Nombre	Capacité	Flotte					
	Unique	Multiples	Unique	Multiples	Limité	Homogène	Avec	Unique	Avec	Facultatif
					Illimité	Hétérogène	Sans	Multiples	Sans	Obligatoire
(Lee2006)	✓		✓	✓	✓	✓	✓	<	✓	✓
(Wen2009)	✓		✓	✓	✓	✓	✓	<	✓	✓
(Liao2010)	✓		✓	✓	✓	✓	✓	✓	✓	✓
(Santos2011)	✓		✓	✓	✓	✓	✓	✓		✓

TABLE 2.5 – Quelques travaux sur le problème de routage avec cross-docking

Travaux étudiés	Critères d'optimisation étudiés	Méthodes de résolution développées
(Lee2006)	Minimisation du nombre de véhicules et du coût du transport	Recherche tabou
(Wen2009)	Réduction du coût du transport	Recherche tabou
(Liao2010)	Minimisation du nombre de véhicules et du coût du transport	Recherche tabou
(Santos2011)	Réduction du coût du transport	Branch-and-Bound et Génération de colonnes

TABLE 2.6 – Critères et méthodes utilisées pour résoudre le cross-docking

Dans (Lee2006), les auteurs ont proposé un modèle mathématique en nombres entiers pour résoudre le VRPCD. Ce modèle s'est avéré être limité pour résoudre des instances de grandes tailles et les auteurs ont donc complété leurs investigations en développant un algorithme basé sur la recherche tabou en utilisant un opérateur de voisinage consistant en une permutation de clients entre les tournées. Pour valider l'algorithme développé, 30 instances aléatoires contenant pour certaines jusqu'à 50 nœuds ont été générées. Pour chacune de ces instances, la solution optimale a été calculée par énumération de toutes les solutions réalisables. Ensuite, les résultats obtenus par l'algorithme ont été comparés aux solutions optimales.

Une contribution sur la résolution par des méthodes approchées du VRPCD est décrite dans (Liao2010). Dans ce papier, les auteurs proposent une heuristique basée sur la recherche avec une stratégie de suppression et réinsertion de requêtes. Pour valider l'algorithme développé, des instances construites dans (Lee2006) ont été utilisées. Les auteurs sont arrivés, selon les instances considérées, à améliorer de l'ordre de 10 à 36 % les solutions initialement trouvées par (Lee2006) tout en réduisant le temps de calcul.

Dans la même optique que le papier précédent, les travaux présentés dans (Wen2009) s'intéressent aussi au VRPCD dans lequel la contrainte de consolidation est relâchée. Ainsi, des véhicules qui arrivent au CD ne transféreront entre eux tout ou partie des produits qu'ils transportent qu'à condition d'engendrer une réduction du coût de transport pour la phase de distribution. Les auteurs ont proposé un modèle mathématique en variables mixtes pour résoudre le problème. Le modèle proposé ne pouvant pas résoudre des instances de grande taille, les auteurs ont développé un algorithme basé sur la recherche tabou. Pour valider l'algorithme, des tests expérimentaux ont été effectués sur des instances réelles contenant pour certaines jusqu'à 200 clients.

Pour résoudre le VRPCD de manière exacte, (Santos2011) ont mis au point un algorithme basé sur Branch and Price. Le VRPCD résolu est le même que celui décrit dans le papier de (Wen2009). Les auteurs ont effectué des tests sur des instances contenant entre 10 et 50 clients et ont pu observer qu'il n'était pas possible de résoudre de manière optimale en moins de 2 heures des instances contenant plus de 10 clients.

L'ensemble des problèmes de transport intégrant des pratiques de transfert de charges entre véhicules tel que le cross-docking, est analysé et présenté de manière détaillée dans l'état de l'art proposé par (Drexel2012).

Les travaux qui viennent d'être présentés sur le cross-dock permettent de se rendre compte de l'intérêt d'introduire une flexibilité au niveau de

l'utilisation des ressources dans le transport de produits. Toutefois, cette flexibilité est partielle dans la mesure où, même s'il est possible d'utiliser au moins deux véhicules pour satisfaire une demande de transport, le passage obligatoire de tous les véhicules au centre de cross-dock est une contrainte forte. Certains auteurs ont essayé d'analyser l'intérêt de disposer d'une flexibilité totale lors du transport d'un produit. Cette flexibilité totale dans le contexte du problème de collecte et livraison signifie la possibilité pour un produit d'être transporté de son point de collecte vers son point de livraison par plus d'un véhicule sans que le passage par un dépôt ou un point de transbordement ne soit systématique pour l'ensemble des véhicules. À notre niveau de connaissance de la littérature, très peu de travaux concernent cet aspect du transport.

#### **2.5.2 Résolution du problème de collecte et livraison avec transbordement**

Les travaux sur le problème de collecte et livraison sont synthétisés dans le tableau 2.7 afin de présenter les éléments caractéristiques des problèmes étudiés, en se basant sur la grille de références précédemment élaborée.

Travaux étudiés	Produits	Dépôts	Véhicules			Fenêtres de temps	Nature du service	Nombre de visites	Synchronisation	Consolidation
			Nombre	Capacité	Flotte					
(Shang1996)	✓	✓	✓	✓	✓	✓	✓	✓	✓	Facultatif
(Ortel2000)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
(Aldaihani2003)	✓		✓	✓	✓	✓	✓	✓	✓	
(Mues2005)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
(Mitrovic2006)	✓	✓		✓	✓	✓	✓	✓	✓	
(Thangiah2007)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
(Fournier2008)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
(Kerivin2008)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
(Lin2008)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
(Cortes2010)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
(Gortz2009)	✓	✓	✓	✓	✓	✓	✓	✓	✓	
(Masson2012)	✓		✓	✓	✓	✓	✓	✓	✓	
(Qu2012)	✓	✓	✓	✓	✓	✓	✓	✓	✓	Obligatoire

TABLE 2.7 – Synthèse de quelques travaux sur le problème de collecte et livraison avec transbordement

Travaux étudiés	Critères d'optimisation étudiés	Méthodes de résolution développées
(Shang1996)	Minimisation du nombre de véhicules, de la durée du transport et des retards	Mini route et heuristique d'insertion
(Ortel2000)	Réduction du coût du transport	Recherche tabou
(Aldaihani2003)	Réduction du coût du transport	Heuristique d'insertion et Recherche tabou
(Mues2005)	Réduction du coût du transport	Heuristique basée sur Génération de colonnes
(Mitrovic2006)	Réduction du coût du transport	Heuristique en deux phases : construction et amélioration
(Thangiah2007)	Minimisation du nombre de véhicules, de la durée du transport et des retards	Mini route, heuristique d'insertion, recherche locale
(Fournier2008)	Réduction du coût du transport	Programmation mathématique et algorithmes de recherche locale
(Kerivin2008)	Réduction du coût du transport	Branch and Cut
(Lin2008)	Réduction du nombre de véhicules et du coût du transport	Solveur
(Cortes2010)	Réduction du coût du transport	Branch and Cut
(Gortz2009)	Minimisation de la date de fin au plus tard	Heuristique d'approximation
(Masson2012)	Réduction du coût du transport	Programmation mathématique et Recherche à voisinage large
(Qu2012)	Minimiser le nombre de véhicules	Métaheuristique GRASP

TABLE 2.8 – Critères et méthodes utilisées pour résoudre le PDPT

### 2.5.2.1 Problème de collecte et livraison avec transbordement statique

Une variante du PDPT sans fenêtre de temps appelée problème de ramassage et distribution avec transbordement et fractionnement (PDPSL – Pickup and Delivery with Split Loads) a été proposée et résolue de manière exacte par (Kerivin2008). Dans ce problème, l'objectif est de réduire le coût du transport. Le fractionnement consiste à autoriser que le point de collecte des produits correspondant à une requête soit visité plusieurs fois par des véhicules qui chargeront chacun une fraction de la demande. Chaque véhicule pourra dès lors déplacer directement sa fraction de produits vers le point de livraison, ou effectuer un transbordement de cette fraction de produits dans un autre véhicule. Les auteurs ont développé un algorithme basé sur la méthode de Branch-and-Cut. Cela leur a permis de résoudre de façon optimale des instances contenant jusqu'à 15 requêtes en moins de 5 heures. L'observation des résultats leur a permis d'indiquer que le couplage du fractionnement et du transbordement pouvait permettre de réaliser des économies sur le coût du transport.

Toujours dans le cadre d'une méthode exacte pour résoudre le PDPT, (Cortes2010) ont fourni un programme mathématique en variables mixtes ainsi qu'un algorithme de type Branch-and-Cut pour résoudre un PDPT avec fenêtre de temps dans le cadre du transport de personnes. L'une des contraintes fortes considérées dans ce travail réside dans le fait que

chaque point de transbordement ne peut être visité au maximum qu'une fois par un même véhicule. Cela permet ainsi d'éviter d'avoir des tournées possédant des sous-cycles, dont les deux extrémités ne correspondent pas au dépôt. L'algorithme ainsi développé a permis de résoudre de manière optimale des instances contenant 6 requêtes, 2 véhicules et un point de transbordement. Toutefois, aucune évaluation des gains apportés par le transbordement n'a été fournie par les auteurs.

(Shang1996) ont modélisé un problème de transport de documents hospitaliers sous la forme d'un PDPT multicritères. Les différents critères à optimiser portent sur la réduction du nombre de véhicules à utiliser, les retards et la durée du transport. Une singularité du problème modélisé est que chaque point de collecte ou de déchargement d'un document peut être visité plusieurs fois et peut être utilisé comme un point de transbordement. Les auteurs ont développé une heuristique d'insertion basée sur la notion de mini route. Une mini route est une route permettant de satisfaire initialement un ensemble de requêtes partageant les mêmes points de collecte ou de livraison. Chacune des minis routes définies est par la suite affectée à un véhicule pour créer une tournée de base. Lorsque plusieurs tournées sont disponibles, l'insertion d'une requête se fait dans celle offrant le meilleur coût. Dans le cadre de leur travail, le transbordement d'un produit n'est autorisé que si sa livraison sans transbordement entraînerait l'utilisation d'un véhicule supplémentaire et si et seulement si cela ne cause pas de temps d'attente au point de transbordement utilisé. Les résultats expérimentaux ont permis de montrer que par rapport à une planification manuelle des tournées, l'heuristique développée permettait de réaliser des gains sur les critères considérés. Toutefois, aucune comparaison des solutions avec et sans transbordement n'a été proposée.

Dans (Ortel2000) l'auteur réalise une étude théorique sur la complexité et sur la résolution exacte de plusieurs problèmes de transport dans lesquels le transfert de charges entre véhicule est autorisé. Ensuite, l'auteur s'est intéressé à un cas particulier de problème de collecte et livraison dans lequel un produit peut être transbordé au maximum une fois dans 2 points de transbordement prédéfinis. Ainsi, il considère trois possibilités pour satisfaire une requête :

- Sans transbordement,
- Avec un transbordement dans le premier emplacement,
- Avec un transbordement dans le deuxième emplacement.

Le problème défini ne pouvant être résolu de manière exacte pour des problèmes de grandes tailles, un algorithme basé sur la recherche tabou couplée à la résolution du « k-star hub problem » (Blasum2007) a été développé. Le « k-star hub problem » est un problème qui consiste à déterminer dans un graphe donné un ensemble de chemins de coût réduit qui permettent de satisfaire un ensemble de requêtes. Dans le graphe considéré, chaque requête peut être satisfaite en transportant directement le produit associé à la requête de son point de collecte vers son point de livraison ou de son point de collecte à son point de livraison via le passage par un point de transbordement. Les tests expérimentaux ont été effectués sur des instances provenant d'un constructeur automobile et contenant pour certaines jusqu'à 70 requêtes. L'analyse des résultats ont permis de mon-

trer l'intérêt de pratiquer le transbordement et l'efficacité de l'algorithme développé.

(Aldaihani2003) ont travaillé sur un problème de transport dont l'objectif était de définir une intégration optimale entre le système de transport public (i.e. ligne de bus) et le système de transport par taxis qui est axé sur du porte à porte pour récupérer des personnes à domicile et les amener à destination. L'objectif de l'étude était de réduire la distance de parcours de tous les véhicules sans dégrader de façon significative la qualité de service des clients caractérisée par le temps d'attente aux arrêts de bus ou les retards. Pour étudier efficacement le problème posé, les auteurs l'ont abordé sous l'angle d'un PDPT avec fenêtre de temps dans lequel chaque personne à transporter possède deux alternatives de trajet. Elle peut être directement transportée de chez elle à sa destination par un taxi. Elle peut alternativement utiliser un transport mixte : elle utilise au maximum les transports en commun (bus) tout en ayant recours à la solution du taxi pour aller des points de départ et de destination aux arrêts de bus sélectionnés par l'usager. Le caractère combinatoire du problème a motivé les auteurs à développer une recherche tabou pour le résoudre. Des tests expérimentaux ont été effectués sur des instances réelles d'une ville de l'état de Californie contenant pour certaines jusqu'à 155 requêtes et 12 véhicules. L'analyse des résultats obtenus a permis de mettre en évidence l'avantage d'une telle intégration en termes de baisse du nombre de véhicules utilisés, et de réduction de la durée du trajet des clients.

Dans (Mitrovic2006) les auteurs construisent une heuristique en deux phases pour résoudre un PDPT de manière approchée avec une flotte de véhicules homogène et de capacité infinie. La première phase de l'heuristique permet de construire une solution initiale. Il s'agit d'une procédure d'insertion de coût minimum à départs multiples. Les départs sont multiples dans le sens où cette phase est constituée de plusieurs itérations, et à chacune d'elles, une solution est construite en générant aléatoirement des tournées réalisables permettant de satisfaire l'ensemble des requêtes. La meilleure solution trouvée durant cette phase est considérée comme solution initiale. Dans la deuxième phase, la solution initiale est améliorée par une procédure qui supprime et réinsère successivement chaque requête. Dans la première et la deuxième phase, l'insertion ou la satisfaction d'une requête se fait suivant deux alternatives. Soit un même véhicule satisfait la requête, soit elle est satisfaite par deux véhicules après passage par un point de transbordement si cela permet de réduire le coût de son transport. Des tests expérimentaux ont été effectués sur des instances générées de façon aléatoire avec la distance de Manhattan et contenant jusqu'à 100 requêtes. L'analyse des résultats laisse apparaître que le transbordement permet de réduire la distance totale avec des gains de l'ordre de 40 %.

Dans sa thèse de doctorat, (Fournier2008) a contribué à l'amélioration d'un outil d'aide à la décision appelé ILOP TPO. Il s'agit d'un logiciel aidant les entreprises à générer des tournées de grandes tailles nécessaire à la satisfaction des demandes de transport. Le processus de résolution utilisé par le logiciel est basé sur la recherche locale et exploite le transbordement comme levier d'optimisation. La contribution de l'auteur a été d'améliorer le mécanisme de résolution tant dans sa capacité à trouver de meilleures solutions que dans la maîtrise du temps de calculs. L'auteur

a proposé une modélisation linéaire du problème pour résoudre des instances de petites tailles avec la technique du Branch and Cut. Pour des instances de grandes tailles, l'auteur propose un mécanisme de coopération entre le logiciel ILOG TPO et une relaxation d'une modélisation en variables mixtes du problème (MIP). La coopération se déroule en deux phases. Dans la première phase, pour chaque ordre de transport, le MIP fournit à ILOG TPO un chemin par lequel il devrait être acheminé en utilisant éventuellement le transbordement. Le chemin proposé est amélioré dans une seconde phase par la recherche locale mise en œuvre dans le moteur du logiciel.

Des heuristiques d'approximation ou à garantie de performances ont été développées dans (Gortz2009) pour résoudre un PDPT sans fenêtre de temps. L'objectif du problème à résoudre était de minimiser le temps de service de chaque véhicule (makespan). Les auteurs ont analysé le cas traitant des capacités de véhicules finies et le cas avec des capacités de véhicules infinies. Aucun calcul expérimental n'a été effectué par les auteurs.

Dans sa thèse de doctorat, (Masson2012) s'intéresse au problème de collecte et livraison avec transbordement pour le transport à la demande. Les travaux de l'auteur concernent deux approches de résolution qui sont la résolution exacte et approchée. Dans l'approche exacte, l'auteur introduit des hypothèses de simplification afin de faciliter la résolution du problème par une technique de génération de colonnes. Un exemple d'une telle hypothèse est qu'un véhicule ne peut visiter plus de deux points de livraison dans sa tournée. Dans la méthode approchée, l'auteur a présenté une heuristique de recherche à voisinages larges dont l'efficacité a déjà été prouvée pour la résolution du PDP sans transbordement. La contribution de l'auteur a consisté à modifier les voisinages existants pour les adapter au PDPT et de proposer de nouveaux voisinages.

Tout récemment, pour résoudre le PDPT, (Qu2012) ont proposé une métahéuristique basée sur la méthode GRASP hybride à une recherche adaptative à voisinage large semblable à celle développée dans (Ropke and Pisinger 2006) pour résoudre le PDPTW. L'une des particularités du problème étudié réside dans le fait qu'un véhicule peut décharger et recharger un même produit au point de transbordement. À chaque itération du GRASP, une solution initiale est construite au moyen d'un algorithme qui permet d'insérer une nouvelle requête suivant deux alternatives, en ne considérant que la meilleure : la première consiste à insérer la requête sans transbordement, la deuxième consiste à choisir parmi tous les points de transbordement celui qui permet de réduire au maximum le coût du transport. Ensuite, cette solution est améliorée par le biais de plusieurs heuristiques qui permettent de supprimer et réinsérer des requêtes dans les tournées. Le choix de l'heuristique à utiliser pour supprimer ou réinsérer une requête est effectué en fonction des performances antérieures de celle-ci. Pour évaluer leur algorithme, les auteurs ont générés des instances contenant jusqu'à 25 requêtes avec un seul point de transbordement. Chaque instance était construite de manière à connaître par avance la solution optimale. Les résultats obtenus par l'algorithme développé par les auteurs avaient en moyenne un écart de 0.18 % par rapport à l'optimum.

### 2.5.2.2 Problème de collecte et livraison avec transbordement dynamique

(Thangiah2007) ont travaillé sur la version dynamique du problème décrit dans (Shang1996) en utilisant la stratégie de reconfiguration basique. Ils ont rajouté en plus un mécanisme de recherche locale pour réduire le nombre de véhicules utilisés. La recherche locale consiste à identifier les tournées possédant le moins de clients, pour supprimer si possible celles-ci par insertion des clients qu'elles contiennent dans les tournées de plus grande longueur i.e. possédant le plus grand nombre de clients.

Un autre papier qui apporte une contribution significative dans la résolution du problème de collecte et livraison avec transbordement dynamique est (Bouros2011). Le problème est considéré comme dynamique du fait de la possibilité d'accepter de nouvelles requêtes de transport pendant que les tournées sont déjà en cours d'exécution. Pour traiter ces nouvelles requêtes, une mise à jour du plan de distribution en cours d'exécution est effectuée à l'aide de la stratégie basique. Pour résoudre le problème posé, les auteurs ont construit un graphe dynamique. Le graphe est dynamique dans la mesure où sa taille peut diminuer ou augmenter au cours du temps. Ainsi, les sommets visités par un véhicule sont progressivement supprimés du graphe et les nouveaux sommets correspondant au point de collecte et livraison des nouvelles requêtes sont ajoutés de la même manière au graphe. La pondération du graphe dynamique ainsi construit repose sur la notion d'action. Les actions possibles sont : le chargement, le transport, le transbordement et le déchargement. À chaque action est associée un coût de réalisation. La pondération d'un arc dépend du nœud aval de celui-ci. Ainsi, si le nœud aval d'un arc consiste à un chargement, le coût de l'arc sera équivalent au coût de l'action correspondante. En résumé, la résolution se passe en trois étapes : arrivée d'une nouvelle requête, mise à jour du graphe dynamique, pondération du graphe via les actions. Les auteurs ont démontré que les algorithmes classiques de recherche du plus court chemin tel que celui de Dijkstra ne pouvaient pas s'appliquer sur le graphe défini et ont proposé une nouvelle heuristique de recherche du plus court chemin. Des expériences menées sur des instances issues du monde réel ont permis aux auteurs de conclure que leur heuristique était relativement plus rapide que les heuristiques classiques.

## 2.6 CONCLUSION

Après l'introduction, ce chapitre dédié à l'état de l'art est structuré en cinq parties. Le deuxième et la troisième partie ont permis de présenter les aspects généraux des problèmes de routage à travers une classification de quelques problèmes ainsi que la présentation des principes généraux des méthodes employées pour les résoudre. La quatrième partie était principalement axée sur le problème de collecte et livraison. La cinquième partie a commencé par mettre en lumière les singularités apportées par la pratique du transbordement, suivi par la présentation d'un ensemble de travaux sur le PDPT ou connexes à celui-ci.

Après avoir analysé les travaux sur le PDPT, il est facile de constater que contrairement aux problèmes de collecte et livraison, peu de travaux ont été réalisés sur le PDPT statique et sur le PDPT dynamique.

À notre niveau de connaissance de la littérature seulement deux articles existent sur le PDPT dynamique. Toutefois, ces dernières années l'intérêt de l'étude du transbordement comme moyen de flexibilité qui peut apporter des gains dans le transport est de plus en plus marqué. Dans tous les papiers recensés, le transbordement est utilisé dans une logique opportuniste. Cela signifie que le transbordement d'un produit n'est pas obligatoire et n'est pratiqué que si cela permet de réduire le coût du transport. Une autre lecture qui peut être faite sur l'analyse des différents travaux sur le transbordement est que la complexité du problème à résoudre pousse généralement à poser des hypothèses restrictives qui tendent à éloigner le problème d'un contexte d'application réelle. C'est le cas par exemple des hypothèses dans lesquelles la capacité des véhicules est supposée infinie ou lorsque les clients n'ont pas de fenêtre de temps.

La synthèse de toutes les analyses précédentes permet d'affiner les différentes hypothèses de travail à considérer dans cette thèse. Ainsi le problème étudié dans le cadre de cette thèse se définit aisément en utilisant les grilles d'analyse proposées. Les éléments renseignés dans le tableau 2.9 circonscrivent le champ d'étude du présent travail.

Travaux étudiés	Produits		Dépôts		Véhicules			Fenêtres de temps	Nature du service	Nombre de visites	Synchronisation	Consolidation
	Nombre	Capacité	Flotte									
Unique	Multiples	Unique	Multiples	Limité	Illimité	Limittée	Homogène	Hétérogène	Avec	Sans	Livraison	Collecte et Livraison
présente thèse	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

TABLE 2.9 – Positionnement de la thèse par rapport à la grille d'analyse définie

Il convient de noter les points particuliers suivants :

- La diversité de la nature des produits transportés n'influe pas sur la modélisation du problème étudié. En effet, ce dernier de par sa nature « one-to-one » est déjà fortement contraint par la définition pour chaque requête de transport d'un unique fournisseur vers un unique destinataire.
- Le nombre de dépôts considéré dans notre travail est principalement limité à un seul lieu. Cependant, les approches de résolution développées dans cette thèse seront appliquées dans le cadre du chapitre 5 à la reconfiguration d'un problème de tournées nécessitant de considérer une évolution vers la résolution d'un problème de type multi-dépôts.
- L'ensemble des clients n'est visité qu'une seule fois dans l'optique d'une collecte ou d'une livraison de produits.
- Le point de transbordement peut être visité plusieurs fois par des véhicules différents. En effet, deux véhicules au moins sont nécessaires pour transborder un produit.
- Le principe de consolidation n'est pas considéré dans notre travail puisqu'applicable aux seuls problèmes de tournées avec Cross-Docking. De ce fait, tous les véhicules ne sont pas obligés de passer au point de transbordement. Le transbordement d'un produit n'est autorisé que si cela entraîne une réduction du coût du transport.
- Si les travaux abordés dans cette thèse se rapprochent de ceux étudiés dans (Mitrovic2006), (Fournier2008), (Masson2012) et (Qu2012), il convient de noter que le transbordement est surtout utilisé soit dans un contexte de tournées de véhicules mais non spécifiquement PDP(Fournier2008), soit orienté transport de personnes à la demande(Masson2012). La fonction objectif ne concerne que la réduction de la distance parcourue (Qu2012). Pour tous les papiers cités, aucun n'utilise la VNS comme métaheuristique permettant de trouver une solution optimale au PDPT.

Les critères d'optimisation et méthodes de résolutions développés sont synthétisés dans le tableau suivant :

Travaux étudiés	Critères d'optimisation étudiés	Méthodes de résolution développées
<ul style="list-style-type: none"> <li>- Planification</li> <li>- Reconfiguration</li> </ul>	<ul style="list-style-type: none"> <li>- Minimisation du nombre de véhicules</li> <li>- Minimisation de la distance</li> </ul>	<ul style="list-style-type: none"> <li>- Modélisation linéaire en variables mixtes et résolution avec le solveur commercial XPRESS MP et Heuristiques</li> <li>- Recherche à voisinage variable (VNS)</li> <li>- Méthode basique utilisée pour la reconfiguration des tournées</li> </ul>

TABLE 2.10 – Positionnement de la thèse par rapport aux critères d'optimisation et méthodes de résolutions.

La fonction objectif et les méthodes de résolution présentées dans ce tableau seront précisément décrites dans la suite de ce mémoire suivant le contexte de résolution choisi : un contexte de résolution exact via la

proposition d'un modèle en programmation linéaire est présenté au chapitre 3, puis un contexte approché utilisant des heuristiques est décrit au chapitre 4.

# MODÉLISATION DU PROBLÈME DE COLLECTE ET LIVRAISON AVEC TRANSBORDEMENT

3

## SOMMAIRE

3.1	INTRODUCTION . . . . .	77
3.2	CARACTÉRISATION DU PDPT . . . . .	77
3.2.1	Le concept de transbordement . . . . .	77
3.2.2	Exemple de problème de tournées avec transbordement .	78
3.3	MODÉLISATION DU PROBLÈME . . . . .	80
3.3.1	Hypothèses du problème étudié . . . . .	80
3.3.2	Formalisation du PDPT . . . . .	80
3.3.3	Du modèle PDP au modèle PDPT . . . . .	81
3.4	TESTS EXPÉRIMENTAUX . . . . .	87
3.4.1	Objectifs de l'expérimentation . . . . .	88
3.4.2	Description de la méthode de génération de problèmes PDPT . . . . .	88
3.4.3	Analyses expérimentales sur la base d'instances simples .	92
3.5	CONCLUSION . . . . .	94



## 3.1 INTRODUCTION

DANS le chapitre 2, le problème de collecte et distribution avec transbordement (PDPT) a été présenté comme un problème connexe du problème PDP faisant partie de la classe des problèmes d'optimisation NP-difficiles. À ce titre, le PDPT est lui-même NP-difficile et l'introduction du transbordement dans le PDP induit des contraintes particulières qu'il nous faut présenter dans ce chapitre.

Pour résoudre des instances de PDPT de grandes tailles, il est nécessaire d'avoir recours à des méthodes approchées heuristiques ou métahéuristiques. Cependant, avant d'envisager de développer de telles méthodes approchées, il nous faut au préalable fixer les hypothèses de travail pour cadrer le périmètre de l'étude et définir précisément les caractéristiques du PDPT. Dans cet objectif, une formalisation mathématique du problème basée sur un modèle de programmation linéaire en variables mixtes est proposée dans ce chapitre. La résolution de ce modèle par une méthode exacte via l'utilisation d'un solveur permet de connaître la solution optimale sur des instances de petites tailles qui serviront à valider le modèle.

La suite de ce chapitre de thèse est divisée en 4 sections : la deuxième section est axée sur la caractérisation du problème de tournées ; elle présente le concept de transbordement appliqué aux tournées de véhicules et fournit un exemple illustratif permettant de mettre en évidence les singularités du problème étudié. La troisième section est consacrée à la présentation du modèle analytique représentant le PDPT en mettant en exergue les différentes contraintes induites par la considération du principe de transbordement. Le modèle PDPT ainsi construit sera positionné par rapport aux travaux de (Cortes2010), qui sont aujourd'hui à notre connaissance, les seuls à proposer un modèle linéaire pour caractériser un tel problème. La quatrième section regroupera les tests expérimentaux effectués sur la base du modèle développé afin de valider ce dernier d'une part, et de tester, d'autre part, les gains qu'amènent le transbordement sur des instances de petites tailles. Une méthode de générations d'instances construite dans ce sens sera proposée. La cinquième et dernière section permet de conclure le chapitre.

## 3.2 CARACTÉRISATION DU PDPT

Cette section débute par un exemple graphique qui permet d'exposer le principe du transbordement appliqué aux tournées de véhicules.

### 3.2.1 Le concept de transbordement

Dans la plupart des problèmes de collecte et distribution (Berbeglia2007), les véhicules partent d'un ou de plusieurs dépôts à vide, visitent un ensemble de clients « fournisseurs » pour y collecter des produits qu'ils vont ensuite distribuer chez des clients « consommateurs ». Il peut arriver, pour des raisons d'optimisation de coûts d'exploitation, ou suite à l'apparition d'aléas telles que les pannes ou une congestion de trafic, qu'un véhicule décharge temporairement tout ou partie de son chargement en un point autre que celui correspondant aux destinataires

des charges transportées. Ce même véhicule peut charger de nouveaux produits en ce lieu, pendant que les produits précédemment déchargés sont pris en charge par un autre véhicule. Afin de mieux comprendre les principes liés au transbordement, intéressons-nous à un exemple, tel que décrit par la figure 3.1. En début de leur tournée respective, le véhicule  $k_1$  (respectivement  $k_2$ ) charge les produits  $p_1$ ,  $p_2$ ,  $p_5$  (respectivement –  $p_3$  et  $p_4$ ). Considérant la nécessité au sens de l'optimalité de réaliser un transbordement en un lieu bien défini, le véhicule  $k_1$  décharge les produits  $p_1$  et  $p_2$ , et le véhicule  $k_2$  fait de même avec le produit  $p_4$ . Une fois les déchargements effectués,  $k_1$  charge le produit  $p_4$  afin de l'amener à destination et  $k_2$  charge les produits  $p_1$  et  $p_2$  pour en faire de même. Cette situation caractérise le transbordement des produits  $p_1$ ,  $p_2$  et  $p_4$ . Notons que le transbordement, lorsqu'il consiste à faire transférer une charge d'un véhicule vers un autre, ne requiert pas obligatoirement la présence simultanée des deux véhicules sur le lieu, la charge pouvant être momentanément stockée.

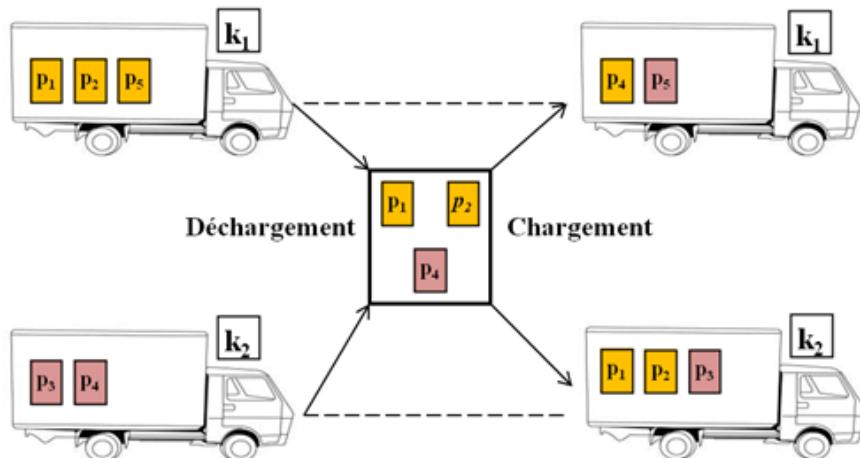


FIGURE 3.1 – Le principe du transbordement

Dans la sous-section suivante, le concept du transbordement appliqué à un contexte de tournée de ramassage et distribution est présenté de manière plus précise à travers un nouvel exemple.

### 3.2.2 Exemple de problème de tournées avec transbordement

Pour illustrer les avantages que peut induire la pratique du transbordement dans un problème de ramassage et distribution, un exemple est présenté dans cette partie.

Cet exemple présenté en figure 3.2 est issu de (Mitrovic2006). Il est composé d'un ensemble de 3 requêtes à satisfaire, sachant que chaque requête est définie par un couple  $(i, i + n)$  où «  $i$  » désigne le client auprès duquel la collecte se fait, «  $i + n$  » le client à livrer et  $n$  représente le nombre de requêtes. Le problème peut être modélisé par un graphe dans lequel :

- Chaque sommet représente un client à visiter, que ce soit pour une collecte ou livraison. Il est caractérisé par une fenêtre de temps, notée  $[borneminimale, bornemaximale]$  précisant l'intervalle de temps pendant lequel le client associé au sommet doit être visité.

- Chaque arc exprime un trajet possible entre deux clients ; la pondération associée à cet arc correspond à la distance qui sépare les deux clients exprimée en unité de distance (U.D.).
- Dans cet exemple, le dépôt de départ et de retour de chaque véhicule ne sont pas identiques.

Ainsi, les trois requêtes à satisfaire,  $1 \leq i \leq n$ ,  $n = 3$ , correspondent aux couples de sommets  $(1, 4)$ ,  $(2, 5)$ ,  $(3, 6)$ . Les clients 2 et 3 (respectivement 4 et 5) sont localisés au même endroit si bien que la distance les séparant est jugée négligeable. La durée de service en chaque sommet de collecte ou de livraison, ainsi qu'au point de transbordement est supposée nulle. Pour satisfaire l'ensemble des requêtes, trois véhicules de vitesse égale à 1 et de capacité supposée infinie sont disponibles. Les deux premiers véhicules  $k_1$  et  $k_2$  sont respectivement localisés chez le client représenté par le sommet 2 pour le premier et chez le client représenté par le sommet 3 pour le second. Le troisième véhicule  $k_3$  se trouve chez le client modélisé par le sommet 1.

- Si le transbordement n'est pas autorisé, 3 véhicules sont nécessaires pour satisfaire l'ensemble des requêtes du fait de la nécessité de respecter les fenêtres de temps. Le véhicule  $k_1$  effectue le trajet de 2 vers 5, le véhicule  $k_2$  charge au sommet 3 et livre au sommet 6, et le véhicule  $k_3$  se déplace après chargement en 1 vers le sommet 4. La distance totale parcourue par tous les véhicules est donc de  $4 + \sqrt{2}$ .
- Si le transbordement est autorisé, seuls deux véhicules sont nécessaires pour satisfaire toutes les requêtes. Le véhicule  $k_1$  initialement situé au sommet 2, effectue la séquence de visite  $\{2, 3, t, 6\}$ , pendant que le véhicule  $k_3$  initialement situé au sommet 1 effectue le trajet  $\{1, t, 5, 4\}$ . Dans cette nouvelle solution,  $k_1$  collecte des produits aux sommets 2 et 3, dépose au point de transbordement  $t$  le produit se destinant au point 5, et termine sa course en livrant le produit chargé en 3 au point 6. Le véhicule  $k_3$  part du point 1 où il récupère une charge, se dirige au point de transbordement  $t$  pour charger le produit préalablement déposé par le véhicule  $k_1$  et termine de livrer son contenu aux sommets 4 et 5. La distance totale parcourue par tous les deux véhicules est de 4

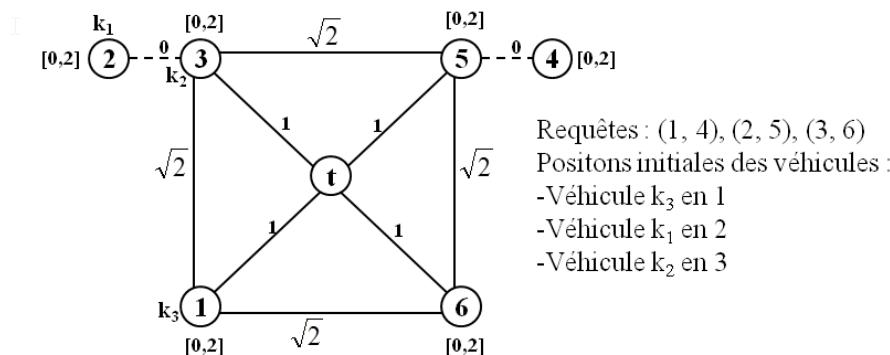


FIGURE 3.2 – Impact du transbordement sur la minimisation des véhicules et de la distance

À travers cet exemple, l'intérêt de pratiquer le transbordement a pu être observé tant sur la réduction de la distance parcourue que sur la ré-

duction du nombre de véhicules utilisés. Afin de pleinement tirer partie des gains potentiels issus de cette pratique, il est nécessaire de comprendre les contraintes régissant son fonctionnement. À cet effet, le problème de collecte et livraison avec transbordement est caractérisé, formalisé et modélisé sous la forme d'un modèle analytique.

### 3.3 MODÉLISATION DU PROBLÈME

Pour modéliser le PDPT, un ensemble d'hypothèses de travail est considéré pour limiter la complexité du problème étudié.

#### 3.3.1 Hypothèses du problème étudié

Le problème étudié dans le cadre de cette modélisation s'appuie sur les hypothèses et caractéristiques suivantes :

- La flotte de véhicules est considérée comme homogène et chaque véhicule est à capacité finie.
- Le nombre de véhicules disponibles est supposé infini.
- Il n'existe qu'un seul dépôt d'où partent et reviennent tous les véhicules.
- Il existe un seul et unique point de transbordement, distinct des lieux de collecte et livraison.
- Chaque produit peut être transbordé une et une seule fois. Cela signifie que durant son transport, de son point de ramassage à son point de livraison, un produit ne peut être déchargé et rechargé au plus une seule fois.
- Tout lieu visité par un véhicule possède une fenêtre de temps. Ainsi, tout véhicule doit arriver en un point à une date comprise dans la fenêtre de temps. Lorsque l'arrivée se fait de manière anticipée, le véhicule doit attendre l'ouverture de la fenêtre de temps pour débuter son service au client : les fenêtres de temps sont toutes « strictes ».
- Le problème à optimiser est étudié sous l'angle d'une minimisation du nombre de véhicules utilisés et de la distance totale parcourue. La fonction objectif utilisée n'intègre pas de composante liée à une pénalité relative à un retard. En effet, les fenêtres de temps sont « strictes » et garantissent la ponctualité de toutes les visites.

#### 3.3.2 Formalisation du PDPT

Le PDP a été présenté et défini de façon formelle au chapitre précédent. Le PDPT étant un problème connexe au PDP, les notations et définitions qui ont été précédemment introduites sont conservées. Ces notations sont rappelées dans les tableaux 3.1 et 3.2 ci-après. Néanmoins, de nouvelles variables et de nouveaux paramètres sont à considérer dans la description formelle du principe du transbordement.

La principale spécificité du PDPT concerne le point de transbordement qui peut autoriser successivement la réalisation d'une opération de chargement et d'une opération de déchargement. Du point de vue de la modélisation, pour séquencer ces deux actions, le point de transbordement est représenté dans le graphe par deux sommets  $o_3$  et  $o_4$ . Cette modélisation

est illustrée en figure 3.3. Lorsqu'un véhicule arrive au point de transbordement, il atteint le sommet  $o_3$  pour y décharger éventuellement un ensemble de produits ; il se dirige ensuite vers le sommet  $o_4$  où il charge éventuellement un ensemble de produits avant de quitter le point de transbordement. Les sommets  $o_3$  et  $o_4$  correspondent à un même lieu – le point de transbordement – si bien qu'ils sont séparés par une distance nulle, et le passage d'un sommet à l'autre se fait instantanément.

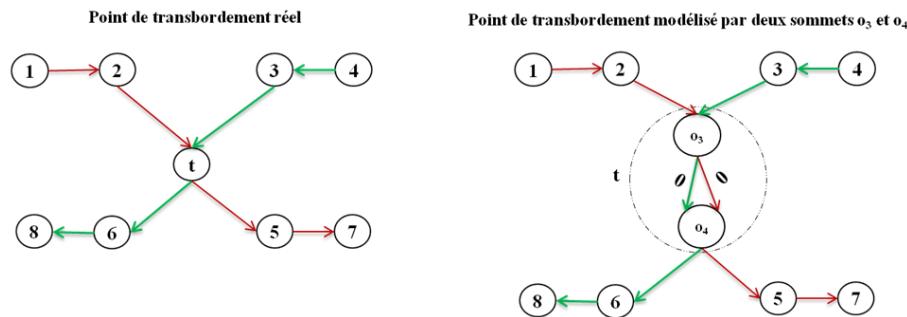


FIGURE 3.3 – Modélisation d'un point de transbordement

Sur la base de cette hypothèse de modélisation, l'ensemble des sommets  $V$  du graphe caractérisant le problème à résoudre devient égal à  $V = N \cup o_1 \cup o_2 \cup o_3 \cup o_4$ . Pour rappel,  $N$  représente l'ensemble des points de collecte et livraison,  $o_1$  et  $o_2$  sont les sommets associés au dépôt.

Le problème de ramassage et distribution avec transbordement est alors formellement caractérisé par un graphe complet non orienté  $G = (V, E)$  tel que  $E = V \times V$  est l'ensemble des arcs de  $G$ .

### 3.3.3 Du modèle PDP au modèle PDPT

Le modèle que nous proposons pour le PDPT (Tchapnga2012a)(Rais2014) est appelé *MO\_PDPT* (Modèle du problème de collecte et livraison avec transbordement). Il comporte un ensemble de paramètres et variables qui sont rappelés ou présentés dans le tableau 3.1.

Notations	Définitions
$t_{ij}$	Durée du trajet entre deux sommets du graphe $G$ avec $(i, j) \in V \times V$
$c_{ij}$	Coût d'utilisation de l'arc $(i, j) \in V \times V$ (assimilé dans notre cas à une distance)
$s_i$	Temps de service associé à chaque point visité $i \in V$
$[a_i, b_i]$	Fenêtre de temps associée au nœud $i \in V$
$l_i/l_{i+n}$	Quantité de produit ramassé/déposé en $i \in P / i + n \in D$
$K$	Ensemble homogène de véhicules de capacité finie
$R_i$	Requêtes de transport $i$
$R$	Ensemble des requêtes

TABLE 3.1 – Tableau des paramètres

Le temps de service  $s_i$  associé aux différentes variables modélisant le dépôt est défini comme suit :

- La durée de service associée aux dépôts  $o_1$  et  $o_2$  est considérée comme étant nulle, i.e.  $s_{o_1} = s_{o_2} = 0$ . En effet, dans une logique

prévisionnelle, il est supposé que les véhicules sont prêts au moment où ils quittent le dépôt  $o_1$  et qu'il n'y a pas de durée d'activité associée à leur retour au dépôt  $o_2$ , une fois leur tournée respective terminée.

- Aux points de transbordement  $o_3$ , (respectivement  $o_4$ ), nous faisons l'hypothèse que le temps de service associé à un produit (chargement ou déchargement) est identique au temps de service de ce même produit chez le client. En conséquence  $s_{o_3}$  (respectivement  $s_{o_4}$ ) est fonction de la quantité de produits à transborder qui sont contenus dans le chargement du véhicule. Ainsi, supposons par exemple qu'un véhicule  $k$  visite le sommet  $o_3$  (respectivement  $o_4$ ) pour y décharger (collecter) trois produits  $P1, P2, P3$ , respectivement chargés chez les clients 1, 2 et 3. La durée de service en  $o_3$  (respectivement  $o_4$ ) sera égale à la somme des temps de service de ces trois produits chez leur fournisseur respectif. Autrement dit :  $s_{o_1} = s_{o_2} = \sum_{i=1}^3 s_i$

Le modèle comporte 6 variables de décision dont les définitions sont rappelées dans le tableau 3.2.

Notations	Définition
$x_{ij}^k$	Variable binaire qui vaut 1 si le véhicule $k \in K$ parcourt l'arc $(i, j) \in V \times V$ et 0 sinon. Cette variable permet de connaître l'ensemble des arcs parcourus sur une tournée
$y_i^k$	Variable binaire qui vaut 1 si le client associé au sommet $i \in N$ est visité par le véhicule $k \in K$ ; et 0 sinon.
$dt_i^k$	Date d'arrivée du véhicule $k \in K$ au sommet $i \in V$ . Cette variable permet de dater le passage de chaque véhicule dans un sommet (client).
$L_i^k$	Charge du véhicule $k \in K$ au départ du sommet $i \in V$ i.e. la charge après service.
$u_i^k$	Variable binaire qui vaut 1 si le véhicule $k \in K$ doit décharger au point de transbordement les produits ramassés en $i \in P$ , et 0 sinon.
$r_i^k$	Variable binaire qui vaut 1 si le véhicule $k \in K$ doit charger au point de transbordement les produits ramassés en $i \in P$ par un autre véhicule, pour les livrer en $(i + n \in D)$ .

TABLE 3.2 – Tableau des variables de décisions

Le modèle de PDP fourni au chapitre 2 permet de satisfaire l'ensemble de requêtes sans transbordement. Pour obtenir le modèle *MO\_PDPT*, il est nécessaire d'introduire les possibilités de transferts de charges entre véhicules dans le modèle PDP. À cet effet, la contrainte (2.8) qui exprime que le point de collecte et le point de livraison associé à une requête soient visités par un même véhicule est relâchée.

- L'équation 2.12 du modèle classique de PDP, permettant de modéliser la nécessaire antériorité entre la collecte d'un produit et sa délivrance au client, doit être adaptée pour intégrer le principe du transbordement. En effet, le lieu de transbordement correspond à un point où une collecte et une livraison doivent être réalisées en séquence par deux véhicules différents, nécessitant en cela que les sommets du graphe associés à ce point puissent être visités plus

d'une fois. Ainsi, la contrainte de précédence 2.12 est remplacée par l'équation 3.1. Dans cette équation, l'utilisation de la somme permet de vérifier que la collecte du produit au point «  $i$  » est réalisée à une date antérieure de celle à laquelle la livraison est effectuée au point «  $i+n$  », quels que soient les véhicules en charge de réaliser ces opérations. Cette équation n'est valide que sur l'hypothèse que la date d'arrivée d'un véhicule ( $dt_i^k$ ) est égale à 0 pour tout client  $i$  non visité par un véhicule  $k$  considéré ; cette hypothèse est exprimée par l'équation 2.11.

$$\sum_{k \in k} dt_i^k \leq \sum_{k \in k} dt_{i+n}^k, \forall i \in P \quad (3.1)$$

- Un autre aspect important du transbordement porte sur le choix des produits à transborder pour réduire le coût du transport. L'équation 3.2 issue de (Wen2009) décrit un tel comportement. Le tableau 3.3 regroupe l'ensemble des combinaisons de valeurs possibles pour les différents termes de l'équation, afin d'en faciliter sa compréhension. Lorsqu'un véhicule  $k$  n'est pas en charge de servir la requête de transport  $(i, i+n)$ , le produit n'est ni déchargé ou rechargé par ce véhicule au point de transbordement  $t$  (cas 1). Si le véhicule  $k$  charge un produit en «  $i$  », et ne livre pas ce produit en «  $i+n$  » (cas 2), le produit a été obligatoirement déposé au point de transbordement pour qu'il soit ramassé et livré par un autre véhicule. Si ce véhicule  $k$  ne collecte pas un produit en «  $i$  » mais en assure sa livraison (cas 3), le produit préalablement déposé par un premier véhicule est chargé par le véhicule  $k$  au point de transbordement. Si le véhicule  $k$  collecte un produit en «  $i$  », et livre ce produit en «  $i+n$  » (cas 4), cela signifie que le véhicule  $k$  n'a ni à charger ni à décharger le produit au point de transbordement. La valeur des variables de décision  $u_i^k$  et  $r_i^k$  est ainsi dépendante de l'affectation des clients à visiter à chaque tournée pour chaque véhicule  $k$ . Il appartiendra au solveur lors de la recherche de la solution optimale de choisir les bonnes valeurs pour les variables  $u_i^k, r_i^k, x_{ij}^k$ .

	$\sum_{j \in V} x_{ij}^k$ ( $k$ collecte en $i$ )	$\sum_{j \in V} x_{i+n,j}^k$ ( $k$ livre en $i+n$ )	$u_i^k$ ( $k$ décharge en $o_3$ )	$r_i^k$ ( $k$ recharge en $o_4$ )
cas 1	0	0	0	0
cas 2	1	0	1	0
cas 3	0	1	0	1
cas 4	1	1	0	0

TABLE 3.3 – Mise en œuvre de la décision de transbordement

$$u_i^k - r_i^k = \sum_{j \in V} x_{ij}^k - \sum_{j \in V} x_{i+n,j}^k, \forall k \in K, \forall i \in P \quad (3.2)$$

- L'équation 3.3 permet d'assurer qu'un produit ne peut être déchargé et rechargé au point de transbordement par un même véhicule.

Ainsi, un véhicule ne peut pas déposer un produit au point de transbordement, pour aller servir d'autres clients avant de revenir ultérieurement le recharger pour l'acheminer vers son destinataire. Cette contrainte, quoi que restrictive mais limitant la combinatoire du problème, évite par ailleurs le stockage temporaire des produits au point de transbordement pouvant induire des coûts supplémentaires. Elle évite également la création de cycles au niveau du point de transbordement qui peuvent subvenir si un produit est chargé et déchargé plusieurs fois en un même sommet.

$$u_i^k + r_i^k \leq 1, \forall k \in K, \forall i \in P \quad (3.3)$$

- L'équation 3.4 exprime la limitation du nombre de transbordements possibles de chaque produit à une seule occurrence.

$$\sum_{k \in K} u_i^k \leq 1, \forall i \in P \quad (3.4)$$

- Le point de transbordement étant modélisé par un ensemble de deux sommets  $\{o_3, o_4\}$  distincts de l'ensemble  $N$  des clients à visiter, il faut déterminer la façon dont les véhicules rentrent et sortent de ce point. L'équation 3.5 exprime que tout véhicule rentre au point de transbordement par le sommet  $o_3$  et parcourt l'arc  $(o_3o_4)$ .

$$\sum_{i \in V} x_{io_3}^k = x_{o_3o_4}^k, \forall k \in K \quad (3.5)$$

- L'équation 3.6 stipule que tout véhicule qui rentre au point de transbordement en ressort obligatoirement par le sommet  $o_4$  après avoir parcouru l'arc  $(o_3o_4)$ .

$$\sum_{i \in V} x_{o_4i}^k = x_{o_3o_4}^k, \forall k \in K \quad (3.6)$$

- Pour assurer la cohérence des dates d'arrivée en n'importe quel point du graphe en prenant en compte la possibilité de faire du transbordement, les équations (3.7, 3.8, 3.9) modélisant les contraintes temporelles existantes au niveau de l'exécution des différents services sont proposées. L'équation (3.7) exprime que la date d'arrivée d'un véhicule chez un client est au moins égale à la somme entre : la date d'arrivée du véhicule chez le client qui le précède, la durée de service chez ce client et le temps nécessaire au parcours de la distance séparant les deux clients. L'équation 3.8) assure que la date d'arrivée d'un véhicule au sommet  $o_4$ , est au moins égale à la somme de deux composantes : la date d'arrivée du dit véhicule au sommet  $o_3$  du point de transbordement et la durée  $s_{o_3} = \sum_{i \in P} u_i^k \cdot s_i$  nécessaire pour décharger tous les produits de ce véhicule. L'équation 3.9 permet de dater l'arrivée d'un véhicule au sommet succédant directement le sommet  $o_4$  du point de transbordement. Cette

date est au moins égale à la somme de trois composantes. La première est la date d'arrivée du véhicule au sommet  $o_4$  du point de transbordement. La seconde est le temps nécessaire pour recharger chaque produit transbordé dans le véhicule. La troisième correspond à la durée nécessaire pour couvrir la distance qui sépare le point de transbordement du noeud qui lui succède dans la tournée.

$$x_{ij}^k = 1 \Rightarrow dt_j^k \geq dt_i^k + s_i + t_{ij}, \forall k \in K, \forall i \in N, \forall j \in N \cup \{o_1, o_2, o_3\} \quad (3.7)$$

$$x_{o_3 o_4}^k = 1 \Rightarrow dt_{o_4}^k \geq dt_{o_3}^k + \sum_{i \in P} u_i^k \cdot s_i, \forall k \in K \quad (3.8)$$

$$x_{o_4 j}^k = 1 \Rightarrow dt_j^k \geq dt_{o_4}^k + \sum_{i \in P} r_i^k \cdot s_i + t_{o_4 j}, \forall k \in K, \forall j \in N \cup o_2 \quad (3.9)$$

- Dans notre problème, la flotte de véhicule est homogène. La capacité finie et identique des véhicules constraint la manière d'exécuter les tournées en fonction des charges à collecter chez les clients. La gestion de la variation de charge des véhicules lors de leur passage chez les différents clients telle que définie dans le modèle classique de PDP n'est plus exacte dans un contexte de transbordement. En effet, la charge d'un véhicule peut être mise à jour alors même que le véhicule n'est pas à un point de collecte ou livraison. Pour garantir l'exactitude du modèle *MO\_PDPT*, il est nécessaire de prendre en compte l'impact du transbordement dans la variation de charge des véhicules. L'équation 3.10 assure que la charge d'un véhicule au départ d'un client est égale à la charge qu'il avait à son arrivée – i.e. la charge au départ du client précédent le client courant dans la tournée, plus la variation de charge liée au service à effectuer chez le client visité. Cette charge est positive en cas de chargement, négative sinon. L'équation 3.11 définit la variation de charge d'un véhicule au point de transbordement. Cette équation est active pour tout véhicule concerné par une action de transbordement. Dans ce cas, sa charge au départ du sommet  $o_4$  est égale à la charge qu'il avait à son arrivée au sommet  $o_3$ , moins la charge des produits qu'il décharge en  $o_3$ , plus la charge des produits qu'il collecte en  $o_4$ .

$$x_{ij}^k = 1 \Rightarrow L_j^k = L_i^k + q_j, \forall i \in N, \forall j \in V, \forall k \in K \quad (3.10)$$

$$x_{o_3 o_4}^k = 1 \Rightarrow L_{o_4}^k = L_{o_3}^k - \sum_{i \in P} u_i^k \cdot l_i + \sum_{i \in P} r_i^k \cdot l_i, \forall k \in K \quad (3.11)$$

- L'équation 3.12 permet de garantir que tout produit déposé au point de transbordement, est obligatoirement rechargé. Associé à l'équation 3.3, l'équation 3.12 permet d'assurer que le recharge est effectué par un véhicule différent de celui qui a déposé le produit.

$$\sum_{k \in K} u_i^k - \sum_{k \in K} r_i^k = 0, \forall i \in P \quad (3.12)$$

- Lorsque le produit est transbordé, il faut assurer la précédence entre sa collecte et son déchargement au point de transbordement, ainsi qu'entre son chargement au point de transbordement et sa livraison à son point de livraison. Afin de généraliser la contrainte de précédence, les équations 3.13, 3.14, 3.15 sont ajoutées au modèle. L'équation 3.13 assure l'antériorité de la date de collecte d'un produit à son point d'enlèvement par rapport à la date de son déchargement au point de transbordement. L'équation 3.14 décrit la même logique temporelle entre la date de chargement du produit au point de transbordement et la date de livraison du dit produit à son destinataire. L'équation 3.15 modélise la précédence temporelle entre opérations de déchargement et chargement au point de transbordement, quels que soient les véhicules en charge d'acheminer le produit vers son destinataire. Cette contrainte assure que la date à laquelle un véhicule  $k_2$  recharge un produit au point de transbordement est au moins supérieure à la somme de trois composantes : la date d'arrivée du véhicule  $k_1$  ayant déchargé le produit au point de transbordement, la durée nécessaire à  $k_1$  pour décharger l'ensemble des produits qui doivent être transbordés et qu'il transporte  $\sum_{k \in K} u_i^{k_1} \cdot s_i$  et le temps nécessaire pour parcourir l'arc  $o_3 o_4$ .

$$u_i^k = 1 \Rightarrow dt_{o_3}^k \geq dt_i^k, \forall k \in K, \forall i \in P \quad (3.13)$$

$$r_i^k = 1 \Rightarrow dt_{o_4}^k \leq dt_{i+n}^k, \forall k \in K, \forall i \in P \quad (3.14)$$

$$u_i^{k_1} + r_i^{k_2} = 2 \Rightarrow dt_{o_4}^{k_2} \geq dt_{o_3}^{k_1} + \sum_{i \in P} u_i^{k_1} \cdot s_i + t_{o_3 o_4}, \forall k_1, k_2 \in K, \forall i \in P \quad (3.15)$$

Enfin, les hypothèses générales qui ont été annoncées au chapitre 2 et ci-dessus, se traduisent de manière concrète par la fonction objectif suivante :

$$\text{Min } P * \sum_{k \in K} \left(1 - x_{o_1 o_2}^k\right) + \sum_k \sum_i \sum_j c_{ij}^k * x_{ij}^k \quad (3.16)$$

La fonction objectif est constituée de deux composantes. La composante  $\sum_{k \in K} (1 - x_{o_1 o_2}^k)$  de la fonction porte sur la minimisation du nombre de véhicules. La composante  $\sum_k \sum_i \sum_j c_{ij}^k * x_{ij}^k$  porte sur la minimisation du coût du transport, plus précisément de la distance totale parcourue. Il s'agit d'une fonction agrégée dans laquelle le coefficient de pondération  $P$  permet de prioriser la minimisation du nombre de véhicules par rapport à la distance parcourue. Ainsi, si deux solutions  $S_1$  et  $S_2$  ont le même nombre de véhicules,  $S_1$  sera déclarée meilleure que  $S_2$  si la distance totale parcourue dans  $S_1$  est inférieure à celle parcourue dans  $S_2$ . Toutefois,

si dans  $S_1$ , le nombre de véhicules utilisés est inférieur au nombre de véhicules utilisés dans  $S_2$ , alors  $S_1$  sera déclarée meilleure quelle que soit la distance totale parcourue dans  $S_2$ . La valeur du coefficient de pondération  $P$  est fixée de manière à ce que  $P$  soit au moins plus grand que la distance totale parcourue pour satisfaire toutes les requêtes. Dans la plupart des problèmes de collecte et livraison, la minimisation du nombre de véhicules est prioritaire par rapport à la réduction de la distance totale parcourue (Bent2006). En effet, le coût d'acquisition d'un véhicule est par exemple plus élevé que le coût du carburant nécessaire à son fonctionnement. Il convient de noter que certaines des équations précédemment présentées doivent être mises sous forme linéaire en s'inspirant de la même technique utilisée pour le modèle PDP décrit au chapitre 2. Le modèle *MO\_PDPT* ayant été construit sur la base du modèle PDP présenté au chapitre 2, une version complète de ce modèle est présentée en annexe A.1 de la thèse.

### 3.3.3.1 Originalité du modèle

Le modèle ainsi proposé se veut en partie original par rapport au seul modèle linéaire porté à notre connaissance dans les travaux issus de la littérature scientifique. Il convient donc de positionner notre proposition vis-à-vis des travaux de (Cortes2010). Cette comparaison peut se faire à plusieurs niveaux.

Dans notre modèle, la fonction objectif est une fonction qui priorise la réduction du nombre de véhicules au détriment de la réduction de la distance totale parcourue. En effet, la composante portant sur le nombre de véhicules est fortement pondérée du fait de l'utilisation du paramètre  $P$ . Dans leur modèle, la fonction objectif porte uniquement sur la réduction de la distance totale parcourue.

Une autre différence concerne la gestion du transbordement. Dans ces travaux, les auteurs ont modélisé l'exécution du transbordement par le biais d'une variable de décision binaire à 3 indices tandis que dans notre modélisation la gestion du transbordement est assurée par le biais de deux variables de décision binaires à deux indices. Par rapport à leur modèle, notre modélisation présente l'avantage de pouvoir connaître de façon immédiate si un produit a été transbordé ou pas. Il suffit à cet effet de consulter la valeur d'une des variables de décision ( $u_i^k, r_i^k$ ) que nous utilisons pour gérer le transbordement et pour déterminer si le produit a été chargé ou déchargé. Dans leur modélisation, la variable utilisée pour le transbordement ne permet pas à elle seule de disposer d'une telle information. Il faut en plus de cette variable examiner toutes les tournées des véhicules. De ce fait, en termes d'utilisation ou d'exploitation, le modèle présenté dans ce chapitre semble être plus efficace.

## 3.4 TESTS EXPÉRIMENTAUX

Le modèle analytique une fois défini, il convient d'en valider son comportement par le biais de la résolution d'un ensemble d'instances de très petites tailles, dont il est aisément de calculer la solution optimale. Cette validation étant acquise, la résolution d'un certain nombre d'instances de PDPT à l'aide d'un tel modèle permet de porter un premier jugement sur

la pertinence de l'utilisation du principe de transbordement en tant que levier d'optimisation.

### 3.4.1 Objectifs de l'expérimentation

La littérature scientifique ne fait pas état de nombreuses instances de problèmes de types PDPT. Lorsque ces instances existent, elles ne sont généralement pas détaillées dans les travaux décrits. L'absence d'instances associée à la nécessité de valider le modèle proposé nous conduit à proposer un ensemble d'instances simples caractérisant les problèmes de faible complexité. Celles-ci sont construites sur la base d'une méthode de génération de problèmes présentée ci-après. Chaque instance est définie de manière à connaître la solution optimale en termes de nombre de véhicules nécessaires et la distance totale parcourue. Chacune des instances générées sera résolue à l'aide du modèle *MO\_PDPT* implémenté dans un solveur commercial. Les résultats ainsi obtenus par le solveur seront comparés aux résultats connus par avance pour chaque instance, afin de tester la validité du modèle et mesurer l'intérêt de pratiquer le transbordement.

### 3.4.2 Description de la méthode de génération de problèmes PDPT

Les clients sont représentés dans un espace cartésien. Ainsi, soient  $(x_A, y_A)$  et  $(x_B, y_B)$  les coordonnées des points *A* et *B* représentant la localisation de deux clients dans cet espace, la distance cartésienne entre ces points est obtenue par le calcul suivant :

$$\text{Distance} = \sqrt{(x_A - x_B)^2 + (y_A - y_B)^2}$$

Pour faciliter le processus de génération d'une instance, les hypothèses suivantes sont considérées :

- La vitesse des véhicules est constante et égale à 1, afin d'assimiler le temps à une distance. Le choix de la vitesse est essentiel dans la définition de valeurs cohérentes pour les bornes des fenêtres de temps associées à chaque client, afin de garantir l'existence d'au moins une solution ;
- Les données caractérisant la distance (et donc le temps) de parcours entre deux points sont indépendantes du sens de visite des points et de valeur constante. Les matrices de temps et de distance sont donc symétriques ;
- Les temps de service au point de transbordement et en tout point de collecte ou livraison sont supposés nuls. Cette hypothèse simplifie la résolution du problème sans limiter les conclusions quant à la validité du modèle.

#### 3.4.2.1 Principe général

Afin de faciliter la description du principe, la figure 3.4 est proposée. Pour générer un ensemble de requêtes dont la solution optimale avec et sans transbordement peut être connue par avance, nous nous appuyons sur le postulat suivant : soit un ensemble de 4 clients  $p_1, d_1, p_2, d_2$ ; définissant deux requêtes de transports :  $(p_1, d_1)$  et  $(p_2, d_2)$ . Ces clients sont positionnés tels que leurs coordonnées dans le plan cartésien définissent deux droites sécantes ; considérant le point d'intersection ( $t$ ) de

ces droites comme le lieu de transbordement, et un ensemble de deux nouveaux clients  $p3$  et  $d3$  définissant une nouvelle requête  $(p3, d3)$ , satisfaisant aux hypothèses suivantes :

- Les nouveaux clients sont distincts des précédents ;
- Les coordonnées de chaque client satisfont l'équation de l'une des deux droites telle que définie précédemment,
- Soit  $(U)$  une droite verticale représentée sur la figure 3.4. En translatant  $(U)$  parallèlement à  $(p1, d2)$  ou  $(p2, d1)$ ,  $p3$  représente le point d'intersection entre  $(U)$  et  $(d2, p2)$ , tandis que  $d3$  représente le point d'intersection entre  $(p1, d1)$  et  $(U)$ .

L'ajout de ce couple de clients ne conduit pas à une modification de la solution optimale dans une recherche de tournée avec transbordement visant à minimiser la distance totale parcourue. Seules des hypothèses ayant trait à la définition de fenêtres de temps imposant la simultanéité de certaines actions ou le non usage du transbordement peuvent influencer la valeur de la solution optimale obtenue.

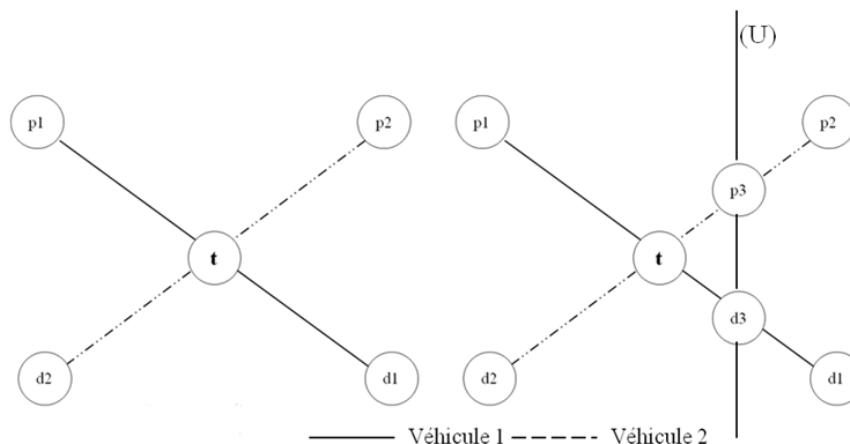


FIGURE 3.4 – Principe de base de construction d'une instance

La figure 3.4 représente un exemple d'insertion d'une nouvelle requête. Le graphique de gauche présente une situation dans laquelle 2 véhicules sont utilisés pour satisfaire deux requêtes. Le premier véhicule suit le trajet en trait plein, collecte un produit en  $p1$  pour le livrer en  $d1$ , tandis que le second véhicule collecte un produit en  $p2$  pour le livrer en  $d2$ . Une requête supplémentaire doit être satisfaite. Elle consiste à insérer le point de collecte  $p3$  et le point de livraison  $d3$  d'un produit dans les tournées existantes en respectant toutes les contraintes ou de créer une tournée spécialement pour la satisfaire en cas d'insertion impossible sans violation de contraintes.

La solution nécessaire à la satisfaction des requêtes  $(p1, d1)$  et  $(p2, d2)$  consiste en l'utilisation de deux véhicules pour une distance totale parcourue de  $DIST$ . Il est possible de satisfaire la requête  $(p3, d3)$  sans varier le coût de cette solution. En effet, le véhicule qui suit le trajet représenté en trait interrompu peut collecter sans détour  $p3$  durant son cheminement pour aller collecter  $p2$ . Après cette collecte, il dépose les deux produits collectés au point de transbordement  $t$ . Après avoir collecté le produit  $p1$ , le véhicule qui suit le trajet en trait plein s'arrête au point de transbordement  $t$ , y charge  $p3$  et  $p2$  pour respectivement les décharger aux points de

labels  $d3$  et  $d2$ . De cette manière, une requête supplémentaire est satisfaite en conservant le coût du transport.

La méthode que nous proposons pour générer les instances consiste alors à systématiser le principe de construction précédemment présenté.

Pour en faciliter la généralisation, les instances générées s'appuient sur l'utilisation d'un carré tel que présenté en figure 3.5, de côté de longueur égale à  $2a$  ( $a > 0$ ) et l'association d'un client à chaque sommet de ce carré satisfait au postulat précédemment énoncé. Les diagonales reliant un couple de sommets sont sécantes en un point définissant le centre du carré, et caractérisant conjointement le dépôt des véhicules et le point de transbordement.

Lorsque le centre du carré est confondu avec l'origine d'un repère orthonormé permettant de caractériser les coordonnées de tout point de l'espace cartésien, les quatre sommets  $A$ ,  $B$ ,  $C$  et  $D$  du carré ont pour coordonnées :  $A(-a, a)$ ,  $B(a, a)$ ,  $C(a, -a)$  et  $D(-a, -a)$ . Le dépôt  $o$  des véhicules, ainsi que le point de transbordement  $t$  sont situés à l'origine du repère orthonormé. L'ajout d'un nouveau couple de points (nouvelle requête), répondant aux hypothèses décrites plus haut, se fait au travers de la définition d'une droite ( $U$ ) parallèle à l'axe des ordonnées du repère orthonormé et positionnée à une distance  $d$  ( $F, E$ ) quelconque de la droite ( $BC$ ). Cette droite ( $U$ ) est alors sécante avec les deux diagonales du carré aux points d'intersection qui déterminent les coordonnées des nouveaux clients à insérer dans le problème.

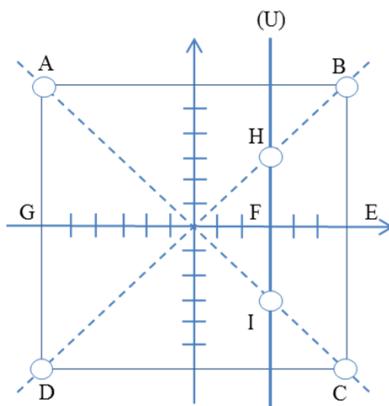


FIGURE 3.5 – Le principe de construction d'une instance

Afin de respecter les logiques de précédence entre opérations de collecte et de livraison, et de pouvoir insérer toute nouvelle requête sans risquer de modifier la valeur de la solution optimale, le carré est divisé en deux zones. Les points de collectes des requêtes sont positionnés dans la zone rectangulaire  $ABEG$ , alors que les points de livraisons correspondant sont positionnés dans la zone rectangulaire  $GECD$ .

La méthode de génération de requêtes se décompose alors en un processus en deux étapes décrites ci-dessous.

#### 3.4.2.2 Génération des 3 premières requêtes

La première étape consiste à générer 3 requêtes. Notons qu'à cette étape, il est possible de générer plus de trois requêtes, toutefois, par souci

de clarté et pour faciliter la compréhension de la méthode proposée, nous fixons le nombre de requêtes initiales à trois. Considérant les règles de construction d'instances définies plus tôt, la première requête de transport considérée consiste à charger une quantité  $l_A$  de produits en  $A$ , et à les transporter en  $C$  ( $l_A = -l_C$ ). La deuxième requête consiste à charger une quantité  $l_B$  de produits en  $B$ , et à les transporter en  $D$  ( $l_B = -l_D$ ). La position des clients concernant la troisième requête est définie conformément aux principes d'utilisation de la droite (U). Après avoir choisi aléatoirement une distance  $d(F, E)$ , l'intersection de cette droite avec les diagonales du carré permet d'obtenir le point de ramassage  $H(l_H)$  et le point de livraison  $I(l_I)$  de la demande ( $l_H = -l_I$ ).

Le temps étant par hypothèse équivalent à la distance, les fenêtres de temps des points  $t, A, B, C, D, H$  et  $I$  sont respectivement :

- $[a_t = 0, b_t = \infty]$  ;
- $[a_A = 0, b_A = a\sqrt{2}]$  ;
- $[a_C = 0, b_C = 3a\sqrt{2}]$  ;
- $[a_B = 0, b_B = a\sqrt{2}]$  ;
- $[a_B = 0, b_B = a\sqrt{2}]$  ;
- $[a_D = 0, b_D = 3a\sqrt{2}]$  ;
- $[a_H = 0, b_H = \sqrt{2}(a - d(F, E))]$  ;
- $[a_I = 0, b_I = 2(a - d(F, E)) + \sqrt{2}(a - d(F, E))]$ .

Avec les fenêtres de temps ainsi définies et en prenant comme hypothèse que les durées de service sont toutes nulles, on peut remarquer que :

1. Si le transbordement n'est pas autorisé, en respectant le principe général précédemment présenté, et en respectant les fenêtres de temps, trois véhicules sont nécessaires pour satisfaire les 3 requêtes. La distance totale parcourue par les 3 véhicules est :  $Dist = d(O, A) + d(A, C) + d(C, O) + d(O, B) + d(B, D) + d(D, O) + d(O, H) + d(H, I) + d(I, O) = 8a\sqrt{2} + 2(\sqrt{2} + 1)(a - d(F, E))$
2. Si le transbordement est autorisé, toujours en respectant le principe général des requêtes présenté, seuls deux véhicules sont nécessaires pour satisfaire les 3 requêtes. La distance totale parcourue pour satisfaire les 3 requêtes est de :  $DistT = d(O, Y) + d(Y, B) + d(B, O) + d(O, D) + d(D, O) + d(O, A) + d(A, O) + d(O, Z) + d(Z, C) + d(C, O) = 8a\sqrt{2}$

La génération de cette base d'instances selon ces principes permet de garantir une solution démontrant l'intérêt du transbordement pour un problème de ramassage et distribution.

#### 3.4.2.3 Génération de requêtes restantes

Pour augmenter la taille des instances, le principe décrit au 3.4.2.2 est généralisé. À cet effet, la droite (U) est translatée parallèlement à (AD) ou (BC) à des distances différentes du point  $E$  et les points de collecte et livraison sont définis conformément au respect des zones ABEG et GECD. Ainsi, la solution optimale, dont la valeur dépend exclusivement de la

longueur du côté du carré considéré, n'est pas altérée par l'ajout des nouvelles requêtes.

### 3.4.3 Analyses expérimentales sur la base d'instances simples

Dans le but d'analyser le comportement du modèle analytique décrit plus haut, et d'en valider le comportement, 24 instances ont été générées. Les 16 premières ont été générées en respectant la méthode de génération d'instances préalablement présentée, et les huit dernières (voir tableau 3.4) ont été générées en utilisant une approche semblable, mais s'appuyant sur un cercle. Toutes les simulations ont été effectuées avec le solveur Xpress MP tournant sur un ordinateur Dell LATITUDE D400, possédant un processeur Intel(R) Pentium(R) cadencé à 1400 MHz avec 1.00 Go de RAM. Le système d'exploitation utilisé est Windows XP. Les instances E1 à E16 sont proposées en annexe A.2.

Le tableau 3.4 regroupe les différents résultats issus de la résolution des différents problèmes générés aléatoirement. La couleur « gris sombre » est utilisée dans la colonne DistT, si  $DistT < Dist$ . La couleur « gris clair » est utilisée dans la colonne NVBT lorsque  $NVBT < NBV$ .

Instances	Nombres de clients	Solution sans transbordement			Solution avec transbordement			% d'amélioration de la solution $G = \frac{Dist - DistT}{DistT} * 100$
		Distance parcourue (Dist)	Nombre de véhicules utilisés (NBV)	Temps CPU	Distance parcourue (DistT)	Nombre de véhicules utilisés (NBVT)	Temps CPU	
E1	3	1585,85	3	2,50	1131,38	2	3,40	40,10
E2	3	1131,38	2	1,50	1131,38	2	3,10	0
E3	3	1546,60	3	2,70	1131,38	2	4,40	36,70
E4	4	1599,70	3	8,90	1131,38	2	12,10	41,30
E5	4	1300,34	3	7,80	1131,38	2	29,40	14,90
E6	4	1590,60	3	8,60	1131,38	2	19,90	40,50
E7	5	1604,56	3	42,30	1131,38	2	58,40	41,80
E8	5	1585,22	3	39	1131,38	2	71,10	40,10
E9	5	1314,54	3	49,60	1131,38	2	110,30	16,10
E10	6	1469,34	3	117	1131,38	2	377,30	29,80
E11	6	1565,92	3	316,20	1131,38	2	1103,40	38,40
E12	6	1541,76	3	168	1131,38	2	251,80	36,20
E13	7	1565,92	3	989	1131,38	2	5701	38,40
E14	7	1435,56	3	2471,60	1131,38	2	4753,30	26,80
E15	7	1493,50	3	2372,20	1131,38	2	> 2000	32
E16	10	1594,90	3	> 30000	1131,38	2	> 50000	40,90
E17	2	682,50	2	0,10	605	2	1,90	3,49
E18	2	800	2	0,10	605	2	0,10	24,30
E19	3	1082,80	3	1,40	605	2	4,10	44
E20	3	-	-	-	605	2	1,50	-
E21	3	1082,80	2	4,10	810	2	13	25,10
E22	4	761,40	2	8,60	625	2	18,60	14,30
E23	5	707,80	3	649,20	630	2	1512,30	10,90
E24	6	1444,20	4	30000	1035	3	37000	28,50
Gain moyen								28,89

TABLE 3.4 – Résolution exacte du PDPT

Les 24 instances sur lesquelles s'est portée l'analyse ont été construites de manière à connaître par avance la solution optimale. Les résultats fournis dans le tableau 3.4 sont issus du solveur Xpress dans lequel le modèle *MO\_PDPT* a été implémenté. Ces résultats sont identiques à ceux attendus pour chaque instance. Cela tend donc à confirmer la validité du modèle. L'observation plus détaillée du tableau permet de motiver la pratique du transbordement dans le transport. En effet, sur les 24 instances considérées, le transbordement permet de réduire le nombre de véhicules utilisés sur 19 instances et permet de réduire la distance parcourue sur 23 instances avec un gain moyen de 28.89%. Toutefois, si les instances étaient construites de manière purement aléatoire, les gains auraient pu être moins intéressants. En effet, les instances utilisées ont été construites de manière à mettre en exergue l'intérêt de l'utilisation du transbordement comme levier d'optimisation. En dernier lieu, dans l'instance E2o, le problème étant très contraint, sans l'application du transbordement, il est impossible de satisfaire l'ensemble des requêtes tout en respectant toutes les contraintes. Dans ce cas particulier, le transbordement apporte une solution là où aucune solution de routage ne peut être trouvée.

### 3.5 CONCLUSION

L'objet de ce chapitre était de proposer un modèle mathématique en variables mixtes pour caractériser et résoudre le problème de collecte et livraison avec transbordement. Pour comprendre le fonctionnement du PDPT, deux exemples ont été proposés. Le premier exemple a permis d'illustrer le principe du transfert de charges entre deux véhicules, tandis que le deuxième exemple a permis de mettre en évidence l'utilisation du transbordement comme levier d'optimisation pour réduire le coût du transport.

Pour cadrer le problème étudié dont la compréhension a été améliorée par ces différents exemples, un ensemble d'hypothèses de travail a été fixé. Sur la base de ces hypothèses, le problème de collecte et livraison avec transbordement a été formalisé, et modélisé par un ensemble d'équations en programmation linéaire.

La littérature scientifique faisant état de très peu d'instances disponibles, une méthode de génération d'instances a été développée. Les instances générées à partir de cette méthode ont permis de tester et valider le modèle tout en mettant l'accent sur les bénéfices issus de la pratique du transbordement dans le transport.

L'observation des résultats obtenus sur les instances permet de se rendre compte de l'explosion combinatoire issue de l'utilisation du modèle proposé pour résoudre des instances de taille modeste. Pour cette raison, il est nécessaire de développer des méthodes approchées pour résoudre des instances de grandes tailles. La définition de ces méthodes fera l'objet du prochain chapitre.

# UNE MÉTHODE DE RÉSOLUTION APPROCHÉE DU PROBLÈME DE RAMASSAGE ET DISTRIBUTION AVEC TRANSBORDEMENT

4

## SOMMAIRE

4.1	INTRODUCTION . . . . .	97
4.1.1	Hypothèses et notations . . . . .	97
4.2	VNS POUR LA RÉSOLUTION DU PDPT . . . . .	99
4.2.1	Fonction objectif . . . . .	99
4.2.2	Algorithme général du VNS . . . . .	100
4.2.3	Détermination d'une solution initiale . . . . .	104
4.2.4	Voisinages . . . . .	106
4.2.5	Recherche locale . . . . .	109
4.3	RÉSULTATS EXPÉRIMENTAUX . . . . .	113
4.3.1	Validation du VNS sur des instances de PDP . . . . .	114
4.3.2	Validation du VNS sur des instances de PDPT . . . . .	116
4.4	CONCLUSION . . . . .	120



## 4.1 INTRODUCTION

**L**e problème de ramassage et distribution avec transbordement (Tchapnga2012b) est un problème NP-difficile. Du fait de l'explosion combinatoire et du temps de calcul nécessaire à l'obtention d'une solution, il n'est pas réaliste de penser utiliser des méthodes de résolution ou modèles exacts pour résoudre de manière optimale des instances de grande taille. Pour surmonter cet obstacle, des méthodes approchées permettant d'obtenir de « bonnes solutions » dans un temps relativement court sont généralement élaborées sur la base d'heuristiques ou de mét-heuristiques.

L'objet du présent chapitre est de proposer une mét-heuristique basée sur la recherche à voisinage variable (VNS : Variable Neighborhood Search) pour résoudre le PDPT. Il convient de noter que le VNS a été utilisé avec succès sur de nombreux problèmes d'optimisation combinatoire et en particulier sur des problèmes de tournées de véhicules (Polacek2004).

Ce chapitre de thèse est divisé en quatre parties. Après avoir rappelé quelques notions préliminaires dans la première partie, la deuxième présente la principale contribution du chapitre portant sur l'adaptation du VNS pour résoudre le PDPT. Dans la troisième partie, des tests expérimentaux sont effectués pour valider d'une part le VNS proposé en montrant sa capacité à trouver la solution optimale pour certains problèmes de taille réduite et pour mettre en évidence d'autre part l'intérêt de la mise en œuvre du transbordement. La quatrième et dernière partie conclut le chapitre.

Afin de présenter de manière claire les contributions du présent chapitre, il est nécessaire de compléter ou modifier les hypothèses et les notations énoncées au chapitre précédent.

### 4.1.1 Hypothèses et notations

Lors du chapitre précédent, la complexité du modèle et la nécessité de disposer de temps de calcul réalisistes nous avaient conduits à considérer un seul point de transbordement. Dans ce chapitre, l'hypothèse concernant l'existence d'un point de transbordement unique est abandonnée. En conséquence, de nouvelles hypothèses relatives au transbordement sont énoncées ci-dessous :

- De multiples lieux de transbordement sont considérés. En effet, la présence de plusieurs lieux de transbordement offre une meilleure flexibilité dans l'organisation du transport.
- Ces lieux de transbordement sont des lieux spécifiques ou des points correspondant à des clients (collecte ou de livraison).
- Un point de transbordement peut faire l'objet de plusieurs visites par des véhicules différents.
- Pour rappel, comme annoncé au chapitre 3, un produit est transbordé au maximum une fois. Cette limitation est fixée pour minimiser les coûts logistiques supplémentaires liés à la pratique du transbordement. En effet, lorsque qu'un produit est transbordé, il est nécessaire de veiller à préserver son intégrité, d'assurer sa sécu-

rité contre un éventuel vol, de décharger, puis recharger le produit, ce qui induit des coûts logistiques conséquents.

L'ajout de ces points de transbordement conduit à considérer de nouvelles définitions présentées ci-dessous :

- $T$  est l'ensemble des points de transbordement tel que  $|T| \geq 1$ . Tout véhicule qui arrive au point de transbordement  $t \in T$  y effectue soit un déchargement, soit un chargement, soit les deux opérations.
- L'ensemble  $V$  de tous les sommets du graphe modélisant le problème à résoudre est mis à jour pour intégrer les points de transbordement supplémentaires et devient  $V = N \cup o_1 \cup o_2 \cup T$ .
- Une route  $r_k$ , également appelée tournée du véhicule  $k$ , est notée  $r_k = \langle x_1, x_2, x_3, \dots, x_m \rangle$ . Cette notation est définie comme suit :
  - Le symbole  $\langle \rangle$  représente un ensemble de points ou nœuds ordonnés par une relation d'ordre. L'indice  $i$  de l'élément  $x_i$  de  $r_k$ , représente l'ordre d'apparition ou la position de cet élément dans la tournée du véhicule  $k$ . Chaque élément de  $r_k$  est associé à un nœud  $v \in V$ .
  - Pour une tournée (ou route)  $r_k (k \in K)$ ,  $nb(r_k)$  représente le nombre de nœuds visités en intégrant le dépôt de départ et d'arrivée.
  - Il convient de noter que dans le cas particulier de ce chapitre, chaque véhicule débute sa tournée au point  $o_1$  et la termine au point  $o_2$ , si bien que  $x_1 = o_1$  et  $x_m = o_2$ . Dans ce cas une tournée est donc de la forme  $r_k = \langle o_1, x_2, x_3, \dots, o_m \rangle$ .
  - Il est supposé que le nombre de véhicules disponibles est supérieur au nombre de requêtes à satisfaire. Ainsi, chaque requête peut toujours être satisfaite.
  - Sur la base de la définition précédente de  $r_k$ ,  $C(r_k) = \sum_{i=1}^{m-1} c_{x_i x_{i+1}}$  est la distance totale parcourue par le véhicule  $k \in K$  pour visiter l'ensemble des clients qui constituent sa tournée. Rappelons que  $c_{x_i x_{i+1}}$  représente le coût du déplacement d'un véhicule entre les nœuds associés aux éléments  $x_i$  et  $x_{i+1}$
  - Une solution  $S$  du problème est constituée d'un ensemble de routes  $S = \{r_1, r_2, \dots, r_n\}$  avec  $n \leq |K|$ . Le nombre de routes dans une solution est donné par  $n = |S|$ .

Conformément aux différentes hypothèses énumérées au chapitre 3 et consolidées par celles présentées ci-dessus, il convient de garder en mémoire qu'il existe deux manières différentes pour satisfaire une requête  $R_i \in R$  : le point de ramassage  $i \in P$  et le point de livraison associé  $i + n \in D$  sont visités soit par un même véhicule  $k \in K$ ; soit par deux véhicules différents  $k_1 \in K$  et  $k_2 \in K$ . Dans ce dernier cas, le véhicule  $k_1$  charge le produit  $l_i$  à l'emplacement défini par  $i \in P$  et le transporte jusqu'au point de transbordement  $t \in T$ . Ensuite, le véhicule  $k_2$  recharge le produit au point  $t$  pour le livrer au point  $i + n \in D$ .

Pour résoudre de manière approchée le problème de collecte et livraison avec fenêtres de temps et transbordement, nous faisons appel dans ce chapitre au VNS qui a été introduit au chapitre 2. L'adaptation de cette méthode de résolution au problème étudié est décrite dans le paragraphe suivant.

## 4.2 VNS POUR LA RÉSOLUTION DU PDPT

La contribution principale du chapitre pour la résolution efficace du PDPT est présentée dans cette partie et porte sur les points suivants :

- La description de la fonction « objectif » ;
- La spécification de l'algorithme général du VNS proposé ;
- La caractérisation de la solution initiale ;
- L'identification de l'ensemble des voisinages exploités ;
- La description de la phase de recherche locale.

### 4.2.1 Fonction objectif

La résolution du PDPT a pour objet, dans notre travail, de minimiser deux objectifs hiérarchisés, à savoir :

1. Le nombre de véhicules utilisés ;
2. La distance totale parcourue.

Ainsi, résoudre le PDPT consiste à trouver une solution  $S$  qui minimise le nombre de véhicules utilisés et en cas d'égalité sur le nombre de véhicules, à préférer la solution avec la plus petite distance totale parcourue.

Il convient de noter qu'au chapitre 3, pour trouver une telle solution, la fonction d'utilité ou d'évaluation linéaire pondérée  $P * f_1 + f_2$  a été utilisée. La fonction  $f_1$  calcule le nombre de tournées ou véhicules utilisés dans une solution ; la fonction  $f_2$  calcule la distance totale dans une solution ;  $P$  est le coefficient de pondération permettant de prioriser la réduction du nombre de véhicules sur la distance totale parcourue.

Une telle fonction d'évaluation souvent nécessaire pour un modèle linéaire présente l'inconvénient du choix du coefficient de pondération. Dans le cadre de ce chapitre, pour surmonter le problème du choix du coefficient de pondération c'est-à-dire rendre l'exploration de l'espace de solution indépendante de  $P$ , la fonction d'évaluation lexicographique  $FE = (f_1, f_2, f_3)$  adaptée de (Bent2006) sera utilisée.  $FE$  est dite lexicographique puisqu'elle permet de classer les solutions obtenues dans un ordre défini. Ainsi, dans le cadre de notre travail, une solution  $S_1$  est déclarée meilleure qu'une solution  $S_2$ , si  $FE(S_1) < FE(S_2)$ , c'est-à-dire si l'une des conditions suivantes sont respectées :

- $f_1(S_1) < f_1(S_2)$  ou
- $f_1(S_1) = f_1(S_2)$  et  $f_2(S_1) \leq f_2(S_2)$  ou
- $f_1(S_1) = f_1(S_2)$ ,  $f_2(S_1) = f_2(S_2)$  et  $f_3(S_1) \leq f_3(S_2)$

Les fonctions  $f_1, f_2, f_3$  sont précisément définies comme suit pour une solution  $S$  :

- La composante  $f_1 = |S|$  calcule le nombre de véhicules.
- La composante  $f_2$  décrite ci-après est calculée comme suit :  $f_2 = -\sum_{k=1}^{|S|} (nb(r_k))^2$ .
- La composante  $f_3 = \sum_{k=1}^{|S|} C(r_k)$  calcule la distance totale parcourue par tous les véhicules.

Afin de réduire le nombre de véhicules nécessaires, le rôle de la composante  $f_2$  est de maximiser le nombre de clients visités dans une même tournée. Dans la définition de  $f_2$ , l'élévation au carré du nombre de nœuds

des routes  $r_k$  est utilisée afin de prioriser ou donner plus de poids aux solutions dans lesquelles il semble potentiellement plus facile de réduire le nombre de véhicules utilisés. Le signe négatif utilisé dans la définition de  $f_2$  assure la cohérence avec la logique de minimisation du coût global. Pour clarifier le propos, et montrer le fonctionnement de  $f_2$ , nous proposons l'exemple ci-dessous.

Soit un problème de transport P dans lequel six requêtes de transport concernant 12 clients doivent être satisfaites. Soit une première solution  $S_1$  pour le problème P défini par trois routes  $S_1 : \{r_1, r_2, r_3\}$  telle que :  $nb(r_1) = 6$ ,  $nb(r_2) = 6$ ,  $nb(r_3) = 6$ . Pour le même problème P, considérons une autre solution  $S_2 : \{r_1, r_2, r_3\}$  telle que  $nb(r_1) = 8$ ,  $nb(r_2) = 6$ ,  $nb(r_3) = 4$ . Durant l'exploration de l'espace des solutions,  $f_1(S_1) = 3$  et  $f_1(S_2) = 3$ , ce qui indique une égalité sur le nombre de véhicules utilisés dans les deux solutions. C'est donc la composante  $f_2$  qui permettra de privilégier la solution  $S_2$ . En effet  $f_2(S_1) = -108$  et  $f_2(S_2) = -116$ . Le choix de  $S_2$  repose sur le postulat qu'il est potentiellement plus facile de réduire le nombre de tournées dans  $S_2$  par l'application d'une transformation élémentaire (i.e. opérateur de voisinage de type suppression/réinsertion de requêtes). En l'occurrence, il suffit pour réduire le nombre de véhicules que les deux requêtes de la tournée  $r_3$  de  $S_2$  soient supprimées de cette tournée pour être réinsérées dans  $r_1$  ou  $r_2$ . Dans  $S_1$ , il est plus difficile de réduire le nombre de tournées, car ces dernières sont équilibrées et possèdent toutes le même nombre de clients.

La fonction d'évaluation nécessaire à l'exploration de l'espace des solutions étant présentée, le pseudo code l'algorithme « VNSPourLePDPT » est proposé.

### 4.2.2 Algorithme général du VNS

Nous introduisons dans un premier temps les caractéristiques spécifiques de notre approche de résolution puis nous décrivons dans un second temps l'algorithme général du VNS mis en œuvre.

#### 4.2.2.1 Spécificités du VNS proposé

Dans le chapitre 2, l'algorithme du VNS de base a été présenté de manière succincte. Pour rappel, il consiste en deux phases principales : la phase d'initialisation et la phase itérative. Cette seconde phase consiste à améliorer une solution initiale à l'aide d'un ensemble de voisinages. Plus précisément chaque itération de la phase itérative comporte deux étapes qui sont la perturbation et la recherche locale. Dans la phase de perturbation, une solution est tirée de façon aléatoire dans le voisinage de la solution courante. Dans la phase de recherche locale, la solution précédente aléatoirement tirée est améliorée par le biais de l'ensemble des voisinages utilisés pour la recherche locale. Deux principales modifications ont été apportées par rapport à ce schéma général.

La première concerne le déclenchement de la recherche locale. En effet à chaque itération du VNS dans sa version de base, la recherche locale est systématiquement exécutée. Cette exécution systématique peut être fortement consommatrice de temps. Afin de limiter le temps de calcul, une stratégie inspirée du VNS réduit (Hansen2001) et permettant de réduire le

temps de calcul en évitant une application systématique de la recherche locale à chaque itération est appliquée. Ainsi, la recherche locale sera appliquée dans le cadre de l'algorithme que nous proposons si et seulement si la solution tirée à l'étape de la perturbation est prometteuse.

Une solution  $S_2$  est dite prometteuse par rapport à une solution  $S_1$  dans les deux cas suivants : (i) le coût de  $S_2$  est inférieur ou égal au coût de  $S_1$ ; (ii) le coût de  $S_2$  est supérieur au coût de  $S_1$ , mais l'écart entre les deux coûts n'excède pas une certaine valeur  $\beta$  donnée en paramètre du problème.

Ainsi,  $S_2$  est dite prometteuse par rapport à  $S_1$  si  $Ecart = \frac{FE(S_2) - FE(S_1)}{FE(S_1)} \leq \beta$  avec  $0.01 \leq \beta \leq 0.09$ . Par exemple, pour un problème donné de minimisation, supposons que  $\beta = 0.05$ ,  $FE(S_2) = 825$  et  $FE(S_1) = 800$ . Avec ces valeurs, l'écart est de  $0.03 < \beta$ . La solution  $S_2$  est donc déclarée prometteuse. Si  $FE(S_2) = 900$  et  $FE(S_1) = 800$ ,  $S_2$  n'est pas déclarée prometteuse. En effet, avec ces valeurs, l'écart est de 0,12.

La seconde modification concerne la recherche locale en elle-même. En effet, la première modification permet de réduire le temps de calcul, mais l'exécution conditionnelle de la recherche locale peut entraîner une perte de performance dans la recherche de la solution optimale. Afin de conserver un bon niveau de diversité dans la recherche locale, celle-ci est structurée en deux niveaux dans le but d'explorer efficacement l'espace des solutions. Le premier niveau est une recherche locale sans transbordement. Le second est une recherche locale avec transbordement spécialement dédiée au contexte de résolution du problème de routage des véhicules. Il est à noter que lors de la recherche locale, les deux niveaux, sans transbordement et avec transbordement, sont toujours successivement exécutés.

Ce paragraphe a été l'occasion de préciser les différences qui existent entre le VNS de base et l'algorithme « VNSPourLePDPT » que nous proposons. Les différentes étapes constituant cet algorithme sont maintenant présentées et chacune de ces composantes sera par la suite expliquée de manière détaillée.

#### 4.2.2.2 Algorithme du VNS pour le PDPT

Le pseudo code du « VNSPourLePDPT » est décrit dans sa globalité dans l'algorithme 3.

---

**Algorithme 3 : VNSPourLePDPT**

---

**Entrées :** Nombre maximum d'itérations –  $MaxIter$ ; Ensemble de requêtes –  $R$ ; Ensemble des points de transbordement –  $T$

**Sorties :** BestSol

```

1 //PHASE D'INITIALISATION;
2 Sol ← InsertionParallèle (R);
3 BestSol ← Sol;
4 N ← ParametrerVoisinagePourLaPerturbation (Sol) ;
5 kmax ← |N|;
6 compteur ← 0;
7 // PHASE ITÉRATIVE;
8 répéter
9   k ← 1;
10  répéter
11    // Étape 1 : Perturbation ;
12    S ← Nk (Sol);
13    //Vérification que S est prometteuse par rapport à Sol;
14    Ecart ←  $\frac{FE(S)-FE(Sol)}{FE(Sol)}$ ;
15    //Étape 2 : Recherche locale ;
16    si Ecart <  $\beta$  alors
17      SolC ← LS (S) //Recherche locale sans
18      transbordement;
19      si FE (SolC) < FE (Sol) alors
20        Sol ← SolC;
21        N ← ParametrerVoisinagePourLaPerturbation (Sol) ;
22        k ← 1;
23        kmax ← |N|;
24        compteur ← 0;
25      sinon
26        k ← k + 1 ;
27      SolC ← Transbordement (R, T, SolC) // Recherche
28      locale avec transbordement;
29      si FE (SolC) < FE (BestSol) alors
30        BestSol ← SolC;
31        compteur ← 0;
32  jusqu'à k = kmax;
33  compteur ← compteur + 1;
34 jusqu'à compteur > MaxIter;
35 Retourne BestSol;
```

---

L'objectif du VNS proposé est de fournir une solution qui permet de satisfaire de manière optimale un ensemble de requêtes de transport. La liste des paramètres de l'algorithme 3 est la suivante :

- MaxIter – ce paramètre permet de définir la condition d'arrêt du VNS et correspond au nombre d'itérations maximum depuis la dernière amélioration de la meilleure solution connue. En effet, à chaque itération, la variable compteur est soit réinitialisée, soit incrémentée d'une unité. La réinitialisation du compteur (ligne 23) est conditionnée par l'amélioration de la solution courante ou par l'amélioration de la meilleure solution connue (ligne 29). L'incrémentation de la variable compteur (ligne 31) a lieu lorsque la solution courante ( $SolC$ ) n'a pas été améliorée.
- R – représente l'ensemble des requêtes à satisfaire
- T – représente l'ensemble des points de transbordement.

Sur la base de ces informations, l'algorithme retournera en ligne 33 la meilleure solution qu'il aura trouvée à l'issue de son exécution.

Dans l'algorithme « VNSPourLePDPT », tant que la condition d'arrêt n'est pas satisfaite, entre la ligne 8 et la ligne 32, les opérations suivantes sont exécutées de manière itérative :

- Une solution  $S$  du PDP est tirée dans le voisinage  $N_k$  de la solution  $Sol$  (ligne 12).  $N_k$  représente le voisinage d'indice  $k$ , i.e. son ordre d'apparition dans l'ensemble  $N$  ( $k_{max} = |N|$ ) des voisinages utilisés pour la perturbation.
- La solution  $S$  subira une recherche locale si et seulement si elle est prometteuse (ligne 16).
- Si  $S$  est prometteuse, elle subit une recherche locale sans transbordement (LS), ce qui permet d'obtenir une solution  $SolC$  (ligne 17).
- Si  $SolC$  est meilleure que la solution courante, sa valeur est affectée à la solution courante (ligne 19).
- Dans le cas où  $SolC$  permet d'améliorer la solution courante, l'ensemble des voisinages utilisés pour la perturbation est mis à jour en ligne 20 via l'algorithme « ParametrerVoisinagePourLaPerturbation ». En effet, cet algorithme a pour entrée la solution courante  $Sol$ . Cette solution ayant été mise à jour, il est donc nécessaire de paramétriser de nouveau les voisinages utilisés pour la perturbation.
- L'indice du prochain voisinage à utiliser pour la perturbation est réinitialisé à la (ligne 21), afin d'intensifier la recherche autour de cette solution. L'initialisation de l'indice  $k$  à 1 permet l'exploitation de tous les voisinages disponibles dans le cas de la perturbation de la solution en cours d'amélioration.
- Lorsqu'une nouvelle solution est découverte, il convient d'explorer l'espace de solution associé à cette solution. Afin de disposer de plus de temps d'analyse, la valeur de la variable « compteur » est réinitialisée à 0 en ligne 23.
- Une tentative d'amélioration de  $SolC$  est effectuée par la recherche locale avec transbordement (ligne 26). Si  $SolC$  améliore la meilleure solution connue, sa valeur est affectée à cette dernière (ligne 28).

Notons que d'autres mécanismes d'amélioration de la solution ayant recours au transbordement, auraient pu être implémentées, comme l'amélioration de la solution  $S$  résultant de la phase de perturbation, plutôt que

la solution SolC. Aucune étude ne permet cependant de conclure précisément, quant au réel impact de la qualité de la solution de départ sur celle de la solution obtenue après amélioration. Nous avons donc arbitrairement choisi d'appliquer un principe d'amélioration fondé sur le transbordement, à la meilleure solution trouvée à l'issue de la recherche locale tentant de résoudre un PDP « classique ».

Dans les parties suivantes, nous présentons en détail les composants principaux de l'algorithme « VNSPourLePDPT ». Ces composants portent sur :

- La détermination de la solution initiale par l'algorithme « Insertion-Parallèle » (ligne 2) ;
- La présentation de l'ensemble  $N$  des voisinages (ligne 4) utilisés pour la perturbation de la solution  $S$  ;
- La description de la recherche locale sans transbordement (ligne 17) et des opérateurs de voisinage afférents ;
- La présentation de l'algorithme « Transbordement » utilisé pour effectuer le transbordement (ligne 26).

#### 4.2.3 Détermination d'une solution initiale

La solution initiale est construite de manière à satisfaire l'ensemble des requêtes sans transbordement. Il s'agit donc d'une solution du problème PDP. L'heuristique employée pour produire la solution initiale est une heuristique d'insertion parallèle (Savelsbergh1990). La méthode d'insertion consiste à traiter une requête de transport à chaque itération et à l'insérer dans une et une seule route. L'heuristique est dite parallèle (algorithme « InsertionParallèle ») dans la mesure où plusieurs routes sont construites en parallèle. En effet, le principe de construction consiste à chaque itération, à tenter de placer toute requête à la meilleure place dans l'une des routes existantes. En d'autres termes, il s'agit de déterminer une tournée et la position du point de collecte et de livraison de la requête dans celle-ci de manière à minimiser le critère d'optimisation retenu sans violer aucune des contraintes exprimées. Si aucune insertion n'est possible dans l'une des routes existantes, une nouvelle route associée à un nouveau véhicule est créée et la requête y est insérée.

L'algorithme de l'heuristique d'insertion parallèle, nécessite l'utilisation de quelques fonctions de bases qui sont décrites dans l'annexe A.3 du chapitre. Ces fonctions sont respectivement :  $VerifTemps(r_k)$ ,  $VerifCapacité(r_k)$ ,  $Initialisation(o_1, o_2, k)$  et  $Insertion(r_k, R_i)$ .

---

**Algorithme 4 : InsertionParallèle**

---

**Entrées :**  $R$  – Ensemble des requêtes ;  $S$  – Solution en cours de construction

**Sorties :**  $S$  – Solution permettant de satisfaire l'ensemble des requêtes ;

```

1 //Si S est initialement vide ;
2 si  $S = \emptyset$  alors
3    $k \leftarrow 0;$ 
4    $r_k \leftarrow \text{Initialisation}(o_1, o_2, k);$ 
5    $S \leftarrow S \cup \{r_k\}$  // Ajout d'une première tournée vide dans  $S$ ;
6 // pour chaque requête, on cherche la meilleure route dans
  laquelle l'insérer ;
7 pour chaque  $R_i$  dans  $R$  faire
8    $Min \leftarrow \infty;$ 
9    $r_{-1} \leftarrow \emptyset;$ 
10  pour chaque route  $r_h$  dans  $S$  faire
11     $r_h \leftarrow \text{Insertion}(r_h, R_i);$ 
12    si  $R_i$  est insérée avec succès dans  $r_h$  et  $C(r_h) < Min$  alors
13       $Min \leftarrow C(r_h);$ 
14       $r_{-1} \leftarrow r_h;$ 
15       $r_h \leftarrow r_h - \{R_i\}$  //  $R_i$  est supprimée de la route  $r_h$ ;
16  si  $Min \neq \infty$  alors
17    // Insertion définitive de la requête ;
18     $r_h \leftarrow r_{-1};$ 
19  sinon
20    // sinon une route supplémentaire est créée pour la
      requête ;
21     $k \leftarrow k + 1;$ 
22     $r_k \leftarrow \text{Initialisation}(o_1, o_2, k);$ 
23     $r_k \leftarrow \text{Insertion}(r_k, R_i);$ 
24     $S \leftarrow S \cup \{r_k\};$ 
25 Retourne  $S$  ;

```

---

Pour insérer la requête  $R_i$  choisie en ligne 7, deux variables sont initialisées. La première  $Min$  initialisée avec une valeur infinie permet de sauvegarder le coût de la route permettant l'insertion à moindre coût de  $R_i$ ; la seconde  $r_{-1}$  permet de sauvegarder une copie de la route dans laquelle  $R_i$  est insérée à moindre coût. Cette variable est initialement vide. Disposant de ces deux variables, entre les lignes 10 et 15, le coût d'insertion de  $R_i$  dans chaque tournée de la solution  $S$  est évaluée. La tournée à évaluer est stockée dans la variable  $r_h$  (ligne 10). À l'issue de cette évaluation, entre les lignes 16 et 19, la requête  $R_i$  est définitivement insérée dans la tournée qui conduit à une dégradation minimale de la fonction objectif. Si aucune des tournées de  $S$  ne peuvent accueillir  $R_i$ , une nouvelle route est spécialement créée pour insérer  $R_i$  (ligne 19 – 24). La dernière route créée est stockée dans la variable  $r_k$ .

L'algorithme « InsertionParallèle » présenté dans la partie précédente permet de construire une solution initiale. À chaque itération du VNS, cette solution est perturbée et améliorée à l'aide d'ensemble d'opérateurs de voisinages.

#### 4.2.4 Voisinages

Comme présenté au chapitre 2, un voisinage d'une solution est un ensemble de solutions obtenu par une série de transformations (ou mouvements) élémentaires. Dans le présent paragraphe, nous nous proposons de décrire les deux types de voisinages mis en œuvre dans la phase de perturbation et de recherche locale.

##### 4.2.4.1 Voisinages utilisés pour la perturbation

Une des étapes de la phase d'initialisation du VNS consiste à paramétrier l'ensemble de voisinages qui sera utilisé dans la phase de perturbation du « VNSPourLePDPT ». Le paramétrage est effectué en ligne 4 et l'exploitation de ces voisinages est effectuée en ligne 12 de l'algorithme cité. Les opérateurs de voisinages utilisés pour la perturbation sont notés  $N_k$  ( $k = 1, \dots, k_{max} = |N|$ ) dans l'algorithme 3, avec  $N$  représentant l'ensemble des voisinages. Nous disposons ainsi d'un ensemble de voisinages dont la taille est connue de manière précise à l'exécution de la ligne 5 de l'algorithme « VNSPourLePDPT » et cette taille est mise à jour à la ligne 22 du même algorithme à chaque découverte d'une nouvelle solution améliorant la solution courante.

Ces opérateurs sont tous basés sur le principe du transfert cyclique. Le transfert cyclique (Cyclic Transfers – TC) est un voisinage initialement proposé par (Thompson1993) pour résoudre les problèmes de tournées de véhicules. L'application du TC au problème de ramassage et distribution consiste à déplacer des requêtes de façon cyclique entre plusieurs routes. Dans le cadre de ce chapitre et toujours avec le souci de maîtriser le temps de calcul, la variante du transfert cyclique utilisée est appelée TC-r-b. La mise en application de l'opérateur TC-r-b se matérialise par la sélection aléatoire de  $r$  routes parmi les routes qui constituent la solution en cours de perturbation. Dans chacune des routes sélectionnées,  $b$  requêtes sont supprimées puis réinsérées dans une autre route et ceci de manière cyclique. À titre d'exemple, considérons une solution  $S$  constituée de six

routes. L'application de TC-3-2 sur  $S$  consiste à sélectionner aléatoirement trois routes  $(r_1, r_2, r_3)$  parmi les six existantes. Deux requêtes ( $b = 2$ ) sont successivement sélectionnées de manière aléatoire dans  $r_1$ , supprimées de  $r_1$  et réinsérées dans  $r_2$ . Le même procédé est appliqué entre  $r_2$  et  $r_3$  puis entre  $r_3$  et  $r_1$ .

Dans ce chapitre, plusieurs voisinages de type TC-r-b sont utilisées pour effectuer l'étape de perturbations. Chaque voisinage est déterminé par le couple de valeurs  $(r, b)$ . L'ensemble des couples nécessaires à la définition des voisinages utilisés pour la perturbation est calculé aux lignes 5 et 6 de l'algorithme 5. Ces deux paramètres sont choisis de manière à garantir que le nombre de routes « r » concerné par le transfert cyclique, ainsi que le nombre de requêtes « b » ne sont jamais supérieurs au nombre de routes et de requêtes présentes dans la solution en cours d'amélioration. Ainsi, les voisinages TC-r-b utilisés pour avoir de la diversité et de la variété dans l'exploration de l'espace des solutions dépendent des valeurs de  $r$  et  $b$  qui sont déterminées dans l'algorithme 5.

---

**Algorithme 5 :** ParametrerVoisinagePourLaPerturbation
 

---

**Entrées :**  $Solution - S$  ;  
**Sorties :** Ensemble des voisinages –  $EVP$  ;

```

1  $EVP \leftarrow \emptyset$  ;
2  $nombreDeroute \leftarrow$  une valeur aléatoirement tirée entre 1 et  $|S|$  ;
3  $nombreDeRequete \leftarrow \min\{\frac{nb(r_m)-2}{2}\}_{m=1}^{m=|S|}$  // nombre de requêtes min
dans une route ;
4  $k \leftarrow 1$  ;
5 pour  $r \leftarrow 1$  à  $nombreDeroute$  faire
6   pour  $b \leftarrow 1$  à  $nombreDeRequete$  faire
7      $N_k \leftarrow TC - r - b$  ;
8      $EVP \leftarrow EVP \cup N_k$  ;
9    $k \leftarrow k + 1$  ;
10 Retourne  $EVP$ ;
```

---

L'explication détaillée de l'algorithme est décrite dans les points suivants :

- L'entrée  $S$  de l'algorithme correspond à la solution à optimiser et la sortie  $EVP$  est l'ensemble des voisinages qui seront utilisés pour perturber  $S$ .
- En ligne 2, un nombre de routes ( $nombreDeroute$ ) est tiré de façon aléatoire entre 1 et le nombre maximum de routes dans  $S$ .
- En ligne 3 le nombre minimum de requêtes dans une route ( $nombreDeRequete$ ) est calculé.
- De la ligne 5 à la ligne 9, l'ensemble des voisinages TC-r-b est paramétré.

Il convient de rappeler que les voisinages utilisés pour effectuer la perturbation d'une solution ne sont pas tous utilisés de manière exhaustive à chaque itération. En effet, à chaque itération « VNSPourLePDPT », un seul de ces voisinages d'indice  $k$  sera utilisé. Il permettra ainsi de disposer

d'une nouvelle solution qui servira de point de départ pour la recherche locale qui sera effectuée avec les voisinages ci-dessous décrits.

#### 4.2.4.2 Voisinages utilisés pour la recherche locale sans transbordement

Dans la mesure où le principe d'amélioration s'appuie sur l'utilisation d'un opérateur de voisinage donné, l'utilisation de plusieurs opérateurs permet d'augmenter les opportunités d'amélioration de la solution en amenant de la variété dans la manière de générer des solutions voisines, et ainsi d'augmenter la probabilité de s'approcher de la solution optimale globale (Mladenovic1997). Dans le cadre de notre recherche, nous considérons trois opérateurs de voisinages présentés ci-dessous. Le but de chacun des voisinages proposés est de déterminer un sous-ensemble de solutions qu'il est possible d'atteindre par une série de transformations données. Les transformations associées à chaque voisinage permettent de le caractériser et sont ci-dessous présentées.

- RNR (Réduire le Nombre de Routes) – Le principe du voisinage RNR consiste à identifier la route contenant le plus petit nombre de requêtes. Chaque requête comprise dans celle-ci est par la suite supprimée pour être insérée à coût minimum dans l'une des autres routes existantes, à condition que toutes les contraintes soient satisfaites. S'il n'est pas possible d'insérer la requête dans l'une des routes différentes de celle d'origine, celle-ci est remise à son emplacement initial. Une description schématique de RNR est fournie dans la figure 4.1 où la requête 2 est retirée de la route  $r_1$  puis réinsérée dans la route  $r_2$ .

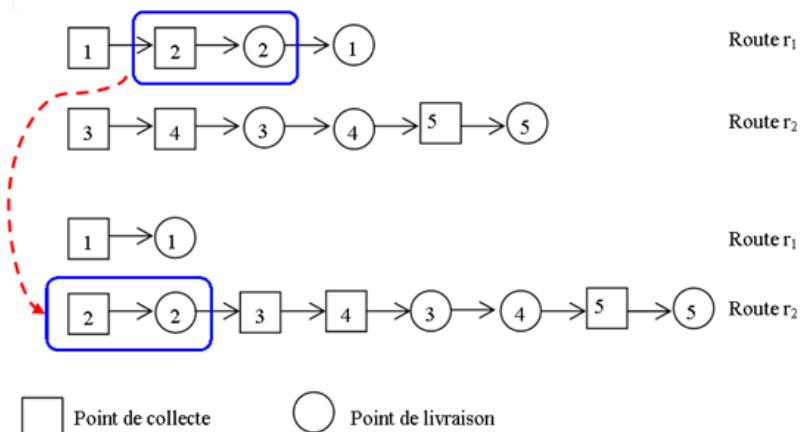


FIGURE 4.1 – Le voisinage RNR

- SWAP – Le fonctionnement du voisinage SWAP consiste en une permutation de requêtes entre deux routes. Le principe de ce voisinage est schématiquement illustré par l'exemple de la figure 4.2. Dans cet exemple, les requêtes  $R_2$  et  $R_5$  sont respectivement supprimées de leur route d'origine  $r_1$  et  $r_2$ . Ensuite, la requête  $R_2$  est réinsérée dans la route  $r_2$  et la requête  $R_5$  est réinsérée dans la route  $r_1$ .

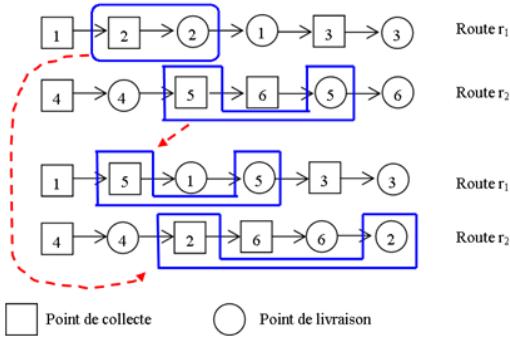


FIGURE 4.2 – Le voisinage SWAP

- PRR (Permutation de Requêtes dans une Route) – Le principe de ce voisinage diffère du voisinage SWAP. Alors que dans SWAP, deux requêtes appartenant à deux routes différentes sont permuteées, dans PRR, la permutation porte sur des requêtes appartenant à une même route. Le processus de permutation consiste à choisir successivement chaque requête de la route dans l'ordre où elle apparaît et tenter une permutation avec les requêtes restantes. Pour exemple, dans la figure 4.3, la requête R<sub>3</sub> est exécutée après la requête R<sub>1</sub>; l'application de PRR permet de perturber cette solution en proposant d'exécuter la requête R<sub>3</sub> avant la requête R<sub>1</sub>.

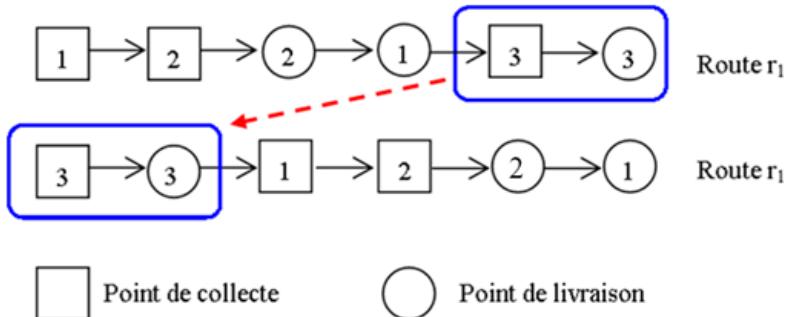


Figure 4.6 – Le voisinage PRR

FIGURE 4.3 – Le voisinage PRR

#### 4.2.5 Recherche locale

La phase de recherche locale dans le « VNSPourLePDPT » consiste en deux algorithmes distincts : la recherche locale sans transbordement et la recherche locale avec transbordement.

##### 4.2.5.1 Recherche locale sans transbordement

La recherche locale sans transbordement utilisée en ligne 17 de l'algorithme « VNSPourLePDPT » est notée LS. Pour rappel, cet algorithme a été présenté au chapitre 2 dans la partie 2.3.2.2 consacrée aux méthodes de résolutions approchées. La mise en œuvre de cet algorithme s'appuie sur l'utilisation des voisinages RNR, SWAP et PRR selon l'ordre de préférence et avec les stratégies d'exploration décrites dans le tableau 4.1. Les

définitions des stratégies d'exploration BI et FI ont été données au chapitre 2. Dans le tableau 4.1, BI et FI signifient respectivement, meilleure amélioration et première amélioration.

Solution S		
Ordre d'utilisation des voisinages	Opérateur de voisinage	Stratégie d'exploration
1	RNR	BI
2	SWAP	FI
3	PRR	BI

TABLE 4.1 – Stratégie d'utilisation des voisinages dans LS

Il convient de noter les points suivants :

- Les voisinages RNR et PRR sont des opérateurs déterministes. Pour le voisinage SWAP, les routes et les requêtes sont aléatoirement sélectionnées.
- L'exploration des voisinages RNR et PRR est moins coûteux en temps de calcul que l'exploration du voisinage SWAP. Pour cette raison, dans un souci de maîtrise de temps de calcul, RNR et PRR utilisent la stratégie d'exploration de type « meilleure amélioration » (BI), tandis que SWAP utilise la stratégie de « première amélioration »(FI).

La solution obtenue à ce stade de l'optimisation est un optimum local qui permet de satisfaire toutes les requêtes mais qui n'intègre pas encore les possibilités de transbordement pouvant potentiellement conduire à une amélioration de solution.

#### 4.2.5.2 Recherche locale avec transbordement

La méthode de transbordement proposée dans l'algorithme « Transbordement » permet de tenter d'améliorer à chaque itération la solution issue de la recherche locale en s'inspirant des travaux de (Mitrovic2006). Le principe général de cette heuristique vise à utiliser le transfert de charges entre véhicules en des points remarquables pour réduire si possible, le nombre de véhicules utilisés ou la distance nécessaire pour satisfaire toutes les requêtes. Elle consiste à retirer chaque requête de la solution de PDP et à la réinsérer en utilisant le transbordement si cela permet de réduire le coût de transport.

---

**Algorithme 6 : Transbordement**

---

**Entrées :** Ensemble des requêtes –  $R$  ; Ensemble des points de transbordement  $T$  ; Solution du PDP –  $S$

**Sorties :** Solution du PDPT –  $Sol$

```

1  $Sol \leftarrow S;$ 
2 pour chaque  $r$ équête  $R_i$  dans  $R$  faire
3    $MeilleureCout \leftarrow FE(S);$ 
4   Supprimer  $R_i$  de  $S$ ;
5    $BestT \leftarrow \emptyset;$ 
6   pour chaque  $t$  dans  $T$  faire
7      $R_e \leftarrow (i, t);$ 
8      $S \leftarrow InsertionParallèle(\{R_e\}, S);$ 
9      $MiseAJourBorneInfFenetreDeTemps;$ 
10     $R_s \leftarrow (t, i + n);$ 
11     $S \leftarrow InsertionParallèle(\{R_s\}, S);$ 
12     $NouveauCout \leftarrow FE(S);$ 
13    Supprimer les requêtes  $R_e$  et  $R_s$  de  $S$  ;
14    si  $NouveauCout < MeilleureCout$  alors
15       $MeilleureCout \leftarrow NouveauCout;$ 
16       $BestT \leftarrow t;$ 
17    si  $BestT \neq \emptyset$  alors
18       $S \leftarrow InsertionParallèle(\{(i, BestT)\}, S);$ 
19       $S \leftarrow InsertionParallèle(\{(BestT, i + n)\}, S);$ 
20      si  $FE(S) < FE(Sol)$  alors
21         $Sol \leftarrow S;$ 
22    sinon
23       $S \leftarrow Sol;$ 
24 Retourne  $Sol;$ 

```

---

Comme annoncé dans l'introduction de cette partie, le but de l'heuristique de transbordement est de tenter d'améliorer une solution de PDP en supprimant et en réinsérant avec transbordement la requête supprimée. Le transbordement d'une requête est effectué en sélectionnant dans l'ensemble  $T$ , le point de transbordement offrant le meilleur coût. Si le transbordement de la requête ne permet pas d'améliorer la solution ou si l'insertion de la requête avec transbordement n'est pas possible, la requête est remise à sa position d'origine. Les principales étapes de l'algorithme peuvent être synthétisées de la manière suivante :

1. En entrée,  $S$  est une solution qui permet de satisfaire toutes les requêtes sans transbordement.
2. La requête  $R_i = (i, i + n)$  est retirée de  $S$  (ligne 4)
3. Le coût de la solution  $S$  après réinsertion de la requête  $R_i$  avec transbordement est évalué entre les lignes 6 à 16. Parmi tous les points de transbordement, le meilleur est utilisé pour séparer la requête  $(i, i + n)$ . Supposons que  $t_1$  et  $t_2$  soient deux points de transbordement. Le point de transbordement  $t_1$  est meilleur que le point de transbordement  $t_2$  si le coût d'insertion des demi-requêtes  $(i, t_1)$  et  $(t_1, i + n)$  est inférieur au coup d'insertion des demi-requêtes  $(i, t_2)$  et  $(t_2, i + n)$ .
4. Entre les lignes 17 et 21, le meilleur point de transbordement sauvegardé dans la variable  $BestT$  est utilisé pour séparer la requête  $(i, i + n)$  en deux demi-requêtes  $(i, BestT)$  et  $(BestT, i + n)$ . Chacune de ces deux demi-requêtes est insérée dans la solution  $S$  (lignes 18 et 19). Si le coût de la solution  $S$  après insertion des deux demi-requêtes est strictement inférieur au coût de la meilleure solution connue  $Sol$ , alors, la valeur de  $S$  est affectée à  $Sol$  (ligne 21).

Deux possibilités existent pour insérer le point de transbordement  $t$  dans la requête  $(i, i + n)$ . Il est possible d'insérer  $(i, t)$  avant  $(t, i + n)$  ou  $(t, i + n)$  avant  $(i, t)$ . Les deux possibilités sont testées et seule celle permettant une réduction plus importante du coût du transport est définitivement exécutée. Afin de ne pas surcharger l'algorithme « Transbordement », seule la possibilité d'insertion de  $(i, t)$  suivie de  $(t, i + n)$  est présentée. L'autre possibilité d'insertion n'est pas présentée. En effet, l'insertion de  $(t, i + n)$  suivie de  $(i, t)$  est effectuée de manière duale à celle de  $(i, t)$  et  $(t, i + n)$ .

Dans l'algorithme « Transbordement », il est nécessaire de garantir l'antériorité du déchargement d'un produit par un premier véhicule vis-à-vis de son rechargement par un second véhicule au point de transbordement.

La mise en œuvre de cette contrainte d'antériorité est effectuée comme suit : lorsque le produit associé à la requête  $(i, i + n)$  est transbordé au point  $t$ , la requête  $(i, t)$  est insérée dans la tournée d'un premier véhicule  $k_1$ . Le produit ainsi déposé au point de transbordement  $t$  par ce premier véhicule est disponible pour être rechargé par un second véhicule au plus tôt à la date  $\text{Max} (a_t, dt_i^{k_1})$ . Pour rappel  $a_t$ , désigne la borne inférieure de la fenêtre de temps du point de transbordement, qui est la date de visite au plus tôt de ce point ;  $dt_i^{k_1}$  désigne la date d'arrivée du véhicule  $k_1$  au

point de transbordement  $t$ . La seconde demi-requête  $(t, i + n)$  sera insérée dans la tournée d'un second véhicule  $k_2$  en s'assurant qu'à l'instant où le véhicule arrivera au point de transbordement, le produit sera disponible. Pour garantir cette contrainte, la borne inférieure de la fenêtre de temps du point de transbordement (ligne 9) est temporairement fixée à  $a_t \leftarrow \text{Max}(a_t, dt_i^{k_1})$  par la fonction « MiseAJourBorneInfFenetreDeTemps ». Elle est remise à sa valeur initiale après l'insertion de la seconde demi-requête. Le schéma 4.4 illustre la modification de cette borne inférieure.

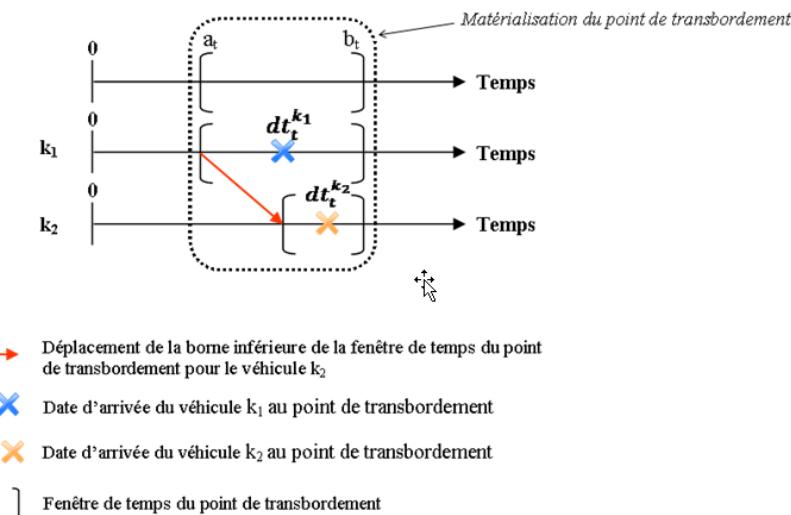


FIGURE 4.4 – Gestion du temps dans un point de transbordement

### 4.3 RÉSULTATS EXPÉRIMENTAUX

L'algorithme « VNSPourLePDPT » a été présenté de manière détaillée au travers des différents éléments nécessaires à son fonctionnement. Afin de juger de la validité et de la qualité de l'algorithme, des tests expérimentaux sont nécessaires et font l'objet de la présente section.

Tous les algorithmes ont été codés en langage java, sur un ordinateur tournant sous Windows 7 et possédant : 2.53 GHz, un processeur Intel(R) Core(TM) 2 Duo CPU avec 3.49 GO de RAM. La présente section est divisée en deux grandes parties. La première permet de valider le VNS sur des instances de PDP issues de la littérature et la seconde partie permet de le valider sur deux types d'instances de PDPT, dont l'un est généré par nos soins (voir chapitre 3) et l'autre est issu de la littérature. La validation en deux étapes est motivée par le fait que d'une part la méthode de résolution développée sur la base du VNS calcule à chaque itération une solution de PDP avant d'évaluer les gains potentiels liés à l'exploitation du transbordement et d'autre part, un algorithme qui résout le PDPT de manière efficace doit pouvoir résoudre le PDP de manière efficace.

Les paramètres du VNS sont définis comme suit :

- La phase de perturbation utilise des opérateurs de transfert cyclique. Du fait du caractère aléatoire de cette étape plusieurs exécutions sont donc nécessaires afin de juger de la stabilité de la méthode proposée.

Après quelques analyses expérimentales, il a été décidé que le VNS serait lancé 4 fois pour les instances de PDP et de PDPT. À la fin de ces quatre exécutions du VNS, la meilleure solution trouvée est retournée pour chaque instance. Tous les paramètres fixés pour l'exploitation de l'algorithme sont définis ci-dessous et ont été choisis de manière expérimentale.

- Le nombre d'itérations autorisées sans amélioration de la solution courante (MaxIter) est fixé à 15000. La durée maximale d'exécution de l'algorithme est fixée à 1200 secondes. Le paramètre  $\beta$  qui permet de définir les solutions prometteuses est fixé à 0,05.

#### 4.3.1 Validation du VNS sur des instances de PDP

Cette première validation repose sur une comparaison des résultats obtenus par le VNS et les meilleurs résultats connus sur des instances de PDP. Les instances<sup>1</sup> utilisées ont été proposées par (Li2001) et les meilleures solutions connues à ce jour pour ces instances sont à notre connaissance présentées dans (Ropke2007). Les instances proposées par (Li2001) sont de trois types : cluster, aléatoire et un mixte des deux (cluster/aléatoire). Ces instances ont été construites à partir des instances de Solomon (Solomon1987) utilisées dans le cadre du VRPTW. La description détaillée de ces instances est fournie dans (Li2001). Notons toutefois qu'une instance de type cluster commence par lc suivi du numéro de l'instance. Une instance de type aléatoire débute par LR suivi du numéro de l'instance. Enfin, une instance de type cluster/aleatoire débute par lrc suivi du numéro de l'instance. Par ailleurs, les clients appartenant aux instances dont le numéro débute par 1 ont une fenêtre de temps dont la largeur est inférieure aux clients des instances dont le numéro par 2. Par exemple, dans l'instance lc101, la largeur de la fenêtre de temps du client 1 est de 55 unités de temps et dans l'instance lc201, le client 1 a une fenêtre de temps dont la largeur est de 160 unités de temps.

Les résultats trouvés par le VNS sur chacune des instances de PDP sont regroupés dans le tableau 4.2 dont les colonnes ont les significations suivantes : « Instance » est la désignation d'une instance. « CPU » représente le temps en minutes qui a été nécessaire à l'obtention de la meilleure solution trouvée. « NBV » et « DIST » sont respectivement le nombre de véhicules et la distance totale trouvée par le VNS pour chacune des instances considérées sans application du transbordement. « BNBV » et « BDIST » sont respectivement le nombre de véhicules utilisés et la distance totale parcourue correspondant à la meilleure solution connue.

NUMERO	INSTANCE	NBV	DIST	CPU	BNBV	BDIST
1	lc101	10	828,9	04	10	828,9
2	lc102	10	828,9	04	10	828,9
3	lc103	9	1035,3	06	9	1035,3
4	lc104	9	860	05	9	860
5	lc105	10	828,9	04	10	828,9
6	lc106	10	828,9	04	10	828,9
7	lc107	10	828,9	04	10	828,9

1. <http://www.sintef.no/Projectweb/TOP/PDPTW/Li-Lim-benchmark/>

8	<i>lc108</i>	9	826,4	05	10	826,4
9	<i>lc109</i>	3	1000,6	07	9	1000,6
10	<i>lc201</i>	3	591,5	16	3	591,5
11	<i>lc202</i>	3	591,5	15	3	591,5
12	<i>lc203</i>	3	585,5	15	3	585,5
13	<i>lc204</i>	3	590,6	14	3	590,6
14	<i>lc205</i>	3	588,8	15	3	588,8
15	<i>lc206</i>	3	588,4	14	3	588,4
16	<i>lc207</i>	3	588,2	16	3	588,2
17	<i>lc208</i>	3	588,3	16	3	588,3
18	<i>lr101</i>	19	1650,8	03	19	1650,8
19	<i>lr102</i>	17	1487,6	04	17	1487,6
20	<i>lr103</i>	13	1292,7	07	13	1292,7
21	<i>lr104</i>	9	1013,3	08	9	1013,3
22	<i>lr105</i>	14	1377,1	05	14	1377,1
23	<i>lr106</i>	12	1252,6	05	12	1252,6
24	<i>lr107</i>	10	1111,3	07	10	1111,3
25	<i>lr108</i>	9	968,9	08	9	968,9
26	<i>lr109</i>	11	1208,9	06	11	1208,9
27	<i>lr110</i>	10	1159,3	07	10	1159,3
28	<i>lr111</i>	10	1108,9	09	10	1108,9
29	<i>lr112</i>	9	1003,7	08	9	1003,7
30	<i>lrc101</i>	14	1708,8	03	14	1708,8
31	<i>lrc102</i>	12	1558	04	12	1558
32	<i>lrc103</i>	11	1258,7	04	11	1258,7
33	<i>lrc104</i>	10	1128,4	06	10	1128,4
34	<i>lrc105</i>	13	1637,6	04	13	1637,6
35	<i>lrc106</i>	11	1424,7	04	11	1424,7
36	<i>lrc107</i>	11	1230,1	03	11	1230,1
37	<i>lrc108</i>	10	1147,4	05	10	1147,4
38	<i>Lc1_6_1</i>	60	1409,6	07	60	1409,6
39	<i>Lc1_6_2</i>	59	14164	10	58	14379,5
40	<i>Lc1_6_3</i>	54	15920,6	50	51	14569,3
41	<i>Lc1_6_4</i>	48	13567,5	80	48	13567,5
42	<i>Lc1_6_5</i>	60	14086,3	05	60	14086,3
43	<i>Lc1_6_6</i>	60	14090,8	05	60	14090,8
44	<i>Lc1_6_7</i>	60	14083,8	05	60	14083,8
45	<i>Lc1_6_8</i>	59	14670,4	30	59	14554,3
46	<i>Lc1_6_9</i>	56	14993,4	80	55	14648,1
47	<i>Lc1_6_10</i>	57	15337,7	65	54	14870,3

TABLE 4.2 – Résultat du VNS sur des instances de PDP

Le tableau 4.2 présente les résultats du VNS sur 47 instances. Les 37 premières instances ont toutes 50 requêtes chacune, tandis que les 10 dernières instances sont constituées de 300 requêtes chacune. Les cases NBV et DIST représentant les solutions trouvées par le VNS sont en grises si et seulement si elles sont identiques aux cases équivalentes BDIST et BNBV. De l'analyse du tableau, nous en retirons les enseignements suivants :

- Le VNS est de qualité satisfaisante pour résoudre des instances de PDP de manière approchée. En effet, sur les 47 instances, le VNS a été capable d'établir une correspondance parfaite entre (NBV, DIST) et (BNV, BDIST) sur 42 instances.
- La résolution des instances avec fenêtres de temps larges consomme plus de temps de calcul que la résolution des instances avec fenêtres de temps réduites. Cette observation est visible dans la colonne « CPU » du tableau 4.2, pour les instances lc201 à lc208.
- La moyenne des écarts de chaque instance entre la meilleure solution et la solution du VNS est de 1,4 % sur le nombre de véhicules et de 1,31 % sur la distance totale. Ces écarts sont calculés en faisant la moyenne de  $\frac{NBV - BNBV}{BNBV} * 100$  et de  $\frac{DIST - BDIST}{BDIST} * 100$ . Les écarts précédemment calculés permettent de conclure qu'en moyenne, le VNS est capable de trouver des solutions dont la différence de coût avec la meilleure solution connue est faible.
- Pour l'instance *Lc1\_6\_2*, le VNS n'a pas trouvé un nombre de véhicules inférieur à celui de la meilleure solution connue. Toutefois, il a été capable de trouver une meilleure distance.

### 4.3.2 Validation du VNS sur des instances de PDPT

La validation du VNS sur les instances de PDPT s'effectue sur deux groupes d'instances. Le premier regroupe les instances générées au chapitre 3 dans le cadre du modèle exact. Le second groupe concerne les instances issues de la littérature.

#### 4.3.2.1 Test du VNS sur les instances générées au chapitre 3

Cette première phase de la validation du VNS vise à vérifier son fonctionnement pour résoudre des instances de PDPT de taille réduite. Celui-ci a été testé sur les problèmes générés suivant la méthode décrite au chapitre 3 et dont la solution optimale est connue pour chacun d'eux. Sur chacune des instances E1 à E24 présentées dans le 3.4.3 du chapitre 3, le VNS a permis de trouver la solution optimale. La solution pour chacune des instances considérées est visualisable dans le chapitre 3. En plus de la vérification du VNS sur de telles instances, l'intérêt de l'utilisation du VNS par rapport au solveur d'équations linéaires, réside dans sa capacité à trouver les solutions optimales en temps négligeable. Cet aspect est visualisable dans le tableau 4.3 dans lequel « TVNS » représente le temps CPU consommé par le VNS en seconde et « Tsolveur » le temps CPU consommé par le solveur en seconde.

Instances	Nombres de requêtes	TVNS	Tsolveur
E1	3	<1	3,40
E2	3	<1	3,10
E3	3	<1	4,40
E4	4	<1	12,10
E5	4	<1	29,40
E6	4	< 1	19,90
E7	5	< 1	58,40

<i>E8</i>	5	< 1	71,10
<i>E9</i>	5	< 1	110,30
<i>E10</i>	6	2	377,30
<i>E11</i>	6	2	1103,40
<i>E12</i>	6	2	251,80
<i>E13</i>	7	3	5701
<i>E14</i>	7	3	4753,30
<i>E15</i>	7	3	> 2000
<i>E16</i>	10	6	> 50000
<i>E17</i>	2	< 1	1,90
<i>E18</i>	2	< 1	0,10
<i>E19</i>	3	< 1	4,10
<i>E20</i>	3	< 1	1,50
<i>E21</i>	3	< 1	13
<i>E22</i>	4	< 1	18,60
<i>E23</i>	5	2	1512,30
<i>E24</i>	6	4	37000

TABLE 4.3 – Temps CPU du VNS comparé au temps CPU du solveur

La phase suivante de validation de la méthode de résolution s'est ensuite appuyée sur les instances issues de la littérature.

#### 4.3.2.2 Validation du VNS sur les instances issues de la littérature

Les rares instances accessibles de la littérature traitant du PDPT que nous ayons trouvé sont proposées par (Qu2012). Chaque instance possède 25 requêtes et un point de transbordement. La taille des instances étant importante pour que celles-ci soient résolues de manière exacte par un solveur, les auteurs ont développés une stratégie pour les générer de manière à connaître par avance la solution optimale pour satisfaire l'ensemble des requêtes qu'elle comporte. À cette fin, les auteurs générèrent au préalable une instance correspondant à un problème PDP avec fenêtre de temps contenant uniquement 6 requêtes. Les auteurs ont développé une méthode de branch and price pour résoudre cette instance de PDP. Un point de transbordement est ensuite rajouté à cette instance en augmentant le nombre de requêtes jusqu'à 25 de manière à conserver la solution optimale du PDP d'origine. L'ajout des requêtes utilise un principe semblable au principe des sécantes décrit au chapitre 3.

Dans le cadre des expérimentations que nous effectuons, l'ensemble des points de transbordement T comporte un seul point dont la position est fournie dans le descriptif des instances proposées par (Qu2012).

Au total, 50 instances de PDPT avec la meilleure solution connue ont été générées dans (Qu2012). Ces instances sont structurées en 5 groupes. Chaque groupe  $PDPT_x (1 \leq x \leq 5)$  est constitué de 10 instances « newdata\_x01 » à « newdata\_x10 ». Dans ce chapitre, l'objectif dans l'utilisation de ces instances est de :

- Montrer que pour certaines d'entre elles, la pratique du transbordement peut permettre de réduire soit le nombre de véhicules utilisés, soit la distance totale parcourue, voire les deux. Ainsi, chacune des instances sera résolue avec et sans transbordement et les deux solutions obtenues seront comparées entre elles.
- Montrer l'efficacité du VNS proposé en dénombrant le nombre d'instances pour lequel le VNS a été capable de trouver la meilleure solution connue.

Le tableau 4.4 représente les résultats obtenus par le VNS sur les instances utilisées. Les colonnes « *NBV* » et « *DIST* » sont respectivement le nombre de véhicules et la distance totale trouvée par le VNS sans transbordement. Les colonnes « *NBVT* » et « *DISTT* » sont respectivement le nombre de véhicules et la distance totale trouvée par le VNS en exploitant le transbordement. Les colonnes « *BNBVT* » et « *BDIST* » sont respectivement le nombre de véhicules et la distance totale de la meilleure solution connue pour chaque instance. Ces valeurs sont identiques pour les instances comprises dans un même groupe et proviennent de (Qu2012). Les colonnes « *GAINV* » et « *GAIND* » représentent respectivement le gain obtenu par application du transbordement sur les instances de *PDP*. Ces gains sont respectivement calculés par  $GAINV = \frac{NBV - NBVT}{NBVT} * 100$  et  $GAIND = \frac{DIST - DISTT}{DISTT} * 100$ . Dans le tableau 4.4, la couleur gris sombre est utilisé lorsque  $NBVT < NBV$ , autrement dit si le transbordement permet une réduction du nombre de véhicules. La couleur gris clair est utilisé lorsque  $DISTT < DIST$  autrement dit si le transbordement permet de réduire la distance parcourue.

L'observation du tableau 4.4 qui synthétise l'ensemble des résultats obtenus par le VNS sur la résolution de 50 instances de *PDPT* permet de mettre en évidence que :

- Le VNS a permis de trouver la meilleure solution connue dans 30 instances sur un total de 50. En effet, sur ces instances,  $(NBVT, DISTT) = (BNVT, BDISTT)$ . Ces résultats démontrent que notre approche de résolution basée sur le VNS est de qualité satisfaisante.
- L'application du transbordement sur les instances de *PDP* permet d'obtenir un gain moyen de 27,3% (colonne *GAINV*) sur le nombre de véhicules utilisés et de 8,5% (colonne *GAIND*) sur la distance totale parcourue.
- Il apparaît clairement que le transbordement permet de réduire le nombre de véhicules dans 30 instances (colonnes *NBVT*) et la distance totale dans 44 instances (colonnes *DISTT*). Il peut donc être déduit que le transbordement peut être un moyen efficace dans certaines circonstances pour optimiser le transport des produits.

Notons que les auteurs des instances que nous utilisons dans ce travail et dont les meilleures solutions sont connues ont aussi résolu les dites instances de manière approchée en utilisant le GRASP. Toutefois, les résultats obtenus dans la résolution de ces instances avec le VNS proposé par nos soins ne peuvent être comparés à ceux obtenus par le GRASP. En effet, les hypothèses de résolutions considérées ne sont pas les mêmes. Dans notre travail, nous supposons que le nombre de véhicules disponibles pour sa-

Name	VNS							(Qu2012)
	NBV	DIST	NBVT	DISTT	GAINV	GAIND	BNBVT	
<i>PDPT1</i>								
newdata_101	3	2849,08	2	2510,94	50	13,4	2 2510,94	
newdata_102	3	2845,87	2	2510,94	50	13,3		
newdata_103	3	2771,15	3	2771,15	0	0		
newdata_104	3	2727,52	2	2510,94	50	8,6		
newdata_105	3	2963,3	3	2945,61	0	0,6		
newdata_106	3	2643,18	2	2510,94	50	5,2		
newdata_107	3	2994,9	2	2510,94	50	19,2		
newdata_108	3	2784,27	3	2717,27	0	2,4		
newdata_109	3	2809,10	3	2809,10	0	0		
newdata_110	3	2820,08	2	2510,94	50	12,3		
<i>PDPT2</i>								
newdata_201	2	2483,19	2	2359,01	0	5,2	2 2359,01	
newdata_202	2	2507,3	2	2359,01	0	6,2		
newdata_203	3	2703,38	2	2359,01	50	14,5		
newdata_204	3	2879,40	2	2359,01	50	22,0		
newdata_205	2	2413,57	2	2359,01	0	2,3		
newdata_206	2	2408,54	2	2359,01	0	2,0		
newdata_207	3	2695,34	2	2359,01	50	14,2		
newdata_208	2	2473,85	2	2359,01	0	4,8		
newdata_209	2	2447,90	2	2359,01	0	3,7		
newdata_210	2	2359,01	2	2359,01	0	0		
<i>PDPT3</i>								
newdata_301	3	2659,78	3	2659,78	0	0	2 2384,84	
newdata_302	3	2735,55	2	2384,84	50	14,7		
newdata_303	3	2665,24	3	2603,25	0	2,3		
newdata_304	3	2654,21	2	2384,84	50	11,2		
newdata_305	3	2657,90	3	2654,38	0	0,1		
newdata_306	3	2663,31	2	2384,84	50	11,6		
newdata_307	3	2652,96	3	2651,19	0	0,0		
newdata_308	3	2672,90	3	2662,02	0	0,4		
newdata_309	2	2678,94	2	2384,84	0	12,3		
newdata_310	3	2676,63	3	2676,31	0	0,01		
<i>PDPT4</i>								
newdata_401	3	2835,08	2	2405,80	50	18,8	2 2405,80	
newdata_402	3	2841,30	2	2405,80	50	19,1		
newdata_403	3	2932,50	2	2405,80	50	22,9		
newdata_404	3	2846,42	2	2405,80	50	19,3		
newdata_405	3	2822,81	2	2405,80	50	18,3		
newdata_406	3	2844,65	2	2405,80	50	19,2		
newdata_407	3	2908,20	2	2405,80	50	21,9		
newdata_408	3	2896,22	2	2405,80	50	21,4		
newdata_409	3	2890,18	2	2405,80	50	21,1		
newdata_410	3	2765,53	2	2405,80	50	15,9		
<i>PDPT5</i>								
newdata_501	4	3953,93	3	3479,03	33,3	13,6	3 3479,03	
newdata_502	4	3580,48	3	3479,03	33,3	2,9		
newdata_503	4	3655,13	3	3479,03	33,3	5,0		
newdata_504	4	3570,66	3	3479,03	33,3	2,6		
newdata_505	4	3598,98	3	3479,03	33,3	3,4		
newdata_506	4	3571,80	4	3571,80	0	0		
newdata_507	4	3536,71	3	3479,03	33,3	1,6		
newdata_508	4	3542,10	4	3538,10	0	0,1		
newdata_509	4	3918,60	3	3479,03	33,3	12,6		
newdata_510	4	3564,00	3	3479,03	33,33	2,4		
Gain moyen					27,3	8,9		

TABLE 4.4 – Résolution du PDPT par le VNS

tisfaire l'ensemble des requêtes n'est pas limité et la stratégie de résolution vise à satisfaire l'ensemble des requêtes avec un nombre de véhicules et une distance minimum. En comparaison, les auteurs cités travaillent sous l'hypothèse d'un nombre de véhicules limités et cherchent à résoudre les problèmes de PDPT de la manière suivante : le nombre maximum de véhicules est initialisé avec les résultats fournis par la solution optimale issue du branch and bound. Par ailleurs le nombre de véhicules étant borné, le GRASP peut ne pas trouver de solution satisfaisante à l'ensemble des requêtes. Le tableau 4.5 permet de mettre en évidence cette différence. Dans ce tableau, les colonnes MRIG (respectivement MRIV) représente le pourcentage de requêtes non satisfaites dans un groupe en utilisant le GRASP (respectivement l'heuristique « VNSPourLePDPT »). Ce pourcentage est obtenu en faisant le rapport entre la somme des requêtes non satisfaites dans un groupe sur le nombre total de requêtes du groupe.

Nom du groupe d'instances	MRIG	MRIV
PDPT1	30%	0
PDPT2	0%	0
PDPT3	20%	0
PDPT4	0%	0
PDPT5	10%	0

TABLE 4.5 – *Mise en évidence de la différence entre le VNS et le GRASP*

#### 4.4 CONCLUSION

Ce chapitre était axé sur la résolution approchée du PDPT. En effet, comme observé au chapitre précédent le PDPT est un problème NP-difficile et l'utilisation de méthodes exactes pour résoudre des instances de grandes tailles s'avèrent impossible. Pour cette raison, l'adaptation du VNS pour résoudre le PDPT a été discutée tout au long de ce chapitre.

Ainsi, le VNS proposé pour résoudre le PDPT a été construit avec un souci d'efficacité et de maîtrise de temps. Pour obtenir cette efficacité, un nouveau voisinage appelé TC-r-b et basé sur le transfert cyclique a été proposé afin de perturber la solution courante à chaque itération. La solution issue de la perturbation a été successivement améliorée par deux algorithmes : la recherche locale classique couplée à l'utilisation de trois voisinages et un algorithme permettant le transbordement de produits entre véhicules durant le transport. Pour maîtriser le temps de calcul, le VNS que nous avons proposé utilise deux notions fondamentales : la première est liée à la notion de solution prometteuse. Ainsi, les seuls voisinages explorés sont ceux de « solutions prometteuses ». Comme expliqué plus haut dans le chapitre, il s'agit d'une stratégie de réduction appelée VNS réduit. La seconde approche consiste à n'utiliser le transbordement que lors d'une phase de post-optimisation, afin d'améliorer la solution de PDP obtenue à chaque itération du VNS.

Enfin, pour valider la méthode proposée, celle-ci a été testée sur des jeux de données de PDP et de PDPT. L'observation des différents résultats a permis de valider le VNS, mettre en évidence son efficacité, ainsi que l'intérêt du transbordement en tant que levier d'optimisation.

# 5

## CADRE D'APPLICATION DES TRAVAUX DE LA THÈSE – LE PROJET PRODIGE

### SOMMAIRE

5.1	INTRODUCTION . . . . .	123
5.2	PRÉSENTATION GÉNÉRALE DU PROJET PRODIGE . . . . .	123
5.2.1	Produit intelligent . . . . .	124
5.2.2	Routage orienté produit . . . . .	125
5.2.3	Architecture matérielle et logicielle . . . . .	126
5.2.4	Intégration des modules Routage et Portail Web . . . . .	130
5.3	CONTEXTE DE TRANSPORT ÉTUDIÉ DANS PRODIGE . . . . .	132
5.3.1	Reconfiguration des tournées dans PRODIGE . . . . .	133
5.3.2	Synthèse des caractéristiques du problème étudié . . . . .	136
5.3.3	Résolution du PDPT – du contexte statique au contexte changeant . . . . .	137
5.4	ÉVOLUTION DE LA MÉTHODE DE RÉSOLUTION . . . . .	139
5.4.1	Prise en compte d'une position quelconque des véhicules	139
5.4.2	Prise en compte d'une charge initiale . . . . .	141
5.4.3	Adaptation de la méthode de résolution . . . . .	142
5.5	VALIDATION EXPÉRIMENTALE . . . . .	143
5.5.1	Expérimentation théorique . . . . .	143
5.5.2	Expérimentation en situation réelle . . . . .	149
5.6	CONCLUSION . . . . .	154



## 5.1 INTRODUCTION

**L**A présente thèse, s'inscrit dans le cadre du projet de recherche PRODIGE (Routage de PRODuits IntelliGEnts) financé par l'Agence Nationale de la Recherche (ANR), s'étant déroulé d'octobre 2009 à mars 2013.

Les développements présentés dans les chapitres précédents définissent les logiques d'optimisation qui ont été implémentées dans la solution logicielle et matérielle PRODIGE visant à rationaliser l'activité de transport et à promouvoir de nouvelles pratiques en logistique collaborative fondées sur le transbordement.

Le projet PRODIGE est donc une opportunité de tester en situation réelle ces développements, afin d'en mesurer l'efficacité en termes de routage.

Ce chapitre de thèse est décomposé en six parties principales. En partie 2, une présentation du projet, et de ses fondements est proposée. En partie 3, un accent particulier est mis sur la présentation de la problématique du transport dans le contexte de PRODIGE. Cette partie est l'occasion d'introduire et d'expliquer de manière détaillée la notion de reconfiguration de tournées dans un contexte un changeant, de synthétiser et caractériser le problème de transport étudié. Le problème étudié dans le cadre de PRODIGE couvre donc un périmètre plus large que celui qui a été considéré dans les chapitres précédents, induisant de fait de nouvelles contraintes. La considération de ces nouvelles contraintes nécessite une évolution des méthodes de résolution présentées au chapitre 4. L'évolution de ces méthodes est analysée en partie 4. En partie 5, une validation expérimentale des méthodes adaptées est effectuée. Elle consiste en une expérimentation théorique et une expérimentation en situation réelle. La partie 6 permet de conclure le chapitre.

## 5.2 PRÉSENTATION GÉNÉRALE DU PROJET PRODIGE

La cible scientifique et technique du projet PRODIGE concerne le sujet très actuel du développement de solutions de « tracking / tracing » des produits tout au long de leur cycle de vie en centrant l'analyse sur le transport qui est l'activité de la chaîne logistique la moins instrumentée. Dans un tel contexte, le projet vise au développement d'une solution s'intéressant à :

- La recherche d'innovations organisationnelles tendant à optimiser les flux et favorisant la logistique collaborative par une meilleure synchronisation des activités menées par les différents acteurs de la chaîne logistique.
- La conception et la réalisation d'une solution technique fondée sur la technologie RFID (Radio Frequency Identification), dont le rôle est de vérifier l'intégrité des chargements dans les véhicules en contrôlant toutes les entrées et sorties de charges.
- La conception et la réalisation d'une solution logicielle « intelligente » et distribuée s'appuyant sur une informatique embarquée à bord de chaque véhicule, permettant de localiser ce dernier et de

dialoguer avec un serveur informatique chez le prestataire de transport.

Une telle solution permet d'assurer les fonctions de « tracing » et « tracking » des produits durant leur transport. La fonction de « tracing » correspond à la mise en œuvre d'une chaîne d'information donnant la faculté à une organisation de retrouver l'historique, l'utilisation ou la localisation d'une entité au moyen d'identifications enregistrées. La fonction de « tracking » permet de suivre les produits en temps réel afin d'optimiser les itinéraires planifiés et de gérer les chargements et déchargements des marchandises. Ces deux fonctions posent des problèmes « classiques », en partie instrumentés par des solutions techniques. La réelle innovation du projet réside alors dans le développement d'une solution fondée sur deux notions fondamentales :

- Le produit est dit « intelligent », i.e. il possède des propriétés lui permettant d'être un des acteurs dans les décisions concernant son acheminement au client.
- Le service à optimiser est le transport du produit et non le guidage des véhicules à proprement parler.

Ces éléments sont brièvement développés dans les paragraphes suivants.

### **5.2.1 Produit intelligent**

#### **5.2.1.1 Définition**

Le fondement même du produit intelligent, abordé du point de vue du contrôle des processus réside dans l'aptitude que l'on souhaite donner à un objet subissant une activité de transformation ou de transport à percevoir son environnement, à participer à l'élaboration des décisions le concernant quant à son devenir.

(Meyer2009) assimilent la notion de produit intelligent, considéré dans sa forme la plus simple, à la recherche d'une connexion entre le monde physique et les systèmes d'information, c'est-à-dire comme une approche facilitatrice de l'alignement du flux d'information avec le flux physique. La caractérisation du produit intelligent, tel qu'il est perçu dans le cadre du projet, s'appuie sur la définition donnée par (Wong2002), à savoir qu'il doit :

1. Posséder un identifiant unique afin qu'il ne puisse subsister aucune ambiguïté sur l'objet manipulé.
2. Pouvoir communiquer avec son environnement. Il doit être en mesure d'acquérir de l'information venant de l'extérieur et d'envoyer en retour des données le caractérisant. Il s'agit ici de mettre en avant l'existence de canaux de communication.
3. Stocker des données le concernant. Un produit intelligent dispose des informations le caractérisant de manière à ce qu'il puisse participer à l'élaboration des décisions quant à son futur proche. Cela tend à distribuer l'information sur les produits plutôt qu'à centraliser ces informations en une seule base de données.
4. Déployer un langage pour mettre en œuvre l'ensemble des caractéristiques énoncées ci-dessus, c'est-à-dire, qu'il doit connaître au-delà

de l'existence de canaux de communication, des règles sémantiques et syntaxiques afin de pouvoir dialoguer avec son environnement.

5. Pouvoir assurer ou participer à une prise de décision sur son devenir. Le produit intelligent est capable de prendre des décisions réflexes telle que la mise en route d'une alarme, ou des décisions plus élaborées pouvant porter sur le choix d'une destination dans le cas d'un routage ; le type de décision implémentée dépend alors du niveau d'instrumentation du produit, i.e. la technologie qui lui est associée pour accueillir les logiques décisionnelles.

Sur la base de la définition précédente, (Wong2002) structurent l'intelligence du produit en deux niveaux. Les produits possédant les propriétés 1 à 3 sont considérés comme instrumentés afin de les rendre *communiquant* et *mémorisant*. Les produits ayant l'ensemble des propriétés 1 à 5 sont considérés comme *intelligents*.

### 5.2.1.2 Adaptation au projet PRODIGE

Le projet PRODIGE s'intéresse au transport de marchandises à l'échelle régionale. L'exploitation de la notion de produit intelligent concerne plus précisément les trois premiers niveaux d'intelligence de la classification de (Wong2002) et s'est donc naturellement faite à l'échelle de la palette et la remorque. Durant le transport, les palettes et la remorque sont assimilées à une entité indivisible et la palette sert d'unité d'encapsulation éphémère d'un ensemble de cartons homogènes ou hétérogènes.

Dans le cadre du projet, la palette est instrumentée afin de répondre aux fonctions de communication, de mémorisation d'informations, via un identifiant unique – souvent donné par un code unique quel que soit le standard de traçabilité adopté – par exemple un code EPC (Electronic Product Code) ou un code type SSCC (Serial Shipping Container Code). L'instrumentation de la palette permet de lui conférer une intelligence informationnelle.

La remorque doit être instrumentée afin de posséder, au-delà des fonctions de communication, mémorisation, les fonctions de perception et décision. L'instrumentation est partiellement déportée vers la cabine du véhicule, afin d'interagir avec le chauffeur.

### 5.2.2 Routage orienté produit

Dans le cadre du transport, le routage orienté produit (Tchapnga2013) caractérise l'abandon d'une approche de traitements dans lequel les décisions et les informations portant sur le routage sont purement centralisées vers une approche dans laquelle les informations et les décisions sont distribuées. La distribution de l'intelligence décisionnelle et de l'information sont effectuées au sein de l'ensemble des éléments constituant le système de transport dont les palettes, la remorque et le tracteur. Dans le cadre de PRODIGE, de manière concrète le routage orienté produit consiste à reconfigurer « dynamiquement » les tournées en privilégiant le respect des contraintes inhérentes aux produits. Ces contraintes logistiques peuvent être stockées dans les tags portés par les palettes. La reconfiguration des

tournées nécessite un suivi permanent de l'activité de transport qui se fait aux :

- Points de contrôle à position fixe mais à horodatage variable : ces points correspondent au lieu de visite par le « produit intelligent » de chaque client de la tournée. La localisation exacte d'un véhicule permet en effet de dater précisément l'événement clé « arrivée chez le client » .
- Points de contrôle à période fixe mais à localisation variable : ces points correspondent à une acquisition périodique de la position du véhicule, par géo-localisation, au cours du déroulement des tournées. L'exploitation d'une telle logique de localisation permet d'anticiper les dérives de temps avant même l'arrivée du véhicule chez le client.

Si la notion de produit intelligent est l'un des fondements du principe de routage orienté produit, elle ne suffit pas en elle-même à s'affranchir des logiques classiques de routage de véhicules. Ainsi le routage orienté produit est plus particulièrement renforcé par l'exploitation du transbordement opportuniste. Le transbordement consiste à utiliser plus d'un véhicule pour acheminer une palette d'un chargeur à son destinataire, donnant ainsi l'aptitude au produit de « passer d'un véhicule à un autre » et à créer une analogie avec l'usager des transports en commun qui adapte son trajet en fonction des conditions de trafic et des opportunités de correspondance entre différents modes de transport. Son caractère « opportuniste » propose une rupture de pratique avec le classique « cross-docking » et qualifie une exécution de transbordement en réponse à la détection de dérives de temps ou d'aléas menant à une reconfiguration possible des tournées dans une recherche systématique d'une meilleure performance. Cette propriété caractérise par ailleurs la possibilité d'exécuter une telle activité en un ensemble de lieux, qu'il s'agisse de clients visités dans les tournées, ou de plateformes ad-hoc, dont le choix relève de logiques d'optimisation spécifiques.

### 5.2.3 Architecture matérielle et logicielle

La nouvelle approche de résolution des problèmes de tournées de véhicules basée sur le concept du produit intelligent et plus particulièrement sur la notion de transbordement nécessite le déploiement d'une architecture matérielle et logicielle. L'architecture matérielle présente la répartition des différents composants matériels ainsi que les interactions qui existent entre eux afin de favoriser et permettre la création d'une chaîne d'information reliant tous les acteurs du transport. L'architecture logicielle présentée décrit de manière schématique les interactions et les connexions qui existent entre les différents modules logiciels dans le but de réaliser les fonctionnalités attendues.

L'architecture matérielle associée à la solution technique développée dans le cadre du projet PRODIGE est représentée par la figure 5.1.

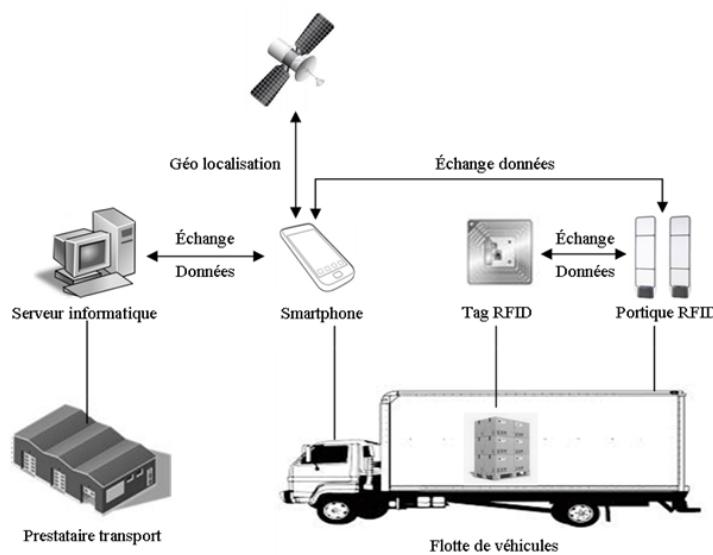


FIGURE 5.1 – Architecture matérielle de PRODIGE

Dans PRODIGE, l'élaboration d'une chaîne d'information allant du produit, en passant par le prestataire en charge du transport, jusqu'au client de l'activité transport, est nécessaire afin de permettre : la localisation permanente des produits, la détection d'aléas de transport et leur traitement à des fins de reconfiguration de trajets. L'élaboration d'une telle chaîne nécessite des développements techniques caractérisés par une instrumentation. Cette instrumentation se base, comme décrit plus tôt, sur le déploiement d'une intelligence informationnelle au niveau des palettes à transporter : cette intelligence est alors mise en œuvre par l'intermédiaire de puces RFID lues par un portique développé dans le cadre du projet pour équiper les véhicules de transport. Elle s'appuie également sur une intelligence décisionnelle déployée au niveau de la remorque, et plus généralement du véhicule dans son ensemble. Elle est supportée par une informatique embarquée (Smartphone) déportée au niveau de la cabine du véhicule, et par une capacité décisionnelle pour le calcul des solutions de routage qui est déportée vers le serveur informatique. En effet, la puissance de calcul disponible au niveau du serveur, favorise une résolution plus rapide des problèmes de routage de grande dimension.

L'architecture logicielle associée à la solution technique développée dans le cadre du projet PRODIGE et supportée par l'architecture matérielle précédemment décrite est représentée par la figure 5.2.

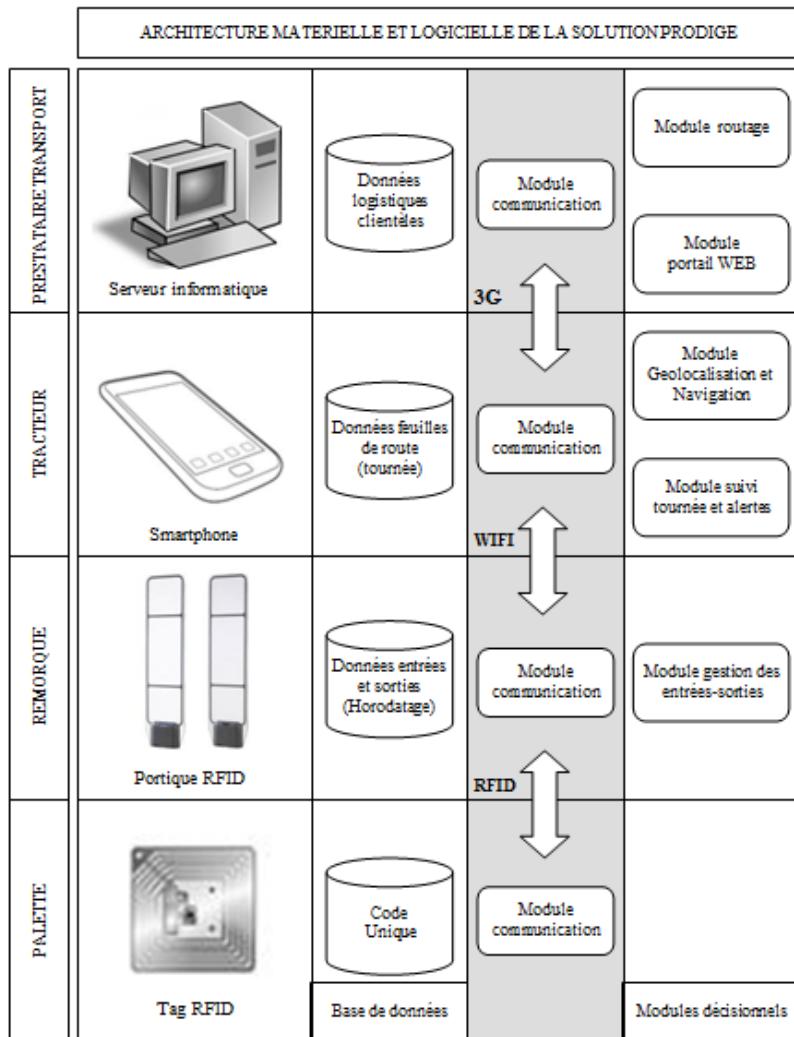


FIGURE 5.2 – Architecture matérielle et logicielle de PRODIGE

Les différents composants logiciels de la solution PRODIGE vont être détaillés dans les paragraphes qui suivent en insistant particulièrement sur les modules décisionnels. Le cycle de vie de l'information qui s'étend du recueil des données sur la palette par le portique via le RFID jusqu'à leurs transmissions successives par WIFI au Smartphone et par 3G au serveur, ainsi que les bases de données afférentes aux composants de l'architecture matérielle sont décrits succinctement dans la figure 5.2. De même les motivations sur les choix techniques du RFID et de la 3G dans le projet PRODIGE ne sont pas prédominantes au regard des objectifs de ce chapitre et sont donc discutées en annexe A.4 du document. La description des différents modules décisionnels se fera en se basant sur la figure 5.2 de haut en bas en présentant chaque composant matériel utilisé et les composants logiciels qui lui sont associés.

Le serveur informatique est localisé chez le prestataire de transport et est hébergé par un serveur d'applications. Le rôle du serveur d'applications est de fournir aux applications une infrastructure afin que celles-ci puissent s'exécuter. Ainsi, la gestion des entrées/sorties au réseau via le protocole « http » est gérée par le serveur, et cela permet aux applications de s'affranchir des contraintes techniques pour se focaliser uniquement

sur les aspects fonctionnels. Deux principales applications sont déployées au niveau du serveur : le module de routage et le module portail web.

- La fonctionnalité principale du module de routage est la mise en œuvre des algorithmes de routage pour générer des tournées optimales afin de satisfaire un ensemble de requêtes de transport.
- Le module portail web consiste en une interface homme-machine permettant aux utilisateurs d'exploiter les différents services offerts par les applications déployées au niveau du serveur. Cet interface visualisable à partir de n'importe quel navigateur web rend accessible en tout point ayant accès à internet les applications telles que le module de routage par exemple. Le module portail web permet au prestataire de transport : de collecter des données logistiques sur les clients du transport, d'appeler le module de routage en lui fournant les entrées nécessaires afin qu'il génère les tournées de véhicules, de visualiser en temps réel la situation du transport, de décider de la reconfiguration d'une tournée suite à l'occurrence d'événements perturbateurs. En résumé, le module portail web est le point d'entrée de la solution PRODIGE et permet d'accéder aux principaux services logiciels qui sont entre autres, la génération et la reconfiguration des tournées, la visualisation en quasi « temps réel » des tournées en cours d'exécution, la notification des tournées achevées et des requêtes de transport satisfaites.

Les tournées générées par le module de routage sont ensuite transmises sous forme de feuille de route à l'application embarquée ou Smartphone situé dans le tracteur ou la cabine et consultable par les chauffeurs. Le Smartphone est basé sur le système d'exploitation Android et permet l'exécution de deux principales applications : le module de géo localisation et de navigation et le module suivi tournée et alertes.

- Le module de géo localisation et de navigation permet de localiser le véhicule en temps réel donc les produits, mais aussi au chauffeur de naviguer vers les différents points de passages obligatoires qui constituent la feuille de route.
- Le module suivi tournée et alertes permet le monitoring des tournées c'est-à-dire le suivi des tournées en temps réel. Durant le transport, tous les événements marquants sont remontés vers le serveur. Les événements marquants peuvent être une dérive de temps suite à l'occurrence d'un événement perturbateur, l'arrivée et le départ chez un client.
- Nous venons de voir que les tournées générées au niveau du serveur par le module de routage sont transmises sous la forme d'une succession de points à visiter similaire à une feuille de route (voir annexe A.6) au Smartphone situé dans le tracteur. L'exécution des tournées définies par ces feuilles nécessitent le chargement et le déchargement des produits dans la remorque et met en œuvre un autre module appelé module de gestion des entrées/sorties. Le module de gestion des entrées/sorties a pour rôle l'identification immédiat via le portique des produits devant être transportés et qui sont chargés dans la remorque. L'information concernant l'identification des produits est enregistrée dans l'application embarquée.

Durant le transport, l'application communique avec le lecteur situé

dans la remorque pour récupérer des informations de diverses natures sur des produits, ici des palettes. Les puces RFID étant passives, la communication ne peut avoir lieu qu'à l'initiative du module de gestion des entrées/sorties et donc du chauffeur.

Nous avons montré que le serveur d'applications hébergeait deux applications essentielles qui sont respectivement le module de routage et le module portail web. Pour fonctionner, le module de routage a besoin d'informations qui proviennent du portail web et sont issues des utilisateurs (prestataire ou clients) qui placent des demandes de transport en spécifiant des requêtes de transport c'est-à-dire le lieu de collecte, de livraison, la quantité, la fenêtre de temps. Le détail de ces informations et la mise en forme préalable de celles-ci par le portail web avant leur transmission aux modules de routage fait l'objet de la partie suivante.

#### **5.2.4 Intégration des modules Routage et Portail Web**

Parmi les informations majeures à transmettre aux algorithmes d'optimisation se trouve la définition des fenêtres de temps. Ces intervalles sont caractérisés par une borne minimale et une borne maximale, dont la valeur est exprimée sous le format « timestamp ». Ce format permet de définir précisément en nombre de secondes une date à partir de la date de référence du 1<sup>er</sup> janvier 1970 0h00m00s. Les avantages d'utilisation d'un tel format sont multiples : il facilite la comparaison de deux dates et permet de calculer aisément la durée séparant ces dernières par une simple soustraction. Cette représentation est par ailleurs universelle et est utilisée dans la plupart des langages et systèmes de gestion de base de données.

Exemple : la date du 14 juillet 2013 12 : 00 correspond au timestamp 1373796000

Toute communication d'information de la solution logicielle PRODIGE vers les algorithmes d'optimisation se fait alors par le biais d'échange de trames de données selon des formats prédéfinis. Les informations en entrée du module de routage sont de trois natures et portent sur la flotte de transport, les requêtes et le distancier.

- Chaque véhicule appartenant à la flotte du transport est caractérisé par un identifiant, une capacité exprimée aussi bien en tonnes qu'en un nombre de palettes, si bien que l'on peut considérer aisément les problèmes de gestion de flotte de véhicules homogènes ou hétérogènes. De plus, chaque véhicule est associé à un lieu de départ et un lieu d'arrivée de tournée, chacun d'entre eux étant caractérisé par un identifiant et une fenêtre de temps selon le format précisé précédemment. La figure 5.3 représente la trame de données pour un véhicule, le nombre de trames échangées étant alors proportionnel au nombre de véhicules composant la flotte.

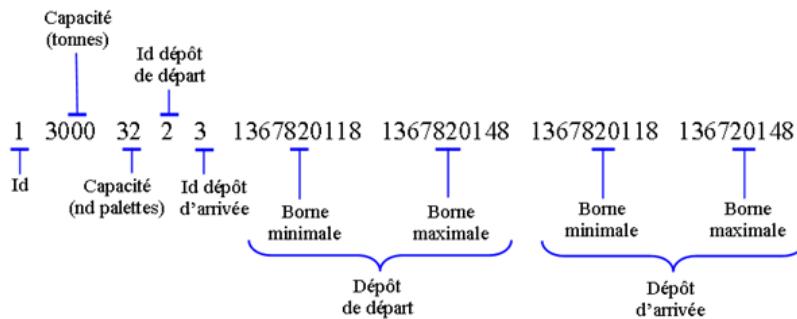


FIGURE 5.3 – Données caractérisant un véhicule

- Les requêtes de transport satisfaites par les véhicules sont caractérisées par un ensemble de données, telles que présentées dans la figure 5.4. Ces données sont transmises dans la même logique de format comprenant un identifiant de l'ordre de transport, des identifiants des opérations de chargement et déchargement. Les lieux de collecte et livraison sont spécifiés par un identifiant, une fenêtre de temps borne minimale, borne maximale et un temps de service correspondant à la durée pendant laquelle un véhicule reste sur le lieu concerné. Les identifiants d'actions de chargement et déchargement sont utilisés en interne par le serveur pour la gestion de la base de données « données logistiques et clientèles ».

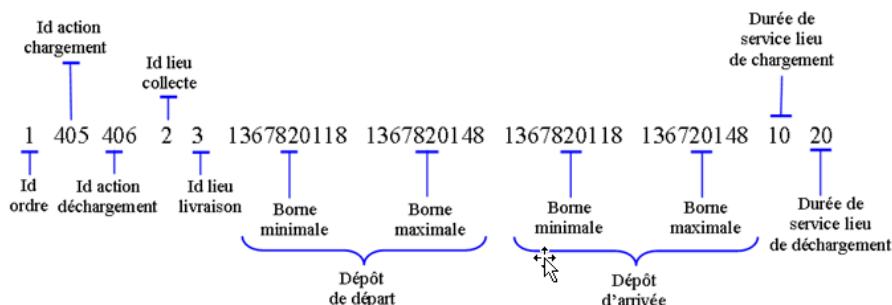


FIGURE 5.4 – Données caractérisant l'ordre de transport

- Le distancier utilisé dans PRODIGE fait appel à un logiciel de cartographie et navigation pour calculer les distances en mètres, et temps de parcours en secondes séparant tout couple de clients à visiter dans le problème à résoudre. Le distancier est donc une donnée d'entrée fournie au module de routage sous la forme d'un ensemble de trames d'information dont le format type est défini dans la figure 5.5.

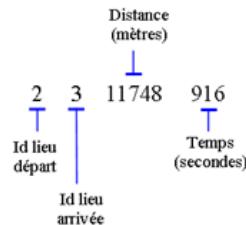


FIGURE 5.5 – Exemple de ligne du distancier

- Les algorithmes une fois renseignés avec ces informations en entrée calculent une solution de routage conformément aux logiques décrites dans le présent travail de thèse. L’organisation des tournées ainsi définie est restituée par le module de routage au module portail web sous la forme d’un ensemble de trames de données caractérisant chaque lieu visité au cours de la tournée, les dates d’arrivée et de départ du véhicule en ce lieu et le type d’action effectuée. Il est à noter que la codification des actions est la même que celle utilisée dans la caractérisation des ordres de transport. La figure 5.6 représente un extrait du résultat de la tournée affectée au véhicule 1, tel que l’algorithme le restitue à la solution logicielle PRODIGE.

Id lieu visité				
Id véhicule	1	1361981728	1361982028	0
		Date arrivée	Date départ	Id action

FIGURE 5.6 – Données en sortie de l’algorithme

### 5.3 CONTEXTE DE TRANSPORT ÉTUDIÉ DANS PRODIGE

La problématique au cœur de PRODIGE est le transport analysé à la fois suivant un contexte statique et un contexte changeant. Un contexte statique correspond à la planification des tournées et un contexte changeant caractérise la reconfiguration des tournées.

L’organisation du transport se décompose par ailleurs classiquement en deux phases : la planification et l’exécution. La planification consiste à organiser dans le temps un ensemble de tournées de véhicules optimales pour satisfaire les demandes de transport, ceci pour une durée limitée. L’exécution concerne la mise en œuvre réelle des tournées planifiées.

Le problème de ramassage et distribution avec transbordement a été analysé dans les chapitres précédents de manière statique. Cela signifie que tous les paramètres étaient supposés invariables dans le temps et connus par avance. Les différentes contributions apportées dans les chapitres précédents, tant sur la modélisation par programmation linéaire en

nombres mixtes de ce problème, que sur sa résolution par des méthodes exactes et approchées appartiennent à la phase de planification.

### 5.3.1 Reconfiguration des tournées dans PRODIGE

Dans un contexte réel de mise en œuvre de l'activité de transport, toutes les informations ne sont pas toujours connues au moment de la phase de planification et peuvent être révélées au fur et à mesure que les véhicules exécutent les activités de transport, i.e. lors de l'exécution des tournées telle que décidée par la planification (Min1998). C'est par exemple le cas d'un nouvel ordre de transport qui peut se présenter. Le processus de distribution peut également ne pas se dérouler selon le plan décidé et devoir faire l'objet de modifications, suite à l'occurrence d'événements pouvant détériorer la qualité de certaines performances attendues tant sur le plan du coût du transport que sur la qualité de service.

La considération de ces événements amène à appréhender le problème de collecte et livraison dans un contexte changeant. Dans un tel contexte, les tournées planifiées ne sont plus figées mais peuvent être reconfigurées en fonction de l'évolution des données du problème à résoudre. La reconfiguration est donc motivée par la volonté de garder le système de transport dans un état proche de son fonctionnement nominal, i.e. garantissant le respect des dates de livraison aux clients au détriment le plus souvent de la distance parcourue. Le changement de contexte évoqué repose sur un aspect événementiel et sur la notion de reconfiguration. À titre d'illustration, et pour mieux fixer les idées, l'exemple suivant est proposé.

#### 5.3.1.1 Exemple de reconfiguration de tournées

Afin de simplifier le propos, l'exemple considéré dans cette partie est restreint à un problème de livraison dans lequel deux véhicules doivent livrer des marchandises à sept clients, notés  $(1, 2, \dots, 7)$ , en partant du dépôt. Le graphe de gauche de la figure 5.7 représente les tournées prévisionnelles des deux véhicules, telles que définies par la résolution du problème « statique » visant à optimiser la distance parcourue, sous contraintes de respect de capacité et fenêtres de temps définies par des bornes minimales et maximales représentées entre crochets. Le graphe de droite permet de visualiser le suivi et l'évolution des deux tournées. La valeur entre parenthèses représente la date d'arrivée du véhicule chez le client  $dt_i^k$  associé au sommet du graphe. La durée de service en chaque point est supposée nulle. Dans cet exemple, les arcs non représentés sont supposés inexistant. C'est par exemple le cas lorsqu'un chemin ne relie pas directement deux villes entre elles.

Dans les figures 5.7 et 5.8, les points 8 et 9 ne sont pas des points remarquables au sens des clients. Il s'agit de points intermédiaires qui représentent la position courante des véhicules à un instant donné. Considérons l'instant de reconfiguration où les sommets 8 et 9 représentent les nouveaux points de départ des véhicules. Il est donc nécessaire de mettre à jour le distancier support au problème d'optimisation afin de déterminer la distance et le temps de parcours qui les séparent des autres points restant à visiter du problème en cours de résolution.

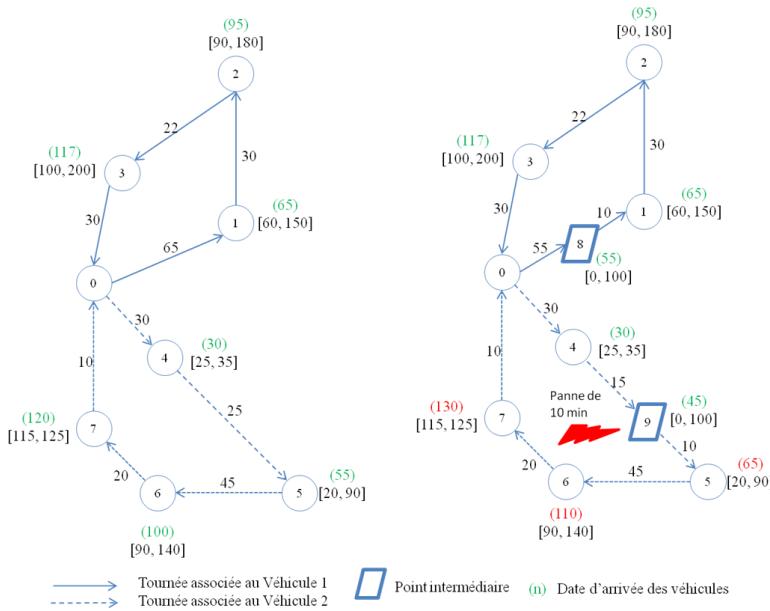


FIGURE 5.7 – Tournée retardée

Supposons que les véhicules débutent tous deux leurs tournées à la date 0, et que nous sommes dans un contexte où les tournées des deux véhicules sont suivies par géo-localisation.

Alors que l'observation de la tournée associée au véhicule 1 (arcs continus) se déroule sans aucun problème, il est constaté un événement perturbateur pour la tournée du véhicule 2. En effet, à la date 45, alors que le véhicule 2 associé à la tournée à arcs discontinus se trouve au point intermédiaire 9, une panne survient. La réparation de cette panne nécessite 10 unités de temps. Si aucune décision de modification de trajet n'est prise concernant la tournée du véhicule 2, celui-ci partira du point 9 à la date 55 et arrivera au point 5 à la date 65 tel que représenté sur le graphe de droite de la figure 5.7. Nous pouvons constater que la tournée du véhicule 2 ne respecte plus l'ensemble des contraintes, puisque le dernier client de la tournée ne peut pas être livré conformément à la fenêtre de temps de ce dernier ; le véhicule 2 arrive en effet à la date 130, alors que la fenêtre de temps est [115, 125].

Dans une telle situation, l'objectif de l'optimisation est de reconfigurer les tournées, en privilégiant le respect des contraintes de fenêtre de temps et de capacité des véhicules, tout en minimisant le nombre de véhicules et la distance parcourue. La reconfiguration des tournées peut alors s'appuyer sur trois principes :

- Modifier la séquence de visite des clients au sein d'une même tournée, permettant ainsi de ne pas impliquer dans la recherche d'une nouvelle solution des véhicules autres que celui subissant la perturbation ;
- Utiliser le principe du transbordement, lorsqu'aucune solution n'est possible au périmètre d'un véhicule. Le produit est alors acheminé à son client par plus d'un véhicule ;
- Si toutefois aucune solution n'est envisageable, le véhicule peut également continuer sa tournée après décision de l'exploitant de la flotte de ne pas livrer l'ensemble des clients dont le service est prévu dans

sa tournée ou de les livrer avec un retard après accord entre le décisionnaire et le client.

Dans le cadre de notre exemple, aucune solution n'est possible sans avoir recours au transbordement. Ainsi, s'il existe un lieu de transbordement, représenté par le sommet 10 sur le graphe de gauche de la figure 5.8, et compte tenu que les contraintes de fenêtre de temps sont ignorées, une solution peut être trouvée : le véhicule 1 ainsi que le véhicule 2 sont déroutés vers le lieu de transbordement associé au sommet 10. Notons qu'après les 10 minutes nécessaires à sa réparation, le véhicule 2 est de nouveau disponible.

Les deux véhicules qui se trouvent alors au sommet 10 opèrent un transbordement. Les charges des clients concernés sont illustrées dans le graphe de droite de la figure 5.8. Le transbordement s'opère de manière détaillée de la façon suivante : une fois arrivé au point 10, un échange de charges permet au véhicule 2 de servir un client (5) de moins dans sa tournée, alors que le véhicule 1 prend en charge la livraison d'un client (5) supplémentaire initialement prévue dans la tournée du véhicule 2. Les contraintes de temps associées aux clients de la tournée réalisée par le véhicule 1 permettent en effet d'absorber le retard induit par la nécessité de servir un nouveau client, pendant que le véhicule 2 sert les clients en respectant les contraintes de temps du fait de l'abandon d'un client.

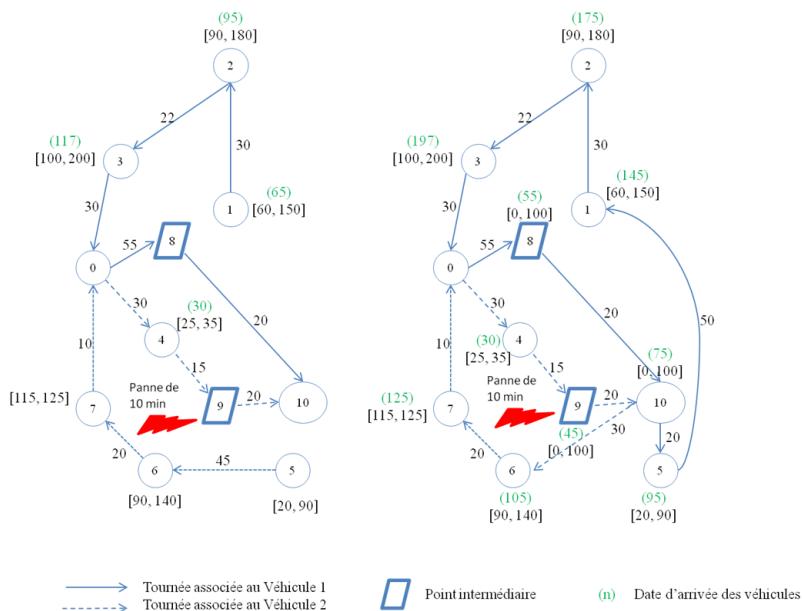


FIGURE 5.8 – Tournées reconfigurées

L'exemple précédent permet de mettre en exergue tous les aspects de transport considérés dans le cadre de PRODIGE. La partie suivante est l'occasion de les formaliser.

### 5.3.2 Synthèse des caractéristiques du problème étudié

Le problème de transport à considérer dans le cadre de PRODIGE est donc un problème de collecte et livraison avec transbordement et reconfiguration. La résolution de ce problème consiste à proposer un ensemble d'algorithmes qui permet la prise en compte :

- Des situations changeantes, telles que celles décrites ci-dessus, qui induisent une différence entre la tournée planifiée et la tournée réellement exécutée due à l'occurrence de perturbations ;
- Des charges initiales dans les véhicules ;
- Une position de départ quelconque des véhicules.

Classiquement, dans les problèmes de tournées reconfigurables, plusieurs événements sont considérés, ceux-ci sont classés en trois catégories (Berbeglia2010). Le premier groupe rassemble les événements en rapport à l'environnement tel que des congestions de trafic, ou encore des déviations de trajet dues aux routes rendues inaccessibles par des travaux ou des conditions climatiques exceptionnelles comme une inondation. Le second groupe concerne les clients de l'activité transport, citons par exemple l'apparition de nouvelles requêtes de transport ou l'annulation de certaines d'entre elles. Le dernier groupe concerne les ressources. Il s'agit d'événements tels les pannes techniques liées au véhicule telles qu'une défaillance du moteur, une crevaison ou bien ceux liés au chauffeur tel qu'un malaise. La majorité des aléas cités se traduisent par un retard dans l'exécution des tournées, souvent constaté en un point remarquable. Dans le cadre de ce travail, seuls ces derniers événements sont considérés.

Ainsi, considérant que les routes initialement planifiées peuvent être amenées à évoluer durant leurs exécutions, nous définissons le problème de tournées reconfigurables comme une succession de problèmes de type PDPT, dont les conditions initiales changent en fonction des événements détectés.

Le tableau 5.1 fait état des différences entre le problème traité dans le cadre de la résolution statique d'un problème de routage et celui correspondant à la reconfiguration des tournées.

Résolution du problème « statique »	Reconfiguration de tournées
Les véhicules partent tous du même dépôt	Chaque véhicule part d'une position qui lui est propre
Chaque véhicule part du dépôt à vide	Les véhicules partent avec une charge initiale non nulle

TABLE 5.1 – Différences entre les deux problèmes traités

Pour résoudre, cette nouvelle variante de notre problème de transport, l'objectif de la partie suivante est de décrire la méthode de résolution qui sera utilisée pour résoudre le PDPT dans un contexte changeant. Il ne

s'agit pas d'une méthode de résolution spécifique, mais d'une adaptation des techniques développées au chapitre 4 pour la résolution des problèmes statiques, afin de planifier les tournées en prenant en considération les nouvelles contraintes présentées ci-dessus. Le choix de l'adaptation réside dans la volonté de réutilisation des composants logiciels existants, ce qui permet d'avoir un gain de temps au niveau de l'élaboration de la méthode, mais aussi d'avoir une heuristique capable de fonctionner simultanément dans un contexte statique et changeant. Préalablement à la présentation de la méthode de résolution du PDPT dans un contexte changeant, des discussions sur l'évolution de la méthode de résolution du contexte statique au contexte changeant sont effectuées dans le paragraphe suivant.

### 5.3.3 Résolution du PDPT – du contexte statique au contexte changeant

La figure 5.9 représente de manière simplifiée les différences qui existent entre une méthode de résolution dédiée à un contexte statique et une méthode de résolution spécifique à un contexte changeant.

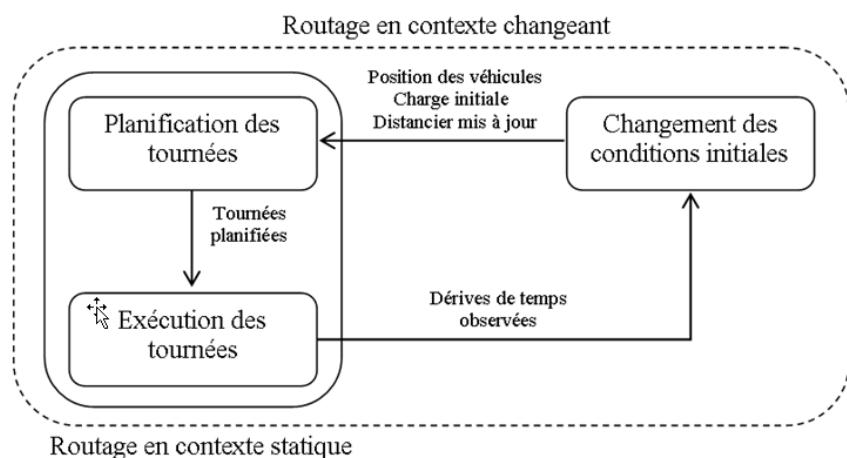


FIGURE 5.9 – *Synthèse de la méthode de résolution*

La méthode de résolution dans un contexte changeant peut se décomposer en deux étapes. La première étape consiste à analyser le contexte de l'événement ayant perturbé la solution issue de la planification. Cette analyse a pour but de déterminer si les conditions initiales du problème ou les hypothèses de départ sur lesquelles était bâtie la solution perturbée ont changé. De cette analyse des données telles que le distancier, la position courante des véhicules, la charge des véhicules, le nombre de clients à servir peuvent potentiellement être mis à jour. La deuxième étape consiste en la résolution du problème de transport sur la base de nouvelles entrées ou avec les anciennes entrées du module de routage mises à jour.

Les changements d'hypothèses évoqués ci-dessus amènent de nouvelles contraintes qu'il convient d'analyser.

#### 5.3.3.1 Analyse des contraintes liées au contexte changeant

Ces contraintes concernent la prise en compte d'une position quelconque des véhicules et la prise en compte d'une charge initiale dans les véhicules.

### 5.3.3.1.1 Prise en compte de la position quelconque d'un véhicule

Dans le problème de collecte et livraison analysé dans ce chapitre, l'hypothèse que chaque tournée de véhicules issue de la phase de planification peut être reconfigurée a été posée. Une reconfiguration suppose que les tournées aient déjà débuté, si bien que chaque véhicule a une position propre et connue. Le problème à résoudre peut être alors assimilé à un problème où chaque véhicule part d'une position propre. À des fins de généralisation, les véhicules dans le nouveau problème traité, n'auront également plus l'obligation de rentrer au dépôt initial, mais pourront terminer leur tournée en un lieu quelconque défini par le gestionnaire de flotte. Cette hypothèse présente plusieurs avantages :

- D'une part, le cas d'un départ et d'un retour à un même lieu pour tous les véhicules tels que abordé jusqu'à présent n'est qu'un cas particulier de la situation générale que nous considérons ici.
- D'autre part, dans un contexte changeant, un aléa peut survenir à n'importe quel moment et retarder un véhicule. Ainsi, au cas où une reconfiguration est décidée, cette position sera considérée comme le nouveau point de départ du véhicule et peut être déterminée par géolocalisation dans un contexte réel ou calculée de manière théorique par la méthode présentée en annexe A.5.

### 5.3.3.1.2 Prise en compte d'une charge initiale dans les véhicules

Dans l'approche statique, nous avons fait l'hypothèse que chaque véhicule part du dépôt avec une charge initiale nulle pour assurer un ensemble de services de collecte et livraison associés à un ensemble de requêtes de transport. Si cette hypothèse est suffisante pour résoudre le PDPT statique, elle ne l'est plus dans un contexte dans lequel les tournées planifiées peuvent changer ou évoluer. En effet, le processus de distribution associé à un véhicule peut être perturbé, alors que le véhicule se trouve à une position quelconque et qu'il a réalisé partiellement sa tournée. Dans un tel contexte, les ordres ou requêtes de transport prises en compte lors de la planification prévisionnelle des tournées sont à classer en trois catégories, afin de caractériser leur état d'avancement et d'identifier le problème de reconfiguration. Nous distinguons les cas suivants :

- Les points de collecte et de livraison associés à une même requête ont déjà été servis par un véhicule. Dans ce cas, la requête est considérée comme entièrement satisfaite et est éliminée du problème en cours de résolution, si bien que le problème de reconfiguration de tournées comportera un nombre moins important d'ordres de transport à traiter.
- Seul le point de collecte de la requête a été visité par le véhicule. Dans ce cas, l'ordre de transport correspondant est en cours de traitement et la charge collectée se trouve dans le véhicule. Il faut donc considérer, dans la manière de caractériser les conditions initiales du problème, l'ensemble des charges associées aux points de collecte déjà visités comme étant dans le véhicule. Les choix techniques de modélisation et résolution du problème visant à prendre en compte les charges initiales seront discutés ultérieurement.

- Le point de collecte et le point de livraison d'une même requête n'ont pas encore été visités. Ces requêtes sont donc à satisfaire, et sont conformes au problème traité dans le cas « statique » si bien qu'il ne sera pas nécessaire d'adapter la méthode de résolution pour la prise en compte de ces dernières.

Dans un contexte de tournées reconfigurables, la méthode utilisée pour effectuer une reconfiguration doit pouvoir trouver une solution optimisée dans un temps de calcul relativement court. En effet, si ce temps est très important, la solution trouvée dans un contexte d'application réelle peut s'avérer inapplicable compte tenu du délai entre l'instant de la prise de décision et celui de son application. L'élaboration de la méthode de résolution doit intégrer cette nouvelle contrainte. Notre choix s'est porté comme annoncé plus haut, sur un enrichissement de l'heuristique développée dans le contexte statique, compte tenu de son efficacité, prouvée par la série de tests effectués, et de l'observation de temps de calcul raisonnables.

## 5.4 ÉVOLUTION DE LA MÉTHODE DE RÉSOLUTION

L'évolution de la méthode de résolution consiste à intégrer dans celle-ci la gestion de nouvelles contraintes précédemment introduites dans le cadre du contexte changeant, tout en garantissant la recherche d'une solution de bonne qualité ou satisfaisante.

### 5.4.1 Prise en compte d'une position quelconque des véhicules

Dans un contexte changeant, tout événement perturbateur susceptible de déclencher une reconfiguration nécessite de prendre en compte la position courante des véhicules. En effet, cette position servira de point de départ pour les nouvelles tournées générées. Le problème engendré par cette nouvelle hypothèse est abordé ci-dessous.

*Problème* : Comment sont initialisées les routes, dans lesquelles les requêtes restantes à satisfaire seront ultérieurement insérées ?

*Solution* : Une telle initialisation a lieu à un instant donné appelé instant de reconfiguration. Il s'agit de l'instant auquel la reconfiguration des tournées est déclenchée. L'algorithme 7 d'initialisation consiste à faire un tri parmi les tournées qui constituent la solution en cours d'exécution. Les tournées achevées ne sont plus considérées. Pour les tournées restantes, la position courante du véhicule est considérée comme le point de départ du véhicule, et le dépôt de retour reste inchangé. Les différentes étapes nécessaires à cette initialisation sont présentées dans l'algorithme « InitialisationRoute ». Pour rappel, la notation  $dt_i^k$  utilisée dans l'algorithme représente la date d'arrivée planifiée du véhicule  $k$  au point  $i$ .

---

**Algorithme 7 : InitialisationRoute**

---

**Entrées :** Solution à reconfigurer –  $S$ , Instant de reconfiguration –  $idr$

**Sorties :** Solution –  $S_1$

```

1  $S_1 \leftarrow \emptyset$ 
2 pour chaque route  $r_k$  dans  $S$  faire
3    $o_1 \leftarrow$  Position courante du véhicule  $k$ ;
4    $o_2 \leftarrow$  Dépôt de retour du véhicule  $k$ ;
5   si  $idr < dt_{o_2}^k$  alors
6      $r_k = \langle o_1, o_2 \rangle$ ;
7      $S_1 \leftarrow S_1 \cup \{r_k\}$ ;
8   sinon
9     Le véhicule  $k$  a fini sa tournée;
10 Retourne  $S_1$ ;
```

---

À l'issue de l'exécution de l'algorithme 7, les véhicules n'ayant pas achevés leur tournée, se trouvent à des positions données qui seront considérées comme leur point de départ respectif dans le cadre d'une reconfiguration des tournées. Les tournées associées à ces véhicules sont utilisées pour initialiser une solution initiale contenant des routes vides i.e., point de départ et dépôt de retour. Notons que la considération de ces nouvelles positions dans le cadre d'une reconfiguration, nécessite obligatoirement la mise à jour du distancier initialement utilisé pour générer les tournées initiales.

#### 5.4.1.1 Mise à jour du distancier

La prise en compte d'une reconfiguration de tournées correspond à une modification du graphe modélisant le problème de transport. Cette évolution se traduit par l'ajout de nouveaux sommets dans le graphe, correspondant à la position courante des véhicules, et à la considération de nouveaux arcs décrivant les alternatives de trajets vers les différents clients à visiter depuis ces nouveaux sommets ; il faut donc renseigner la longueur de chaque nouvel arc, assimilée à la distance parcourue ou un temps de parcours. Ces données, sont essentielles pour l'exploitation de tout algorithme de routage de véhicules.

Pour fixer les idées, considérons le distancier fourni dans le tableau 5.2 et qui correspond au graphe de gauche de la figure 5.7.

Sommet de départ	Sommet d'arrivée	Distance
0	1	65
0	4	30
1	2	30
2	3	22
3	0	30
4	5	25
5	6	45
6	7	20
7	0	10

TABLE 5.2 – Distancier du problème initial

Dans le cadre d'une reconfiguration, une mise à jour du distancier précédent est proposée dans le tableau 5.3. Ce distancier correspond au graphe de droite de la figure 5.8. Dans le distancier proposé, seuls les nouveaux arcs sont représentés. Les arcs non représentés sont identiques à ceux présentés dans le tableau 5.2.

Sommet de départ	Sommet d'arrivée	Distance
0	8	55
8	10	20
4	9	15
9	10	20
10	5	20
5	1	50

TABLE 5.3 – Une partie du distancier du problème reconfiguré

#### 5.4.2 Prise en compte d'une charge initiale

Dans un contexte changeant, au moment où survient un aléa, conduisant à la perturbation d'une tournée initialement planifiée, le véhicule se trouve dans une position donnée. Comme détaillée dans la partie précédente cette position constitue le point de départ de la tournée à reconfigurer pour le véhicule concerné. Toutefois, contrairement au contexte statique où les véhicules partaient à vide de leur dépôt de départ, à cette position, le véhicule peut être chargé. Cela signifie que pour certaines requêtes, le point de collecte peut déjà avoir été visité. Il convient donc d'identifier de manière précise le statut de chaque requête de transport. Cette identification peut être formalisée par le problème suivant :

*Problème* : Comment identifier les requêtes de transport à satisfaire et distinguer celles qui sont en cours de traitement des autres requêtes déjà satisfaites ?

*Solution* : Pour montrer les différentes étapes qui permettent la mise en application de cette nécessité, nous proposons l'algorithme « PriseEnCompteChargeInitiale ». Pour rappel, une requête de transport notée  $R_i = (i, i + n)$  consiste à collecter des produits au sommet  $i$ , pour les livrer au sommet  $i+n$ . Dans l'algorithme qui suit, contrairement à  $dt_i^k$  qui représente la date d'arrivée planifiée du véhicule  $k$  au point  $i$ ,  $dtE_i^k$  représente la date d'arrivée effective du véhicule  $k$  au point  $i$ .

---

**Algorithme 8 : PriseEnCompteChargeInitiale**

---

**Entrées :** Solution à reconfigurer –  $S$ , Instant de reconfiguration –  $idr$

**Sorties :** Ensemble des requêtes à reconSIDéRer –  $RE$

```

1  $RE \leftarrow \emptyset$ 
2 pour chaque route  $r_k$  dans  $S$  faire
3    $S_k \leftarrow$  Ensemble des requêtes satisfaites dans  $r_k$ ;
4    $c_k \leftarrow$  Position courante du véhicule  $k$ ;
5   pour chaque Requête  $R_i = (i, i + n)$  dans  $S_k$  faire
6     si  $idr > dtE_{i+n}^k$  alors
7       supprimer la requête  $R_i = (i, i + n)$ ;
8     sinon si  $idr < dtE_i^k$  et  $idr < dtE_{i+n}^k$  alors
9        $RE \leftarrow RE \cup R_i$ ;
10    sinon si  $idr > dtE_i^k$  et  $idr < dtE_{i+n}^k$  alors
11      Créer la requête  $R_{c_k} = (c_k, i + n)$ ;
12       $RE \leftarrow RE \cup R_{c_k}$ ;
13 Retourne  $RE$ ;
```

---

L'algorithme « PriseEnCompteChargeInitiale » a une entrée qui correspond à la solution à reconfigurer. La sortie de l'algorithme est l'ensemble des requêtes ou ordre de transport restant à exécuter dans le cadre de la reconfiguration.

Dans chaque route  $r_k$  constituant la solution courante, et pour chaque requête  $R_i = (i, i + n)$  initialement affectée à cette route, les situations suivantes sont traitées :

- La requête est complètement satisfait, si bien qu'elle n'est plus à considérer (ligne 6 – 7).
- La requête est non exécutée, et doit être servie par le véhicule considéré (ligne 8 – 9).
- La requête est en cours de traitement, si bien que la charge a été collectée mais non livrée à son destinataire (ligne 10 – 12).

La prise en compte des contraintes précédentes permet de disposer d'un ensemble de tournées initialisées dans lesquelles seront insérées les requêtes à satisfaire. Les méthodes algorithmiques proposées au chapitre 4 sont alors enrichies pour intégrer les raisonnements inhérents à la résolution du PDPT dans un contexte changeant.

#### 5.4.3 Adaptation de la méthode de résolution

L'algorithme général pour résoudre le problème de collecte et livraison avec transbordement dans un contexte changeant est identique sur le principe de résolution au VNS présenté au chapitre 4 et appelé « VNS-PourLePDPT ». La principale différence concerne la génération de la solution initiale présentée dans l'algorithme « SolutionInitialeDansLecontexte-Changeant » ci-dessous.

---

**Algorithme 9 : SolutionInitialeDansLecontexteChangeant**

---

**Entrées :** Solution à reconfigurer –  $S$ , Instant de reconfiguration –  $idr$

**Sorties :** Solution initiale –  $S_1$

- 1  $S_1 \leftarrow InitialisationRoute(S, idr);$
  - 2  $R \leftarrow PriseEnCompteChargeInitiale(S, idr);$
  - 3  $S_1 \leftarrow Parallèle(S_1, R) ;$
  - 4 Retourne  $S_1;$
- 

L'algorithme précédent est structuré en trois étapes qui synthétisent les différentes opérations qui permettent de construire une solution initiale afin de résoudre le PDPT dans un contexte changeant. Cette solution peut être ensuite optimisée par l'algorithme « VNSPourLePDPT » déjà présentée au chapitre 4. L'étape 1 consiste en l'initialisation d'un ensemble de routes dans lesquelles les requêtes restantes à faire doivent être insérées. L'ensemble des dites requêtes est déterminé à l'étape 2. La méthode permettant d'insérer ces requêtes dans les routes précédemment initialisées est l'algorithme parallèle appelé à la ligne deux et identique sur le principe à l'algorithme de même nom présenté et analysé au chapitre 4.

Préalablement à l'intégration de la solution proposée dans le module de routage de PRODIGE, celle-ci doit d'abord être validée. La validation consiste à analyser le comportement de chacun des algorithmes considérés dans le cadre du contexte changeant.

## 5.5 VALIDATION EXPÉRIMENTALE

La démarche de validation expérimentale des algorithmes proposés se déroule en deux étapes. Une première étape consiste à les valider de manière théorique afin de vérifier les comportements attendus des algorithmes implémentés avant de les intégrer dans le système d'information de PRODIGE pour une expérimentation pratique ou réelle.

### 5.5.1 Expérimentation théorique

À notre niveau de connaissance de la littérature, il n'existe pas d'instances théoriques ou réelles permettant de valider la méthode de résolution proposée. Ne disposant pas de telles instances et ne disposant pas non plus d'un simulateur permettant des comparaisons expérimentales, la validation de cette méthode se fera dans un premier temps sur la base d'instances théoriques définies dans un format conforme à celui utilisé au chapitre 4. Pour ces instances, un ensemble de scénarios sont définis de manière à connaître, a priori, les solutions optimales et mettre en évidence l'impact du comportement à tester. Ces instances sont ensuite soumises à l'algorithme de reconfiguration et les résultats retournés sont comparés avec ceux attendus. Les scénarios testés portent sur les contraintes apportées par le contexte changeant. Les comportements à valider sont les suivants :

- La prise en compte d'une position quelconque des véhicules

- L'impact de la reconfiguration des tournées.

### 5.5.1.1 Validation d'une tournée à origine et destination quelconque

La première expérimentation a pour objectif de juger de la capacité de la méthode proposée à résoudre un problème de transport dans lequel les lieux de départ et d'arrivée des véhicules ne sont pas tous identiques. Nous proposons un exemple volontairement simple ne comportant que deux requêtes. Ce problème est illustré par le graphe représenté dans la figure 5.10 ci-après. Les points du graphe ont respectivement pour coordonnées : 1(−100; 100), 2(100; 100), 3(100; −100), 4(−100; −100), 5(0; 150), 6(0; 200) et 7(0; −200). Dans le graphe suivant tous les chemins non représentés sont supposés inexistantes.

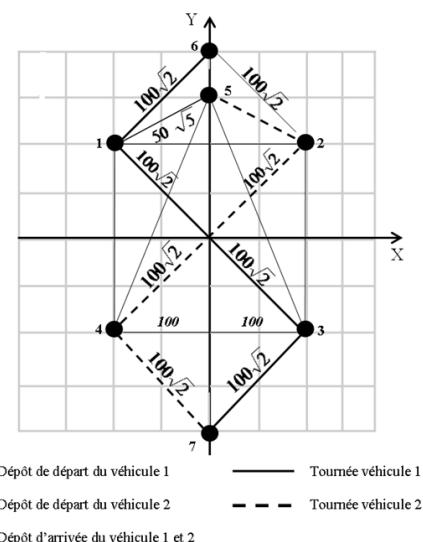


FIGURE 5.10 – Exemple de tournée avec dépôt de départ et d'arrivée différents pour deux véhicules

Le problème à résoudre dans cet exemple consiste à satisfaire les requêtes  $R_1 = (1, 3)$  et  $R_2 = (2, 4)$ , par l'exploitation de 2 véhicules. Le premier véhicule est initialement localisé au point 6 et le second au point 5. Les deux véhicules doivent terminer leur tournée respective au point 7. Les fenêtres de temps en tout point du graphe sont de  $[0, 1200]$ . Les fenêtres de temps sont prises volontairement larges pour simplifier la résolution. La durée de service en chaque noeud visité est nulle. La quantité de produits à transporter pour satisfaire chaque requête est de 100. La vitesse et la capacité de chaque véhicule sont respectivement de 1 et 150.

L'exécution de l'algorithme « VNSPourLePDPT » adapté pour le contexte changeant permet de trouver une solution à deux véhicules pour satisfaire l'ensemble des requêtes, pour une distance totale de  $700\sqrt{2} + 50\sqrt{5}$ . Cette solution est représentée dans le tableau 5.4. La colonne « charge » représente la charge du véhicule au départ du noeud visité.

Véhicule 1			
Sommet	Date d'arrivée	Date de départ	Charge après service
6	0	0	0
1	$100\sqrt{2}$	$100\sqrt{2}$	100
3	$300\sqrt{2}$	$300\sqrt{2}$	0
7	$400\sqrt{2}$	$400\sqrt{2}$	0
véhicule 2			
Sommet	Date d'arrivée	Date de départ	Charge après service
5	0	0	0
2	$50\sqrt{5}$	$50\sqrt{5}$	100
4	$200\sqrt{2} + 50\sqrt{5}$	$200\sqrt{2} + 50\sqrt{5}$	0
7	$300\sqrt{2} + 50\sqrt{5}$	$300\sqrt{2} + 50\sqrt{5}$	0

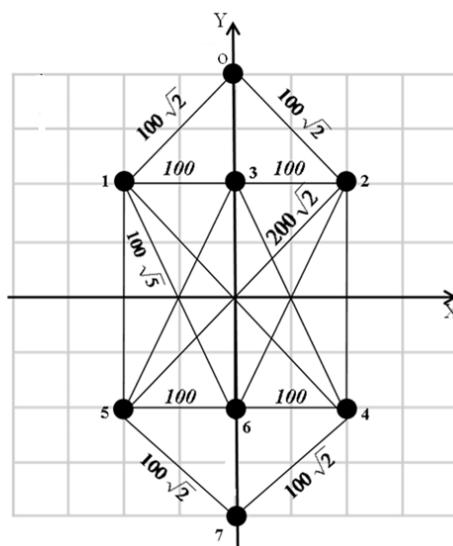
TABLE 5.4 – Tournées de véhicules avec dépôts différents

L'algorithme a donc trouvé dans cet exemple élémentaire, une solution conforme aux résultats attendus en intégrant les origines et destinations quelconques des tournées.

### 5.5.1.2 Validation du principe de reconfiguration

Le deuxième cas de validation a pour objectif de présenter un exemple de reconfiguration des tournées avec prise en charge d'une position quelconque des véhicules comme point de départ et d'une charge initiale.

Le problème de transport proposé comme exemple est illustré par le graphe de la figure 5.11. Les points de ce graphe ont respectivement pour coordonnées : o(0;200), 1(-100;100), 2(100;100), 3(0;100), 4(100;-100), 5(-100;-100), 6(0;-100), 7(0;-200).



0 Dépôt de départ des véhicules 1 et 2

7 Dépôt d'arrivée des véhicules 1 et 2

FIGURE 5.11 – Représentation des clients

Le problème à résoudre consiste à saisir trois requêtes de transport respectivement représentées par les couples de clients (1,4), (2,5) et (3,6)

par une flotte homogène de trois véhicules de capacité infinie et de vitesse constante égale à l'unité. Chaque véhicule utilisé pour satisfaire les requêtes débute sa tournée en 0 et la termine en 7. Pour satisfaire une requête, 20 produits doivent être chargés et déplacés du point de collecte vers le point de livraison en respectant les contraintes de fenêtres de temps associées à chaque client. La fenêtre de temps des clients 0, 7, 1, 2, 3, 4, 5 est de [0, 800]. Elle est de [0, 450] pour le client 6 afin de permettre de mettre plus facilement en évidence l'impact d'un aléa sur le non respect d'une contrainte de fenêtre de temps. La durée de service en tous les points est supposée nulle, afin de faciliter le calcul.

Le tableau 5.5 synthétise la solution fournie à l'issue de l'étape de planification pour satisfaire l'ensemble des trois requêtes. La date d'arrivée en chaque sommet visité (client) est notifiée. La colonne « charge » représente la charge du véhicule à son départ du nœud visité. La distance totale parcourue dans cette solution pour satisfaire l'ensemble des clients est de 1230,6 unités de distance.

Véhicule 1			
Sommet	Date d'arrivée	Fenêtre de temps	Charge après service
0	0	[0, 800]	0
3	100	[0, 800]	20
1	200	[0, 800]	40
6	$200 + 100\sqrt{5} \approx 424$	[0, 450]	20
4	$300 + 100\sqrt{5} \approx 524$	[0, 800]	0
7	$300 + 100\sqrt{5} + 100\sqrt{2} \approx 665$	[0, 800]	0
Véhicule 2			
Sommet	Date d'arrivée	Fenêtre de temps	Charge après service
0	0	[0, 800]	0
2	$100\sqrt{2} \approx 141$	[0, 800]	100
5	$300\sqrt{2} \approx 424$	[0, 800]	0
7	$400\sqrt{2} \approx 566$	[0, 800]	0

TABLE 5.5 – Récapitulatif des tournées prévisionnelles des véhicules

Cette solution peut être visualisée dans la figure 5.12-1. Le premier véhicule effectue le trajet représenté en trait interrompu. Le second effectue le trajet en trait plein.

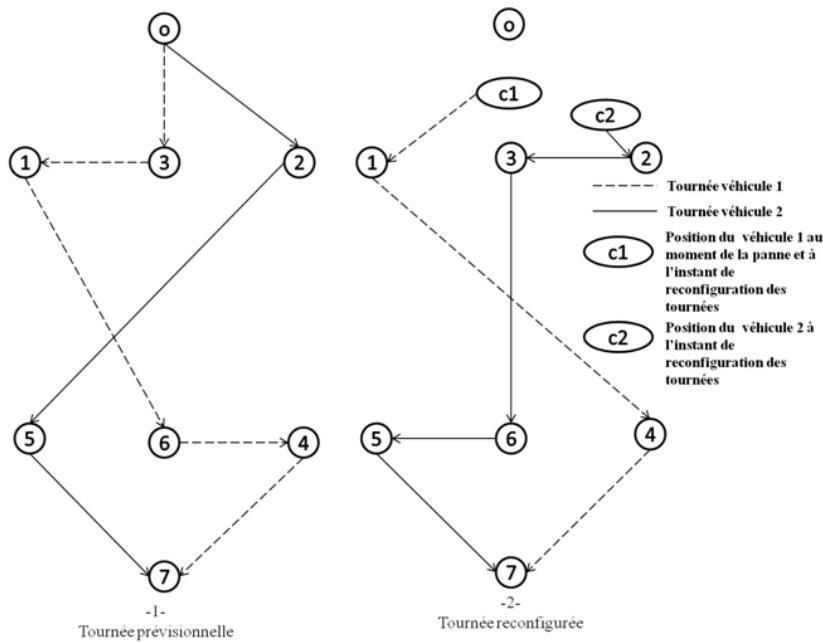


FIGURE 5.12 – Représentation des tournées

Durant l'exécution de ces deux tournées, à la date 50, le véhicule 1 est l'objet d'un incident, par exemple une panne, alors qu'il se trouve à la position c1 sur la figure 5.12-2. La réparation de cette panne nécessite 50 unités de temps. Le véhicule 2 n'ayant subi aucun incident, sa tournée prévisionnelle n'est pas contrariée et se déroule comme prévu. Le véhicule 1 repart de c1 à la date 100 et le temps de retard se propage au reste de sa tournée telle que décrite dans le tableau 5.6. Dans le tableau 5.6, NS signifie non significatif.

Véhicule 1			
Sommet	Date d'arrivée	Fenêtre de temps	Charge après service
c1	100	NS	0
3	150	[0, 800]	20
1	250	[0, 800]	40
6	250 + 100 $\sqrt{5}$ ≈ 474	[0, 450]	20
4	350 + 100 $\sqrt{5}$ ≈ 574	[0, 800]	0
7	350 + 100 $\sqrt{5}$ + 100 $\sqrt{2}$ ≈ 715	[0, 800]	0
Véhicule 2			
o	0	[0, 800]	0
2	100 $\sqrt{2}$ ≈ 141	[0, 800]	100
5	300 $\sqrt{2}$ ≈ 424	[0, 800]	0
7	400 $\sqrt{2}$ ≈ 566	[0, 800]	0

TABLE 5.6 – Propagation du retard dans la tournée du véhicule 1

L'analyse du tableau 5.6 permet de constater que la tournée du véhicule 1 ne peut être réalisée du fait de la violation de la contrainte de fenêtre de temps au sommet 6. Il importe donc d'effectuer une mise à jour ou une reconfiguration de cette tournée afin de la rendre réalisable. Pour réaliser cette reconfiguration, il est nécessaire de déterminer la position courante des deux véhicules. En effet, comme expliqué plus haut dans le chapitre, les véhicules débuteront leur tournée respective à partir de ces positions. Dans un contexte réel, de telles positions sont aisément disponibles du fait de la géolocalisation. Dans un contexte théorique, il est nécessaire de disposer d'une méthode permettant de les simuler. La méthode utilisée pour calculer les dites positions simulées est présentée dans l'annexe A.5 du document.

Dans l'exemple traité, la mise à jour des positions des véhicules et des ordres de transport est effectuée comme suit :

- L'instant de reconfiguration des tournées est déterminé à 100 unités de temps. Il correspond à la date où les deux véhicules sont disponibles. En effet, à cette date, le véhicule 1 a été réparé.
- La position courante de chaque véhicule doit être calculée. Nous appellerons  $c_1$  cette position pour le véhicule 1 (voir figure 5.12-2). Il en est de même du point  $c_2$  pour le véhicule 2 sur la même figure. Le point  $c_1$ , respectivement  $c_2$ , est assimilé au point de départ du véhicule 1, respectivement du véhicule 2 dans la nouvelle solution à calculer. En appliquant les formules de calcul de la position courante présentées en annexe A.5, nous obtenons les coordonnées suivantes  $c_1 : [0 + 50 * \sin(180^\circ); 200 + 50 * \cos(180^\circ)] = [0; 150]$  et  $c_2 : [0 + 100 * \cos(45^\circ); 200 - 100 * \sin(45^\circ)] = [70,76; 129,28]$ .
- Aucun produit n'ayant encore été collecté à l'arrivée de l'aléa, les ordres initiaux sont conservés et correspondent à (1,4), (2,5) et (3,6).

Rappelons que la prise en compte des points  $c_1$  et  $c_2$  entraîne une mise à jour du distancier associé au problème. Ce distancier est facilement déductible de la figure 5.12-2.

Avec ces nouvelles informations, le problème de transport est de nouveau résolu avec la méthode de résolution proposée dans ce chapitre. La solution issue de cette reconfiguration est décrite dans le tableau 5.7.

Véhicule 1			
Sommet	Date d'arrivée	Fenêtre de temps	Charge après service
$c_1$	100	NS	0
1	$100 + 50\sqrt{5} \approx 212$	[0, 800]	20
4	$100 + 50\sqrt{5} + 200\sqrt{2} \approx 495$	[0, 800]	0
$o_2$	$100 + 50\sqrt{5} + 300\sqrt{2} \approx 636$	[0, 800]	0
Véhicule 2			
$c_2$	100	NS	0
2	$100 + 29\sqrt{2} \approx 141$	[0, 800]	20
3	$200 + 29\sqrt{2} \approx 241$	[0, 800]	40
6	$400 + 29\sqrt{2} \approx 541$	[0, 450]	20
5	$500 + 29\sqrt{2} \approx 541$	[0, 800]	0
7	$500 + 129\sqrt{2} \approx 682$	[0, 800]	0

TABLE 5.7 – *Effet de la reconfiguration sur une tournée*

Le tableau 5.7 permet d'observer l'intérêt de la reconfiguration sur une tournée suite à l'occurrence d'un aléa. Il peut aussi être observé que, la reconfiguration apporte un coût supplémentaire notamment en temps de service. En effet, initialement dans le tableau 5.5, le véhicule 2 achève sa tournée à la date  $400\sqrt{2}$ . La reconfiguration entraîne un décalage de cette date qui passe de  $400\sqrt{2}$  à  $500 + 129\sqrt{2}$  tel que donné dans le tableau 5.7. Toutefois, elle permet de satisfaire l'ensemble des requêtes, ce qui était compromis du fait de la perturbation.

Les algorithmes qui viennent d'être validés permettent de dérouler des tests grandeur nature, pour vérifier le bon comportement des solutions de routage telles que définies dans ce travail.

### 5.5.2 Expérimentation en situation réelle

La démarche suivie pour exécuter les tests grandeur nature dans le cadre de PRODIGE a été la suivante :

- Un ensemble de scénarios a été défini manuellement, en fonction des comportements à observer. Le choix des clients à visiter parmi un ensemble de lieux préalablement définis et le choix des fenêtres de temps ont été réalisés de manière à favoriser ces comportements. Ainsi, la solution optimale correspondant à chaque scénario est connue par avance. Il suffit alors de suivre en temps réel l'exécution des tournées générées par le module de routage de PRODIGE, pour les comparer avec les solutions attendues et juger de la validité de l'algorithme.
- Les données caractérisant les problèmes à traiter ont été saisies dans la base de données de PRODIGE via le module portail web. Cette base de donnée est appelée « Données logistiques clientèles ». Elles concernent les ordres de transport et les véhicules.
- De la base de donnée « Données logistiques clientèles », un distancier est généré et fournit en entrée du module de routage afin d'être utilisé par les algorithmes d'optimisation.

- Les scénarios ont été effectués au moyen de deux véhicules. Afin de générer des retards dans les tournées, les véhicules sont volontairement arrêtés pour une durée donnée, à l'instant souhaité en un lieu défini par avance. La durée, le lieu et l'instant d'arrêt ont été calculés par avance de manière à valider la solution proposée par PRODIGE.

La démarche de validation ayant été présentée, nous pouvons rentrer dans le détail des expérimentations effectuées.

### 5.5.2.1 Mise en œuvre réelle de la solution PRODIGE

Les algorithmes développés étant intégrés dans la solution logicielle de PRODIGE, les scénarios d'expérimentations réels sont présentés. Les données présentées dans ce qui suit correspondent aux résultats donnés par la solution PRODIGE en situation réelle d'exploitation d'une flotte de deux véhicules de grande capacité chacun, si bien que le cumul de charges à transporter reste très inférieur à la capacité de chaque véhicule. Deux scénarios sont retranscrits ci-dessous.

#### 5.5.2.1.1 Éléments de caractérisation des scénarios

Les tests grandeur nature se sont déroulés dans la périphérie de Bordeaux, dans un périmètre géographique restreint de manière à ne pas avoir des tournées trop longues en temps et en distance. Ainsi, nous avons considéré un ensemble de 12 lieux correspondants à des clients, en plus du dépôt appelé entrepôt GLS où se situent les véhicules au départ des tournées. Ces lieux sont caractérisés par des coordonnées géographiques (longitude, latitude) pouvant être « injectées » dans le module de géolocalisation et navigation afin de permettre le calcul du distancier. Le tableau 5.8 synthétise ces informations.

ID	Longitude	Latitude	Ville	Nom client
1	-0.582836	44.7079	MARTILLAC	Entrepôt GLS
49	-0.53097	44.7224	MARTILLAC	LIDL Martillac Parking
50	-0.517774	44.6864	LA BREDE	Simply Market La brède Parking
51	-0.596765	44.6589	SAUCATS	Intermarché Saucats Parking
52	-0.580991	44.7128	MARTILLAC	Technopôle Montesquieu Parking
53	-0.598044	44.7272	LEOGNAN	Stade de Léognan
54	-0.621504	44.7235	LEOGNAN	Lac Bleu
55	-0.504749	44.7159	ST. MEDARD D.	Gare de Saint-Médard-d'Eyrans
56	-0.451276	44.7014	BEAUTIRAN	Gare de Beautiran
57	-0.543121	44.7138	MARTILLAC	Centre-ville de Martillac
58	-0.552371	44.7146	MARTILLAC	La breyra – Martillac
59	-0.606501	44.7134	LEOGNAN	Château Fieuzal Parking
60	-0.601963	44.6976	LEOGNAN	Aérodrome Saucats Parking

TABLE 5.8 – Définition des clients potentiels

La figure 5.13 permet d'observer la distribution spatiale des lieux considérés. Les scénarios élaborés sur la base de ces données donneront lieu à des tournées qui n'excéderont pas deux heures de temps par véhicule.



FIGURE 5.13 – Distribution spatiale des clients

#### 5.5.2.1.2 Scénario 1 – Transbordement

Ce premier scénario concerne l'élaboration des tournées de deux véhicules en relation avec deux requêtes de transport. Les tournées respectives doivent débuter et s'achever au point 52. L'objectif est de montrer la capacité de la solution PRODIGE à trouver des solutions optimisées pour satisfaire une demande de transport en exploitant le transbordement. Afin de faciliter la lecture des données, nous présentons ces dernières dans les tableaux 5.9 et 5.10. Ainsi, le problème se caractérise par :

ID	Point Collecte	Point Livraison
300	54	55
301	51	49

TABLE 5.9 – Définition des requêtes

Lieux	Fenêtres de temps	
	Borne Min	Borne Max
49	14 : 33	14 : 58
51	14 : 08	14 : 33
52	13 : 54	23 : 00
54	14 : 08	14 : 33
55	14 : 33	15 : 06

TABLE 5.10 – Définition des fenêtres de temps

La solution PRODIGE met en œuvre deux véhicules effectuant un transbordement qui permet de satisfaire les deux requêtes. Pour rappel, l'algorithme retourne la meilleure solution trouvée qui correspond selon

les cas, soit à une solution avec transbordement comme c'est le cas dans cet exemple, soit à une solution sans transbordement. Les tournées respectives des deux véhicules sont représentées dans le tableau 5.11 constitué des colonnes suivantes :

- Le temps de trajet qui sépare deux points consécutifs dans une tournée. Par exemple, le temps de trajet entre le point 52 et le point 54 est de 901 secondes.
- Les fenêtres de temps associées à chaque lieu. Ces fenêtres sont exprimées par des « timestamp » mais traduites dans le tableau 5.11 dans le format heures : minutes pour faciliter la lecture.
- La date d'arrivée du véhicule chez chaque client.
- La date à laquelle le véhicule quitte le client (départ), considérant un temps de service chez le client de 5 minutes.

Lieux	Temps de trajet	Fenêtre de temps		Arrivée	Départ	
		Borne Min	Borne Max			
52					13 : 53	Tournée du véhicule 1
54	901	14 : 08	14 : 33	14 : 08	14 : 13	
52	901	13 : 54	23 : 00	14 : 28	14 : 33	
52	0	13 : 54	23 : 00	14 : 33		
52					13 : 54	Tournée du véhicule 2
51	818	14 : 08	14 : 33	14 : 08	14 : 13	
52	818	13 : 54	23 : 00	14 : 33	14 : 38	
49	611	14 : 33	14 : 58	14 : 48	14 : 53	
55	334	14 : 33	15 : 06	14 : 59	14 : 04	
52	898	13 : 54	23 : 00	15 : 19		

TABLE 5.11 – Tournées générées par le module de routage

Un transbordement s'effectue au point 52 de la façon suivante : un premier véhicule part de son dépôt (i.e. point 52) et se dirige au point 54 pour y charger un produit qu'il décharge ensuite au dépôt. Un second véhicule quitte le dépôt, se rend au point 51 pour collecter un produit. Du point 51, il se rend au dépôt pour charger le produit précédemment déposé par le premier véhicule. Il termine ensuite sa tournée en livrant les deux produits chargés respectivement au point 49 et au point 55. Notons que dans l'exemple proposé le transbordement s'effectue au dépôt. Nous sommes donc dans un problème similaire au cross-docking. Toutefois, la généralité de l'approche de résolution n'est pas affectée.

### 5.5.2.1.3 Scénario 2 – Reconfiguration

Le second scénario concerne l'élaboration de tournées en relation avec 4 requêtes de transport. Les données associées à ces requêtes sont données dans les tableaux 5.12 et 5.13. L'objectif de ce scénario est de montrer l'impact de la reconfiguration de tournées dans un contexte réel, notamment comme moyen curatif pour préserver la réalisabilité d'une solution suite à l'occurrence de perturbations. Tout comme dans le scénario précédent, nous présentons ces dernières sous la forme de tableaux synthétiques.

La solution PRODIGE nous permet d'obtenir la tournée initiale, telle

ID	Point Collecte	Point Livraison
302	53	55
303	60	57
304	58	51
305	50	54

TABLE 5.12 – Définition des requêtes

Lieux	Fenêtres de temps	
	Borne Min	Borne Max
1	09 : 00	13 : 00
50	11 : 02	11 : 15
51	11 : 40	12 : 14
53	10 : 08	10 : 13
54	11 : 45	12 : 30
55	10 : 30	12 : 00
57	10 : 30	11 : 00
58	10 : 30	10 : 50
60	10 : 15	10 : 25

TABLE 5.13 – Définition des fenêtres de temps

que représentée par le tableau 5.14. L'explication de ce tableau est identique à celui du tableau 5.11.

Lieux	Temps de trajet	Arrivée			Départ
		Borne Min	Borne Max	Heure	
1					10 : 00
53	433	10 : 08	10 : 13	10 : 08	10 : 13
60	289	10 : 15	10 : 25	10 : 17	10 : 22
58	683	10 : 30	10 : 50	10 : 34	10 : 39
57	148	10 : 30	11 : 00	10 : 41	10 : 44
55	440	10 : 30	12 : 00	10 : 54	10 : 59
50	676	11 : 02	11 : 15	11 : 10	11 : 15
54	1685	11 : 45	12 : 30	11 : 43	11 : 50
51	1096	11 : 40	12 : 14	12 : 08	12 : 13
1	914	09 : 00	13 : 00	12 : 28	

TABLE 5.14 – Tournée initiale sans retard

Après avoir visité le point 58, alors que le véhicule se déplace en direction du point 57, des ralentissements imprévus le font arriver à ce point avec un retard de 10 minutes par rapport à la date de visite planifiée. À partir du point 57, le retard constaté se propage au reste de la tournée tel que décrit dans le tableau 5.15.

Lieux	Temps de trajet	Arrivée			Départ
		Borne Min	Borne Max	Heure	
57	148	10 : 30	11 : 00	10 : 51	10 : 56
55	440	10 : 30	12 : 00	11 : 04	11 : 09
50	676	11 : 02	11 : 15	11 : 20	11 : 25
54	1685	11 : 45	12 : 30	11 : 53	11 : 58
51	1096	11 : 40	12 : 14	12 : 16	12 : 21
1	914	09 : 00	13 : 00	12 : 36	

TABLE 5.15 – *Tournée initiale impactée par le retard*

L'observation du tableau 5.15 permet de se rendre compte que sans aucune reconfiguration de la tournée en cours d'exécution, celle-ci ne peut se terminer sans violation de contraintes de fenêtre de temps. En effet, la fenêtre de temps des clients 50 et 51 n'est pas respectée rendant de fait la tournée irréalisable. Pour rendre la tournée réalisable et assurer un service de qualité, le décisionnaire décide d'une reconfiguration de la tournée à partir des points 57 et à l'instant 10 : 51. Cet instant correspond au temps d'arrivée du véhicule au point 51. La reconfiguration de la tournée issue de l'utilisation de l'algorithme, dont les principes ont été décrits plus tôt, conduit à une nouvelle séquence de visites décrite dans le tableau 5.16.

Lieux	Temps de trajet	Arrivée			Départ
		Borne Min	Borne Max	Heure	
57	148	10 : 30	11 : 00	10 : 51	10 : 56
50	669	11 : 02	11 : 15	11 : 07	11 : 12
55	676	10 : 30	12 : 00	11 : 24	11 : 29
51	1490	11 : 40	12 : 14	11 : 53	11 : 58
54	1096	11 : 45	12 : 30	12 : 17	12 : 22
1	1011	09 : 00	13 : 00	12 : 39	

TABLE 5.16 – *Reconfiguration de la tournée*

La nouvelle tournée issue de la reconfiguration respecte l'ensemble des contraintes de temps, mais a conduit à modifier la séquence initiale. En effet, les clients 50 et 55 ont été intervertis dans la séquence, ainsi que 51 et 54 afin de livrer prioritairement les clients ayant des contraintes de rendez-vous assez strictes. Notons toutefois que bien que tous les clients soient au final servi, la reconfiguration induit pour cet exemple une dégradation du temps de service du véhicule. En effet, le véhicule termine sa tournée à 12 : 39 : 02 au lieu de la terminer à 12 : 28 : 30

## 5.6 CONCLUSION

Le présent chapitre axé sur la résolution en contexte réel du problème de collecte et livraison avec reconfiguration peut être globalement synthétisé en deux parties.

Dans un premier temps, la présentation du projet PRODIGE a été effectuée. Les recherches menées au niveau de PRODIGE trouvent des applications dans plusieurs métiers de la chaîne logistique. La présente thèse étant axée sur le transport, des différents champs d'applications étudiés au sein de PRODIGE, seule la partie transport a été analysée de manière

détaillée. De cette analyse, il s'est avéré nécessaire de faire évoluer les méthodes de résolution développées au chapitre 4 afin de les adapter à un contexte de transport changeant et d'activités réelles.

Dans un second temps, la prise en compte de la dimension changeante de l'activité du transport a été effectuée et consiste en l'identification des nouvelles contraintes apportées par le contexte changeant, la mise à niveau des heuristiques existantes et la validation de celles-ci. Le caractère théorique du travail, et le manque d'instances clairement identifiées dans la littérature a nécessité l'utilisation d'instances théoriques afin de juger de la capacité de la solution développée à proposer des alternatives de reconfigurations des tournées satisfaisantes pour la décisionnaire. Une validation supplémentaire des méthodes développées dans cette thèse dans un contexte de transport réel via le projet PRODIGE a été effectuée. Les différents tests réalisés sur le réseau routier de la périphérie de Bordeaux, ont permis de montrer la validité des différentes approches et solutions développées. Il a notamment été observé que le transbordement couplé à la reconfiguration des tournées pouvaient permettre pour certaines instances de réduire le coût du transport mais aussi de mettre à jour les tournées afin de garantir au mieux, suite à l'occurrence d'événements perturbateurs, la satisfaction des requêtes de transport.



# CONCLUSION GÉNÉRALE

**L**e projet PRODIGE est un projet de recherche à vocation industrielle couvrant des cibles à la fois technique et scientifique. Sur le plan scientifique, le projet vise « la recherche d'innovations organisationnelles tendant à optimiser les flux et favorisant la logistique collaborative par une meilleure synchronisation des activités menées par les différents acteurs de la chaîne logistique ». PRODIGE fournit donc un cadre d'étude et une structure informationnelle basée sur une instrumentation de l'activité transport qui repose sur l'utilisation des nouvelles technologies de l'information et de la communication (NTIC). Cette thèse qui s'inscrit donc dans ce contexte vise à contribuer au développement d'algorithmes et de modèles d'optimisation pour l'élaboration de tournées de véhicules dans le cadre du transport routier de marchandises et plus spécifiquement pour les problèmes de collectes et de livraisons avec transbordement.

Ainsi, la contribution principale de ce travail a été de concevoir, spécifier et développer un système d'aide à la décision qui puisse assister un gestionnaire de flotte de véhicules ou un décisionnaire du transport dans sa prise de décision. La prise de décision consiste en la génération d'un ensemble de tournées de véhicules permettant d'assurer la satisfaction d'un ensemble de requêtes de transport dans un problème de collectes et livraisons avec transbordement et fenêtres de temps. Pour rappel, le transbordement est utilisé en un ou plusieurs lieux spécifiques dits points de transbordement, comme levier d'optimisation dans la génération des tournées et n'est mis en œuvre que pour la réduction effective du coût du transport. En outre, le système d'aide à la décision doit pouvoir générer les tournées dans un contexte de transport changeant, l'activité étant soumis à de nombreuses perturbations qui induisent des retards de livraison dans les tournées et nécessitent des reconfigurations de ces tournées. Pour présenter cette contribution, la thèse a été organisée en cinq chapitres.

Le chapitre 1 a permis de mettre en évidence l'importance du transport dans la chaîne logistique. Il a été aussi l'occasion d'énoncer un référentiel d'indicateurs orienté transport routier, défini et validé par interviews lors du projet PRODIGE, afin d'en extraire les mesures de performances les plus pertinentes pour l'optimisation de tournées. L'état de l'art réalisé dans le chapitre 2 s'est focalisé sur l'analyse et la classification des principaux travaux réalisés pour la modélisation et la résolution des problèmes de collectes et livraisons. Cette analyse a montré que le principe de transbordement n'était que très faiblement étudié dans cette classe de problèmes de routage, ce qui nous a amené à centrer notre recherche sur le sujet, et la classification formulée a servi de support à la caractérisation plus fine du problème étudié et de la méthode de résolution choisie. La méthodologie suivie pour caractériser et résoudre les problèmes de

collecte et livraison avec fenêtres de temps et transbordement s'est alors articulé en trois phases :

- La première phase, présentée dans le chapitre 3, a consisté à modéliser le problème au travers d'un programme mathématique linéaire, sur la base d'un certain nombre d'hypothèses :
  - les véhicules sont homogènes, i.e. de même capacité, et en nombre infini ;
  - les activités de collecte et livraison de produits sont soumises à une prise de rendez-vous chez le client, caractérisée par des fenêtres de temps au sein desquelles ces activités doivent être strictement engagées ;
  - l'acheminement d'un produit de son fournisseur à son client peut requérir l'usage de plus d'un véhicule, lors de transbordements pouvant s'opérer en un point spécifique. Par ailleurs, ce point peut être visité par plus d'un véhicule ;
  - La fonction objectif orientant la recherche de solutions optimales consiste en une minimisation du nombre de véhicules et de la distance parcourue, sans intégrer de pénalités en cas de retard sur rendez-vous, puisque les fenêtres de temps sont à respecter strictement.
- L'incapacité d'une méthode exacte à résoudre un problème de collectes et livraisons avec transbordement de taille réaliste dans un temps de calcul raisonnable, nous a conduit à développer une méthode de résolution approchée. Ainsi, le chapitre 4 fait l'objet de la présentation d'une métaheuristique s'appuyant sur le principe de recherche à voisinages variables (VNS – Variable Neighborhood Search) adaptée à la résolution des problèmes de transbordement. Un tel développement a permis de lever entre autre l'hypothèse de l'unicité du point de transbordement, pour considérer qu'un transbordement pouvait s'opérer en plusieurs lieux spécifiques ou auprès des clients visités dans une tournée. La validation de la méthode s'est faite en deux temps :
  - La première étape de validation s'est concentrée sur une résolution de problème de collectes et livraisons sans transbordement, à partir d'instances publiées dans la littérature. Afin de vérifier la qualité de la méthode proposée, nous avons ainsi comparé les résultats obtenus par rapport aux meilleurs résultats connus pour les instances choisies. L'efficacité de la méthode développée dans le présent travail a pu être ainsi mise en exergue : la majorité des résultats obtenus à l'aide de notre approche de résolution correspondent aux meilleures solutions publiées dans la littérature.
  - La deuxième étape de validation s'est appuyée d'une part sur l'ensemble des instances définies dans le chapitre 3, et d'autre part sur les rares instances issues de la littérature scientifique. Dans les deux cas, notre approche de résolution a fourni des solutions de bonne qualité, et l'intérêt du transbordement à des fins d'optimisation a été une fois de plus prouvé.
  - La dernière phase a résidé dans l'application de la méthode de résolution développée en situation réelle. En ce sens, les expérimentations de terrain réalisées dans le cadre du projet PRODIGÉ ont offert

cette opportunité, mais ont également mené à une adaptation de la méthode. En effet, le problème de routage a été abordé sous l'angle d'un contexte changeant : toutes les informations nécessaires à la planification des tournées ne sont pas obligatoirement bien identifiées et sont révélées au fur et à mesure de l'exécution de l'activité de transport ; l'occurrence d'événements perturbateurs peut par exemple empêcher l'exécution normale des tournées planifiées et les rendre irréalisables du fait du non-respect de certaines contraintes. Ainsi, le chapitre 5, après avoir décrit plus précisément le cadre du projet PRODIGE, s'applique à décrire les nouvelles hypothèses du problème à résoudre, centré sur une recherche de reconfiguration de tournées, et les adaptations de la méthode de résolution développée en chapitre 4 qui ont été nécessaires pour intégrer ce contexte changeant. Ces nouveaux développements ont été testés dans le cadre du projet PRODIGE sur un certain nombre de cas d'applications réelles dans la région bordelaise, dont deux sont particulièrement décrits dans ce chapitre.

La complexité du problème à résoudre nous a amené à poser un certain nombre d'hypothèses. Les perspectives de cette recherche visent donc d'une part à lever certaines de ces hypothèses tout en prenant en considération de nouvelles hypothèses de travail et d'autre part à améliorer l'efficacité des méthodes de résolution :

- Le modèle linéaire en variables mixtes proposé peut être généralisé de manière à supporter plusieurs points de transbordement. Par ailleurs, afin de s'approcher d'un cas industriel, il est utile à l'avenir de pouvoir considérer une flotte de véhicules hétérogènes. En ce qui concerne la méthode de résolution elle-même et pour améliorer les performances de la méthode exacte, il est pertinent d'utiliser des techniques de coupes et d'ajouter des inégalités valides ainsi qu'exploiter des calculs de bornes. Ceci est d'autant plus pertinent que ces approches sont inexistantes pour le problème considéré, du fait de l'intérêt relativement récent sur le sujet.
- Concernant la méthode approchée, dans un premier temps, de nouveaux voisinages permettant une exploration plus efficace de l'espace de solutions peuvent être définis pour les approches heuristiques. L'efficacité recherchée est caractérisée par la capacité du voisinage à analyser l'espace de solutions de manière diversifiée dans l'espérance d'aboutir à des solutions s'approchant davantage de l'optimal. Dans un second temps il convient de reconsiderer la stratégie de résolution qui consistait principalement dans une première phase en la génération d'une solution initiale sans transbordement dans laquelle le transbordement était intégré dans une seconde phase dite de post optimisation. Cette stratégie a été utilisée dans un souci de maîtrise de temps de calcul. Une évolution du travail effectué consiste à déterminer le meilleur compromis entre temps d'exécution et l'utilisation du transbordement dès la génération de la solution initiale.

En plus des perspectives présentées ci-dessus, des extensions supplémentaires ont été développées dans le cadre de la thèse et nécessitent une validation à grande échelle. Ces extensions portent sur la considération de

nouveaux événements, la générnicité de la solution de transport proposée et sont présentées ci-dessous.

- Les solutions développées permettent d'avoir des tournées de coûts satisfaisants dans un contexte statique. Dans un tel contexte, la recherche à voisinages variables a été adaptée afin de résoudre des problèmes de collectes et livraisons avec transbordement et reconfiguration dans lesquels les tournées peuvent débuter en des points quelconques et avec la possibilité pour les véhicules de débuter leur tournée au dépôt avec une charge initiale non nulle. Le contexte changeant est caractérisé entre autres par l'occurrence d'événements perturbateurs. Nous avons centré l'analyse au chapitre 5 autour d'un seul type d'événement, à savoir le retard, considérant que de nombreux aléas de transport (panne, congestion de trafic, etc) avaient pour conséquence un non-respect des dates de livraisons. Néanmoins, notre méthode, par son adaptation à des fins de reconfiguration, permet la prise en compte de nouveaux clients ou l'annulation d'un ordre de transport : chaque reconfiguration s'appuyant sur une résolution d'un problème statique à conditions initiales changeantes, l'ajout ou le retrait d'une requête de transport à chaque occurrence de calcul ne pose pas de problème majeur. Cependant, ces caractéristiques n'ont pas été réellement étudiées, ni prouvées dans le présent travail, et la prise en compte de ces nouveaux événements ne se fait pas en temps réel, ce qui peut être handicapant dans une application où la réactivité de l'outil décisionnel est déterminante pour l'optimisation de l'activité de transport.
- Les modifications apportées à notre métaheuristique, prenant en considération une position quelconque des véhicules en début de tournée et une charge initiale, nous laissent croire que l'approche de calcul développée, quoi que non optimisée au sens du temps de calcul pour la résolution de tels problèmes, peut traiter des problèmes de type VRP (Vehicle Routing Problems). En effet, les requêtes de transport définissant initialement un point de collecte et un point de livraison peuvent être initialisées de manière à considérer que le point de collecte de chacune d'elles est le dépôt d'où partent les véhicules. Quelques tests ont été réalisés en ce sens, mais dans un nombre insuffisant, pour juger de la réelle qualité de la méthode, quant à son application pour la résolution de cette classe de problèmes.

# ANNEXES

A

## SOMMAIRE

A.1	MODÈLE COMPLET DU PDPT . . . . .	163
A.2	EXEMPLES D'INSTANCES DE PDPT . . . . .	165
A.3	FONCTION DE BASES . . . . .	172
A.4	CHOIX TECHNIQUES ET LIMITES DU RFID . . . . .	174
A.5	GÉNÉRATION D'UNE POSITION COURANTE CORRESPONDANT À UN RETARD SIMULÉ . . . . .	175
A.6	EXEMPLE DE TOURNÉES TYPES GÉNÉRÉES DANS PRODIGE . . .	176



## A.1 MODÈLE COMPLET DU PDPT

$$\text{Min } P * \sum_{k \in K} \left(1 - x_{o_1 o_2}^k\right) + \sum_k \sum_i \sum_j c_{ij}^k * x_{ij}^k \quad (\text{A.1})$$

$$\sum_{j \in P \cup \{o_2\}} x_{o_1 j}^k = 1, \forall k \in K \quad (\text{A.2})$$

$$\sum_{j \in D \cup \{o_1\}} x_{j o_2}^k = 1, \forall k \in K \quad (\text{A.3})$$

$$\sum_{j \in V} x_{ij}^k = y_i^k, \forall k \in K, \forall i \in N \quad (\text{A.4})$$

$$\sum_{i \in V} x_{ij}^k = y_j^k, \forall k \in K, \forall j \in N \quad (\text{A.5})$$

$$\sum_{k \in K} y_i^k = 1, \forall i \in N \quad (\text{A.6})$$

$$\sum_{j \in V} x_{ji}^k - \sum_{i \in V} x_{ij}^k = 0, \forall k \in K, \forall i \in N \quad (\text{A.7})$$

$$\sum_{i \in V} x_{io_3}^k = x_{o_3 o_4}^k, \forall k \in K \quad (\text{A.8})$$

$$\sum_{i \in V} x_{o_4 i}^k = x_{o_3 o_4}^k, \forall k \in K \quad (\text{A.9})$$

$$u_i^k - r_i^k = \sum_{j \in V} x_{ij}^k - \sum_{j \in V} x_{i+n, j}^k, \forall k \in K, \forall i \in P \quad (\text{A.10})$$

$$u_i^k + r_i^k \leq 1, \forall k \in K, \forall i \in P \quad (\text{A.11})$$

$$\sum_{k \in K} u_i^k \leq 1, \forall i \in P \quad (\text{A.12})$$

$$\sum_{k \in K} u_i^k - \sum_{k \in K} r_i^k = 0, \forall i \in P \quad (\text{A.13})$$

$$x_{ij}^k = 1 \Rightarrow dt_j^k \geq dt_i^k + s_i + t_{ij}, \forall k \in K, \forall i \in N, \forall j \in N \cup \{o_1, o_2, o_3\} \quad (\text{A.14})$$

$$x_{o_3 o_4}^k = 1 \Rightarrow dt_{o_4}^k \geq dt_{o_3}^k + \sum_{i \in P} u_i^k \cdot s_i, \forall k \in K \quad (\text{A.15})$$

$$x_{o_4 j}^k = 1 \Rightarrow dt_j^k \geq dt_{o_4}^k + \sum_{i \in P} r_i^k \cdot s_i + t_{o_4 j}, \forall k \in K, \forall j \in N \cup o_2 \quad (\text{A.16})$$

$$u_i^k = 1 \Rightarrow dt_{o_3}^k \geq dt_i^k, \forall k \in K, \forall i \in P \quad (\text{A.17})$$

$$r_i^k = 1 \Rightarrow dt_{o_4}^k \leq dt_{i+n}^k, \forall k \in K, \forall i \in P \quad (\text{A.18})$$

$$u_i^{k_1} + r_i^{k_2} = 2 \Rightarrow dt_{o_4}^{k_2} \geq dt_{o_3}^{k_1} + \sum_{i \in P} u_i^{k_1} \cdot s_i + t_{o_3 o_4}, \forall k_1, k_2 \in K, \forall i \in P \quad (\text{A.19})$$

$$a_i \cdot y_i^k \leq dt_i^k \leq b_i \cdot y_i^k, \forall i \in V, \forall k \in K \quad (\text{A.20})$$

$$\sum_{k \in k} dt_i^k \leq \sum_{k \in k} dt_{i+n}^k, \forall i \in P \quad (\text{A.21})$$

$$dt_{o_1}^k = 0, \forall k \in K \quad (\text{A.22})$$

$$x_{ij}^k = 1 \Rightarrow L_j^k = L_i^k + q_j, \forall i \in N, \forall j \in V, \forall k \in K \quad (\text{A.23})$$

$$x_{o_3 o_4}^k = 1 \Rightarrow L_{o_4}^k = L_{o_3}^k - \sum_{i \in P} u_i^k \cdot l_i + \sum_{i \in P} r_i^k \cdot l_i, \forall k \in K \quad (\text{A.24})$$

$$0 \leq L_i^k \leq Q, \forall i \in V, \forall k \in K \quad (\text{A.25})$$

$$L_i^k = 0, \forall k \in K, \forall i \in o_1 \cup o_2 \quad (\text{A.26})$$

$$x_{ij}^k, y_i^k \in \{0, 1\}, \forall k \in K, \forall (i, j) \in V \times V \quad (\text{A.27})$$

## A.2 EXEMPLES D'INSTANCES DE PDPT

Dans cette annexe, 16 instances correspondant aux problèmes de  $E1$  à  $E16$  au format de fichier texte, qui ont été fournies en entrée du logiciel Xpress MP sont présentées. Chaque fichier est structuré de la manière suivante :

- Les coordonnées de chaque point du problème d'optimisation au format  $(x, y)$  sont proposées. Elles permettent de générer le distancier du problème. Pour un point de collecte  $C_i$  ( $i \leq n$ ) , le point de livraison correspondant est donné par  $C_{i+n}$ .
- La demande (respectivement la durée de service) en chaque point est fournie par le vecteur  $LI$  (respectivement  $SI$ ).
- La borne inférieure et la borne supérieure en chaque point sont respectivement fournies par les vecteurs  $AI$  et  $BI$ .
- La capacité du véhicule est définie par  $Q$ .
- Le nombre de véhicules considéré dans le problème d'optimisation est défini par  $K$ .
- Le coefficient de pondération de la fonction objectif est donné par  $P$ .

### Instances E1

```
! Instance de test Pour le transbordement de taille n = 3
!o1, o2, o3, o4 (0,0) !c1 (-100 ; 100)
!c2 (100 ; 100)
!c3 (94 ; 94)
!c4 (100 ; -100)
!c5 (-100 ; -100)
!c6 (94 ; -94)
LI : [0, 0, 0, 0, 14, 8, 4, -14, -8, -4]
Q : 26
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
BI : [566, 566, 566, 566, 142.421, 142.421, 133.936, 425, 4250, 566]
K : 4
P=3000
```

### Instances E2

```
! Instance de test Pour le transbordement de taille n = 3
!o1, o2, o3, o4 (0,0)
!c1 (-100 ; 100)
!c2 (100 ; 100)
!c3 (1 ; 1)
!c4 (100 ; -100 )
!c5 (-100 ; -100)
!c6 (1 ; -1)
LI : [0, 0, 0, 0, 5, 20, 18, -5, -20, -18]
Q : 43
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
BI : [566, 566, 566, 566, 142.421, 142.421, 2.414, 425, 425, 566]
K : 4
```

$P=3000$

#### Instances E3

! Instance de test Pour le transbordement de taille  $n = 3$

!o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)

!c<sub>1</sub> (-100 ; 100)

!c<sub>2</sub> (100 ; 100)

!c<sub>3</sub> (86 ; 86)

!c<sub>4</sub> (100 ; -100)

!c<sub>5</sub> (-100 ; -100)

!c<sub>6</sub> (86 ; -86)

LI : [0, 0, 0, 0, 4, 3, 8, -4, -3, -8]

Q : 15

SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

AI : [0, 0, 0, 0, 0, 0, 0, 0, 0]

BI : [566, 566, 566, 566, 142.421, 142.421, 122.622, 425, 425, 566]

K : 4

$P=3000$

#### Instances E4

! Instance de test Pour le transbordement de taille  $n = 4$

!o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)

!co (0 ; 0)

!c<sub>1</sub> (-100 ; 100)

!c<sub>2</sub> (100 ; 100)

!c<sub>3</sub> (53 ; 53)

!c<sub>4</sub> (97 ; 97)

!c<sub>5</sub> (100 ; -100)

!c<sub>6</sub> (-100 ; -100)

!c<sub>7</sub> (53 ; -53)

!c<sub>8</sub> (97 ; -97)

LI : [0, 0, 0, 0, 6, 20, 2, 19, -6, -20, -2, -19]

Q : 47

SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

AI : [0, 0, 0, 0, 0, 0, 0, 0, 0]

BI : [566, 566, 566, 566, 142.421, 142.421, 75.953, 138.179, 425, 425, 566, 566]

K : 4

$P=3000$

#### Instances E5

! Instance de test Pour le transbordement de taille n=4

!o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)

!c<sub>1</sub> (-100 ; 100)

!c<sub>2</sub> (100 ; 100)

!c<sub>3</sub> (32 ; 32)

!c<sub>4</sub> (35 ; 35)

!c<sub>5</sub> (100 ; -100)

!c<sub>6</sub> (-100 ; -100)

!c<sub>7</sub> (32 ; -32)

!c<sub>8</sub> (35 ; -35)

LI : [0, 0, 0, 0, 15, 2, 2, 14, -15, -2, -2, -14]

Q : 33

SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

BI : [566, 566, 566, 566, 142.421, 142.421, 46.255, 50.497, 425, 425, 566, 566]

K : 4

P=3000

#### **Instances E6**

! Instance de test Pour le transbordement de taille n=4

!o1, o2, o3, o4 (0, 0)

!c1 (-100; 100)

!c2 (100; 100)

!c3 (95; 95)

!c4 (10; 10)

!c5 (100; -100)

!c6 (-100; -100)

!c7 (95; -95.0)

!c8 (10; -10)

LI : [0, 0, 0, 0, 3, 19, 17, 3, -3, -19, -17, -3]

Q : 42

SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

BI : [566, 566, 566, 566, 142.421, 142.421, 135.350, 15.142, 425, 425, 566, 566]

K : 4

P=3000

#### **Instances E7**

! Instance de test Pour le transbordement de taille  $n = 5$

!o1, o2, o3, o4 (0, 0)

!c1 (-100; 100)

!c2 (100; 100)

!c3 (6; 6)

!c4 (56; 56)

!c5 (98; 98)

!c6 (100; -100)

!c7 (-100; -100)

!c8 (6; -6.0)

!c9 (56; -56.0)

!c10 (98; -98)

LI : [0, 0, 0, 0, 16, 6, 14, 10, 9, -16, -6, -14, -10, -9]

Q : 55 SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

BI : [566, 566, 566, 566, 142.421, 142.421, 9.485, 80.196, 139.593, 425, 425, 566, 566]

K : 4

P=3000

#### **Instances E8**

! Instance de test Pour le transbordement de taille  $n = 5$

!o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)  
!c<sub>1</sub> (-100; 100)  
!c<sub>2</sub> (100; 100)  
!c<sub>3</sub> (15; 15)  
!c<sub>4</sub> (43; 43)  
!c<sub>5</sub> (94; 94)  
!c<sub>6</sub> (100; -100)  
!c<sub>7</sub> (-100; -100)  
!c<sub>8</sub> (15; -15)  
!c<sub>9</sub> (43; -43)  
!c<sub>10</sub> (94; -94)  
DI : [18, 20, 10, 13, 16]  
LI : [0, 0, 0, 0, 18, 20, 10, 13, 16, -18, -20, -10, -13, -16]  
Q : 77  
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
BI : [566, 566, 566, 566, 142.421, 142.421, 22.213, 61.811, 133.936, 425, 425, 566, 566, 566]  
K : 4  
P=3000

**Instances E9**

! Instance de test Pour le transbordement de taille  $n = 5$   
!o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)  
!c<sub>1</sub> (-100; 100)  
!c<sub>2</sub> (100; 100)  
!c<sub>3</sub> (26; 26)  
!c<sub>4</sub> (25; 25)  
!c<sub>5</sub> (38; 38)  
!c<sub>6</sub> (100; -100)  
!c<sub>7</sub> (-100; -100)  
!c<sub>8</sub> (26; -26)  
!c<sub>9</sub> (25; -25)  
!c<sub>10</sub> (38; -38)  
LI : [0, 0, 0, 0, 5, 12, 4, 8, 9, -5, -12, -4, -8, -9]  
Q : 38  
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
BI : [566, 566, 566, 566, 142.421, 142.421, 37.770, 36.355, 54.74, 425, 425, 566, 566, 566]  
K : 4  
P=3000

**Instances E10**

! Instance de test Pour le transbordement de taille  $n = 6$   
!o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)  
!c<sub>1</sub> (-100; 100)  
!c<sub>2</sub> (100; 100)  
!c<sub>3</sub> (44; 44)  
!c<sub>4</sub> (15; 15)

```

!c5 (23 ; 23)
!c6 (70 ; 70)
!c7 (100 ; -100)
!c8 (-100 ; -100)
!c9 (44 ; -44 )
!c10 (15 ; -15 )
!c11 (23 ; -23 )
!c12 (70 ; -70 )
LI : [0, 0, 0, 0, 9, 3, 13, 20, 10, 19, -9, -3, -13, -20, -10, -19]
Q : 74
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
BI : [566, 566, 566, 566, 142.421, 142.421, 63.225, 22.213, 33.527, 99.995, 425,
425, 566, 566, 566, 566]
K : 4
P=3000

```

### Instances E11

```

! Instance de test Pour le transbordement de taille n = 6
!o1, o2, o3, o4 (0, 0)
!c1 (-100; 100 )
!c2 (100; 100 )
!c3 (7; 7 )
!c4 (71; 71 )
!c5 (90; 90 )
!c6 (77; 77 )
!c7 (100; -100 )
!c8 (-100; -100 )
!c9 (7; -7 )
!c10 (71; -71 )
!c11 (90; -90 )
!c12 (77; -77 )
LI : [0, 0, 0, 0, 12, 19, 16, 1, 9, 6, -12, -19, -16, -1, -9, -6]
Q : 63
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
BI : [566, 566, 566, 566, 142.421, 142.421, 10.899, 101.409, 128.279, 109.894,
425, 425, 566, 566, 566, 566]
K : 4
P=3000

```

### Instances E12

```

! Instance de test Pour le transbordement de taille n = 6
!o1, o2, o3, o4 (0, 0)
!c1 (-100; 100 )
!c2 (100; 100 )
!c3 (27; 27 )
!c4 (32; 32 )
!c5 (72; 72 )
!c6 (85; 85 )

```

!c<sub>7</sub> (100 ; -100 )  
 !c<sub>8</sub> (-100 ; -100 )  
 !c<sub>9</sub> (27 ; -27 )  
 !c<sub>10</sub> (32 ; -32 )  
 !c<sub>11</sub> (72 ; -72 )  
 !c<sub>12</sub> (85 ; -85)  
 LI : [0, 0, 0, 0, 9, 20, 6, 12, 17, 6, -9, -20, -6, -12, -17, -6]  
 Q : 70  
 SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
 AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
 BI : [566, 566, 566, 566, 142.421, 142.421, 39.184, 46.255, 102.823, 121.208,  
 425, 425, 566, 566, 566, 566]  
 K : 4  
 P=3000

#### Instances E13

! Instance de test Pour le transbordement de taille  $n = 7$   
 !o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)  
 !c<sub>1</sub> (-100 ; 100 )  
 !c<sub>2</sub> (100 ; 100 )  
 !c<sub>3</sub> (88 ; 88 )  
 !c<sub>4</sub> (68 ; 68)  
 !c<sub>5</sub> (90 ; 90)  
 !c<sub>6</sub> (43 ; 43)  
 !c<sub>7</sub> (19 ; 19)  
 !c<sub>8</sub> (100 ; -100)  
 !c<sub>9</sub> (-100 ; -100)  
 !c<sub>10</sub> (88 ; -88)  
 !c<sub>11</sub> (68 ; -68)  
 !c<sub>12</sub> (90 ; -90)  
 !c<sub>13</sub> (43 ; -43)  
 !c<sub>14</sub> (19 ; -19 )  
 LI : [0, 0, 0, 0, 7, 10, 15, 13, 14, 18, 15, -7, -10, -15, -13, -14, -18, -15]  
 Q : 92  
 SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
 AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
 BI : [566, 566, 566, 566, 142.421, 142.421, 125.451, 97.167, 128.279, 61.811,  
 27.870, 425, 425, 566, 566, 566, 566]  
 K : 4  
 P=3000

#### Instances E14

! Instance de test Pour le transbordement de taille n=7  
 !o<sub>1</sub>, o<sub>2</sub>, o<sub>3</sub>, o<sub>4</sub> (0, 0)  
 !c<sub>1</sub> (-100 ; 100 )  
 !c<sub>2</sub> (100 ; 100 )  
 !c<sub>3</sub> (9 ; 9 )  
 !c<sub>4</sub>(63 ; 63 )  
 !c<sub>5</sub> (57 ; 57)  
 !c<sub>6</sub> (25 ; 25 )

```

!c7 (61 ; 61 )
!c8 (100 ; -100 )
!c9 (-100 ; -100 )
!c10 (9 ; -9 )
!c11 (63 ; -63.)
!c12 (57 ; -57)
!c13 (25 ; -25)
!c14 (61 ; -61)
LI : [0, 0, 0, 0, 2, 8, 11, 17, 5, 11, 19, -2, -8, -11, -17, -5, -11, -19]
Q : 73
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
BI : [566, 566, 566, 566, 142.421, 142.421, 13.728, 90.95, 81.61, 36.355, 87.267,
425, 425, 566, 566, 566, 566, 566.0]
K : 4
P=3000

```

**Instances E15**

! Instance de test Pour le transbordement de taille  $n = 7$

```

!o1, o2, o3, o4 (0, 0)
!c1 (-100 ; 100)
!c2 (100 ; 100)
!c3 (21 ; 21)
!c4 (53 ; 53)
!c5 (30 ; 30)
!c6 (75 ; 75)
!c7 (25 ; 25)
!c8 (100 ; -100)
!c9 (-100 ; -100)
!c10 (21 ; -21)
!c11 (53 ; -53)
!c12 (30 ; -30)
!c13 (75 ; -75)
!c14 (25 ; -25)
LI : [0, 0, 0, 0, 10, 16, 8, 14, 3, 16, 2, -10, -16, -8, -14, -3, -16, -2]
Q : 69
SI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
AI : [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
BI : [566, 566, 566, 566, 142.421, 142.421, 30.698, 75.953, 43.426, 107.066,
36.355, 425, 425, 566, 566, 566, 566, 566]
K : 4
P=3000

```

**Instances E16**

! Instance de test Pour le transbordement de taille  $n = 8$

```

!o1, o2, o3, o4 (0, 0)
!co (0, 0)
!c1 (-100 ; 100)
!c2 (100 ; 100)
!c3 (83 ; 83)

```

### A.3 FONCTION DE BASES

L'ensemble des fonctions de base nécessaires à la mise en place de l'algorithme « Insertionparallèle » sont les suivantes :

- $\text{VerifTemps } (r_k)$  est une fonction qui vérifie que la contrainte de fenêtre de temps n'est pas violée pour chaque client visité par le véhicule  $k \in K$  durant sa route  $r_k$ . Le pseudo code de la fonction est présenté dans l'algorithme 10.

---

**Algorithme 10 :** VerifTemps

**Entrées :**  $Route - r_k = \langle o_1, x_1, \dots, x_{i-1}, x_i, \dots, o_2 \rangle$  ;

**Sorties** : Booléen – vrai si aucune fenêtre de temps n'est violée et faux sinon ;

- 1 pour chaque  $x_j$  dans  $r_k$  faire
- 2    si  $dt_j^k > b_j$  alors
- 3     Retourne faux ;
- 4 Retourne vrai;

Cette fonction s'assure que la date d'arrivée du véhicule  $k$  chez chaque client  $j$  de la tournée  $r_k$  est inférieure ou égale à la borne supérieure de sa fenêtre de temps. Si les contraintes de temps sont respectées, la fonction renvoie la valeur booléenne « vrai » sinon, un renvoi de « faux » caractérise la violation d'au moins une contrainte de temps parmi les clients constituant la tournée.

- VerifCapacité ( $r_k$ ) est une fonction qui permet d'assurer que la contrainte de capacité propre au véhicule  $k \in K$  est respectée.

---

**Algorithme 11 : VerifCapacité**

**Entrées :** Route –  $r_k = \langle o_1, x_1, \dots, x_{j-1}, x_j, \dots, o_2 \rangle$ ; Capacité du véhicule

–  $Q$

**Sorties :** Booléen – vrai si la capacité du véhicule est respectée et faux sinon;

```

1  $q \leftarrow 0$ ;
2 pour chaque  $x_j$  dans  $r_k$  faire
3    $q \leftarrow q + l_j$ ;
4   si  $q > Q$  alors
5     Retourne faux;
6 Retourne vrai;
```

---

La fonction calcule de manière progressive la variation de charge du véhicule en charge d'exécuter la tournée  $r_k$ . Pour chaque client visité dans la tournée, la charge du véhicule quittant ce dernier est égale à sa charge lors de son arrivée en ce nœud, plus ou moins une quantité de produits selon que le client visité correspond à un lieu de collecte ou de livraison (ligne 3 – algorithme 3). Cette charge est alors comparée à la capacité du véhicule de manière à identifier les clients d'une tournée qui violent cette contrainte. La fonction retourne « faux » si la contrainte de capacité n'est pas respectée lors du service d'un client de la tournée, « vrai » sinon.

- Initialisation ( $o_1, o_2, k$ ) est une fonction qui permet d'initialiser la route du véhicule  $k$  avec le point de départ et de retour du dépôt (voir algorithme 12).

---

**Algorithme 12 : Initialisation**

**Entrées :** Dépôt de départ des véhicules –  $o_1$ ; Dépôt d'arrivée des véhicules –  $o_2$ ; Indice du véhicule –  $k$

**Sorties :** Nouvelle tournée –  $r_k$

```

1  $r_k \leftarrow \langle o_1, o_2 \rangle$ 
2 Retourne  $r_k$ ;
```

---

- Insertion ( $r_k, R_i$ ) est une heuristique d'insertion d'une requête au meilleur coût. Le pseudo code de cette fonction est présenté dans l'algorithme 13 et fait appel aux fonctions précédemment présentées.

---

**Algorithme 13 : Insertion**

---

**Entrées :**  $Route - r_k = \langle o_1, x_1, \dots, x_{j-1}, x_j, \dots, o_2 \rangle$ ; Requête –  $R_i = (i, i+n)$ ; Capacité du véhicule  $k - Q$

**Sorties :** Route –  $r_k$ ;

```

1  $Best \leftarrow \emptyset$ ;
2  $Bcost \leftarrow \infty$ ;
3 pour  $j \leftarrow 1$  à  $nb(r_k) - 1$  faire
4    $r_1 \leftarrow r_k$ ;
5   Insérer la demi requête ( $i$ ) en position  $j$  dans  $r_1$ ;
6   si  $VerifCapacité(r_1, Q) = vrai$  et  $VerifTemps(r_1) = vrai$  alors
7     pour  $l = j + 1$  à  $nb(r_1) - 1$  faire
8        $r_2 \leftarrow r_1$ ;
9       Insérer la demi requête ( $i+n$ ) en position  $l$  dans  $r_2$ ;
10      si  $VerifCapacité(r_2, Q) = vrai$  et  $VerifTemps(r_2) = vrai$  et
11         $C(r_2) < Bcost$  alors
12           $Bcost \leftarrow C(r_2)$ ;
13           $Best \leftarrow r_2$ ;
14
15  $r_k \leftarrow Best$ ;
16 Retourne  $r_k$ ;
```

---

Pour insérer la requête  $R_i$  de nœud de collecte ( $i$ ) et de nœud de livraison ( $i+n$ ) dans la tournée  $r_k$  du véhicule  $k$ , il convient d'évaluer toutes les positions d'insertions réalisables de ( $i$ ) et ( $i+n$ ) et de sélectionner la meilleure. Cette recherche de la meilleure insertion possible est une recherche exhaustive standard constituée de deux itérations imbriquées. La première itération externe (ligne 3) balaye toute les positions d'insertion du point de collecte ( $i$ ). La seconde itération interne (ligne 9) balaye toutes les positions d'insertion du point de livraison ( $i+n$ ) une fois que le point de collecte a été fixé. L'insertion du point de livraison ne peut s'effectuer que postérieurement à la collecte.

Les contraintes de capacités et de fenêtre de temps sont vérifiées avant chaque insertion d'un point de collecte ou de livraison. La variable «  $Bcost$  » (respectivement «  $Best$  ») sauvegarde le coût d'insertion minimum (respectivement la route de coût d'insertion minimum).

#### A.4 CHOIX TECHNIQUES ET LIMITES DU RFID

Le choix de la technologie RFID, quant à elle, s'est initialement imposée pour deux raisons : cette technologie est dans une phase de déploiement et de nombreux logisticiens croient en sa capacité à faciliter la maîtrise de l'activité logistique ; de plus, si cette technologie est couteuse, elle reste parmi les moins chères des technologies sans contact basées sur des solutions RF. Les différentes séries de tests réalisées ont permis cependant de montrer l'actuelle immaturité de cette technologie appliquée à un contexte de transport.

Les essais de lecture réalisés en champ libre, i.e. sans obstacle au déplacement des ondes électromagnétiques, ont en effet montré des perfor-

mances correctes mais son utilisation dans un milieu confiné telle que la remorque s'est avérée plus complexe, compte tenu du taux d'erreurs de lecture. Le diagnostic effectué a mis en exergue les difficultés suivantes :

- Le problème de directivité des antennes, les effets de rebonds en milieu fermé, notamment lorsque les parois sont protégées par des plaques métalliques ;
- La proximité de matériaux absorbants de nature métallique, carnée ou encore liquide des produits sur les palettes ;
- Les parasites dus à l'utilisation de bandes de fréquences relativement sollicitées par beaucoup d'équipements radio fréquences,
- La vitesse de franchissement du portique qui varie selon que l'on utilise chariot à gaz ou transpalette.

Néanmoins, cette technologie est en perpétuel développement, et de nouvelles solutions techniques arrivent régulièrement sur le marché. Il est donc possible de croire rapidement à son exploitation en contexte de transport routier et donc à la viabilité de la solution développée dans le cadre du projet dans les années futures.

La 3G ou troisième génération représente une génération de normes de téléphonie mobile offrant des débits plus rapides que ceux de la génération précédente telle que le GSM.

## A.5 GÉNÉRATION D'UNE POSITION COURANTE CORRESPONDANT À UN RETARD SIMULÉ

Dans un contexte changeant, le principe de résolution présenté au chapitre 5 fait état de changement de conditions initiales liées la plupart du temps à l'observation de dérives de temps dans l'exécution des tournées planifiées. Cette observation peut survenir alors que le véhicule se trouve à une position quelconque (position courante) de sa tournée entre deux points  $A : (x_A, y_A)$  et  $B : (x_B, y_B)$  faisant partie de cette tournée. La figure A.1(1) représente le problème projeté dans un repère orthonormé.

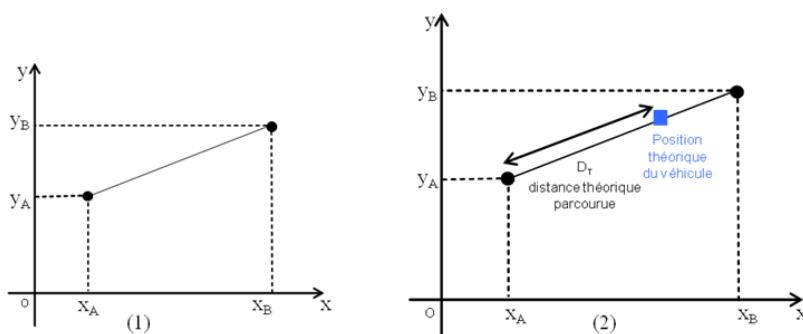


FIGURE A.1 – Position des clients dans un repère euclidien orthonormé

Connaissant la vitesse théorique du véhicule, il est donc possible d'estimer la distance théorique parcourue par ce dernier au bout de  $m$  minutes au départ du client  $A$ , tel que représenté par la figure A.1(2).

La génération d'un aléa consiste à choisir arbitrairement une valeur pour la distance réelle parcourue par le véhicule, qui soit inférieure à la

distance théorique, décrivant ainsi le retard qu'a pu prendre le véhicule, du fait de congestion de trafic par exemple  $DR < DT$ . Le cas de génération aléatoire du retard est représenté par la figure A.2(1).

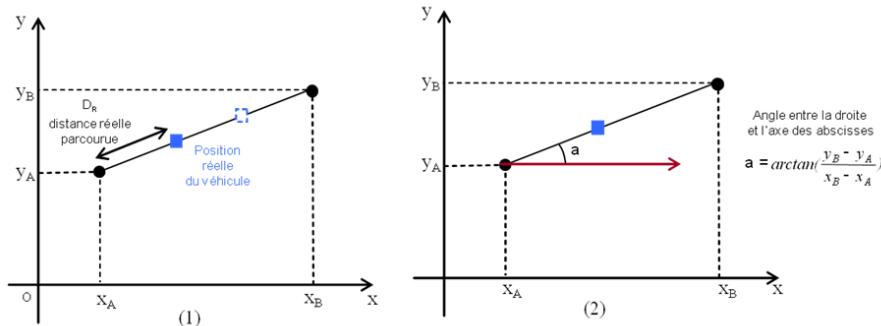


FIGURE A.2 – Position réelle du véhicule en cas de congestion et calcul d'angle

### DÉMARCHE

#### *Première étape – Calculer la pente de la droite*

Considérons l'équation de la droite  $y = p.x + q$ . La pente et l'origine de la droite sont calculées de la manière suivante :

$$\begin{aligned} - p &= \frac{(Y_B - Y_A)}{(X_B - X_A)} \\ - q &= Y_A - X_A \left( \frac{Y_B - Y_A}{X_B - X_A} \right) \end{aligned}$$

*Deuxième étape – Calculer l'angle « a » qui existe entre la droite et l'axe des abscisses, tel que illustré par la figure A.2(2)*

L'angle « a » qui existe entre la droite et l'axe des abscisses se calcule à l'aide de l'équation suivante :

$$- a = \arctang(p) = \arctang\left(\frac{Y_B - Y_A}{X_B - X_A}\right)$$

#### *Troisième étape – Calcul des coordonnées du véhicule*

L'angle que fait la droite avec l'axe des abscisses étant déterminé, il est possible de retrouver les coordonnées  $(x_R, y_R)$  du véhicule dans le repère euclidien, en utilisant les fonctions trigonométriques. Ainsi on aura :

- $X_R = X_A + D_R \cdot \cos(a)$
- $Y_R = Y_A + D_R \cdot \sin(a)$

## A.6 EXEMPLE DE TOURNÉES TYPES GÉNÉRÉES DANS PRODIGE

Dans PRODIGE, une feuille de route peut être visualisée sous forme de tournée telle que représentée dans la figure A.4 en se servant d'une cartographie. Cela permet au chauffeur d'avoir un aperçu global du trajet qu'il va effectuer avec son véhicule. D'autres informations sont aussi associées à cette feuille telles que les collectes, les livraisons, les produits ou unités d'expédition à transporter. Un exemple de représentation de ces informations est donné dans la figure A.5.

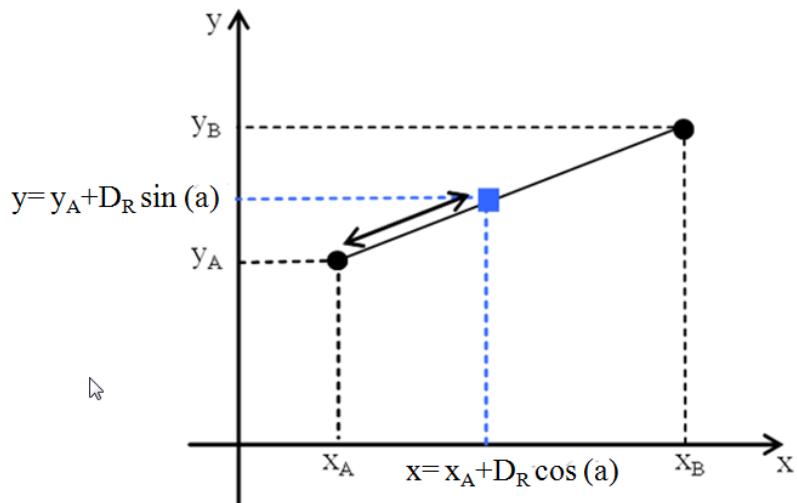


FIGURE A.3 – Position réelle du véhicule en cas de congestion et calcul d'angle

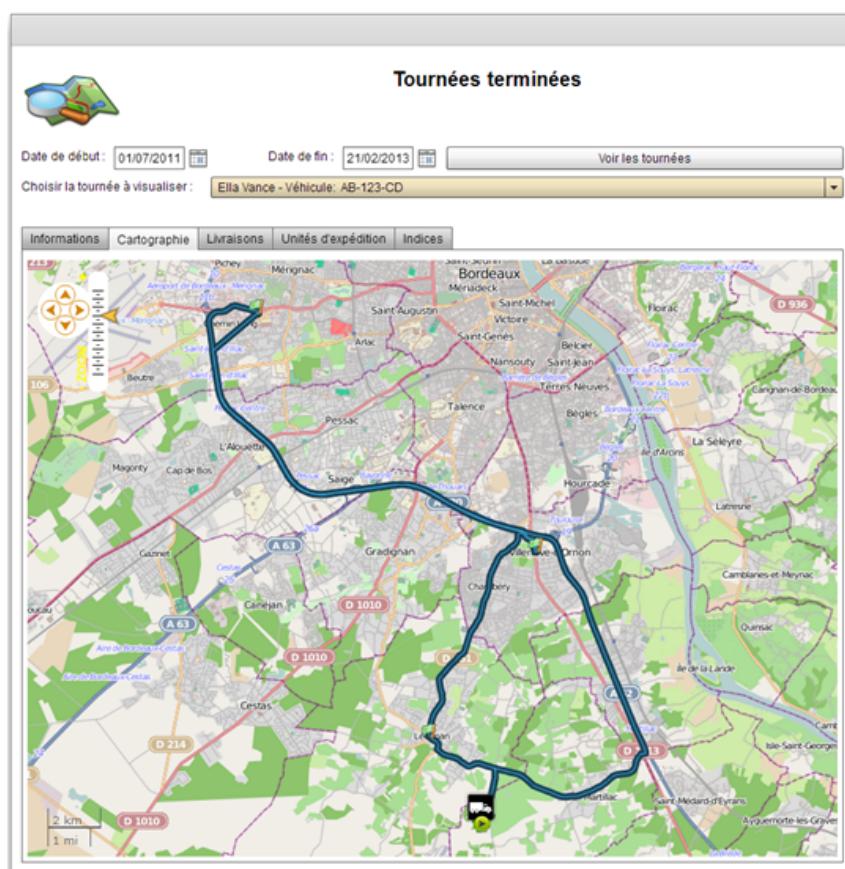


FIGURE A.4 – Exemple de feuille de route représentée sous forme de tournées

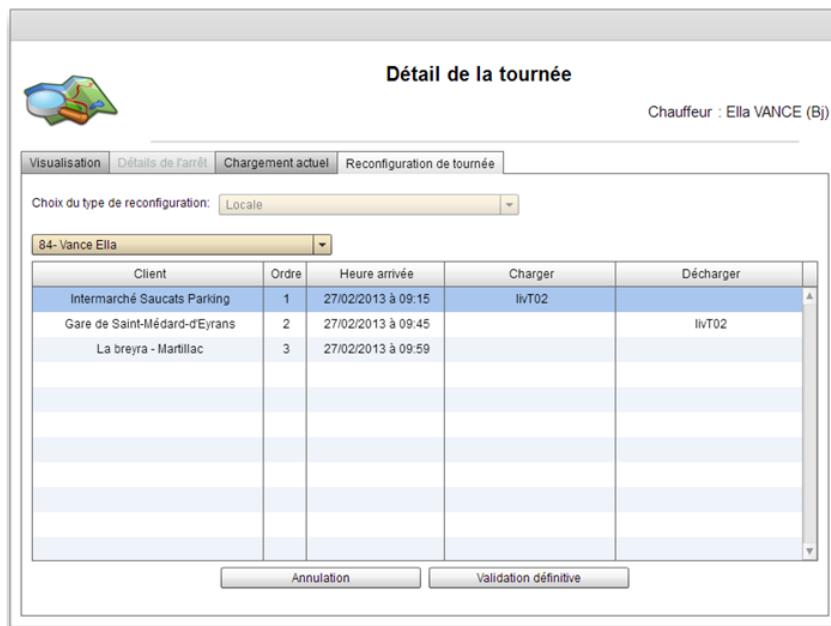


FIGURE A.5 – Détail des informations associées à une tournée

Dans la figure précédente, le chauffeur sait qu'il doit visiter trois clients. Pour chaque client la date et l'heure d'arrivée est spécifiée ainsi que l'opération qui y sera effectuée. Cette opération peut être un chargement, un déchargement. Si aucune opération n'est associée à un lieu visité, ce lieu correspond à un dépôt.

# BIBLIOGRAPHIE

- [Aldaihani2003] Aldaihani Majid and Dessouky Maged M. Hybrid scheduling methods for paratransit operations. *Computers & Industrial Engineering*, 45(1) :75 – 96, 2003. (Cité pages 65, 66 et 68.)
- [Beale1979] Beale E.M.L. Branch and bound methods for mathematical programming systems. In E.L. Johnson P.L. Hammer and B.H. Korte, editors, *Discrete Optimization II Proceedings of the Advanced Research Institute on Discrete Optimization and Systems Applications of the Systems Science Panel of NATO and of the Discrete Optimization Symposium co-sponsored by IBM Canada and SIAM Banff, Aha. and Vancouver*, volume 5 of *Annals of Discrete Mathematics*, pages 201 – 219. Elsevier, 1979. (Cité page 40.)
- [Bent2006] Bent Russell and Hentenryck Pascal Van. A two-stage hybrid algorithm for pickup and delivery vehicle routing problems with time windows. *Computers & Operations Research*, 33(4) :875–893, 2006. (Cité pages 53, 54, 55, 87 et 99.)
- [Berbeglia2007] Berbeglia Gerardo, Cordeau Jean-François, Gribkovskaia Irina, and Laporte Gilbert. Static pickup and delivery problems : a classification scheme and survey. *TOP*, 15(1) :1–31, 2007. (Cité pages 34, 37 et 77.)
- [Berbeglia2010] Berbeglia Gerardo, Cordeau Jean-François, and Gilbert Laporte. Dynamic pickup and delivery problems. *European Journal of Operational Research*, 202(1) :8–15, 2010. (Cité pages 56, 57, 58 et 136.)
- [Bernadet1997] Bernadet M. *Le transport routier de marchandise : fonctionnement et dysfonctionnements*. Economica, 1997. (Cité page 10.)
- [Blasum2007] Blasum Ulrich, Hochstättler Winfried, Oertel Peter, and Woeginger Gerhard J. Steiner diagrams and k-star hubs. *Journal of Discrete Algorithms*, 5(3) :622 – 634, 2007. Selected papers from Ad Hoc Now 2005 Ad Hoc Now 2005. (Cité page 67.)
- [Bouros2011] Bouros Panagiotis, Sacharidis Dimitris, Dalamagas Theodore, and Sellis Timos. Dynamic pickup and delivery with transfers. In Dieter Pfoser, Yufei Tao, Kyriakos Mouratidis, MarioA Nascimento, Mohamed Mokbel, Shashi Shekhar, and Yan Huang, editors, *Advances in Spatial and Temporal Databases*, volume 6849 of *Lecture Notes in Computer*

- Science*, pages 112–129. Springer Berlin Heidelberg, 2011. (Cité page 70.)
- [Boysen2010] Boysen Nils and Fliedner Malte. Cross dock scheduling : Classification, literature review and research agenda. *Omega*, 38(6) :413 – 422, 2010. (Cité pages 17 et 61.)
- [Branche2014] Branche François. Transport de messagerie. *Techniques de l'ingénieur Transport et logistique*, base documentaire : TIB123DUO.(ref. article : ag8151), 2014. fre. (Cité page 12.)
- [Branke2005] Branke Jürgen, Middendorf Martin, Noeth Guntram, and Dessouky Maged M. Waiting strategies for dynamic vehicle routing. *Transportation Science*, 39(3) :298–312, 2005. (Cité page 57.)
- [Braysy2002] Bräysy Olli and Gendreau Michel. Tabu search heuristics for the vehicle routing problem with time windows. *Top*, 10(2) :211–237, 2002. (Cité page 47.)
- [Brewer2001] Brewer Ann M., Button Kenneth J., and Hensher David A. *Handbook of logistics and supply-chain management / edited by Ann M. Brewer, Kenneth J. Button, David A. Hensher*. Pergamon Amsterdam ; New York, 2001. (Cité page 18.)
- [Cazier2007] Cazier J.A. and Poluha R.G. *Application of the SCOR Model in Supply Chain Management*. Cambria Press, 2007. (Cité page 21.)
- [Chiang1996] Chiang Wen-Chyuan and Russell RobertA. Simulated annealing metaheuristics for the vehicle routing problem with time windows. *Annals of Operations Research*, 63(1) :3–27, 1996. (Cité page 47.)
- [Chopra2007] Chopra S. and Meindl P. *Supply chain management : strategy, planning, and operation*. Pearson Prentice Hall, 2007. (Cité page 8.)
- [Cordeau2003] Cordeau Jean-François and Laporte Gilbert. A tabu search heuristic for the static multi-vehicle dial-a-ride problem. *Transportation Research Part B : Methodological*, 37(6) :579–594, 2003. (Cité page 47.)
- [Cortés2010] Cortés Cristián E., Matamala Martín, and Contardo Claudio. 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. (Cité pages 65, 66, 77 et 87.)
- [Dakin1965] Dakin R. J. A tree-search algorithm for mixed integer programming problems. *The Computer Journal*, 8(3) :250–255, 1965. (Cité page 40.)

- [Dantzig1959] Dantzig George B. and Ramser J.H. The truck dispatching problem. *Management Science*, 10(6) :80–91, 1959. (Cité pages 2 et 31.)
- [Desaulniers2005] Desaulniers G., Desrosiers J., and Solomon M.M. *Column Generation*. Cahiers du GERAD. Springer, 2005. (Cité page 41.)
- [Dode2010] Dodé Ludovic. La gestion des transports et les tms. [http://www.acteos.com/build/client/Article\\_Presse/PDF/20100616-TIL-dossier.special.TMS.pdf](http://www.acteos.com/build/client/Article_Presse/PDF/20100616-TIL-dossier.special.TMS.pdf), 2010. (Cité page 15.)
- [Dominguez2004] Dominguez H. and Lashkari R. S. Model for integrating the supply chain of an appliance company : a value of information approach. *International Journal of Production Research*, 42(11) :2113–2140, 2004. (Cité page 13.)
- [Drexel2012] Drexel Michael. Applications of the vehicle routing problem with trailers and transshipments. *European Journal of Operational Research*, (0) :–, 2012. (Cité page 63.)
- [Dumas1991] Dumas Yvan, Desrosiers Jacques, and Soumis François. The pickup and delivery problem with time windows. *European Journal of Operational Research*, 54(1) :7–22, 1991. (Cité pages 53, 54 et 57.)
- [Dupas2013] Dupas Rémy, Hsu Tiente, and Gonçalves Gilles. Chapter 14. metaheuristics for the solution of vehicle routing problems in a dynamic context. In P. Siarry B. Jarboui and J. Teghem, editors, *Metaheuristics for production scheduling*. ISTE WILEY, 2013. (Cité page 52.)
- [Feillet2010] Feillet Dominique, Gendreau Michel, Medaglia Andrés L., and Walteros Jose L. A note on branch-and-cut-and-price. *Operations Research Letters*, 38(5) :346–353, 2010. (Cité page 41.)
- [Fournier2008] Fournier Sylvain. *Outils pour des Problèmes Industriels de Tournées de Véhicules avec Transbordement*. Phd, Université Joseph-Fourier - Grenoble I, 2008. (Cité pages 65, 66, 68 et 73.)
- [Galbreth2008] Galbreth Michael R., Hill James A., and Handley Sean. An investigation of the value of cross-docking for supply chain management. *Journal of Business Logistics*, 29(1) :225–239, 2008. (Cité page 17.)
- [Gendreau2010] Gendreau Michel and Potvin Jean-Yves. *Handbook of Metaheuristics*. Springer Publishing Company, Incorporated, 2nd edition, 2010. (Cité page 39.)
- [Glover1989] Glover Fred. Tabu search – part i. *ORSA Journal on Computing*, 1(3) :190–206, 1989. (Cité page 46.)

- [Glover2001] Glover Fred. Tabu search. In *Encyclopedia of Operations Research and Management Science*, pages 821–827. Springer US, 2001. (Cité page 46.)
- [Glover2006] Glover Fred and Martí Rafael. Tabu search. In Enrique Alba and Rafael Martí, editors, *Metaheuristic Procedures for Training Neural Networks*, volume 36 of *Operations Research/Computer Science Interfaces Series*, pages 53–69. Springer US, 2006. (Cité page 46.)
- [Goldberg1989] Goldberg David E. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989. (Cité page 47.)
- [Gortz2009] Gortz IngeLi, Nagarajan Viswanath, and Ravi R. Minimum makespan multi-vehicle dial-a-ride. In Amos Fiat and Peter Sanders, editors, *Algorithms - ESA 2009*, volume 5757 of *Lecture Notes in Computer Science*, pages 540–552. Springer Berlin Heidelberg, 2009. (Cité pages 65, 66 et 69.)
- [Hansen2001] Hansen Pierre and Nenad Mladenović. Variable neighborhood search : Principles and applications. *European Journal of Operational Research*, 130(3) :449 – 467, 2001. (Cité page 100.)
- [Harvey1995] Harvey William D. and Ginsberg Matthew L. Limited discrepancy search. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1, IJCAI'95*, pages 607–613, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. (Cité page 55.)
- [Hax1984] Hax A.C. and Candea D.I. *Production And Inventory Management*. Prentice Hall College Division, 1984. (Cité page 14.)
- [Ho2004] Ho S.C. and Haugland D. A tabu search heuristic for the vehicle routing problem with time windows and split deliveries. *Computers & Operations Research*, 31(12) :1947–1964, 2004. (Cité page 47.)
- [Holland1992] Holland John H. *Adaptation in Natural and Artificial Systems : An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge, MA, USA, 1992. (Cité page 47.)
- [Insee2008] Insee. Emploi dans les transports routiers de marchandises, 2008. (Cité page 10.)
- [Kalantari1985] Kalantari Bahman, Hill Arthur V., and Arora Sant R. An algorithm for the traveling salesman problem with pickup and delivery customers. *European Journal of Operational Research*, 22(3) :377–386, 1985. (Cité pages 53 et 54.)
- [Kaplan2003] Kaplan R.S. and Norton D.P. *Le tableau de bord prospectif*. Eyrolles- Éd. d'Organisation, 2003. (Cité page 21.)

- [Kerivin2008] Kerivin H.L.M., Lacroix M., Mahjoub A.R., and Quilliot A. The splittable pickup and delivery problem with reloads. *European Journal of Industrial Engineering*, 2(2) :112–133, 2008. (Cité pages 65 et 66.)
- [Kirkpatrick1983] Kirkpatrick Scott, Gelatt C. D., and Vecchi M. P. Optimization by simulated annealing. *Science*, 220(4598) :671–680, 1983. (Cité page 47.)
- [Kirkpatrick1984] Kirkpatrick Scott. Optimization by simulated annealing : Quantitative studies. *Journal of Statistical Physics*, 34(5-6) :975–986, 1984. (Cité page 47.)
- [Kreng2008] Kreng Victor B. and Chen Fang-Tzu. The benefits of a cross-docking delivery strategy : a supply chain collaboration approach. *Production Planning & Control*, 19(3) :229–241, 2008. (Cité page 17.)
- [Lanneau2006] Lanneau K. Analyse et contrôle du cout logistique dans un contexte d'asymétrie informationnelle : cas des entreprises des départements français d'amérique. <http://www.intercostos.org/documentos/Lanneau.pdf>, 2006. (Cité page 16.)
- [Laporte1992] Laporte Gilbert. The vehicle routing problem : An overview of exact and approximate algorithms. *European Journal of Operational Research*, 59(3) :345–358, 1992. (Cité page 49.)
- [Lawler1966] Lawler E. L. and Wood D. E. Branch-and-bound methods : A survey. *Operations Research*, 14(4) :699–719, 1966. (Cité page 40.)
- [Lee1993] Lee Hau L. and Billington Corey. Material management in decentralized supply chains. *Operations Research*, 41(5) :835–847, 1993. (Cité pages 8 et 9.)
- [Lee2006] Lee Young Hae, Jung Jung Woo, and Lee Kyong Min. Vehicle routing scheduling for cross-docking in the supply chain. *Computers & Industrial Engineering*, 51(2) :247–256, 2006. (Cité pages 62 et 63.)
- [Li2001] Li H. and Lim A. A metaheuristic for the pickup and delivery problem with time windows. In *Tools with Artificial Intelligence, Proceedings of the 13th International Conference on*, pages 160–167, 2001. (Cité pages 53, 54, 55, 56 et 114.)
- [Liao2010] Liao Ching-Jong, Lin Yaoming, and Shih Stephen C. Vehicle routing with cross-docking in the supply chain. *Expert Syst. Appl.*, 37(10) :6868–6873, 2010. (Cité pages 62 et 63.)
- [Lin2008] Lin C.K.Y. A cooperative strategy for a vehicle routing problem with pickup and delivery time windows. *Computers & Industrial Engineering*, 55(4) :766–782, 2008. (Cité pages 65 et 66.)

- [Little1963] Little John D. C., Murty Katta G., Sweeney Dura W., and Karel Caroline. An algorithm for the traveling salesman problem. *Operations Research*, 11(6) :972–989, 1963. (Cité page 54.)
- [Man1999] Man K. F., Tang K. S., and Kwong S. *Genetic Algorithms : Concepts and Designs with Disk*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2nd edition, 1999. (Cité page 47.)
- [Masson2012] Masson Renaud. *Problèmes de collectes et livraisons avec transferts*. PhD thesis, École des Mines de Nantes - Nantes, 2012. (Cité pages 65, 66, 69 et 73.)
- [Mentzer2001] Mentzer John T., DeWitt William, Keebler James S., Min Soonhong, Nix Nancy W., Smith Carlo D., and Zacharia Zach G. Defining supply chain management. *Journal of Business Logistics*, 22(2) :1–25, 2001. (Cité page 13.)
- [Meyer2009] Meyer Gerben G., Främling Kary, and Holmström Jan. Intelligent products : A survey. *Computers in Industry*, 60(3) :137–148, 2009. (Cité page 124.)
- [Min19981] Min Hokey, Jayaraman Vaidyanathan, and Srivastava Rajesh. Combined location-routing problems : A synthesis and future research directions. *European Journal of Operational Research*, 108(1) :1–15, 1998. (Cité page 133.)
- [Minoux1983] Minoux M. *Programmation mathematique : theorie et algorithmes*. Collection technique et scientifique des télécommunications. Dunod, 1983. (Cité page 40.)
- [Mitrovic2004] Mitrović-Minić Snežana, Krishnamurti Ramesh, and Laporte Gilbert. Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B : Methodological*, 38(8) :669–685, 2004. (Cité pages 53, 54 et 58.)
- [Mitrovic2004a] Mitrović-Minić Snežana and Laporte Gilbert. Waiting strategies for the dynamic pickup and delivery problem with time windows. *Transportation Research Part B : Methodological*, 38(7) :635–655, 2004. (Cité pages 53, 54 et 58.)
- [Mitrovic2006] Mitrović-Minić Snežana and Laporte Gilbert. The pickup and delivery problem with time windows and transshipment. In *INFOR*, volume 44, pages 217–227, 2006. (Cité pages 65, 66, 68, 73, 78 et 110.)
- [Mladenovic1997] Mladenović Nenad and Hansen P. Variable neighborhood search. *Computers & Operations Research*, 24(11) :1097–1100, 1997. (Cité pages 44 et 108.)
- [Mues2005] Mues Christopher and Pickl Stefan. Transshipment and time windows in vehicle routing. In *Proceedings of the 8th International Symposium on Parallel Architectures,Algorithms and Net-*

- works, ISPAN '05, pages 113 – 119, Washington, DC, USA, 2005. IEEE Computer Society. (Cité pages 65 et 66.)
- [Nakao2008] Nakao Yoshitaka and Nagamochi Hiroshi. Worst case analysis for pickup and delivery problems with transfer. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, E91-A(9) :2328–2334, 2008. (Cité page 17.)
- [Nanry2001] Nanry William P and Barnes J Wesley. Solving the pickup and delivery problem with time windows using reactive tabu search. *Transportation Research Part B : Methodological*, 34(2) :107–121, 2000. (Cité pages 53, 54 et 55.)
- [Nemhauser1988] Nemhauser George L. and Wolsey Laurence A. *Integer and Combinatorial Optimization*. Wiley-Interscience, New York, NY, USA, 1988. (Cité page 39.)
- [New1995] New Stephen J and Payne Philip. Research frameworks in logistics : three models, seven dinners and a survey. *International Journal of Physical Distribution & Logistics Management*, 25(10) :60–77, 1995. (Cité pages 8 et 9.)
- [OCDE2001] OCDE. *Indicateurs de performance pour le secteur routier : Résumé des essais sur le terrain*. Recherche en matière de transport routier et intermodal. OCDE, 2001. (Cité page 21.)
- [Ombuki2006] Ombuki Beatrice, Ross BrianJ., and Hanshar Franklin. Multi-objective genetic algorithms for vehicle routing problem with time windows. *Applied Intelligence*, 24(1) :17–30, 2006. (Cité page 47.)
- [Ortel2000] Oertel Peter. *Routing with Reloads*. PhD thesis, University of Cologne, 2000. (Cité pages 65, 66 et 67.)
- [Parragh1] Parragh Sophie N., Doerner KarlF., and Hartl RichardF. A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(1) :21–51, 2008. (Cité page 37.)
- [Parragh2] Parragh Sophie N., Doerner KarlF., and Hartl RichardF. A survey on pickup and delivery problems. *Journal für Betriebswirtschaft*, 58(2) :81–117, 2008. (Cité page 37.)
- [Peng2008] Peng Wen and Zhou Chang-Yu. Solving vehicle routing problem using ant colony and genetic algorithm. In De-Shuang Huang, II Wunsch, DonaldC, DanielS Levine, and Kang-Hyun Jo, editors, *Advanced Intelligent Computing Theories and Applications. With Aspects of Contemporary Intelligent Computing Techniques*, volume 15 of *Communications in Computer and Information Science*, pages 23–30. Springer Berlin Heidelberg, 2008. (Cité page 47.)
- [Polacek2004] Polacek Michael, Hartl Richard F., Doerner Karl, and Reimann Marc. A variable neighborhood search for the multi depot vehicle routing problem with time windows. *Journal of Heuristics*, 10(6) :613–627, 2004. (Cité page 97.)

- [Prive2004] Privé Julie. *Solving a Vehicle Routing Problem Arising in Soft Drink Distribution*. Document de travail. Centre for Research on Transportation (C.R.T.), 2004. (Cité page 36.)
- [Qu2012] Qu Yuan and Bard Jonathan F. A grasp with adaptive large neighborhood search for pickup and delivery problems with transshipment. *Computers & Operations Research*, 39(10) :2439–2456, 2012. (Cité pages 65, 66, 69, 73, 117, 118 et 119.)
- [Rais2014] 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. (Cité page 81.)
- [Ropke2006] Ropke Stefan and Pisinger David. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4) :455–472, 2006. (Cité pages 53 et 56.)
- [Ropke2007] Ropke Stefan, Cordeau Jean-François, and Laporte Gilbert. Models and branch-and-cut algorithms for pickup and delivery problems with time windows. *Networks*, 49(4) :258–272, 2007. (Cité pages 53, 54 et 114.)
- [Santos2011] Santos Fernando Afonso, Mateus Geraldo Robson, and da Cunha Alexandre Salles. A branch-and-price algorithm for a vehicle routing problem with cross-docking. *Electronic Notes in Discrete Mathematics*, 37(0) :249–254, 2011. (Cité pages 62 et 63.)
- [Savelsbergh1990] Savelsbergh M.W.P. A parallel insertion heuristic for vehicle routing with side constraints. *Statistica Neerlandica*, 44(3) :139–148, 1990. (Cité pages 48 et 104.)
- [Savelsbergh1998] Savelsbergh Martin and Sol Marc. Drive : Dynamic routing of independent vehicles. *Operations Research*, 46(4) :474–490, 1998. (Cité pages 53, 54 et 57.)
- [Shang1996] Shang Jen S. and Cuff Carolyn K. Multicriteria pickup and delivery problem with transfer opportunity. *Computers & Industrial Engineering*, 30(4) :631–645, 1996. (Cité pages 17, 65, 66, 67 et 70.)
- [Shaw2003] Shaw T. *Performance measures of operational effectiveness for highway segments and systems*. Synthesis of highway practice. Transportation Research Board, 2003. (U.S.). (Cité page 19.)
- [Simchi2003] Simchi-Levi D., Kaminsky P., and Simchi-Levi E. *Designing and Managing the Supply Chain : Concepts, Strategies, and Case Studies*. Irwin/McGraw-Hill series in operations and decision sciences. McGraw-Hill/Irwin, 2003. (Cité page 13.)

- [Solomon1987] Solomon Marius M. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2) :254–265, 1987. (Cité page 114.)
- [Stadtler2005] Stadtler Hartmut. Supply chain management and advanced planning - basics, overview and challenges. *European Journal of Operational Research*, 163(3) :575 – 588, 2005. Supply Chain Management and Advanced Planning. (Cité page 13.)
- [Stadtler2010] Stadtler Hartmut and Kilger Christoph. *Supply Chain Management and Advanced Planning : Concepts, Models, Software, and Case Studies*. Springer Publishing Company, Incorporated, 4th edition, 2010. (Cité page 14.)
- [Still2009] Still Claus and Westerlund Tapiro. Extended cutting plane algorithm. In Christodoulos A. Floudas and Panos M. Pardalos, editors, *Encyclopedia of Optimization*, pages 959–966. Springer US, 2009. (Cité page 40.)
- [Stillb2009] Still Claus and Westerlund Tapiro. Sequential cutting plane algorithm. In Christodoulos A. Floudas and Panos M. Pardalos, editors, *Encyclopedia of Optimization*, pages 3471–3476. Springer US, 2009. (Cité page 40.)
- [Tayur1999] Tayur S., Ganeshan R., and Magazine M. *Quantitative Models for Supply Chain Management*. International Series in Operations Research & Management Science. Springer US, 1999. (Cité pages 8 et 9.)
- [Tchapnga2012a] Tchapnga Takoudjou Rodrigue, Jean-Christophe Deschamps, and Rémy Dupas. A mip formulation for the pickup and delivery problem with time windows and transhipment. In Borangiu Theodor, Dolgui Alexandre, Dumitrasche Ioan, and Filip Florin Gheorghe, editors, *14th IFAC Symposium on Information Control Problems in Manufacturing*, volume 4 of 5, Bucharest, Roumanie, 2012. International Federation of Automatic Control. (Cité page 81.)
- [Tchapnga2012b] Tchapnga Takoudjou Rodrigue, Jean-Christophe Deschamps, and Rémy Dupas. A hybrid multistart heuristic for the pickup and delivery problem with and without transhipment. In *Proceedings of MOSIM'12*, Bordeaux, France, 2012. SS07A2 - Optimization logistic Systems SS07A2 - Optimization logistic Systems. (Cité page 97.)
- [Tchapnga2013] Tchapnga Takoudjou Rodrigue, Deschamps Jean-Christophe, and Dupas Rémy. Opportunistic and dynamic reconfiguration of vehicle routing problem controlled by the intelligent product. In Christos Emmanouilidis, Marco Taisch, and Dimitris Kiritsis, editors, *Advances in Production Management Systems. Competitive Manufacturing for Innovative Products and Services*, volume 398 of *IFIP Advances in Information and Communication Technology*, pages 608–613. Springer Berlin Heidelberg, 2013. (Cité page 125.)

- [Thangiah2007] Thangiah SamR., Fergany Adel, and Awan Salman. Real-time split-delivery pickup and delivery time window problems with transfers. *Central European Journal of Operations Research*, 15(4) :329–349, 2007. (Cité pages 65, 66 et 70.)
- [Thompson1993] Thompson Paul M. and Psaraftis Harilaos N. Cyclic transfer algorithm for multivehicle routing and scheduling problems. *Operations Research*, 41(5) :935–946, 1993. (Cité page 106.)
- [Toth2002] Toth P. and Vigo D. *The Vehicle Routing Problem*. SIAM Monographs on Mathematical Modeling and Computation. Society for Industrial and Applied Mathematics (SIAM), 2002. (Cité page 49.)
- [Tsui1990] Tsui Louis Y. and Chang Chia-Hao. A microcomputer based decision support tool for assigning dock doors in freight yards. *Computers & Industrial Engineering*, 19 :309–312, 1990. (Cité page 60.)
- [Van2012] Van Belle Jan, Valckenaers Paul, and Cattrysse Dirk. Cross-docking : State of the art. *Omega*, 40(6) :827–846, 2012. (Cité page 61.)
- [Vander1993] Van der Bruggen L. J. J., Lenstra J. K., and Schuur P. C. Variable-depth search for the single-vehicle pickup and delivery problem with time windows. *Transportation Science*, 27(3) :298–311, 1993. (Cité pages 53 et 54.)
- [Wen2009] Wen Min, Larsen Jesper, Clausen Jens, Cordeau Jean-François, and Laporte Gilbert. Vehicle routing with cross-docking. *Operational Research Society. Journal*, 60(12) :1708–1718, 2009. (Cité pages 62, 63 et 83.)
- [Wong2002] Wong C.Y., McFarlane D., A. Ahmad Zaharudin, and Agarwal V. The intelligent product driven supply chain. In *Systems, Man and Cybernetics, 2002 IEEE International Conference on*, volume 4, page 6, 2002. (Cité pages 124 et 125.)
- [Yu2008] Yu Wooyeon and Egbelu Pius J. Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. *European Journal of Operational Research*, 184(1) :377–396, 2008. (Cité page 60.)

**Titre** Méthodes de modélisation et d'optimisation par recherche à voisinages variables pour le problème de collecte et de livraison avec transbordement

**Résumé** La présente thèse se déroule dans le cadre du projet ANR PRODIGE et est axée sur la recherche de stratégies permettant l'optimisation du transport en général et du transport routier de marchandises en particulier. Le problème de transport support de cette étude est le problème de collecte et livraison avec transbordement. Ce problème généralise plusieurs problèmes de transports classiques. Le transbordement y est utilisé comme levier de flexibilité et d'optimisation. Pour analyser et résoudre ce problème, les analyses sont effectuées suivant trois axes : le premier axe concerne l'élaboration d'un modèle analytique plus précisément d'un modèle mathématique en variables mixtes. Ce modèle permet de fournir des solutions optimales au décisionnaire du transport mais présente l'inconvénient de nécessiter un temps de résolution qui croît exponentiellement avec la taille du problème. Cette limitation est levée par le deuxième axe d'étude qui permet de résoudre le problème de transport étudié par une méthode d'optimisation approchée tout en garantissant des solutions satisfaisantes. La méthode utilisée est une métaheuristique inspirée de la recherche à voisnages variables (VNS). Dans le troisième axe, l'ensemble des résultats obtenus dans la thèse sont testés en situation de transports réels via le projet PRODIGE.

**Mots-clés** Élaboration de tournées de véhicules – Problèmes de collectes et livraisons avec transbordement – Programmation Linéaire en variables mixtes – Recherche à voisnages variables (VNS)

**Title** Modeling method and optimization by the variable neighborhood search for the pickup and delivery problem with transshipment

**Abstract** The thesis is conducted under the ANR project PRODIGE and it is focused on seeking strategies allowing the optimization of transport in general and road freight transport in particular. The transportation problem support for this study is the pickup and delivery problem with transshipment. This problem generalizes several classical transportation problems. Transshipment is used as optimization and flexibility leverage. To study and solve this problem, analyzes are performed along three axes : the first objective concerns the development of an analytical model, more accurately a mathematical model with mixed variables. This model allows providing optimal solution to the decision maker, but has the disadvantage of requiring a time resolution that grows exponentially with the size of the problem. This limitation is overcome by the second line of the study that solves the transportation problem studied by an approximate optimization method while ensuring satisfactory solutions. The method used is a mataheuristic broadly followed the variables neighborhoods research principles. In the third objective, the overall results obtained in the thesis are tested in real transport situation via the PRODIGE project.

**Keywords** Vehicle routing problems –Pickup and delivery problem with transshipment – Mixed integer linear programming – Variable neighborhood search – Intelligent product and RFID

## UNITÉ DE RECHERCHE

Laboratoire de L'Intégration du Matériau au Système (IMS) – UMR 5218 CNRS. 351, cours de la Libération 33405 Talence cedex France.