

Une méthode génétique pour la résolution du problème dynamique de routage de véhicules avec temps de parcours variables

THESE

Présentée et soutenue publiquement le 12 décembre 2008
en vue de l'obtention du

Doctorat de l'Université d'Artois

(Spécialité : Génie Informatique)

Par

ZHAO Xin

Composition du jury

<i>Président:</i>	Daniel JOLLY	Professeur à l'Université d'Artois (Directeur du LGI2A)
<i>Rapporteurs :</i>	Tahar KECHARDI	Professeur à l'University College of Dublin
	Dominique BREUIL	Docteur d'Etat, Directeur de la Recherche à l'EIGSI
<i>Examineurs :</i>	Gilles GONCALVES	Professeur à l'Université d'Artois (Directeur de thèse)
	Rémy DUPAS	Professeur à l'Université de Bordeaux 1 (Co-directeur de thèse)
	Daniel JOLLY	Professeur à l'Université d'Artois (Directeur du LGI2A)
	Tienté HSU	Maître de Conférences à l'Université d'Artois

Mis en page avec la classe thloria.

Remerciements

Cette thèse n'aurait vu le jour sans la confiance, la patience et la générosité de mon directeur de thèse, monsieur **Gilles Goncalves**, que je veux vivement remercier. Je voudrais aussi le remercier pour le temps et la patience qu'il m'a accordé tout au long de ces années. Les conseils et les aides qu'il m'a divulgués tout au long de la rédaction, ont permis la production de cette thèse.

Je remercie particulièrement monsieur **Daniel Jolly** directeur du LGI2A, de m'avoir accueilli dans son laboratoire de recherche.

Je remercie monsieur le professeur **Tahar Kechadi** de l'University College of Dublin et monsieur **Dominique Breuil** directeur de la recherche à l'EIGSI, de me faire l'honneur d'être rapporteurs de ce mémoire.

Mes plus sincères remerciements vont également à monsieur **Rémy Dupas**, qui en agissant au titre de co-directeur a fortement enrichi ma formation. Ses conseils et ses commentaires auront été fort utiles.

Je tiens à remercier tout particulièrement monsieur **Tienté Hsu** pour son intuition et ses nombreux conseils. Je souhaite également remercier tous les membres de LGI2A.

Je remercie en particulier **Ning Li**, de m'avoir supporté et aidé.

Enfin, je tiens à remercier de tout cœur mon père **Yingcheng Zhao** et ma mère **Min Wu**. Merci à vous deux, pour votre soutien, de m'avoir permis d'entreprendre ces études et de poursuivre dans la voie de la recherche.

Table des matières

Table d'abréviations	1
Introduction générale	3
Chapitre 1 Le problème de routage de véhicules et ses principales extensions	7
1.1 Introduction	7
1.2 Formulation	8
1.3 Variantes du problème VRP	10
1.4 Domaines d'applications	12
1.5 Les méthodes de résolution du VRP	12
1.5.1 Les méthodes exactes	13
1.5.1.1 Les méthodes de recherche arborescente (Branch & Bound) . . .	13
1.5.1.2 Programmation Linéaire en nombres entiers (Branch & Cut) . .	15
1.5.1.3 Programmation Dynamique	15
1.5.2 Les heuristiques	15
1.5.2.1 Algorithme de gain (Clarke & Wright)	15
1.5.2.2 Méthode d'insertion séquentielle	16
1.5.2.3 Algorithme groupe en premier, route en second	16
1.5.2.4 Algorithme route en premier, groupe en second	17
1.5.2.5 Algorithme de recherche locale	17
1.5.3 Les métaheuristiques	17
1.5.3.1 Les algorithmes génétiques	18
Codage	18
Opérateurs de reproduction	20
1.5.3.2 Recuit simulé	21
1.5.3.3 La méthode de recherche tabou	22
1.5.3.4 La colonie de fourmis	24
1.6 Le problème d'élaboration dynamique de tournées de véhicules DVRP	26
1.6.1 Introduction	26

1.6.2	Une classification des problèmes VRP dynamiques	27
1.6.2.1	VRP statique	28
1.6.2.2	VRP dynamique	29
1.6.2.3	Caractéristiques des problèmes statiques et dynamiques	29
1.6.3	Le degré de dynamisme	32
1.6.3.1	Dynamisme sans fenêtres de temps (dod)	32
1.6.3.2	Degré de dynamisme effectif - edod	33
1.6.3.3	Degré de dynamisme avec fenêtres de temps - $edod_{tw}$	33
1.7	Conclusion	34
Chapitre 2 Modèles de calcul de temps de trajet dépendant du temps		35
2.1	Introduction	35
2.2	Evaluation du temps de trajet	36
2.2.1	Temps de trajet discret	36
2.2.2	Temps de trajet continu	37
2.3	Modèles de calcul pour évaluer les temps de trajets	37
2.3.1	Modèles variables en fonction de la distance à parcourir	37
2.3.2	Modèles variables en fonction du temps	39
2.3.3	Modèle basé sur des horizons différents	41
2.3.4	Modèle basé sur des ratios de vitesse	41
2.3.5	Modèle basé sur des files d'attente	42
2.4	Conclusion	46
Chapitre 3 Une gestion dynamique de la tournée du réparateur avec temps de trajet dépendant du temps		47
3.1	Introduction	47
3.1.1	Formulation du DTRP	48
3.1.2	Formulation du PDTRP	48
3.1.3	Les applications de problème DTRP	49
3.2	Approches de résolution du DTRP	50
3.2.1	Approche de Bertsimas et Van Ryzin	50
3.2.2	Approche de Kilby, Prosser et Shaw	50
3.2.3	Approche de Irani, Lu, Regan	51
3.3	Proposition d'un modèle de résolution de DTRP avec temps de trajet dépendant du temps	51
3.3.1	Un modèle de vitesse de trajet dépendant du temps	52
3.3.2	Matrice des profils de vitesse	53

3.3.3	Les stratégies modifiées	55
3.4	Plate-forme et résultats	56
3.4.1	La plate-forme développée	56
3.4.2	Résultats et analyses de la simulation	57
3.5	Conclusion	58
Chapitre 4 Le problème DVRPTW avec temps de trajet dépendant du temps		63
4.1	Introduction	64
4.2	Le problème VRPTW	64
4.2.1	Formulation du VRPTW	65
4.2.2	Revue de littérature sur le VRPTW	66
4.2.2.1	Méthodes exactes	66
4.2.2.2	Méthodes approchées	67
4.3	Le problème DVRPTW avec temps de trajet dépendant du temps	69
4.4	Algorithme Génétique pour le problème DVRPTW	69
4.4.1	Introduction	69
4.4.2	Le codage de la solution	70
4.4.3	Fonction d'évaluation	71
4.4.4	Décodage d'un chromosome	72
4.4.5	Population initiale	74
4.4.6	Opérateurs de Sélection	74
4.4.7	Opérateurs de croisement (Crossover)	75
	PMX (Partially Mapped Crossover)	75
	MX1T/D	76
4.4.8	Opérateurs de mutation	78
4.4.9	Méthode de remplacement par l'élitisme	80
4.5	Plate-forme de Simulation Dynamique	80
4.5.1	Le composant Algorithme Génétique (AG)	81
4.5.2	Le composant Gestion d'Événements (GE)	81
4.5.3	Gestion du temps	82
4.5.4	Planification des tournées	82
4.5.5	L'évènement « fin de service »	84
4.5.6	L'évènement « arrivée d'un nouveau client »	84
4.6	Résolution du DVRPTW avec temps de trajet dépendant du temps	87
4.6.1	Les paramètres de l'algorithme génétique	87
4.6.2	Les profils de vitesse dépendant du temps	87
4.6.3	Benchmark utilisé	88

4.6.4	Méthode de validation	90
4.6.5	Les résultats de simulation	90
4.7	Conclusion	92

Chapitre 5 Le problème DVRPTW temps réel avec temps de trajet dépendant du temps 95

5.1	Introduction	95
5.2	Le problème DVRPTW temps réel avec temps de trajet dépendant du temps . .	96
5.2.1	Technologies temps réel	96
5.2.2	Revue de la littérature	97
5.2.2.1	Modèle à base de profils de vitesses	97
5.2.2.2	Modèle à base de profils de temps de trajet	98
5.3	Approche de résolution du DVRPTW temps réel avec temps de trajet dépendant du temps	99
5.3.1	Méthode de calcul de temps de trajet réel	100
5.3.2	Le modèle de prévision	100
5.3.3	Calcul des temps de trajet intégrant la prévision	102
5.4	Environnement de simulation	103
5.4.1	Fichiers de vitesses	103
5.4.2	Synchronisation du planificateur	103
5.4.3	Architecture du simulateur	104
5.4.4	Gestion d'Événements	105
5.4.4.1	L'évènement « Vitesse mesurée »	105
5.4.4.2	L'évènement « Arrivée effective chez un client »	107
5.4.4.3	L'évènement « Fin de service »	107
5.4.4.4	L'évènement « Arrivée d'un nouveau client »	107
5.4.5	Déroulement d'une simulation	107
5.4.5.1	Phase statique basée sur les profils historiques	109
5.4.5.2	Phase dynamique basée sur les profils de planification	109
5.4.5.3	Phase réelle basée sur les profils de vitesses réelles	109
5.5	Expériences et résultats	109
5.5.1	Génération des vitesses réelles	109
5.5.2	Réglage du facteur de lissage β	111
5.5.3	Méthode d'évaluation de la qualité des solutions	112
5.5.4	Résultats	114
5.5.4.1	Expérimentations basées sur des scénarios de congestions	114

5.5.4.2	Expérimentations basées sur la variation du facteur de ralentissement	117
5.5.4.3	Résultats sur l'ensemble des problèmes tests du benchmark . . .	118
5.5.4.4	Analyse des résultats selon le degré de dynamisme	118
5.6	Conclusions	119
Conclusions générales et perspectives		121
Annexes		125
Annexe A Résultats des problèmes de PDTRP avec temps de trajet dépendant du temps		127
A.1	Exemple : Le benchmark utilisé	127
A.2	Résultats complets	128
Annexe B Résultats des problèmes de DVRPTW avec temps de trajet dépendant du temps		145
B.1	Réglages des paramètres de l'algorithme génétique	145
B.2	Exemple : Le problème C101 dynamique	147
Annexe C Résultats des problèmes de DVRPTW temps réel avec temps de trajet dépendant du temps		151
C.1	Résultats pour les neuf problèmes C1 et pour les scénarios D2, D3 et D4	151
C.2	Résultats pour les neuf problèmes C1 et pour les trois scénarios V1, V2 et V3 . .	155
C.3	Résultats pour toutes les catégories de problèmes de test avec la distribution D2 et le scénario V1	158
Bibliographie		165
Publications personnelles		177
Index		179

Table des figures

1.1	Le problème VRP	8
1.2	Les méthodes de résolution du VRP	14
1.3	Fusion dams (Clarke & Wright)	16
1.4	Groupe en premier, route en second	17
1.5	Le problème DVRP	27
1.6	Le cas d'échec	29
1.7	Deux scénarios ayant des "dod" équivalents	32
2.1	Modèles de vitesse constante et de vitesse variable	36
2.2	Fonction de temps de trajet sur un arc	37
2.3	Fonction de vitesse par morceaux et temps de trajet continu	38
2.4	Changement de la vitesse en fonction du temps	40
2.5	Calcul de temps de trajet à la prévision	42
2.6	Types de profil de vitesses variables	43
2.7	Représentation de flux de trafic par Files d'Attente	44
2.8	Relation entre vitesse et flux pour le modèle M/M/1	45
2.9	Description du calcul du temps de trajet	46
3.1	Profil de vitesse	52
3.2	Exemple de calcul de temps de trajet	53
3.3	Exemples de temps de trajet	54
3.4	Exemple de graphe avec des profils de routes différents	54
3.5	Deux modes de résolution	57
3.6	Gain de « temps de trajet » pour la stratégie NN, 30 clients	59
3.7	Gain de « temps d'attente » pour la stratégie NN, 30 clients	59
3.8	Gain de « temps de trajet » pour la stratégie PartNN2, 30 clients	59
3.9	Gain de « temps d'attente » pour la stratégie PartNN2, 30 clients	60
3.10	Gain de « temps de trajet » pour la stratégie NN, 50 clients	60
3.11	Gain de « temps d'attente » pour la stratégie NN, 50 clients	60
3.12	Gain de « temps de trajet » pour la stratégie PartNN2, 50 clients	61
3.13	Gain de « temps d'attente » pour la stratégie PartNN2, 50 clients	61
4.1	Représentation directe	71
4.2	Représentation indirecte	71
4.3	Le chromosome représentant les clients qui restent à servir à l'instant t	72
4.4	L'ensemble des tournées représentant les clients servis ou en cours de service à l'instant t	72
4.5	Le décodage d'un chromosome	74

4.6	Principe du croisement	76
4.7	Principe de la mutation	78
4.8	L'opérateur de mutation 1-Opt	79
4.9	L'opérateur de mutation Or-Opt	79
4.10	L'évolution générale de la population	80
4.11	Simulateur pour le DVRPTW	81
4.12	Durée de la simulation	82
4.13	La simulation de la journée de service	82
4.14	Diagramme UML montrant la séquence d'interactions entre les composants AG et GE, à la fin d'une génération	83
4.15	Procédure d'évènement fin de service	85
4.16	Procédure d'évènement nouveau client	86
4.17	Les jeux de tests de Solomon pour le VRPTW	89
5.1	Les différents types de discrétisation de la vitesse	98
5.2	Modification en temps réel des profils de temps de trajet	98
5.3	Tournées de véhicules et nœuds critiques	100
5.4	Principe d'élaboration du profil utilisé pour la planification de tournées	102
5.5	Synchronisation des temps d'arrivée	104
5.6	Simulateur pour le DVRPTW temps réel avec de temps de trajet dépendant du temps	105
5.7	Diagramme de séquence entre les composants AG et GE du simulateur	106
5.8	Profil de vitesses réelles avec perturbations	110
5.9	Un exemple de profil de vitesses avec une congestion et des perturbations	111
5.10	Les différentes distributions de congestion	111
5.11	Deux modes de résolution	114

Liste des tableaux

1	Table d'abréviations	2
1.1	Caractéristiques du problème VRP	10
1.2	Différentes applications du VRP	13
1.3	Classification du VRP	28
1.4	Le problème DVRP et ses extensions dans la littérature	31
2.1	Profil basé sur les vitesses de conception	42
2.2	Caractéristique des principaux modèles de files d'attente	44
3.1	Matrice des profils de vitesse	55
3.2	Profils de vitesse de trajet pour les scénarios 1 à 3	55
3.3	Jeu de 100 problèmes à 30 clients pour un <i>dod</i> 50% : stratégie NN	58
4.1	Les objectifs choisis pour la résolution du VRPTW	68
4.2	Le réglage de chaque groupe des benchmarks statiques	87
4.3	Profils de vitesse pour les scénarios 1 à 3	88
4.4	Résultats pour le benchmark statique	91
4.5	Résultats pour le benchmark dynamique	93
5.1	Variation du pourcentage d'erreur en fonction du paramètre β et du profil de route	113
5.2	Quatre scénarios d'affectation des congestions sur les 3 types de routes	114
5.3	Résultats du problème C1 pour le scénario D1	115
5.4	Résultats du problème C1 pour les 4 scénarios de distribution des congestions	116
5.5	Profils de vitesse pour les scénarios V1 à V3	117
5.6	Résultats du problème C1 pour les 3 scénarios de variation de vitesses de chaque période	117
5.7	Résultats des 6 catégories de problèmes pour les scénarios D2 et V1	118
A.1	Le benchmark de 50 clients et 30% de <i>dod</i>	127
A.2	Jeu de 100 fichiers de 30 clients pour la stratégie NN	128
A.3	Jeu de 100 fichiers de 30 clients pour la stratégie PartNN2	132
A.4	Jeu de 100 fichiers de 50 clients pour la stratégie NN	135
A.5	Jeu de 100 fichiers de 50 clients pour la stratégie PartNN2	139
B.1	Réglages des paramètres de l'algorithme génétique pour le type 1 de problèmes dynamiques	145
B.2	Réglages des paramètres de l'algorithme génétique pour le type 2 de problèmes dynamiques	146

B.3	Fichier de données du problème dynamique C101Dyn (Format texte)	147
C.1	Résultats du problème C1 pour le scénario D2	152
C.2	Résultats du problème C1 pour le scénario D3	153
C.3	Résultats du problème C1 pour le scénario D4	154
C.4	Résultats du problème C1 pour le scénario V1	155
C.5	Résultats du problème C1 pour le scénario V2	156
C.6	Résultats du problème C1 pour le scénario V3	157
C.7	Résultats du problème C1	158
C.8	Résultats du problème R1	159
C.9	Résultats du problème RC1	160
C.10	Résultats du problème C2	161
C.11	Résultats du problème R2	162
C.12	Résultats du problème RC2	163

Table des algorithmes

1.1	Pseudo-code d'un algorithme génétique de base	19
1.2	Pseudo-code de la règle de Metropolis	21
1.3	Pseudo-code de l'algorithme du recuit simulé	22
1.4	Pseudo-code de la méthode de recherche tabou	23
1.5	Pseudo-code de la méthode de colonie de fourmis	24
2.1	Calcul du temps de trajet	40
3.1	Stratégie NN basée sur le temps de trajet	55
3.2	Stratégie PartNN2 basée sur le temps de trajet	56
4.1	Algorithme d'insertion (décodage d'un chromosome)	73
4.2	Algorithme de traitement de l'évènement « fin de service »	84
4.3	Algorithme de traitement de l'évènement « nouveau client »	86
5.1	Calcul du temps de trajet intégrant la prévision	103
5.2	Algorithme de traitement de l'évènement « Vitesse mesurée »	107
5.3	Algorithme de traitement de l'évènement « Arrivée effective chez un client »	108
5.4	Algorithme de traitement de l'évènement « fin de service »	108

Table d'abréviations

Abréviation	Description
RO	Recherche Opérationnelle
AE	Algorithmes Evolutionnistes
AG	Algorithme Génétique
RT	Recherche Tabou
PE	Plans d'Expériences
PC	Plan d'expériences Complet
PF	Plan d'expériences Fractionnaire
GIS	Graphical Information Systems
GPS	Global Positioning System
DTP	Dynamics Transports Problems
VRP	Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time Windows
VRPB	Vehicle Routing Problem with Backhauls
DVRP	Dynamic Vehicle Routing Problem
DVRPTW	Dynamic Vehicle Routing Problem with Time Windows
SVRP	Stochastic Vehicle Routing Problem
PVRP	Periodic Vehicle Routing Problem
MDVRP	Multi-Depot Vehicle Routing Problem
TRP	Traveling Repairman Problem
DTRP	Dynamic Traveling Repairman Problem
TSP	Traveling Salesman Problem
DTSPW	Dynamic Traveling Salesman Problem with Time Windows
ADTSPW	Apriori Dynamic Traveling Salesman Problem with Time Windows
VRPPD	Vehicle Routing Problem with Pick-up and Delivery
DPDPTW	Dynamic Pick-up and Delivery Problem with Time Windows
OVRP	Open Vehicle Routing Problem
VRPHF	Vehicle Routing Problem with Heterogeneous Fleet
VRPSD	Vehicle Routing Problem with Split Delivery
<i>m</i> -VRP	Vehicle Routing Problem with Limited Number of Vehicles
dod	Degré de dynamisme (Degree Of Dynamism)
edod	Degré de dynamisme effectif (Effective Degree Of Dynamism)
edod-tw	Degré de dynamisme effectif pour les problèmes à fenêtre de temps
IP	Idle Point
FCFS	First Come First Serve

A suivre ⇒

Table d'abréviations

Abréviation	Description
NN	Nearest Neighbor
PART	The partitioning policy
SQM	Stochastic Queue Median
GEN	GENeration strategy
HI-REQ	HIghest expected number of immediate REQuests
DF	Drive-First
WF	Wait-First
DW	Dynamic Waiting
ADW	Advanced Dynamic Waiting
PFIH	Push Forward Insertion Heuristic
PMX	Partially Mapped Crossover
MX1T/D	Merge Heuristic (Time, Distance)
GE	Gestion d'Événements

TABLEAU 1: Table d'abréviations

Introduction générale

Notre travail se consacre à l'étude du Problème de Tournées de Véhicules (PTV ou VRP pour Vehicle Routing problem en anglais). Ce problème a été étudié avec d'autant plus d'intérêt ces dernières décennies puisque les coûts de transport représentent une part importante du coût final imputé au client. Avec les fluctuations du prix du baril de pétrole il devient essentiel de maîtriser les coûts de transports dans les schémas des chaînes logistiques globales. En plus de l'intérêt économique évident, la recherche de solution optimale ou quasi optimale pour ce problème constitue un enjeu intéressant pour la communauté scientifique de la Recherche Opérationnelle (RO). Ces problèmes sont appelés NP-difficile car ils peuvent rarement être résolus par un algorithme polynomial en temps d'exécution. L'intérêt scientifique s'est porté plus particulièrement sur l'apport de nouvelles méthodes d'optimisation combinatoire permettant la résolution de ces problèmes en des temps de résolution acceptables.

La plupart des travaux se sont concentrés sur les cas statiques et déterministes dans lesquels toutes les informations sont supposées connues de façon certaine au moment de la planification des itinéraires. Cependant dans les applications courantes, des informations dynamiques et/ou stochastiques apparaissent alors que l'itinéraire planifié est en cours d'exécution. L'ère de l'optimisation où l'information peut être reçue et traitée en temps réel grâce aux technologies de plus en plus performantes des communications a amplifié cette évolution vers la nécessité d'une réponse de plus en plus réactive à une demande immédiate.

D'autre part, la plupart des planificateurs de tournées existant font l'hypothèse d'une vitesse de véhicule constante, évaluée sur une valeur moyenne pour établir des temps de trajets prévisionnels. Dans les problèmes réels, les vitesses sont variables au cours du temps car sujettes à des variations périodiques (période de fort trafic) ou à des aléas (pannes). Il est donc intéressant de choisir un nouveau modèle de calcul de temps de trajet pour s'adapter à cet environnement dynamique. Le problème que nous traitons donc dans cette thèse considère à la fois l'apparition de nouvelles demandes de service en cours de journées et des temps de trajet variables en fonction du trafic pour la période de temps considérée. Ce modèle consiste à se rapprocher de la réalité en considérant que les trajets ne sont plus effectués à vitesse constante sur toute la journée, mais seulement sur des plages horaires prédéfinies (i.e. vitesse constante par morceaux). Ces plages correspondent aux différentes périodes d'une journée (heures de pointes ou heures creuses).

En outre, les problèmes de décision en temps réel jouent un rôle de plus en plus important dans plusieurs secteurs car les technologies de communication et d'information permettent aujourd'hui d'obtenir et de traiter rapidement l'information en temps réel. La variabilité des conditions de trafic routier ne peut être ignorée si on veut réaliser une optimisation des routes de bonne qualité.

Les objectifs de cette thèse peuvent être résumés dans les points suivants :

- Intégrer dans une gestion *dynamique* de tournées de véhicule, un modèle de calcul de temps de trajet prenant en compte *la variabilité de la vitesse* d'un véhicule en fonction de

- la période de temps considérée. Vérifier que le modèle à vitesse variable donne des résultats meilleurs que le modèle à vitesse constante à l'aide d'un simulateur ;
- Etendre le modèle précédent pour tenir compte des congestions de trafic qui peuvent être suivies en temps réel. Pour cela, on définira un profil de vitesse variable en temps réel qui intègre ces *informations trafic routier* ;
- Modifier une plate forme de résolution/simulation Java de type évolutionniste (basée sur un Algorithme Génétique) pour y évaluer ces modèles.
- Vérifier que les performances que l'on pouvait attendre de ce type de méthode sont comparables voire supérieures à celles des meilleures techniques utilisées à ce jour.

Le présent document est structuré de la façon suivante :

Le premier chapitre présente la problématique du **VRP**. Dans ce chapitre, nous rappelons les différents éléments qui composent ce problème, ainsi que les contraintes à satisfaire et l'objectif à optimiser. La formulation mathématique de ce problème est ensuite rappelée. Nous rappelons la plupart des variantes de ce problème trouvées dans la littérature, puis nous présentons un état de l'art sur les méthodes qui sont utilisées pour la résolution de ce problème. Nous présentons aussi la version dynamique du problème VRP. Une comparaison entre les deux versions (statique et dynamique) est présentée. Ensuite, la notion de *degré de dynamisme* d'un problème dynamique est rappelée et une classification des problèmes dynamiques **DVRP** (Dynamic Vehicle Routing Problem) est proposée à partir de cette notion. Nous terminons ce chapitre par une revue sur la littérature concernant les méthodes de résolution du problème VRP dynamique et de quelques unes de ses extensions.

Le deuxième chapitre est consacré aux modèles de calcul de temps de trajet permettant de s'adapter à un environnement où les vitesses sont variables au cours du temps. Ces modèles se rapprochent de la réalité en considérant que les trajets ne sont plus effectués à vitesse constante sur toute la journée, mais seulement sur des plages horaires prédéfinies. Nous présentons plus en détail le modèle de Ichoua [Ichoua et al. 2003] pour l'étendre dans les chapitres suivants pour intégrer la prise en compte d'information temps réel.

Le troisième chapitre se focalise sur le problème dynamique de la tournée du réparateur. Nous avons adopté le modèle de calcul de temps de trajet précédent pour traiter ce type de problème. Nous avons adapté les stratégies classiques NN et Part en modifiant la fonction objectif qui consiste à minimiser le temps de trajet total de véhicule et à minimiser le temps d'attente des clients. Les résultats obtenus montrent que le modèle à vitesse variable basé sur des profils historiques présente un intérêt légitime pour prendre en compte les fluctuations de la vitesse d'un véhicule. La prise en compte de cette connaissance est donc significative.

Le quatrième chapitre présente une étude basée sur ce modèle de temps de trajet dépendant du temps utilisé pour la résolution du VRPTW dynamique c'est-à-dire avec des fenêtres de temps. Nous utilisons cette fois ci une approche évolutionniste qui utilise l'algorithme génétique comme méthode de résolution de type métaheuristique. Une plateforme Java a été modifiée pour intégrer le modèle de temps de trajet dépendant du temps. Deux types d'évènements sont issus de ce problème dynamique : L'évènement « **arrivée d'un nouveau client** » et l'évènement « **fin de service d'un client** ». Nous présentons l'architecture logicielle du prototype réalisé pour gérer les tournées de véhicules et simuler les évènements à traiter pendant la journée de service. Les solutions respectives obtenues pour le modèle à vitesse constante et le modèle à vitesse variable sont ensuite évaluées et comparées. Nous remarquons que le modèle à vitesse variable donne des

résultats meilleurs que le modèle à vitesse constante lorsque l'on replonge les solutions de celui-ci dans un environnement à vitesse variable.

Dans le cinquième chapitre, nous voulons prendre en compte les congestions du trafic routier grâce à des informations remontées en temps réel au planificateur des tournées de véhicule. Nous proposons un modèle de prévision simple pour calculer dynamiquement un nouveau profil de vitesse variable qui prend en compte les variations du trafic routier. Ce modèle a été intégré dans la plate forme Java de simulation AG pour évaluer son apport.

Nous avons du modifier profondément la plate forme de simulation pour y considérer de nouveaux événements comme : l'évènement « **vitesse mesurée** » qui remonte en temps réel la vitesse d'un véhicule, l'évènement « **arrivée chez un client** » qui indique l'arrivée effective d'un client en se basant sur des vitesses réelles, l'évènement « **arrivée d'un nouveau client** » et l'évènement « **fin de service d'un client** ». Trois types de profils de vitesses ont été définis pour ce faire : le profil de vitesse statique basé sur des historiques, le profil de vitesse prévisionnelle basé sur un modèle de prévision et le profil de vitesse réelle basé sur un relevé des vitesses réelles du véhicule. Différentes distributions de congestion ont été définies pour évaluer et comparer les apports respectifs de chacun des modèles. L'analyse des résultats montre que notre modèle de prévision temps réel fournit de bonnes performances pour le problème DVRPTW. Par rapport à un modèle sans remontée d'informations trafic routier.

Une conclusion de ces travaux est donnée, puis des perspectives terminent ce mémoire de thèse.

Remerciements

Les travaux présentés dans ce papier entre dans le cadre d'un projet TAT-MOST du CPER subventionné par l'Etat, la Région Nord-Pas de Calais et le FEDER.

Chapitre 1

Le problème de routage de véhicules et ses principales extensions

Sommaire

1.1	Introduction	7
1.2	Formulation	8
1.3	Variantes du problème VRP	10
1.4	Domaines d'applications	12
1.5	Les méthodes de résolution du VRP	12
1.5.1	Les méthodes exactes	13
1.5.2	Les heuristiques	15
1.5.3	Les métaheuristiques	17
1.6	Le problème d'élaboration dynamique de tournées de véhicules DVRP	26
1.6.1	Introduction	26
1.6.2	Une classification des problèmes VRP dynamiques	27
1.6.3	Le degré de dynamisme	32
1.7	Conclusion	34

1.1 Introduction

Le problème de routage de véhicules (VRP : Vehicle Routing Problem) aussi appelé Problème de Tournées de Véhicules (PTV) est un nom générique donné à une classe de problèmes dans lesquels un certain nombre de clients doit être visités par un certain nombre de véhicules (i.e. tournées de véhicules). Ces tournées doivent être organisées de façon à optimiser un certain nombre de critères tels que par exemple, la distance totale parcourue et les temps d'attente des clients. Il y a de nombreuses variantes de ce problème intégrant une ou plusieurs des contraintes suivantes : capacité des véhicules, périodes de temps pour visiter les clients, limites technologiques du système ou des situations géographiques des clients.

Le problème de base de tournées de véhicules peut se définir ainsi :

Etant donné un graphe dont les nœuds représentent les clients et les arcs représentent les routes (i.e. lien de communication entre les clients), le problème consiste à concevoir des itinéraires pour un ensemble des véhicules disponibles, afin d'une part de respecter les contraintes et d'autre

part d'optimiser une fonction objectif donnée [Eilon et al. 1971]. Un exemple est donné en figure 1.1. Il décrit l'instance d'un VRP servant 9 clients sur 3 tournées.

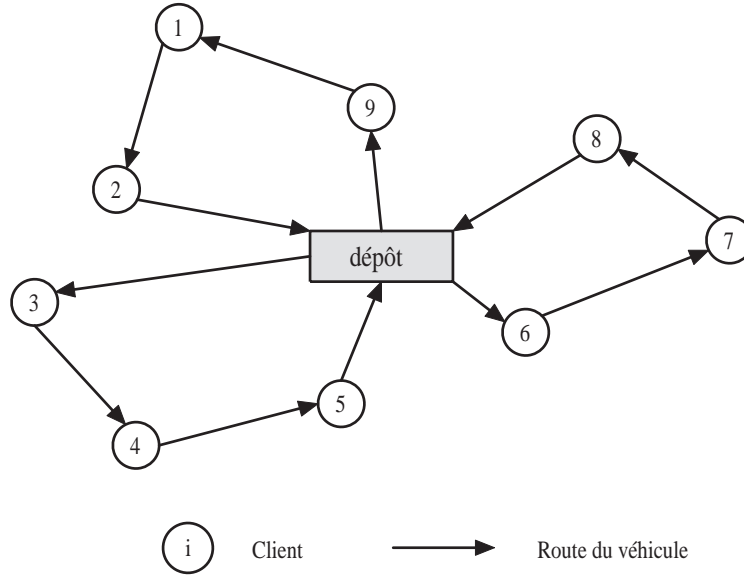


FIG. 1.1 – Le problème VRP

L'objectif de ce chapitre est d'introduire le concept général et la résolution du problème VRP. Il est décomposé en deux parties principales. La première partie contient l'introduction au problème VRP, sa formulation mathématique, puis les principales variantes du VRP rencontrées dans la littérature et enfin les méthodes utilisées pour la résolution de ce problème. La deuxième partie est consacrée au problème du VRP de type dynamique qui est au centre de nos préoccupations.

1.2 Formulation

La version standard du VRP considère une contrainte de capacité (CVRP, Capacited Vehicle Routing Problem). Il peut être représenté sous la forme d'un graphe orienté et valué $G = (N, A)$ où :

- N représente les positions des clients et du dépôt,
- A représente les arcs entre deux clients $i, j \in N$.

Plus spécifiquement, nous avons un ensemble $C = \{1, \dots, n_c\}$ de clients qui doivent obtenir une livraison de marchandise provenant du dépôt. L'ensemble des positions de ces clients ou nœuds est défini par l'ensemble $N = C \cup \{0, n_c + 1\}$ où 0 et $n_c + 1$ représentent le dépôt (aller et retour). Une demande positive de produit d_i est associée à chaque client i appartenant à C .

Une flotte de véhicules $V = \{1, \dots, n_v\}$ est disponible au dépôt et chaque véhicule possède la même capacité (flotte homogène) Q telle que $Q \geq \max d_i, \forall i \in N$. Pour tous les clients i et $j, \forall i, j \in N$, nous connaissons le coût $c_{i,j}$ de transport direct entre i et j (proportionnel à la distance à parcourir). Pour trouver l'ordre de visite des clients, nous définissons les variables de décisions comme suit :

$$x_{i,j}^v = \begin{cases} 1 & \text{si le véhicule } v \in V \text{ visite le client } j \text{ après le client } i, \\ 0 & \text{sinon.} \end{cases}$$

En définissant y_i comme étant la charge résiduelle du véhicule après avoir desservi le client $i \in C$. Il nous est possible d'écrire formellement le modèle de VRP. Il s'agit d'optimiser la fonction suivante :

$$\text{Min} \sum_{v \in V} \sum_{i \in N} \sum_{j \in N} c_{i,j} x_{i,j}^v \quad (1.1)$$

Avec les contraintes :

$$\sum_{v \in V} \sum_{j \in N} x_{i,j}^v = 1, \quad \forall i \in C \quad (1.2)$$

$$\sum_{i \in N} x_{i,h}^v - \sum_{j \in N} x_{h,j}^v = 0, \quad \forall h \in C, v \in V \quad (1.3)$$

$$\sum_{j \in N} x_{0,j}^v = 1, \quad \forall v \in V \quad (1.4)$$

$$\sum_{j \in N} x_{j,n+1}^v = 1, \quad \forall v \in V \quad (1.5)$$

$$x_{i,j}^v = 1 \Rightarrow y_i - d_j = y_j, \quad \forall i, j \in N, v \in V \quad (1.6)$$

$$y_0 = Q, \quad 0 \leq y_i, \quad \forall i \in C \quad (1.7)$$

$$x_{i,j}^v \in \{0, 1\}, \quad \forall i, j \in N, v \in V \quad (1.8)$$

La fonction de coût de la solution $X = (x_{i,j}^v), \forall i, j \in N, \forall v \in V$ est définie par :

$$\text{coût}(X) = \sum_{v \in V} \sum_{i \in N} \sum_{j \in N} c_{i,j} x_{i,j}^v \quad (1.9)$$

Le nombre de véhicules utilisés par la solution X , est défini par :

$$\text{Nb véhicules}(X) = \sum_{v \in V} \sum_{j \in C} x_{0,j}^v \quad (1.10)$$

La fonction objectif (équation 1.1) représente le nombre de véhicules utilisés pour les trajets effectués et la somme des coûts s'y rapportant.

La formulation du problème nécessite de satisfaire certaines contraintes :

- L'équation 1.2 assure qu'on part une et une seule fois de chaque client, avec un seul véhicule.
- L'équation 1.3 assure que le véhicule qui arrive chez un client est le même que celui qui part de ce client.
- L'équation 1.4 assure que chaque véhicule ne sort qu'une seule fois du dépôt.

- L'équation 1.5 assure le retour unique au dépôt pour chaque véhicule (ou tournée). Il n'y a pas de restriction sur le nombre de véhicules, mais un coût c_v est affecté à chaque véhicule utilisé. On impose une valeur de c_v suffisamment grande pour réduire principalement le nombre de véhicules et pour réduire au minimum dans un deuxième temps les coûts de transport.
- Les équations 1.6 et 1.8 définissent les contraintes de capacité et d'intégrité.
- Les équations 1.9 et 1.10 sont des fonctions de mesure qui permettent respectivement de quantifier la solution selon la distance totale parcourue, ainsi que le nombre de véhicules utilisés.

Le problème d'élaboration de tournées de véhicules est un problème NP-difficile, c'est-à-dire qu'il n'existe pas à ce jour un algorithme déterministe pouvant résoudre ce problème en temps polynomial [Garey et al. 1979]. Pour des problèmes comportant un grand nombre de clients (> 100 clients), des méthodes approchées sont nécessaires pour les résoudre.

1.3 Variantes du problème VRP

Le tableau 1.1 présente les caractéristiques des extensions du problème de tournées de véhicules qui permettent de décrire de nombreuses situations réelles.

Caractéristiques	Options possibles
Nombre de véhicules disponibles	- un - plusieurs
Type de véhicule	- homogène - hétérogène
Capacité de véhicule	- finie - infinie
Dépôts	- un - plusieurs
Demandes des clients	- statiques (connues en avance) - dynamiques (apparaissent au cours de temps) - stochastiques (les demandes suivent des lois aléatoires) - fenêtre de temps.
Service proposé	- ramassage ou livraison - ramassage et livraison - ramassage avant livraison
Période considérée	- jour - semaine - périodique

TABLEAU 1.1 – Caractéristiques du problème VRP

Ces caractéristiques permettent de décrire les diverses variantes du VRP qui apparaissent dans la littérature :

1. **VRPTW** (Vehicle Routing Problem with Time Windows) : le problème d'élaboration de tournées de véhicules avec fenêtres de temps est le problème le plus étudié. C'est un

problème de VRP classique auquel on a ajouté une contrainte supplémentaire : chaque client doit fournir une limite de temps supérieure et inférieure entre lesquelles le service doit commencer [Cordeau et al. 2002]. Nous nous intéresserons plus particulièrement à ce problème dans le cadre de cette thèse. C'est pourquoi nous présenterons plus en détail celui-ci dans la suite.

2. **VRPB** (Vehicle Routing Problem with Backhauls) est une prolongation du VRP classique qui inclut un ensemble de clients à qui des produits doivent être livrés et un ensemble de fournisseurs dont les marchandises doivent être amenées au dépôt. En outre, sur chaque tournée, toutes les livraisons doivent être effectuées avant que les marchandises puissent être ramassées pour éviter de réarranger les charges sur le véhicule [Jacobs-Blecha et al. 1993].
3. **DVRP** (Dynamic Vehicle Routing Problem) : dans le VRP classique, toutes les demandes pour tous les clients sont connues à l'avance (lors de la planification des tournées). Cependant, de nombreux problèmes de notre vie pratique incluent au moins un (ou plusieurs) caractère dynamique comme par exemple l'apparition d'un nouveau client en cours de journée. Dans ce cas, le décideur doit changer la planification des véhicules en réponse aux nouvelles demandes [Kilby et al. 1998a] qui arrivent au fil du temps. Nous allons présenter ce problème en détail dans les sections suivantes car il est au centre de nos travaux de thèse.
4. **SVRP** (Stochastic Vehicle Routing Problem) : le problème VRP est dit stochastique lorsque au moins un élément du problème est aléatoire [Gendreau et al. 1999]. Autrement dit, un élément du problème ne peut être connu avec certitude. Ce peut être les demandes (quantité à livrer ou à ramasser) des clients, le temps ou le coût du transport, ou bien l'ensemble des clients à visiter. Le problème avec demandes stochastiques est le plus étudié de cette catégorie. Il est alors supposé que la demande suit une loi de distribution connue (généralement une loi normale).
5. **PVRP** (Periodic Vehicle Routing Problem) : le problème de tournées de véhicules multi-périodique (PTVMP) consiste à livrer pour un ensemble de clients, la quantité demandée d'un ou plusieurs produits sur un horizon de temps donné [Cordeau et al. 2001]. Dans ce problème, chaque client détient un stock de produits dont il consomme une certaine quantité par jour. La tâche principale est de planifier les jours de livraison pour chaque client sur un horizon de temps prédéterminé et ensuite d'organiser les tournées de la flotte de véhicules afin d'effectuer les livraisons nécessaires. Cette planification doit respecter les contraintes sur les niveaux de stocks à maintenir chez les clients et en même temps elle doit minimiser le coût total de livraison sur l'ensemble de l'horizon considéré. Le problème ainsi présenté est une extension du problème de tournées de véhicules classique par l'ajout de la notion de période et par la nécessité d'intégrer des éléments de la gestion des stocks dans la planification des transports.
6. **MDVRP** (Multi-Depot Vehicle Routing Problem) : dans ce type de problème il y a plusieurs dépôts disponibles qui sont distribués géographiquement. Chaque véhicule part et revient vers son dépôt initial [Fischetti et al. 1999].
7. **TSP** (Traveling Salesman Problem) : le problème du voyageur de commerce est un cas particulier du problème VRP sans contrainte de capacité ; il consiste en la détermination d'un parcours de coût minimal (distance, temps, etc.) pour un seul véhicule partant d'une localité, visitant n autres localités et revenant à son point de départ [Rego et al. 1994].

8. **TRP** (Traveling Repairman Problem) : le problème de la tournée du dépanneur (TRP) est une variante du problème d'élaboration de tournées de véhicule VRP. L'objectif est de minimiser la latence, autrement dit, la somme pondérée des temps de service des clients [Krumke et al. 2003]. Nous nous sommes intéressés également à ce problème dans nos travaux de thèse. Nous présenterons celui-ci plus en détail dans les sections suivantes.
9. **VRPPD** (Vehicle Routing Problem with Pick-up and Delivery) : le problème de ramassage et de livraison (VRPPD) a les mêmes caractéristiques qu'un problème VRP dans lequel chaque client doit fournir deux positions géographiques différentes : la première indique le lieu de ramassage du produit et la seconde indique le lieu de livraison du produit. Ce type de problème introduit une contrainte de précédence : pour chaque tournée, l'opération de ramassage chez un client doit précéder l'opération de livraison [Mechti 1995].
10. **OVRP** (Open Vehicle Routing Problem) : la caractéristique importante de ce problème, qui le distingue du VRP de base, est que les véhicules ne sont pas obligés de retourner au dépôt ou lorsqu'ils sont obligés d'y retourner, ils revisitent les clients qui leurs sont affectés dans l'ordre inverse, par conséquent, les itinéraires des véhicules ne sont pas des chemins fermés mais restent ouverts [Eglese et al. 2005].
11. **VRPHF** (Vehicle Routing Problem with Heterogeneous Fleet) : ce type de problème cherche à satisfaire les objectifs du VRP mais en utilisant une flotte de véhicules de types différents. En fait, nous pouvons distinguer ces véhicules par leurs capacités, leurs vitesses ou leurs coûts de déplacement, etc. [Prins 2002] et [Taillard 1999].
12. **VRPSD** (Vehicle Routing Problem with Split Delivery) : chaque client peut être visité plus d'une fois si cela est nécessaire. Autrement dit, la demande de client peut être divisée sur plusieurs tournées. Contrairement à ce qui est habituellement supposé dans le problème classique (VRP) la demande de chaque client peut être plus grande que la capacité des véhicules [Archetti et al. 2006].
13. **m-VRP** (Vehicle Routing Problem with Limited Number of Vehicles) : dans ce problème, nous cherchons à résoudre le problème VRP en utilisant seulement un nombre limité de véhicules [Lau et al. 2003]. En sachant que ce nombre est supposé illimité (ou très grand) dans le VRP de base et que dans le cas présent, on cherche à le minimiser.

1.4 Domaines d'applications

Le VRP permet de modéliser de nombreux types d'applications dans le domaine du transport et de la distribution. Les exemples donnés dans le tableau 1.2 peuvent concerner la conception, la reconfiguration d'un réseau de transport, et la gestion quotidienne de la collecte et de la livraison de produits. Le VRP avec fenêtres de temps (VRPTW) qui prend en compte des contraintes temporelles, permet d'approcher les problèmes réels d'une manière plus fine et plus réaliste que le problème VRP de base. Cependant, les diminutions incessantes des délais, la recherche des réductions de coûts, les règlements liés à des facteurs sociaux et environnementaux, imposent de prendre en compte dans la modélisation le contexte dynamique qui est une caractéristique incontournable des problèmes réels rencontrés dans le domaine du transport et de la distribution.

1.5 Les méthodes de résolution du VRP

Les méthodes permettant de résoudre le problème VRP peuvent être classées en deux parties. On distingue les méthodes exactes et les méthodes approchées. Ces dernières sont aussi divisées

Secteur économique	Applications
Industrie automobile	Distribution de pièces de rechange
La livraison des produits	Carburant, gaz naturel, béton
Transport de nourriture	Grands détaillants ou petits magasins
La livraison de nourriture aux particuliers	Lait, aliments surgelés , plats préparés livrés à domicile
Vente au détail	La livraison d'appareils
Santé	Médicaments aux pharmacies
Presse	Des journaux et des magazines
Secteur bancaire	La livraison d'argent aux banques, perception d'argent des commerçants et courriers
Secteur public	Poubelles domestiques; poubelles publiques, nettoyage des rues, sablage des routes en hiver
Fabriquant	Organisation des mouvements d'une flotte de robots de transport
Industrie	Approvisionnement en parties et marchandises parmi différents emplacements
Agriculture	Collecte d'animaux, lait, céréales, livraison de nourriture animale
Industrie de transport	Entreprises de collecte et de livraison

TABLEAU 1.2 – Différentes applications du VRP

en deux sous groupes : les heuristiques et les métaheuristiques. Elles sont répertoriées dans la figure 1.2

Les principales méthodes de résolution sont présentées dans l'état de l'art ci-après.

1.5.1 Les méthodes exactes

Les méthodes exactes qui fournissent la meilleure solution sont limitées à de petites instances de problèmes. Dès que le nombre de variables de décision augmente, les temps de résolution avec les méthodes exactes augmentent rapidement. Nous pouvons rappeler que cette complexité de résolution vient du fait que le problème appartient à la classe des problèmes NP-difficiles.

Certaines approches exactes permettent cependant de réduire le nombre de solutions examinées. Il s'agit principalement des approches de type Branch and Bound, ou de type Branch and Cut et celles issues de la programmation dynamique.

1.5.1.1 Les méthodes de recherche arborescente (Branch & Bound)

C'est une méthode qui utilise un arbre de recherche représentant l'espace des solutions et élague des parties de cet arbre [Aronson 1996] pour accélérer la recherche. Elle peut être employée pour trouver les solutions exactes au VRP, mais en général la résolution nécessite un temps exponentiel en fonction de la taille du problème. Elle construit incrémentalement les routes nœud par nœud. Le problème est transformé sous la forme d'un TSP en créant des dépôts artificiels (égal au nombre de véhicules disponibles) et en les ajoutant au problème initial. Un branchement dans l'arbre est créé en incluant ou en excluant un arc d'une route. La difficulté de ces méthodes

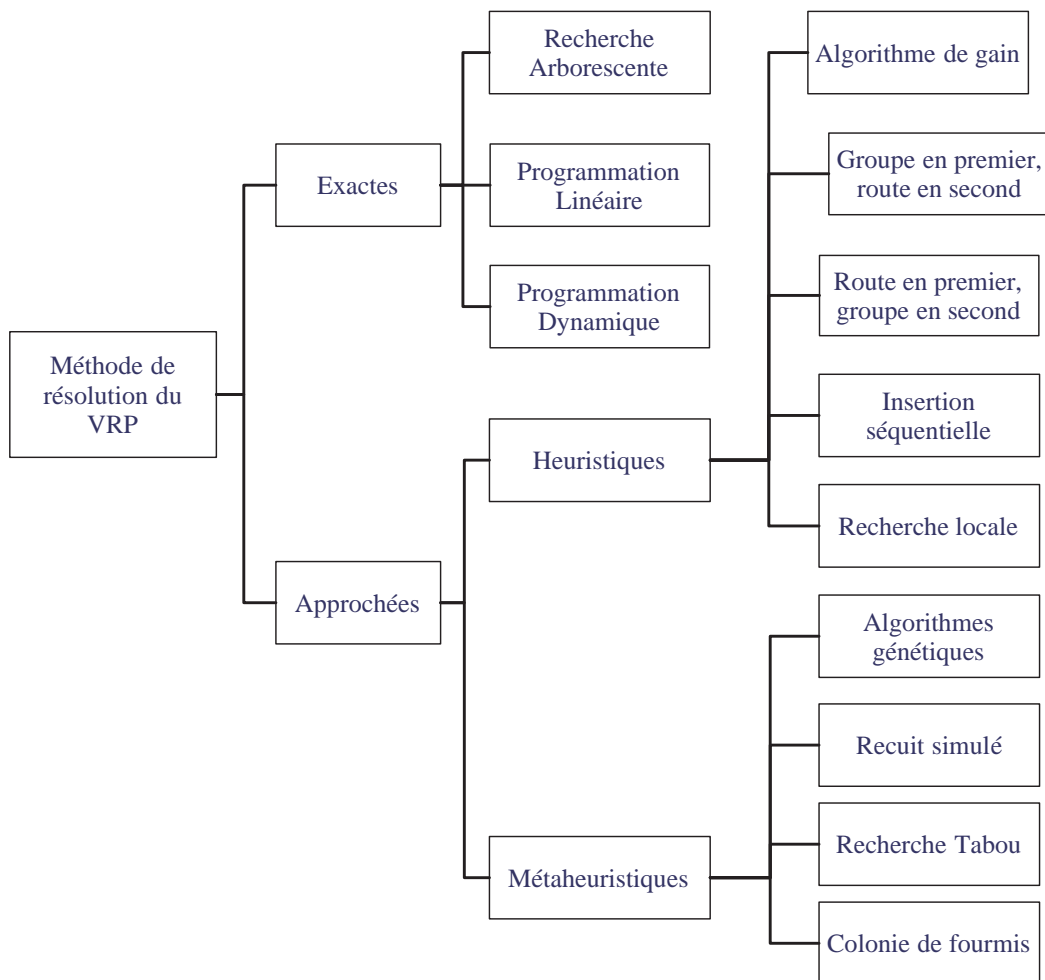


FIG. 1.2 – Les méthodes de résolution du VRP

est l'obtention d'une borne de bonne qualité pour améliorer la phase d'élagage. Des méthodes comme la relaxation Lagrangienne [Miller 1995] donne de meilleures bornes que les techniques de relaxation classiques (contrainte de capacité, contrainte de connectivité) [Laporte et al. 1986].

1.5.1.2 Programmation Linéaire en nombres entiers (Branch & Cut)

Lorsque le nombre de contraintes est trop élevé, les méthodes précédentes ne sont plus applicables. Les méthodes de type *Branch & Cut* constituent une des approches les plus performantes pour résoudre le VRP de façon exacte. Un solveur PL (Programmation Linéaire) est utilisé pour tenter de trouver une solution optimale entière qui respecte les contraintes du problème PLNE (Programmation Linéaire en Nombres Entiers). Dans le cas contraire, une phase de décomposition (i.e. Branch) du problème en 2 sous problèmes est nécessaire et la phase de coupes est relancée sur ces sous problèmes [Toth et al. 2002].

1.5.1.3 Programmation Dynamique

Cette méthode est basée sur le principe de Bellman qui dit : « que toute solution du problème initial de taille N contient la solution optimale du sous-problème de taille $N - 1$ ». Dans la pratique, on commence à résoudre une famille de problèmes de taille 1, puis on passe à l'étape suivante pour résoudre une famille de problèmes de taille 2. Après un certain nombre d'étapes, on retrouve le problème initial de taille N . À chaque étape, des états intermédiaires doivent être considérés et correspondent à une famille de problèmes à résoudre. Pour que l'approche soit faisable, le nombre d'états intermédiaires et le nombre d'étapes doivent être le plus petit possible. Il est parfaitement connu que seul un très petit nombre d'instances de problèmes d'Optimisation Combinatoire peuvent être résolus par la programmation dynamique (10 à 25 clients dans notre cas) [Rego et al. 1994].

1.5.2 Les heuristiques

Le recours aux méthodes approchées est souvent nécessaire pour la résolution de plus grandes instances de problèmes. Plusieurs méthodes ont été proposées dans la littérature. Elles peuvent être divisées en deux classes : d'une part les algorithmes spécifiques à un problème donné, ou heuristiques et d'autre part les algorithmes génériques potentiellement applicables à une grande variété de problèmes d'optimisation connus sous le nom de métaheuristiques.

Des méthodes heuristiques ont été conçues pour explorer seulement des parties de l'espace de recherche, se concentrant dans ces parties qui semblent promettre une amélioration des solutions. De ce fait, le temps requis pour obtenir une solution est réduit et la solution trouvée est souvent non-optimale, mais constitue déjà une bonne amélioration. Une heuristique utilise en général les caractéristiques particulières d'un problème et les exploite afin de trouver une solution intéressante.

1.5.2.1 Algorithme de gain (Clarke & Wright)

L'algorithme des gains de Clarke et Wright [Clarke et al. 1964] développé pour le VRP, est sans doute la méthode la plus connue. Dans cet algorithme, le nombre de véhicules est une variable de décision. Il nécessite le calcul du gain possible entre (i, j) lié à la fusion de routes comme suit :

Au départ on crée n_c routes $(0, i, 0)$ (un client i par tournée, 0 représentant le dépôt) pour $i = 1, \dots, n_c$. Puis on regarde le gain réalisé par des fusions possibles de routes. Pour calculer le gain lié à une fusion de routes on évalue $S_{i,j} = c_{i,0} + c_{0,j} - c_{i,j}$ pour $i, j = 1, \dots, n_c$ et $i \neq j$, $c_{i,j}$ étant le coût de l'arc (i, j) .

On trie ensuite les gains possibles par ordre décroissant et on choisit la fusion réalisant le gain le plus grand. Pour les étapes suivantes, on cherchera à fusionner des routes deux à deux soit au début ou à la fin de tournée. Par exemple si on considère une route $(0, i, \dots, j, 0)$ on va chercher à déterminer le meilleur gain $S_{k,i}$ ou $S_{j,l}$ parmi toutes les fusions possibles qui peuvent être réalisées en formant une route réalisable (respectant les contraintes du problème) par la fusion de la route courante et d'une autre route se terminant par $(k, 0)$ ou commençant par $(0, l)$. L'algorithme s'arrête dès qu'il n'est plus possible de trouver une fusion améliorante. La représentation d'une fusion étant donnée en figure 1.3.

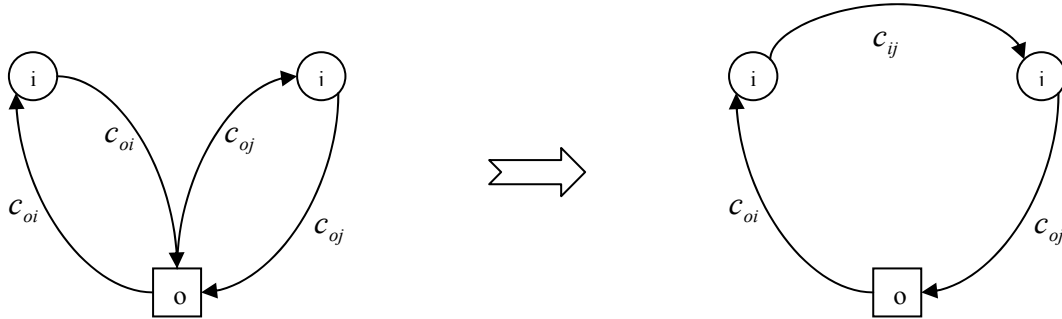


FIG. 1.3 – Fusion dams (Clarke & Wright)

1.5.2.2 Méthode d'insertion séquentielle

Les algorithmes d'insertion sont constitués de deux phases, la première sélectionne le prochain nœud à insérer et la deuxième réalise l'opération d'insertion.

L'heuristique d'insertion du plus proche voisin choisi séquentiellement et de façon répétitive l'arc le plus court qui n'a pas encore été sélectionné pour étendre le cycle (i.e. route) courant. Plusieurs variantes incluent des critères de sélection plus subtils tel que le plus grand angle formé entre les arcs. Pour sa part, l'insertion de Mole et Jameson [Mole et al. 1976] construit une seule route à la fois en insérant un arc à la fois, tandis que la construction parallèle de [Christofides et al. 1979] construit alternativement en séquentiel et en parallèle les différentes routes.

1.5.2.3 Algorithme groupe en premier, route en second

Probablement les plus connus, les algorithmes de type "groupe en premier, route en second" se basent sur une approche géométrique permettant de former des groupes de clients pour ensuite trouver des routes à l'intérieur de ces groupes [Gillett et al. 1974]. La figure 1.4 représente cet algorithme : on effectue un balayage circulaire autour du dépôt, afin de former des groupes de nœuds voisins ayant des angles proches.

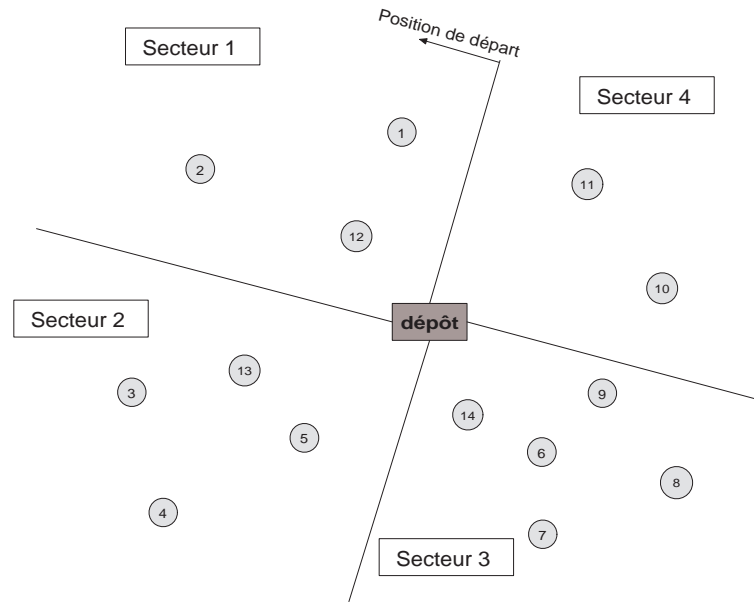


FIG. 1.4 – Groupe en premier, route en second

1.5.2.4 Algorithme route en premier, groupe en second

[Beasley 1983] a utilisé ce genre d'algorithme qui effectue d'abord la construction de cycles comportant de très grandes routes généralement non réalisables. Ces grands parcours sont, par la suite, divisés en parcours plus petits réalisables, formant ainsi des solutions au VRP.

1.5.2.5 Algorithme de recherche locale

L'application de la recherche locale permet de trouver de meilleures solutions dans un voisinage immédiat si le minimum local n'a pas été atteint [Breedam 1996]. Elle consiste à faire des permutations entre arcs à l'intérieur de la même tournée ou entre tournées. Les recherches locales les plus connues sont de la famille des $\lambda - Opt$ ($2 - Opt$, $3 - Opt$, $n - Opt$).

Dans la méthode heuristique d'amélioration de tournée $\lambda - Opt$ ($\lambda = 2, 3, 4, \dots, n$) proposée par [Lin 1965], il s'agit d'enlever λ arcs pour remettre les chaînes associées dans la meilleure combinaison possible. Une variation intéressante est le $\alpha - \lambda - Opt$ [Lin 1965], où seulement les α plus courts gains sont explorés.

1.5.3 Les métaheuristiques

Les métaheuristiques se sont développées de manière considérable ces trente dernières années. Elles partagent en général trois particularités :

- Elles partent d'une ou plusieurs solutions initiales générées aléatoirement dans la plupart des cas, ou construites à partir d'une heuristique dans certains cas.
- Elles mémorisent des solutions ou des caractéristiques des solutions visitées durant le processus de recherche afin de ne pas tomber dans des solutions déjà visitées ou dans un optimum local.

- Elles utilisent une procédure qui permet de créer une nouvelle solution à partir de la solution courante et des informations mémorisées. Il s’agit d’un élément essentiel de la recherche car ceci permet une meilleure exploration de l’espace des solutions.

On y distingue les algorithmes génétiques, le recuit simulé, la recherche tabou, la recherche par dispersion et l’optimisation des colonies de fourmis.

1.5.3.1 Les algorithmes génétiques

Les principes fondamentaux de ces algorithmes ont été exposés par [Holland 1975]. Ces algorithmes s’inspirent du fonctionnement de l’évolution naturelle, notamment la sélection de Darwin, et de la reproduction selon les règles de Mendel. La sélection naturelle, que Darwin appelle “élément propulseur” de l’évolution, favorise les individus d’une population qui sont le mieux adaptés à un *environnement*. La sélection est suivie de la procréation. Elle est réalisée à l’aide de croisements et de mutations au niveau du patrimoine génétique des individus (ou “génotypes”) qui est constitué d’un ensemble de gènes. Ainsi deux individus “parents”, qui se croisent, transmettent une partie de leur patrimoine génétique à leurs descendants. Le génotype de l’enfant fait que celui-ci est plus ou moins bien adapté à l’environnement. S’il est bien adapté, il a une plus grande chance de se reproduire dans la génération future. Au fur et à mesure des générations, on sélectionne les individus les mieux adaptés, et l’augmentation du nombre des individus bien adaptés fait évoluer la population entière vers un optimum (local ou global).

Dans les algorithmes génétiques, nous essayons de simuler le processus d’évolution d’une population. Nous partons d’une population de N solutions du problème représentées par des individus (chromosomes). Cette population choisie aléatoirement est appelée population parent. Le degré d’*adaptation* d’un individu à l’environnement est exprimé par la valeur de la fonction coût $f(x)$ (ou *fitness*), où x est la solution que l’individu représente. Nous disons qu’un individu est d’autant mieux adapté à son environnement, que le coût de la solution qu’il représente est plus faible (pour un problème de minimisation). Au sein de cette population, intervient alors la sélection au hasard d’un ou deux parents qui produisent une nouvelle solution, à l’aide des *opérateurs génétiques*, tels que le *croisement* et la *mutation*. La nouvelle population, obtenue par le choix de N individus parmi les parents et les enfants de l’étape courante, est appelée génération suivante. En itérant ce processus, nous produisons une population plus riche en individus mieux adaptés. L’algorithme (1.1) montre le pseudo-code de l’algorithme génétique de base où, POP représente la population courante et POP’ représente la population des individus générés par l’étape de reproduction.

Cet algorithme comporte trois phases distinctes :

1. La sélection de la population d’individus la mieux adaptée pour contribuer à la reproduction de la génération suivante (version artificielle de la sélection naturelle) ; elle peut être mise en œuvre sous plusieurs formes algorithmiques.
2. La phase de reproduction, qui exploite essentiellement les opérateurs de croisement et de mutation sur les individus sélectionnés précédemment
3. La stratégie de remplacement des populations parent et enfant pour la génération suivante. Elle pourra être mise en œuvre sous plusieurs formes.

Codage Chaque individu de la population est une solution potentielle, cet individu est représenté par un *chromosome*. Ce chromosome est constitué de *gènes* qui peuvent prendre des

Algorithme 1.1 Pseudo-code d'un algorithme génétique de base

POP, POP' : deux tableaux de taille N
 Initialiser la population POP
 Evaluer la population POP
 POP' est initialement VIDE
 Rechercher $x \mid f(x) = \min_i[f(x_i)], 1 \leq i \leq N$
 $f_{min} = f(x)$
 $x_{min} = x$
JUSQU'À conditions d'arrêt satisfaites
 Evaluer la population POP
 REPETER //phase de reproduction génétique
 sélection des individus
 croisement des individus
 mutation des individus
 JUSQU'À POP' soit remplie par les nouveaux individus
 Sélectionner la nouvelle population à partir de (POP, POP')
 Rechercher $x \mid f(x) = \min_i[f(x_i)], 1 \leq i \leq N$
 SI $f(x) < f_{min}$
 $f_{min} = f(x)$
 $x_{min} = x$
 FIN DE SI
REPETER
Résultat (x_{min}, f_{min})

valeurs appelées *allèles*. Il existe plusieurs manières de coder une solution au problème VRP (voir chapitre 4).

Opérateurs de reproduction La phase de reproduction exploite principalement deux opérateurs : **le croisement** et **la mutation**. Elle comporte aussi l'opération de sélection et l'opération de production de la génération suivante, à partir des populations parent et enfant.

– *Sélection*

La sélection consiste à choisir les individus qui vont participer à la reproduction de la population future. La fonction de sélection choisit, de façon déterministe ou selon une méthode probabiliste (roulette, tournoi, etc.), un individu pour qu'il participe à l'étape de reproduction pour former la nouvelle population.

– *Croisement*

Le principal opérateur agissant sur la population des parents est le croisement, qui est appliqué avec une certaine probabilité appelée taux ou probabilité de croisement P_c (typiquement proche de l'unité). Le croisement consiste à choisir deux individus tirés au hasard dans la population courante, et à définir aléatoirement un ou plusieurs points de croisement. Les nouveaux individus sont alors créés en interchangeant les différentes parties de chaque chaîne. Cet opérateur permet de bien **exploiter** le domaine de variation des individus, et de diriger la recherche vers des régions intéressantes de l'espace d'étude en utilisant la connaissance déjà présente dans la population courante.

– *Mutation*

L'opération de mutation protège les algorithmes génétiques des pertes prématurées d'informations pertinentes. Elle permet d'introduire une certaine information dans la population, qui a pu être perdue lors de l'opération de croisement. Ainsi elle participe au maintien de la diversité, utile à une bonne **exploration** du domaine de recherche. L'opérateur de mutation s'applique avec une certaine probabilité, appelée taux ou probabilité de mutation P_m , typiquement faible.

– *Élitisme*

Cette méthode consiste à copier un ou plusieurs des meilleurs chromosomes dans la nouvelle génération avec un pourcentage prédéfini appelé "*taux d'élitisme*". Ensuite, on génère le reste de la population selon l'algorithme de reproduction usuel. Cette méthode améliore considérablement les algorithmes génétiques, car elle évite de perdre les meilleures solutions trouvées jusqu'ici [Davis 1991].

L'efficacité de l'algorithme génétique dépend fortement du choix de ces paramètres de contrôle (les paramètres ci-dessus), dont le réglage reste très empirique ; elle dépend aussi fortement de la représentation choisie pour le codage des solutions et des opérateurs de reproduction utilisés. Les principaux paramètres sont les suivants :

1. la taille de la population N ,
2. le type et taux de croisement,
3. le type et le taux de la mutation,
4. le type de la sélection,
5. le taux d'élitisme.

Il existe beaucoup de travaux relatifs au VRP et à ses extensions ; parmi eux nous pouvons citer les travaux de [Rego et al. 1994] qui présentent plusieurs méthodes de résolution (heuristiques et métaheuristiques) du problème VRP et TSP de [Pereira et al. 2002] qui proposent un

algorithme génétique avec une nouvelle représentation chromosomique du VRP et de [Prins 2004] et [Baker et al. 2003] qui utilisent l'algorithme génétique comme méthode de résolution avec aussi une nouvelle représentation des solutions.

1.5.3.2 Recuit simulé

Cette méthode trouve ses origines dans la *thermodynamique*. Elle est issue d'une analogie entre le phénomène physique de refroidissement lent d'un corps en fusion, qui le conduit à un état solide de basse énergie. Il faut abaisser lentement la température, en marquant des paliers suffisamment longs pour que le corps atteigne *l'équilibre thermodynamique* à chaque palier de température. Pour les matériaux, cette énergie basse se manifeste par l'obtention d'une structure régulière comme dans les cristaux ou l'acier. L'analogie exploitée par le recuit simulé consiste à considérer une fonction f à minimiser comme fonction d'énergie. Une solution x peut être considérée comme un état donné de la matière dont $f(x)$ est l'énergie. Le recuit simulé exploite généralement le critère défini par l'algorithme de Metropolis (voir le pseudo-code dans l'algorithme 1.2) pour l'acceptation d'une solution obtenue par perturbation de la solution courante. Pour une "température" T donnée, à partir d'une solution courante x , on considère une transformation élémentaire qui changerait x en $s(x)$. Si cette perturbation induit une diminution de la valeur de la fonction objectif f , $\Delta f = f(s(x)) - f(x) < 0$, elle est acceptée, ceci dans le cas d'une minimisation. Dans le cas contraire, si $\Delta f = f(s(x)) - f(x) \geq 0$, la perturbation est acceptée avec une certaine probabilité $p = \exp \frac{-\Delta f}{T}$ qui décroît avec le temps [Thangiah et al. 1994].

Algorithme 1.2 Pseudo-code de la règle de Metropolis

```

SI  $f(s(x)) \leq f(x)$ 
     $f(x) = f(s(x))$ 
     $x = s(x)$ 
SINON
     $p = \exp \frac{-\Delta f}{T}$ 
     $r =$  solution aléatoire entre  $[0, 1]$ 
    SI  $r \leq p$ 
         $f(x) = f(s(x))$ 
         $x = s(x)$ 
    FIN SI
FIN SI

```

Le paramètre de contrôle T de cet algorithme est la "Température" du système, qui influe sur la probabilité d'acceptation d'une solution moins bonne. A une température élevée, la probabilité d'acceptation d'un mouvement quelconque tend vers 1 : presque tous les changements sont acceptés. L'algorithme équivaut alors à une marche aléatoire dans l'espace des solutions. Cette température est diminuée lentement au fur et à mesure du déroulement de l'algorithme pour simuler le processus de refroidissement des matériaux, et sa diminution est suffisamment lente pour que l'équilibre thermodynamique soit maintenu (voir le pseudo-code de l'algorithme du recuit simulé 1.3) :

La génération des voisins $S(x)$ se fait par la technique de la recherche locale déjà présentée dans ce chapitre.

Algorithme 1.3 Pseudo-code de l'algorithme du recuit simulé

```

 $x$  = solution aléatoire
 $f_{min} = f(x)$ 
 $x_{min} = x$ 
Initialiser  $T$  à une température assez élevée
REPETER
  REPETER
    générer un voisin  $s(x) \in \text{voisinage } S(x)$ 
    appliquer la règle de Metropolis
    SI  $f(x) < f_{min}$ 
       $f_{min} = f(x)$ 
       $x_{min} = x$ 
    FIN DE SI
  JUSQU'A équilibre thermodynamique atteint
  faire décroître la température  $T$ 
JUSQU'A conditions d'arrêt satisfaites

```

L'efficacité du recuit simulé dépend fortement du choix de ses paramètres de contrôle, dont le réglage reste lui aussi très empirique. Les principaux paramètres de contrôle sont les suivants :

- la valeur initiale de la température T ,
- la fonction de décroissance de la température T ,
- le critère de changement du palier de température,
- les critères d'arrêt.

Le rôle de la température T au cours du processus de recuit simulé est très important. Une forte décroissance de température risque de piéger l'algorithme dans un minimum local, alors qu'une faible décroissance au début du processus entraîne une convergence très lente de l'algorithme. Un compromis pour adapter la décroissance de la température à l'évolution du processus consiste à utiliser une variation logarithmique. La loi logarithmique de décroissance de la température, qui assure la convergence théorique du recuit simulé, est la suivante :

$$T_k = \frac{\mu}{\text{Log}(1 + k)}$$

Où k est le nombre de paliers de température effectués, et μ est une constante positive. En pratique, on adopte souvent une décroissance géométrique $T_{k+1} = \alpha.T_k$, avec ($0 < \alpha < 1$), car la loi précédente induit un temps de calcul prohibitif.

Nous trouverons dans [Rego et al. 1994] une utilisation de cette méthode pour résoudre le problème VRP.

1.5.3.3 La méthode de recherche tabou

La recherche tabou **RT** est une métaheuristique originalement développée par [Glover 1986]. Elle est basée sur des idées simples, mais elle est néanmoins très efficace. Cette méthode combine une procédure de recherche locale avec un certain nombre de règles et de mécanismes permettant à celle-ci de surmonter l'obstacle des optima locaux, tout en évitant les problèmes de cycle. Elle a été appliquée avec succès pour résoudre de nombreux problèmes difficiles d'optimisation combinatoire dont le problème de routage de véhicules (VRP).

Dans une première phase, la méthode de *recherche tabou* part d'une solution quelconque x appartenant à l'ensemble de solutions X , puis on se déplace vers une solution $s(x)$ située dans le voisinage $S(x)$ de x . L'algorithme va explorer itérativement un sous-ensemble de l'espace de solutions X . Afin de choisir le meilleur voisin $s(x)$ dans le voisinage $S(x)$, l'algorithme évalue la fonction objectif f en chaque point $s(x)$, et retient le voisin qui améliore la valeur de la fonction objectif f ou si ce n'est pas possible celui qui la dégrade le moins.

L'originalité de la méthode de recherche tabou par rapport aux méthodes de recherches locales réside dans le fait que l'on retient le meilleur voisin, même si celui-ci est plus mauvais que la solution d'où l'on vient. Ce caractère autorisant les dégradations de la fonction objectif évite à l'algorithme d'être piégé dans un minimum local. Cependant il induit un risque de création de cycles. En effet, lorsque l'algorithme quitte un minimum local par acceptation de la dégradation de la fonction objectif, il peut revenir sur la même trace déjà suivie, à l'itération suivante. Pour régler ce problème, l'algorithme a besoin d'une mémoire pour conserver la trace des dernières meilleures solutions déjà visitées. Ces solutions sont déclarées *tabous*, d'où le nom de la méthode. Elles sont stockées dans une liste de longueur L donnée, appelée *liste tabou*. Une nouvelle solution n'est acceptée que si elle n'appartient pas à cette liste tabou. Ce critère d'acceptation d'une nouvelle solution évite la formation de cycles de longueur inférieure ou égale à la longueur de la liste tabou. Il dirige l'exploration de la méthode vers des régions du domaine de solutions non encore visitées. Le pseudo-code de l'algorithme tabou classique est présenté dans l'algorithme 1.4.

Algorithme 1.4 Pseudo-code de la méthode de recherche tabou

```
 $x$  = solution aléatoire  
 $f_{min} = f(x)$   
 $x_{min} = x$   
 $TABOU$  est VIDE // liste de solutions  $s(x)$ , de longueur  $L$   
REPETER  
  Générer 1 solution TEL QUE  $s_i(x) \in \text{voisinage } S(x)$  et  $\{x, s_i(x)\} \notin TABOU$   
   $f(s(x)) = \min_i [f(s_i(x))]$ ,  $1 \leq i \leq N$   
  Ajouter  $(\{x, s_i(x)\}, TABOU)$   
   $x = s(x)$   
  SI  $f(x) < f_{min}$   
     $f_{min} = f(x)$   
     $x_{min} = x$   
  FIN DE SI  
JUSQU'A conditions d'arrêt satisfaites
```

On élimine à chaque itération la solution tabou la plus ancienne, en la remplaçant par la nouvelle solution retenue. Mais le codage d'une telle liste est encombrant, car il faudrait garder en mémoire tous les éléments qui définissent une solution. Pour pallier cette contrainte, il est possible de remplacer la liste tabou de solutions interdites par une liste de "*transformations interdites*", en interdisant la transformation inverse d'une transformation faite récemment.

Cette méthode métaheuristique est largement utilisée pour le problème VRP [Rego et al. 1994], [Gulay et al. 1999] et [Rego 2001]. Nous trouvons aussi une version parallèle qui traite ce problème [Caricato et al. 2003].

1.5.3.4 La colonie de fourmis

Les algorithmes de fourmis sont issus des travaux de Colorni, Dorigo et Maniezzo [Colorni et al. 1992] au début des années 90. Ils reposent sur une analogie avec le comportement collectif des fourmis à s'organiser pour la recherche de nourriture.

Schématiquement, les fourmis explorent leur environnement immédiat en laissant derrière elles des traces chimiques, la *phéromone*. Elles se servent de la phéromone pour se guider et tendent naturellement à suivre les traces existantes. En l'absence de phéromone, leur exploration devient complètement aléatoire. Elles possèdent donc une vision très limitée de leur environnement. Une fois la nourriture trouvée, elles se servent des traces qu'elles viennent de déposer pour retrouver le chemin du retour vers le nid. Durant le trajet, elles laissent de nouveau sur leur passage de la phéromone, en quantité proportionnelle à l'intérêt de la source de nourriture (pondération). Le chemin est donc fortement imprégné de phéromones et constitue une piste de choix pour la colonie de fourmis. Plus la source de nourriture a été jugée intéressante, plus les fourmis auront tendance à suivre ce chemin. Ainsi, peu à peu, les traces vers les sources de nourriture seront de plus en plus marquées. Il est intéressant de noter que les fourmis ont la remarquable propriété de suivre naturellement le chemin le plus court vers la source de nourriture. En effet, [Goss et al. 1989] a observé qu'en présence de deux routes possibles vers la nourriture, les fourmis adoptent rapidement la plus courte. Ceci est dû au fait que le retour à l'embranchement des fourmis ayant pris la bonne option s'opère avant celles ayant effectué le mauvais choix. De fait, dans l'intervalle de temps, la bonne route se trouve favorisée par une plus grande concentration de phéromones. Le biais est ensuite renforcé par le phénomène d'évaporation de la phéromone, qui tend à effacer progressivement les traces les moins fréquentées. Colorni, Dorigo et Maniezzo [Colorni et al. 1992] ont adapté ce principe au domaine de l'Optimisation Combinatoire. Pour ce faire, ils ont associé le voisinage du nid à l'espace des solutions. Chaque solution s'apparente à une source de nourriture dont la qualité est fournie par la fonction d'évaluation. Chaque fourmi est assimilée à un processus répétitif de construction de solutions. La construction est guidée par un ensemble de données globales "*la phéromone*". Cet ensemble est typiquement une mémoire sur les attributs de solution, régulièrement mise à jour par les processus de construction (en fonction de la qualité de la solution) et par un mécanisme simulant l'évaporation de la phéromone. Le problème majeur de ce modèle réside dans sa grande sensibilité vis-à-vis du principe de la phéromone. En fait, la qualité des résultats fournis par la métaheuristique dépend très fortement de l'importance accordée à la phéromone. Lorsque la pondération est trop élevée, le mécanisme de construction tend à favoriser la génération de solutions trouvées dans les itérations précédentes. A l'inverse, lorsqu'il est trop faible, il tend à produire des solutions aléatoires.

Le pseudo-code de l'algorithme de la colonie de fourmis est présenté dans l'algorithme 1.5 :

Algorithme 1.5 Pseudo-code de la méthode de colonie de fourmis

1. Initialiser les traces,
 2. Tant qu'un critère d'arrêt n'est pas satisfait, répéter en parallèle pour chacune des p fourmis :
 - (a) Construire une nouvelle solution à l'aide des informations contenues dans les traces et une fonction d'évaluation partielle (i.e. heuristique).
 - (b) Evaluer la qualité de la solution.
 - (c) Mettre à jour les traces.
-

Nous pouvons améliorer la performance en tenant en compte des modifications suivantes qui touchent pratiquement tous les aspects de la métaheuristique :

1. le processus de génération de solutions s'est vu intégrer une phase d'amélioration itérative afin de fournir des solutions de meilleure qualité.
2. l'actualisation des traces a évolué pour permettre l'application explicite de stratégies de fouille ou de prospection. Seules les meilleures solutions donnent lieu à une mise à jour des traces.
3. la coordination des différentes fourmis a été sensiblement améliorée par l'apparition d'une fourmi particulière, la reine. Cette dernière prend en charge la gestion de la mémoire collective de la colonie (les traces) et prend les décisions stratégiques en fonction de l'évolution globale de la recherche. Elle constitue de fait une sorte d'agent "intelligent".

L'application de ces différents aménagements du modèle initial a permis d'augmenter sensiblement les performances de la métaheuristique. Ceci est particulièrement vrai pour le problème VRP [Gambardella et al. 2003].

La complexité du VRP exige des méthodes métaheuristicques pour la plupart des exemples réels. Dans les sections précédentes, nous avons largement examiné l'évolution des méthodologies métaheuristicques pour le problème VRP. La qualité des solutions obtenues avec ces différentes techniques est bien souvent meilleure que les heuristiques traditionnelles. En même temps, les métaheuristicques exigent plus de temps de calcul et sont plus complexes à mettre en application et à calibrer [Bräysy et al. 2001].

Le rapport de recherche du GERAD [Cordeau et al. 2004] compare les 9 meilleures méthodes actuelles pour le VRP. Ces méthodes sont :

1. **GTS** (Granular Tabu Search) [Toth et al. 2003] : Cette méthode utilise une recherche tabou et élimine les tronçons de taille supérieure à un seuil de granularité dans le graphe représentant la région de service.
2. L'heuristique **VRTR** (Variable-length neighbourhood list Record-to-Record) [Li et al. 2004] basée sur l'heuristique des gains de [Clarke et al. 1964] et l'opérateur de recherche locale 2-Opt.
3. **UTSA** (Unified Tabu Search Algorithm) [Cordeau et al. 1997] : Cette méthode utilise principalement la recherche tabou, elle utilise uniquement une solution initiale et à partir de cette solution elle construit la liste tabou (client réinsertion).
4. **VLNS** (Very Large Neighbourhood Search) [Özlem Ergun et al. 2006] : Cette méthode est basée sur l'opérateur de recherche locale 2-Opt, appliqué sur plusieurs routes.
5. **L'algorithme génétique hybride** de [Prins 2004] : Cette méthode hybride utilise un algorithme génétique combiné avec une recherche locale. Le chromosome est représenté par une séquence de clients et un algorithme de plus court chemin est utilisé pour découper de manière optimale cette séquence en tournées. L'opérateur de mutation consiste en une recherche locale (client réinsertion, tronçons réinsertion, 2-Opt, 2-Opt sur plusieurs routes, etc.) appliquée sur l'ensemble des tournées.
6. L'algorithme "**Bone Route**" proposé par [Tarantilis et al. 2002] est basé sur le couplage d'une procédure constructive (la méthode des gains modifiée) pour obtenir une solution initiale, suivie d'une recherche tabou qui utilise l'opérateur 2-Opt pour obtenir les solutions voisines.

7. **AGES** (Active Guided Evolution Strategies) [Mester et al. 2005] combine des stratégies d'évolution avec l'opérateur de recherche locale 2-Opt.
8. **L'algorithme génétique hybride** [Berger et al. 2004] qui est basé sur l'utilisation d'un algorithme génétique couplé avec une recherche locale (2-Opt, etc.).
9. L'heuristique **D-Ants** [Reimann et al. 2004], cette méthode utilise l'heuristique des gains modifiée avec la méthode des colonies de fourmis.

La comparaison entre ces méthodes montre que l'algorithme génétique de [Prins 2004], AGES [Mester et al. 2005] et Bone Route [Tarantilis et al. 2002] sont les meilleures pour toutes les instances des benchmarks traités (les 14 instances de Christofides et les 20 instances à grande taille de Golden).

1.6 Le problème d'élaboration dynamique de tournées de véhicules DVRP

1.6.1 Introduction

D'un point de vue général, les problèmes de décision en temps réel jouent un rôle de plus en plus important dans plusieurs secteurs ou domaines d'application car les technologies de communication et d'information permettent aujourd'hui d'obtenir et de traiter rapidement l'information en temps réel [Séguin et al. 1997]. Plus précisément, un problème d'optimisation peut être considéré comme dynamique si les deux principales conditions suivantes sont vérifiées : [Bianchi 2000].

1. L'information concernant le problème est dépendante du temps (i.e. loi qui dépend du temps).
2. Les solutions sont déterminées et réactualisées dans le temps en fonction de l'arrivée de nouvelles informations, ce qui implique qu'il n'y a pas de solution a priori. La seule détermination a priori concerne la stratégie employée qui spécifie quelles actions doivent être prises en fonction de l'état actuel du système.

Dans le domaine du transport, les systèmes de positionnement globaux (GPS), les systèmes d'information géographiques (GIS), les systèmes intelligents d'autoroutes et autres systèmes embarqués, ont augmenté l'intérêt des méthodes et des stratégies pour la résolution du problème d'élaboration de tournées de véhicules dans un contexte "*en ligne*" où l'échange d'information peut se faire en temps réel avec un outil d'aide à la décision. Par conséquent, les problèmes dynamiques de tournée et d'affectation de véhicules sont apparus comme un intense domaine de recherche dans la communauté de recherche opérationnelle [Haines et al. 1982] [Powell et al. 1995] [Psaraftis 1995]. A titre d'exemple, le caractère dynamique de ces problèmes apparaît lorsqu'un ensemble de véhicules est disponible sur un horizon de temps particulier tandis que les nouvelles demandes de service apparaissent en temps réel. A chaque nouvelle demande, la solution courante peut être modifiée pour intégrer le service de la nouvelle demande aux tournées existantes. Ainsi, dans un problème VRP de type dynamique, toutes les informations concernant la planification ne sont pas forcément connues quand le processus d'élaboration des tournées commence et celle-ci peut changer après que les tournées sont en cours de réalisation.

Par contre, un problème ne peut pas être considéré comme dynamique si :

1. l'information change en suivant une évolution temporelle connue à l'avance,
2. l'information ci-dessus n'est pas connue avec certitude mais avec une certaine probabilité d'apparition connue.

Dans la figure 1.5, nous avons un exemple simple de problème dynamique où deux véhicules doivent servir un ensemble de clients. Les clients statiques sont représentés par des nœuds noirs, alors que les clients dynamiques (i.e. nouveaux clients) sont représentés par les nœuds blancs. Les arcs en trait fin représentent la route planifiée pour chaque véhicule. Les deux arcs en trait plein épais indiquent les trajets en cours des véhicules lorsque les demandes dynamiques sont reçues. Les nouveaux clients doivent être insérés dans les routes déjà construites en tenant compte des clients non encore visités. Les modifications de trajet sont représentées en trait pointillé dans la figure 1.5.

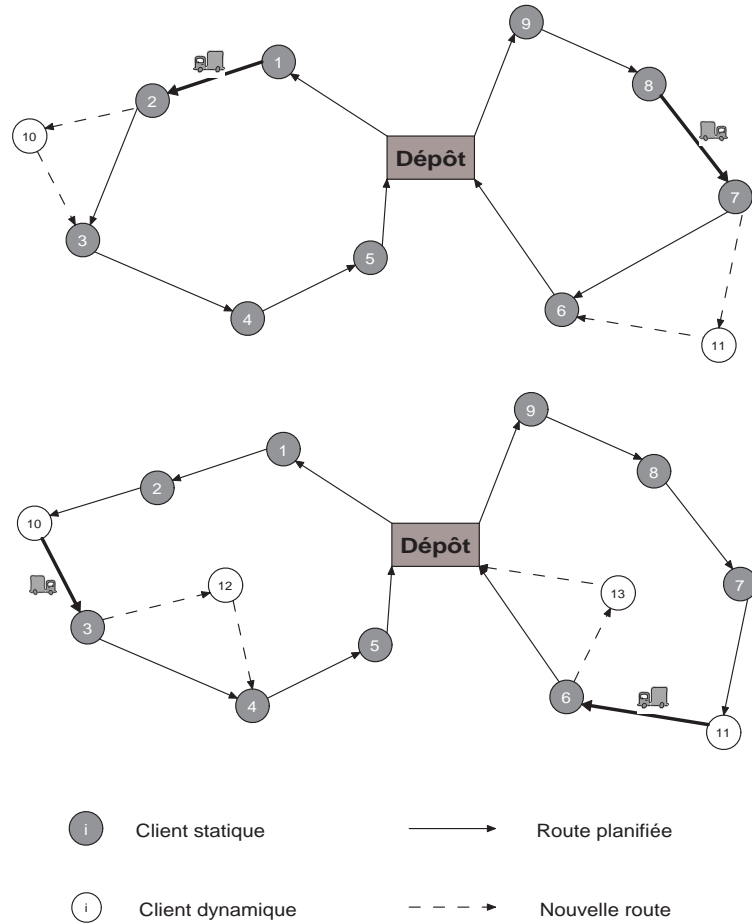


FIG. 1.5 – Le problème DVRP

1.6.2 Une classification des problèmes VRP dynamiques

Quatre types de problèmes de VRP sont distingués dans la littérature [Ghiani et al. 2003] et sont représentés sur le tableau 1.3.

Un problème est dit *statique* si toutes les données nécessaires à la planification ne dépendent pas du temps, sinon il est *dynamique*. Le caractère *déterministe* traduit le fait que toutes les données sont parfaitement connues (i.e. sans incertitude). Le VRP statique peut être déterministe ou stochastique. Le VRP dynamique, également connu sous le nom du VRP en temps réel, peut également être déterministe ou stochastique. [Ghiani et al. 2003] a classé le VRP en quatre types :

- Dans le VRP déterministe et statique dit aussi VRP classique, toutes les données sont connues à l’avance avec certitude et ne dépendent pas du temps.
- Dans les problèmes déterministes et dynamiques, les données sont connues avec certitude et seuls quelques éléments dépendent du temps.
- Dans les problèmes stochastiques et statiques, l’incertitude peut porter sur la quantité de la demande, sur le temps de service chez le client, ou sur le temps de trajet. Le temps n’est pas pris en compte explicitement dans les données du problème.
- Dans les problèmes stochastiques et dynamiques, les données incertaines sont représentées par des variables stochastiques, de plus, des éléments dépendent du temps. Par exemple, dans [Bent et al. 2003], une partie des clients arrivent dans la journée, et le temps d’arrivée des clients suit une certaine loi de distribution.

	Déterministe	Stochastique
VRP statique	<ul style="list-style-type: none"> - Données nécessaires ne dépendent pas du temps - Toutes les données sont connues avant l’exécution 	<ul style="list-style-type: none"> - Données nécessaires ne dépendent pas du temps - Les données sont mises à jour en se basant sur l’information en temps réel
VRP dynamique	<ul style="list-style-type: none"> - Données nécessaires dépendent du temps - Toutes les données sont connues avant l’exécution 	<ul style="list-style-type: none"> - Données nécessaires dépendent du temps - Les données sont mises à jour en se basant sur l’information en temps réel

TABLEAU 1.3 – Classification du VRP

1.6.2.1 VRP statique

Ce type de problème peut être *déterministe* ou *stochastique*. Dans le VRP *statique déterministe* ou VRP classique, toutes les données sont connues à l’avance et le temps n’est pas pris en compte explicitement. Par contre dans le problème *statique stochastique*, l’incertitude sur les données peut être modélisée par des variables stochastiques. L’incertitude peut porter sur la quantité de la demande, le temps de service chez le client, ou le temps de trajet d’un tronçon routier. Si certaines données sont incertaines, il devient impossible de satisfaire les contraintes avec (100%) de réussite. Si l’incertitude affecte uniquement les contraintes mais que la fonction objectif est *déterministe*, il est possible que les contraintes soient satisfaites avec une probabilité donnée ou un taux de réussite donné. Dans une approche générale proposée par [Gendreau et al. 1999], une solution est construite sur la base de ces données incertaines sur la quantité à ramasser ou à livrer. Cette solution représente une liste des clients à visiter par un seul véhicule (une seule tournée). On doit alors prendre une décision, après chaque client servi, qui consiste par exemple soit à retourner au dépôt soit à se déplacer vers le client suivant de la tournée. Une mauvaise décision peut entraîner pendant l’exécution de la tournée, un défaut de capacité et conduire à un retour au dépôt prématuré. (voir la figure 1.6).

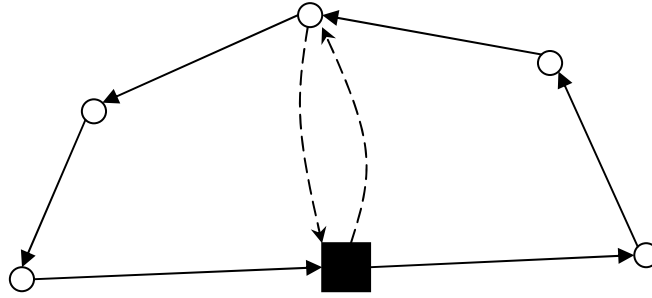


FIG. 1.6 – Le cas d'échec

1.6.2.2 VRP dynamique

Un problème dynamique peut également être *déterministe* ou *stochastique*. Dans les *problèmes dynamiques déterministes*, les données sont connues à l'avance avec certitude et seuls quelques éléments d'information dépendent du temps [Larsen 2001]. Dans des *problèmes dynamiques stochastiques* [Bent et al. 2003], également connus sous le nom du problème VRP en *temps réel*, les données incertaines sont représentées par des variables stochastiques. Par exemple, l'arrivée des demandes des clients peut suivre une distribution de type *loi de Poisson*.

Les événements qui mènent à une modification de la planification peuvent être par exemple :

1. L'arrivée d'une nouvelle demande.
2. La panne d'un véhicule.
3. L'apparition d'une congestion de trafic suite à des travaux ou à un accident sur une route.

Chaque événement doit être traité selon les stratégies choisies par le planificateur de la flotte de véhicules. En général, quand une nouvelle demande est reçue, il doit décider s'il peut la satisfaire le jour même, ou si il doit la retarder pour le jour suivant ou plus radicalement la rejeter définitivement. Si la demande est acceptée, elle est temporairement affectée à une position dans la tournée actuelle d'un véhicule donné. Par conséquent, c'est au moment où le véhicule atteint sa destination courante que le planificateur lui assignera sa prochaine destination.

1.6.2.3 Caractéristiques des problèmes statiques et dynamiques

Nous allons revenir sur les caractéristiques les plus importantes :

- **Le temps** : Il constitue la distinction principale. Certains problèmes sont « open-end », c'est-à-dire, n'ont pas de limites de temps. Une hypothèse de résolution de ces problèmes est qu'un équilibre est atteint par le système avec une valeur qui fluctue autour de cet équilibre. Ces problèmes sont généralement modélisés par des files d'attente. C'est le cas du problème dynamique du réparateur ou DTRP. Beaucoup de problèmes ont un horizon fini, soit un jour ou une semaine et ne demandent pas de condition d'équilibre de tournée comme dans le DVRPTW et le DRIVE.
- **L'information future** : Dans un système dynamique, l'information dans l'état initial peut être connue en partie a priori, et en partie révélée dynamiquement ou mise à jour. Donc, pour un problème dynamique, à un instant donné, il n'y a aucune incertitude relative aux informations reçues à cet instant. Au contraire, l'information future peut être inconnue ou partiellement connue dans une approche stochastique.

- **La fonction objectif** : Elle ne se définit pas simplement dans les problèmes dynamiques. On peut considérer un problème dynamique comme une série de problèmes statiques à résoudre mais en tenant compte de la progression du temps dans la fonction objectif au moyen d'une contrainte temporelle.
- **La stratégie** : Les stratégies utilisées dans le cas dynamique doivent considérer un nouveau sous-problème statique soit à chaque fois qu'un nouveau client apparaît, soit attendre un ensemble de nouveaux clients à traiter. On peut ajouter à cela des contraintes spatiales sur l'ensemble de clients que l'on cherche à regrouper. Dans la littérature, on distingue deux types de travaux : soit on applique des stratégies différentes au problème dynamique posé, soit on fixe une seule stratégie.

Dans le tableau (1.4) nous présentons pour chaque problème dynamique : le nom du problème, les informations qui concernent la partie statique du problème, les informations concernant la partie dynamique, la méthode utilisée pour la résolution et finalement la référence bibliographique.

Problème	Info. A-priori	Info. Dynamiques	Méthode	Référence
DSVRP	m véhicules avec capacités	Quantités des demandes	Markov	[Laporte et al. 1989]
DTRP	1 véhicule sans capacité	nouveaux clients	stratégies (SQM)	[Bertsimas et al. 1991]
DRIVE	- m véhicules hétérogènes, - n dépôts.	nouveaux clients	Branch & Price	[Savlesberg et al. 1991]
DTRP	m véhicules avec et sans capacité	nouveaux clients	stratégies	[Bertsimas et al. 1993b]
DTRP	m véhicules avec et sans capacité	nouveaux clients	stratégies	[Bertsimas et al. 1993a]
VSPTS	- m véhicules avec capacité, - n clients	le temps de trajet	Heuristiques	[Park et al. 1997]
DVRP	- m véhicules avec capacité, - % de clients.	nouveaux clients	Insertion	[Kilby et al. 1998a]
DPDPTW	- m véhicules sans capacité, - % de clients.	nouveaux clients	Recherche Tabou Parallèle	[Gendreau et al. 1998]
DVRPTW	- m véhicules sans capacité, - % de clients.	nouveaux clients	Recherche Tabou Parallèle	[Gendreau et al. 1999]
DTSP	1 véhicule sans capacité	Supprimer et insérer de nouveaux clients	ACO	[Guntsch et al. 2000]
DVRPTW	- m véhicules avec capacité, - % de clients.	nouveaux clients	Insertion	[Larsen 2001]
ADTSPTW	- 1 véhicule sans capacité, - % de clients.	nouveaux clients	Stratégies	[Larsen et al. 2004]
DVRP	- m véhicules sans capacité, - % de clients.	nouveaux clients	Colonie de fourmis	[Montamanni et al. 2003]
DPDPTW	m véhicules avec capacité.	nouveaux clients	- Insertion, - Recherche Tabou.	[Mitrovic-Minic et al. 2004]
TDVRP	- m véhicules avec capacité, - n clients	le temps de trajet	Recherche Tabou Parallèle	[Ichoua et al. 2003]

TABLEAU 1.4: Le problème DVRP et ses extensions dans la littérature

Les abréviations des différents types de problèmes (première colonne du tableau 1.4) sont répertoriées dans le glossaire.

1.6.3 Le degré de dynamisme

Contrairement à un problème statique d'élaboration de tournées de véhicules nous supposons que la performance dépend non seulement du nombre de clients et de la distribution spatiale de ces derniers, mais également du nombre d'évènements dynamiques et des instants où ces évènements ont lieu réellement. Par conséquent, une mesure pour décrire le dynamisme du système est d'une grande utilité quand nous voulons examiner la performance d'un algorithme spécifique dans des conditions variables.

Le but de cette section est de présenter les mesures existantes données par [Larsen 2001], pour décrire le dynamisme d'un problème DVRP avec et sans fenêtres de temps.

1.6.3.1 Dynamisme sans fenêtres de temps (dod)¹

Nous examinons les mesures décrivant le dynamisme d'un problème DVRP sans fenêtres de temps. Dans un tel problème, trois paramètres seulement sont appropriés :

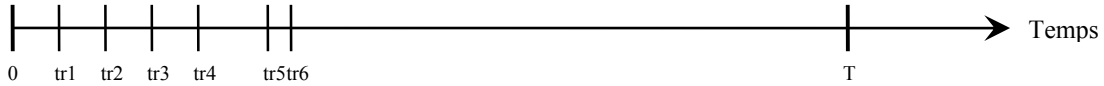
1. Le nombre de clients statiques.
2. Le nombre de clients dynamiques (appelés aussi demandes immédiates).
3. Les instants d'arrivée des clients dynamiques.

Dans ce cas le degré de dynamisme (dod : Degree Of Dynamism) est défini comme ci-dessous [Lund et al. 1996] :

$$dod = \frac{\text{le nombre de clients dynamiques}}{\text{le nombre total de clients}} \quad (1.11)$$

Cependant, cette mesure ne tient pas compte des temps d'arrivée des demandes dynamiques ou immédiates. Ceci signifie que deux systèmes différents, un dans lequel les demandes immédiates sont reçues au début de l'horizon de planification et un autre dans lequel elles se produisent tout au long de la journée, seront perçus comme étant équivalents (voir la figure 1.7).

Scénario A



Scénario B

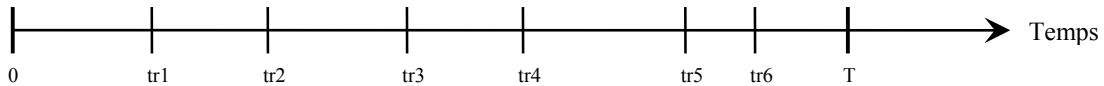


FIG. 1.7 – Deux scénarios ayant des “dod” équivalents

¹Degree Of Dynamism

Naturellement, dans la pratique ces deux scénarios auront des comportements très différents. Cette figure (1.7) illustre deux scénarios de DVRP dans lesquels les instants de réception des demandes immédiates diffèrent considérablement. Nous remarquons que dans le scénario *A*, chacune des six demandes immédiates est reçue relativement tôt pendant l'horizon de planification. Par contre, dans le scénario *B*, les demandes sont uniformément distribuées sur tout l'horizon de planification.

Du point de vue des performances, il est clair qu'avoir un nombre élevé de demandes en file d'attente améliore la qualité de la solution produite en ce qui concerne l'objectif de minimiser la distance totale parcourue. Par conséquent, la distance parcourue prévue serait éventuellement plus courte dans le scénario *A* que dans le scénario *B*, étant donné que le planificateur connaît à l'instant tr_6 toute l'information sur les positions des clients à servir.

1.6.3.2 Degré de dynamisme effectif - $edod^2$

On considère maintenant un scénario dans lequel l'horizon de planification commence au temps 0 et finit au temps T . Les demandes statiques sont reçues avant le commencement de l'horizon de planification. Le temps de réception de la demande numéro i est noté $tr_i : 0 < tr_i \leq T$. Le nombre de demandes immédiates reçues dynamiquement pendant l'horizon de planification est noté n_{imm} et le nombre de demandes statiques est noté n_{adv} . Le nombre total des demandes, n_{tot} est égal à $n_{tot} = n_{adv} + n_{imm}$. Le degré de dynamisme effectif, noté $edod$, est défini par [Larsen 2001] :

$$edod = \frac{\sum_{i=1}^{n_{imm}} (\frac{tr_i}{T})}{n_{tot}} \quad (1.12)$$

Le degré de dynamisme effectif représente alors un pourcentage moyen du retard de chaque demande reçue par rapport au retard maximal admissible défini par T .

1.6.3.3 Degré de dynamisme avec fenêtres de temps - $edod_{tw}^3$

Finalement, nous avons déjà vu que pour le problème VRPTW, chaque client doit fournir deux limites de temps :

- e_i : est la borne inférieure de la fenêtre de temps du client i .
- l_i : est la borne supérieure de la fenêtre de temps du client i .

Le degré de dynamisme effectif peut alors être étendu pour intégrer cette notion et montrer l'urgence de la demande. Pour chaque demande i , le terme $l_i - tr_i$ représente l'horizon de décision pour traiter la demande. Plus celui-ci sera petit plus la décision sera urgente à prendre. Une valeur moyenne du caractère d'urgence peut alors être définie par :

$$edod_{tw} = \frac{1}{n_{tot}} \sum_{i=1}^{n_{tot}} \left(\frac{T - (l_i - tr_i)}{T} \right) \quad (1.13)$$

²Effective Degree Of Dynamism

³Effective Degree of Dynamism for problems with Time Windows

1.7 Conclusion

Dans ce chapitre, nous avons présenté le problème d'élaboration de tournées de véhicules (le problème VRP). Nous avons décrit les différentes extensions de ce problème et introduit les méthodes de résolution utilisées pour ce problème.

Parmi ces extensions, une famille importante de problèmes concerne les problèmes de type VRP dynamique. Ce type de problème est fréquent dans de nombreux domaines d'application de la vie courante. C'est pourquoi nous nous y intéressons plus particulièrement dans le cadre de cette thèse.

Dans le chapitre suivant, nous présentons une autre extension au cœur de nos travaux qui porte sur la prise en compte de temps de trajets variables. Cette extension concerne à la fois les problèmes du VRP de base avec ou sans fenêtre de temps ainsi que les problèmes de VRP dynamiques. Notre travail de thèse portera précisément sur le développement d'une approche de résolution de ce type de problème.

Chapitre 2

Modèles de calcul de temps de trajet dépendant du temps⁴

Sommaire

2.1	Introduction	35
2.2	Evaluation du temps de trajet	36
2.2.1	Temps de trajet discret	36
2.2.2	Temps de trajet continu	37
2.3	Modèles de calcul pour évaluer les temps de trajets	37
2.3.1	Modèles variables en fonction de la distance à parcourir	37
2.3.2	Modèles variables en fonction du temps	39
2.3.3	Modèle basé sur des horizons différents	41
2.3.4	Modèle basé sur des ratios de vitesse	41
2.3.5	Modèle basé sur des files d'attente	42
2.4	Conclusion	46

2.1 Introduction

La plupart des travaux de recherche sur le problème DVRP concerne la prise en compte de demandes de clients dynamiques. Dans ces travaux, la vitesse est toujours supposée constante (valeur moyenne) ; c'est à dire, que le temps de trajet entre deux clients quelconques ne dépend que de la distance qui sépare ces deux clients. Cette distance, supposée connue, est déterminée généralement par un algorithme de plus court chemin. Cependant, dans les problèmes réels de tournées de véhicules, les vitesses sont variables au cours du temps car sujettes à des aléas. Il est donc intéressant de choisir un nouveau modèle pour s'adapter à cet environnement.

Le problème que nous traitons dans ce mémoire considère une extension particulière du DVRP (Dynamic Vehicle Routing Problem) dans laquelle les temps de trajets sont supposés variables. Ce modèle consiste à se rapprocher de la réalité en considérant que les trajets ne sont plus effectués à vitesse constante sur toute la journée, mais seulement sur des plages horaires prédéfinies (i.e. vitesse constante par morceaux). Ces plages correspondent aux différentes périodes d'une journée (heures de pointes ou heures creuses).

⁴Time dependent travel time

La figure (2.1) décrit les deux modèles qui sont basés d'une part sur la vitesse constante et d'autre part sur la vitesse variable (constante par morceaux). Il apparaît clairement que le modèle de vitesse variable est plus proche de la situation réelle représentée en traits pointillés (obtenue à l'aide d'historiques par exemple).

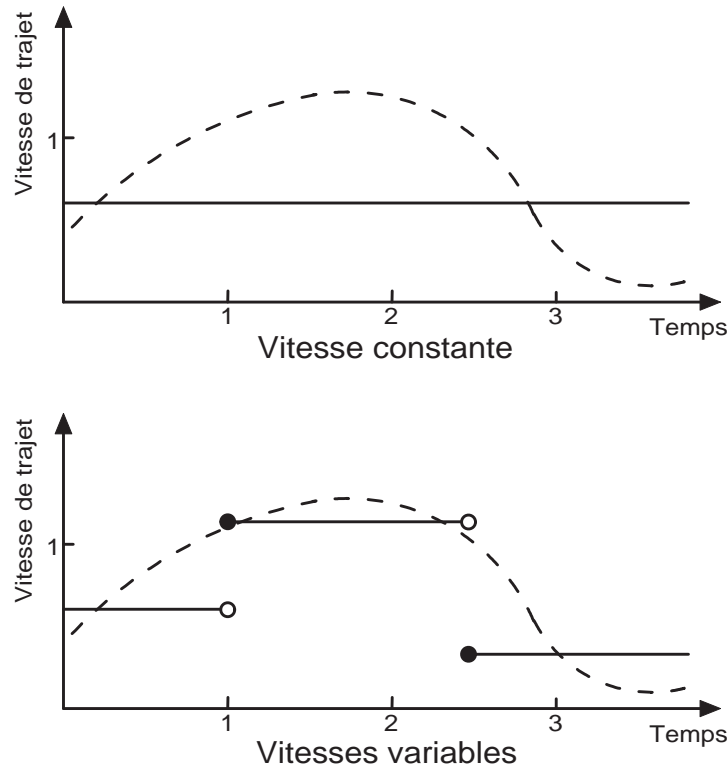


FIG. 2.1 – Modèles de vitesse constante et de vitesse variable

Ce chapitre est constitué de deux parties principales. La première partie présente les types de fonctions utilisées pour évaluer les temps de trajet. La deuxième partie présente les différents modèles de calcul de temps trajets variables utilisés dans le domaine du VRP.

2.2 Evaluation du temps de trajet

Il existe différentes fonctions pour évaluer le temps de trajet d'un véhicule dans les modèles dépendant du temps. Elles peuvent être classées principalement en temps de trajet discret et temps de trajet continu.

2.2.1 Temps de trajet discret

De nombreux articles tels que [Malandraki et al. 1992], [Hill et al. 1992] et [Malandraki et al. 1996], ont utilisé un modèle de fonction en escalier pour définir un temps de trajet discret. La période de temps est divisée en plusieurs intervalles et le temps de trajet est une fonction par morceaux. Pour illustrer ceci, considérons la figure (2.2) qui représente une fonction du temps de trajet sur un arc entre 2 nœuds i et j dont la distance est 1.

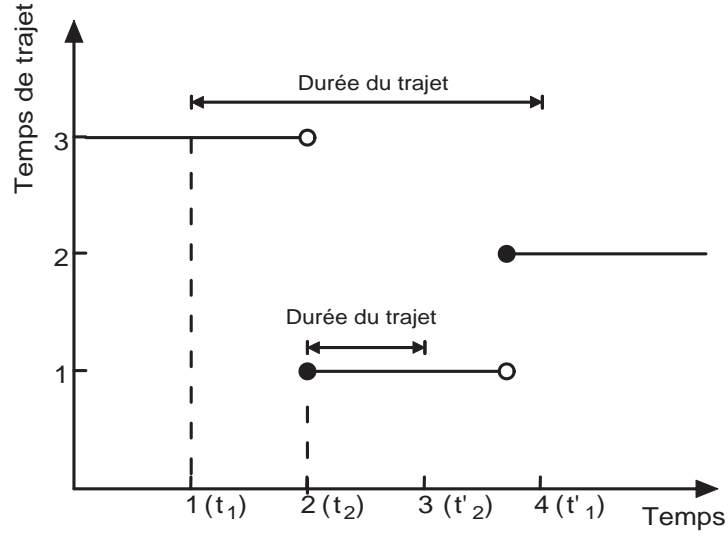


FIG. 2.2 – Fonction de temps de trajet sur un arc

Si le véhicule part du nœud i à l'instant $t_1 = 1$, il atteindra le nœud j à l'instant $t'_1 = 4$; mais il arrivera à l'instant $t'_2 = 3$ ($< t'_1$) si il part du nœud i à l'instant $t_2 = 2$ ($> t_1$).

2.2.2 Temps de trajet continu

Pour éviter le défaut du modèle ci-dessus, [Ichoua et al. 2003], [Jung et al. 2001] et [Haghani et al. 2005] ont établi une fonction continue pour le temps de trajet représentée en figure (2.3). Elle est obtenue à partir d'une fonction vitesse en escalier et permet d'assurer que le calcul des temps de trajet respecte la propriété FIFO : un véhicule A qui part vers la même destination que le véhicule B arrivera avant ce véhicule B parti plus tard.

En fait, les fonctions de vitesse de trajet dépendant du temps et les fonctions de temps de trajet dépendant du temps sont équivalentes puisqu'il est toujours possible de déduire la vitesse de trajet du temps de trajet, et réciproquement. La présentation des fonctions de temps de trajet montre la nécessité d'un ajustement de la vitesse quand le véhicule passe la frontière entre deux périodes de temps consécutives. Dans le paragraphe suivant, nous présentons certains modèles existants dans ce domaine.

2.3 Modèles de calcul pour évaluer les temps de trajets

Pour les problèmes VRP dynamiques ou temps réel, une question clé est le calcul des temps de trajet sur les arcs en fonction de la période de temps dans la journée.

Ainsi, il est nécessaire de faire un bilan des différentes approches permettant d'évaluer des temps de trajet, suivant les types de données utilisées.

2.3.1 Modèles variables en fonction de la distance à parcourir

[Kolesar 1975] a proposé le modèle non-linéaire suivant pour estimer le temps de déplacement du véhicule de l'endroit i à j avec d_{ij} la distance à parcourir :

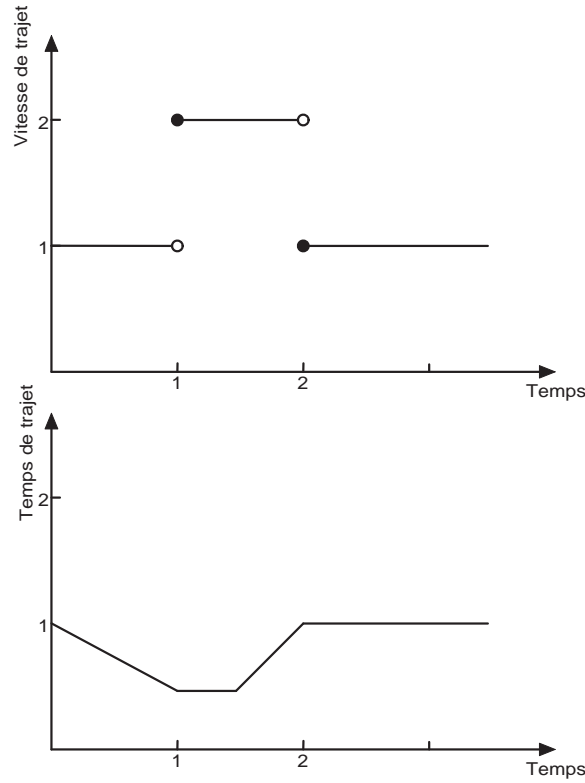


FIG. 2.3 – Fonction de vitesse par morceaux et temps de trajet continu

$$t_{i,j} = \begin{cases} 2\left(\frac{d_{ij}}{\alpha}\right)^{\frac{1}{2}}, & \text{pour } d_{ij} \leq 2d_c, \\ \frac{d_{ij}}{v_c} + \frac{v_c}{\alpha}, & \text{pour } d_{ij} > 2d_c. \end{cases} \quad (2.1)$$

Dans ce modèle, les véhicules accélèrent (ou ralentissent) à une constante d'accélération $\alpha \text{ km/h}^2$ et ont une vitesse constante de $v_c \text{ km/h}$ après la distance d_c , qui est la distance exigée pour que le véhicule atteigne sa vitesse de croisière qui est égale à $\frac{v_c^2}{2\alpha}$. [Kolesar 1975] suppose que les paramètres (α , d_c et v_c) sont connus. [Larson et al. 1981] a proposé le suivant comme bonne approximation du modèle précédent :

$$t_{ij} = \frac{d_{ij}}{v_c} + \frac{v_c}{\alpha} \quad (2.2)$$

[Braca et al. 1997] a présenté un modèle très détaillé d'évaluation de temps de trajet pour un problème de cheminement d'autobus scolaires de la ville de New York. Leur méthode considère les facteurs suivants :

- nombre de ruelles ;
- largeur de rue ;
- rue à sens unique ou bi-directionnelle ;
- règlements de stationnement (par exemple, double stationnement) ;
- type de rue (avenue/rue) ;
- nombre de signes d'arrêt et/ou de feux de signalisation ;
- volume du trafic ;

- type de secteur (résidentiel ou affaires) ;
- direction de rue ;
- état physique des rues

La collecte de ces informations est une tâche très fastidieuse et longue. Les auteurs ont rassemblé environ 30 points de repères pour la plupart des rues.

Une limitation des approches mentionnées ci-dessus est le nombre très élevé d'évaluations nécessaires pour estimer les temps de trajet. Le volume de stockage des données est très important également.

L'approche réalisée par [Ratliff et al. 1999] diffère de ceci en quatre points :

1. Une mesure des erreurs absolues est utilisée, alors que les modèles précédents emploient des critères de moindres carrés. La programmation linéaire utilisée est plus flexible en manipulant les contraintes qui sont souvent nécessaires afin de représenter un problème réel ;
2. Un modèle de vitesse moyenne constante par morceaux est utilisée en fonction de la distance à parcourir. Une fonction par morceaux linéaire est simple et facile à mettre en application dans la pratique ;
3. La collecte d'information est modérée et peut être effectuée électroniquement ;
4. La variabilité des vitesses de trajets est considérée.

Ainsi, des mesures ont été effectuées dans deux grandes villes américaines, Atlanta et New York ; Ces mesures ont montré que la vitesse de parcours selon la distance à parcourir n'est pas constante. Jusqu'à une certaine distance de parcours appelé « point de rupture », la vitesse croît linéairement, après ce point, la vitesse peut être considérée comme constante quelque soit la distance.

2.3.2 Modèles variables en fonction du temps

Comme mentionné précédemment, la littérature liée aux problèmes dépendant du temps de trajet de véhicule est relativement rare. En fait, les seuls papiers dont nous rendons compte dans cette catégorie sont ceux de [Malandraki et al. 1992], [Hill et al. 1992] et [Malandraki et al. 1996]. Ils ont été brièvement introduits dans la section 2.2.1.

Nous présentons ici plus en détail le modèle de [Ichoua et al. 2003] qui sera mis en œuvre dans la partie expérimentale de ce mémoire. La caractéristique principale de ce modèle est qu'il ne suppose pas que la vitesse reste constante dans le temps et qu'elle est uniforme sur tout le réseau routier. Ce modèle considère une vitesse de trajet variable dans le temps et ajuste celle-ci lorsqu'on passe d'une période à une autre. Dans ce modèle, il est associé également à chaque type de route empruntée une vitesse moyenne qui peut changer à chaque période de temps pour tenir compte des phénomènes de congestion cyclique.

Ce principe du modèle de [Ichoua et al. 2003] est illustré dans la figure 2.4 où le trajet d'un véhicule partant de i pour se rendre au point j est considéré. L'horizon est supposé divisé en 3 périodes (T_j) , $1 \leq j \leq 3$ avec des vitesses différentes associées à chaque période. Le véhicule part du nœud i au temps $t_0 \in T_1$ et se déplace à la vitesse V_{1T1} jusqu'à ce qu'il atteigne le point i_1 à la frontière entre les périodes T_1 et T_2 . Ensuite sa vitesse sera de V_{1T2} jusqu'à ce qu'il atteigne le point i_2 , à l'instant séparant les périodes T_2 et T_3 . Enfin, sa vitesse sera de V_{1T3} jusqu'à ce qu'il atteigne le nœud destination j .

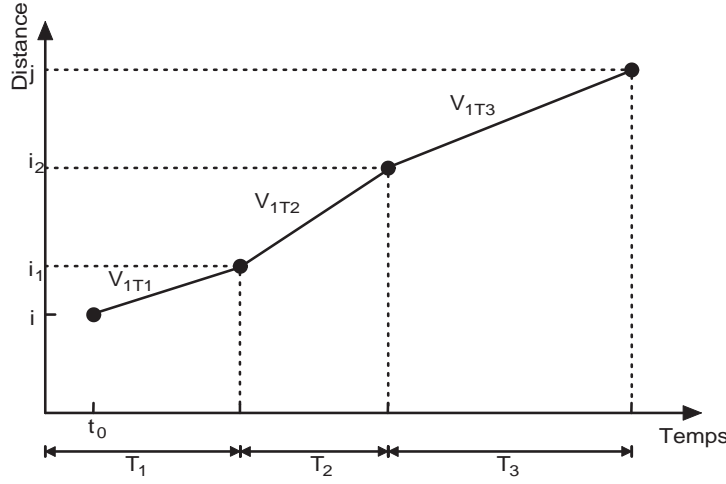


FIG. 2.4 – Changement de la vitesse en fonction du temps

L'algorithme (2.1) décrit la procédure pour calculer le temps de trajet entre deux nœuds quelconques i et j . Nous supposons que le véhicule quitte le nœud i au temps $t_0 \in P_k =]\bar{t}_k, \bar{t}_k]$ et que l'arc (i, j) appartient au type de routes c , $1 \leq c \leq C$. Nous supposons également que d_{ij} est la distance entre i et j , et V_{cTk} est la vitesse de trajet associée au type d'arc c et à la période de temps Tk . En notant t le temps courant et t' le temps d'arrivée, voici la procédure pour calculer le temps de trajet entre n'importe quelle paire de nœuds i et j :

Algorithme 2.1 Calcul du temps de trajet

Données matrices des profils historiques

$t \leftarrow t_0$

$d \leftarrow d_{ij}$

$t' \leftarrow t + (d/V_k)$

Tant Que ($t' > \bar{t}_k$) **Faire**

$d \leftarrow d - V_k(\bar{t}_k - t)$

$t \leftarrow \bar{t}_k$

$t' \leftarrow t + (d/V_{k+1})$

$k \leftarrow k + 1$

Fin Tant Que

Retourner ($t' - t_0$)

Où :

- t : temps courant
- t_0 : instant de départ. $t_0 \in]\bar{t}_k, \bar{t}_k]$
- d : distance restante
- d_{ij} : distance entre les clients i et j
- t' : temps d'arrivée au client
- V_k : vitesse à la période k
- \bar{t}_k : temps de la borne supérieure à la période k

Dans ce modèle, la vitesse de trajet V_{ct} est une fonction par morceaux suivant la période

du jour. Par conséquent, le temps de trajet entre i et j est une fonction continue (figure 2.3). La complexité de l'algorithme est en $O(k)$ avec k le nombre de périodes de vitesse différente à considérer dans une journée.

2.3.3 Modèle basé sur des horizons différents

Le modèle de [Ichoua et al. 2003] a utilisé des vitesses de trajet différentes pour les 3 périodes de la journée les plus significatives (début de journée, milieu de journée, fin de journée). Néanmoins il n'a pas considéré des conditions de fonctionnement imprévues qui perturbent l'environnement des véhicules (congestion du trafic, pannes de véhicule). Donc il ne peut pas s'adapter précisément à une application réelle.

Le modèle proposé par [Potvin et al. 2006] s'appuie sur trois types différents de prévision pour obtenir le temps de trajet entre 2 clients et ainsi améliorer le modèle de [Ichoua et al. 2003] :

1. **Prévisions à long terme** : Ces prévisions sont connues à l'avance et ne sont pas sujettes à des variations stochastiques. Elles sont basées sur des temps de trajet qui varient selon la période de temps (matin, midi, après midi). Elles respectent la règle FIFO. C'est donc le modèle de [Ichoua et al. 2003] précédent.
2. **Prévisions à court terme** : Quand le véhicule est prêt à quitter le client courant, le temps de trajet pour la prochaine destination est modifié par une quantité uniforme aléatoire positive ou négative. Cette modification est générée par la connaissance des informations relatives à l'état du trafic routier à l'instant courant. Celle-ci reste néanmoins simulée par un nombre aléatoire.
3. **Perturbation dynamique** : Le temps de trajet est perturbé par des événements imprévus ; ces événements ne sont pas prévisibles ; par exemple un accident ou une panne. Ils ne seront connus qu'une fois le véhicule arrivé à la destination. Le système ne peut le prendre en compte lors du calcul de la date d'arrivée planifiée.

Il convient de constater que le modèle de [Ichoua et al. 2003] a considéré uniquement le premier de ces trois éléments.

Les temps de trajet sont calculés comme mentionné en figure 2.5 en utilisant plusieurs paramètres pour prendre en compte les trois types de prévision mentionnés ci-dessus. $[e_0, l_0]$ représente le temps d'ouverture de dépôt ; t est le temps de trajet moyen ; 1,25, 0,5, 1,25 sont les facteurs de variation pour la prévision à long terme. ε_1 , ε_2 et ε_3 sont les paramètres pour la prévision à court terme qui sont uniformément distribués entre $[-0, 1, +0, 1]$. ρ_1 , ρ_2 et ρ_3 sont les paramètres pour la perturbation dynamique qui sont produits par une loi normale avec le paramètre d'espérance mathématiques 0 et un écart type faible (1 et 4) ou un écart type important (16 et 32) et concernent seulement le retard par rapport à la planification courante. Ainsi ces valeurs seront affectées à une valeur nulle si les résultats obtenus pour ces valeurs sont négatifs.

Par exemple si $\varepsilon_1 = 0,05$, $\rho_1 = 0,2$, le temps de trajet du matin est donc égal à : $40 \times 1,25 \times (1 + 0,05 + 0,2) = 62,5$ au lieu de 40 en utilisant uniquement le modèle de [Ichoua et al. 2003].

2.3.4 Modèle basé sur des ratios de vitesse

Dans le modèle de [Ichoua et al. 2003], le temps de trajet est une fonction continue, mais la vitesse de trajet est une fonction par morceaux suivant la période du jour. C'est une approximation des conditions rencontrées dans la réalité. [Haghani et al. 2005] ont proposé des vitesses de trajet basées sur une fonction continue permettant de mieux approcher les situations réelles.

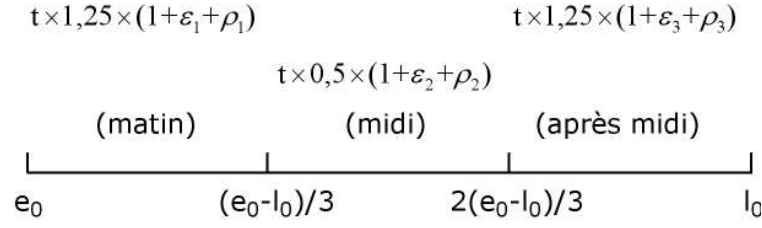


FIG. 2.5 – Calcul de temps de trajet à la prévision

Le réseau de transport fictif utilisé est constitué d'un seul dépôt, de 382 nœuds et de 1398 arcs. Les arcs ont été classés en trois types selon leur fonction (autoroute, voie rapide ou route). Chaque type d'arc a deux types de profil de vitesse correspondant aux caractéristiques de flux de trafic dans une journée (avoir deux valeurs de vitesse de conception). Cette hypothèse est identique au modèle de [Ichoua et al. 2003], mais la différence est la suivante : six types de ratio de vitesse variable pendant la journée ont été créés fictivement tels que représentés en figure 2.6 pour décrire la variation de vitesse de trajet. Le ratio est égal au pourcentage de la valeur à la vitesse réelle divisée par la vitesse de conception. Ces ratios correspondent séparément aux trois types de route et six types de profil de vitesse. Les types de profil et les vitesses de conception sont représentés dans le tableau 2.1.

Type de route	Profils	Vitesse de conception
Autoroute(freeway)	1, 2	50, 60 MPH
Voie rapide(arterial)	3, 4	40, 50 MPH
Route(collector)	5, 6	30, 40 MPH

TABLEAU 2.1 – Profil basé sur les vitesses de conception

Chaque arc est caractérisé par une vitesse de conception. À tout moment de la journée, la vitesse de trajet sur un arc est calculée selon la figure 2.6 et le tableau 2.1. Par exemple dans le tableau (2.1) et la figure (2.6), l'arc entre les nœuds 56 et 78 est une route dont la fonction est qualifiée de « voie rapide ». Cet arc est de type 3 (profil 3) et la vitesse limite est 45MPH. Le ratio de vitesse pour cet arc à 10H est 90,0%. La vitesse est donc égale à $45 \times 90,0\%$ soit : 40,5MPH.

Ensuite, les temps de trajet entre les nœuds sont calculés en utilisant un algorithme de plus court chemin dépendant du temps; ces temps servent d'entrées sous la forme d'une matrice Origine-Destination pour l'algorithme de résolution du VRP.

2.3.5 Modèle basé sur des files d'attente

Pour déterminer le temps de trajet sur l'arc (i, j) , il est nécessaire de connaître la distance entre (i, j) et la vitesse de trajet sur cet arc au temps p . La distance étant constante, la méthode de détermination de la vitesse de trajet est donc importante pour le calcul du temps de trajet.

Dans le modèle présenté précédemment, [Ichoua et al. 2003] a choisi une vitesse moyenne constante (40 km/h) et a obtenu les autres vitesses pour les différentes périodes de temps en introduisant un facteur multiplicatif α . Selon le même principe, [Potvin et al. 2006] a utilisé plusieurs paramètres ε et ρ pour ajuster les vitesses de trajet. [Haghani et al. 2005] a remplacé

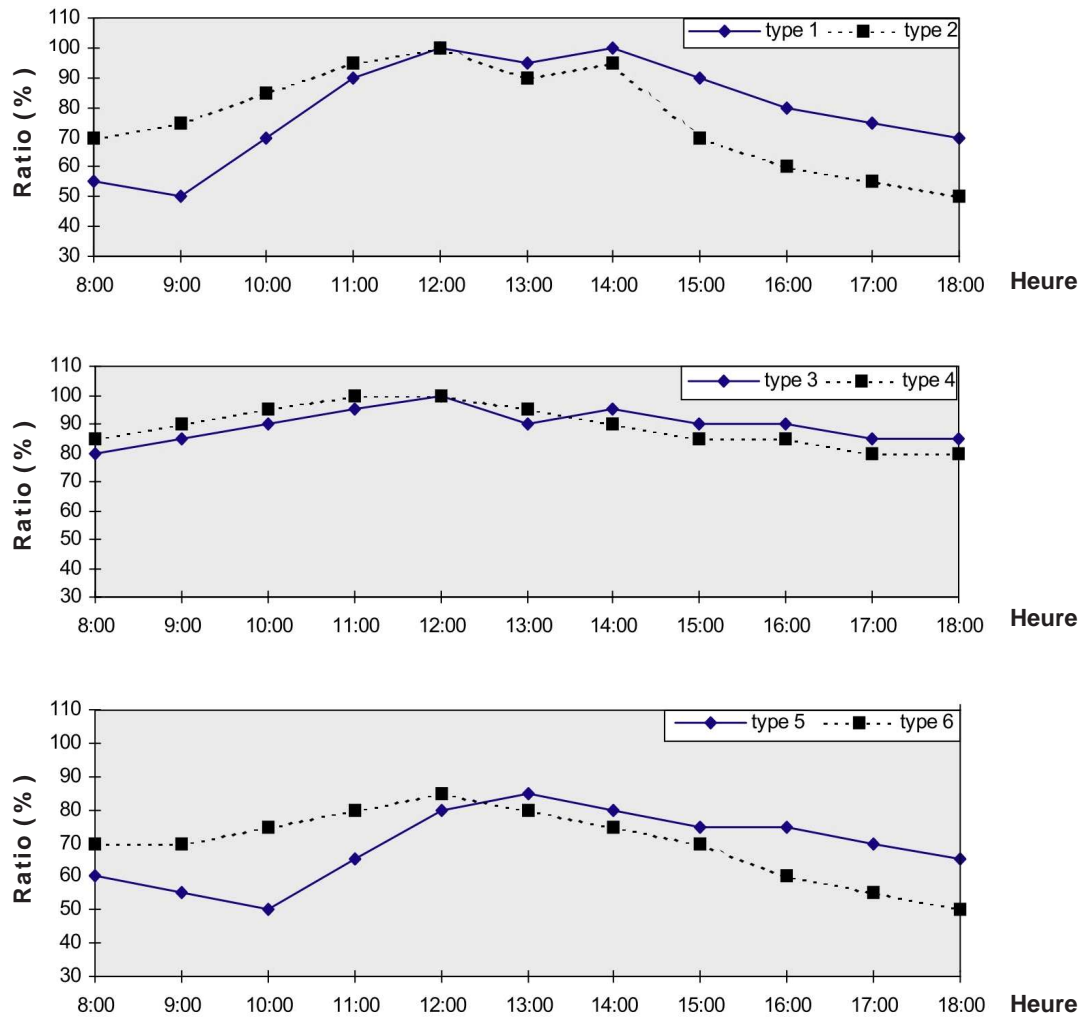


FIG. 2.6 – Types de profil de vitesses variables

la vitesse moyenne et ces paramètres (ε et ρ) par une vitesse limite et 3 graphiques de ratio de vitesse. L'insuffisance de ces modèles est qu'il y a un écart de vitesses entre ces valeurs théoriques et les valeurs réelles.

Par contre, [Woensel et al. 2002] a obtenu des vitesses dépendantes du temps à partir de données d'observations réelles et en utilisant la théorie des files d'attente et de flux de trafic.

Avant de calculer la vitesse de trajet, il faut présenter le modèle de [Vandaele et al. 2000], [Woensel et al. 2001] et [Heidemann et al. 1996] basé sur les files d'attente et décrit à la figure (2.7). Les routes sont divisées en segments dont la longueur est égale à l'espace minimal qui est utilisé par un seul véhicule dans cette route. On définit k_j comme la densité maximum du trafic. La longueur du segment est alors égale à $\frac{1}{k_j}$ et elle définit l'espace minimal requis par un véhicule circulant sur cette route. Chaque segment de route est considéré comme une station (serveur de files d'attente), dans lequel des véhicules arrivent à un certain taux λ et sont servis par le serveur avec un autre taux μ .

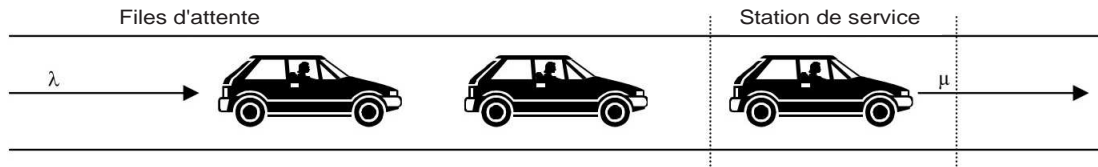


FIG. 2.7 – Représentation de flux de trafic par Files d'Attente

On distingue quatre principaux modèles dans la théorie des files d'attente qui sont présentés dans le tableau (2.2). [Vandaele et al. 2000] a exploité ces différents modèles dans ses travaux. Le modèle M/M/1 (en notation de Kendall simplifiée) est considéré comme le modèle de base. Les lois de distribution qui caractérisent le processus des arrivées et de service sont de type loi de Poisson.

Modèle	Processus d'arrivées	Processus de service
M/M/1	Poisson	Poisson
M/G/1	Poisson	Général
G/G/1	Général	Général
G/G/1 dépendent de l'état	Général	Général

TABLEAU 2.2 – Caractéristique des principaux modèles de files d'attente

Dans la théorie du flux de trafic, les équations les plus importantes sont l'interdépendance du flux de trafic q , de la densité de trafic E et de la vitesse v comme mentionné dans l'équation 2.3 et l'intensité du trafic ρ comme mentionné dans l'équation 2.4. Dans le modèle M/M/1, la vitesse efficace v est donnée par l'équation 2.5 ([Heidemann et al. 1996]).

$$q = Ev \quad (2.3)$$

$$\rho = \frac{k}{k_j} \quad (2.4)$$

$$v = v_f(1 - \rho) \quad (2.5)$$

$$k_j \cdot v^2 - k_j \cdot v_f \cdot v + v_f \cdot q = 0 \quad (2.6)$$

Où :

- v_f : vitesse de flux libre (vitesse maximum de la route) ;
- k_j : densité de trafic maximum (exprimée en nombre de véhicules par km) ;
- q : flux mesuré (exprimé en nombre de véhicules par heure)

Si la densité de trafic est faible, les véhicules ne se gênent pas les uns les autres, ce qui mène à des vitesses efficaces plus élevées. Plus le nombre de véhicules arrivant sur la route est important, plus la vitesse efficace v diminue. Donc, d'après les équations 2.3, 2.4 et 2.5 nous pouvons obtenir l'équation 2.6 et construire le graphique de la relation entre la vitesse et le flux comme mentionné en figure 2.8 pour le modèle M/M/1 ([Vandaele et al. 2000]).

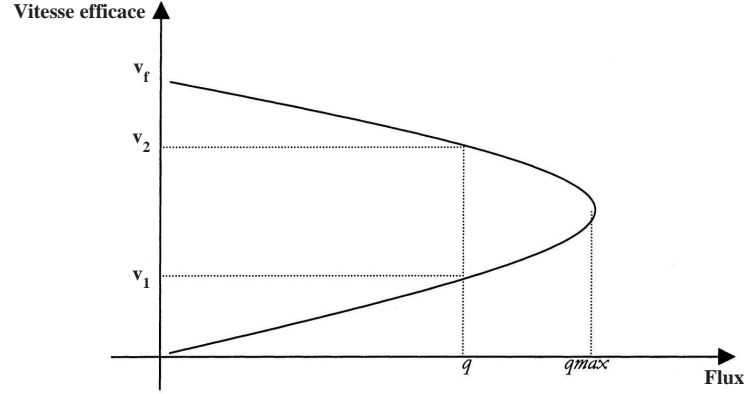


FIG. 2.8 – Relation entre vitesse et flux pour le modèle M/M/1

Les vitesses efficaces ' v'_1 ' et ' v'_2 ' sont explicitement définies en fonction de l'écoulement ' q ' comme présenté dans l'équation 2.7 :

$$v_{1,2} = \frac{k_j \times v_f \pm \sqrt{k_j \cdot v_f \cdot [k_j \cdot v_f - 4q]}}{2k_j} \quad (2.7)$$

Par conséquent, nous pouvons calculer le temps de trajet suivant la formule de [Kerbache et al. 2005] (voir l'équation 2.8 et la figure 2.9). T_{ij} est la somme des temps nécessaires pour aller de i à j . Le nombre de périodes k est une fonction de la distance entre i et j et des vitesses dans les différentes périodes ; Les termes fractionnaires ϕ montrent que le véhicule ne parcourt pas complètement la première et la dernière période. Cette fraction dépend de la vitesse et de la distance qui reste à parcourir dans la dernière période.

$$T_{ij} = \varphi \Delta P_{premier} + (k - 2) \Delta P + \phi \Delta P_{dernier} \quad (2.8)$$

Où :

- ΔP : durée ;
- $\varphi, \phi \in [0, 1]$: fractions de ΔP

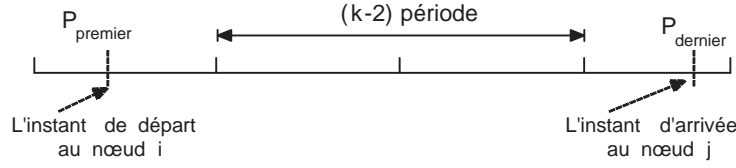


FIG. 2.9 – Description du calcul du temps de trajet

La vitesse du véhicule est variable suivant les périodes. Elle dépend de la densité et du flux de trafic et du modèle de file d'attente (voir l'équation 2.7).

Avec le procédé ci-dessus, le temps de trajet T_{ij} du nœud i au nœud j en partant à l'instant p_0 peut être facilement déterminé en se basant sur la distance d_{ij} à parcourir et la vitesse v_{pij} pour les différentes périodes de temps p obtenues à partir du modèle de files d'attente. Le modèle de calcul des temps de trajet est identique au modèle de [Ichoua et al. 2003] (voir l'algorithme 2.1).

2.4 Conclusion

Evaluer le temps de trajets d'un véhicule sur un tronçon routier est primordial pour la planification des tournées d'une flotte de véhicules. Utiliser un modèle de calcul de temps de trajet qui dépend du temps au cours de la journée est une caractéristique importante parce qu'elle correspond aux conditions réelles où les temps de déplacement sont soumis à des variations dans le temps. Ces variations résultent soit d'événements prévisibles tels que les phénomènes de congestion aux heures de pointe soit d'événements imprévisibles comme les accidents, les pannes de véhicules, etc.

Dans ce chapitre, nous avons présenté différents types de fonctions de temps de trajet et les modèles de calcul de temps de trajet dépendant du temps. Dans nos travaux nous avons pris le modèle [Ichoua et al. 2003] parce qu'il intègre des vitesses variables et respecte la propriété FIFO. Nous avons étendu ce modèle dans les chapitres suivants pour intégrer la prise en compte d'informations temps réel.

Chapitre 3

Une gestion dynamique de la tournée du réparateur avec temps de trajet dépendant du temps

Sommaire

3.1	Introduction	47
3.1.1	Formulation du DTRP	48
3.1.2	Formulation du PDTRP	48
3.1.3	Les applications de problème DTRP	49
3.2	Approches de résolution du DTRP	50
3.2.1	Approche de Bertsimas et Van Ryzin	50
3.2.2	Approche de Kilby, Prosser et Shaw	50
3.2.3	Approche de Irani, Lu, Regan	51
3.3	Proposition d'un modèle de résolution de DTRP avec temps de trajet dépendant du temps	51
3.3.1	Un modèle de vitesse de trajet dépendant du temps	52
3.3.2	Matrice des profils de vitesse	53
3.3.3	Les stratégies modifiées	55
3.4	Plate-forme et résultats	56
3.4.1	La plate-forme développée	56
3.4.2	Résultats et analyses de la simulation	57
3.5	Conclusion	58

3.1 Introduction

Le problème dynamique de la tournée du réparateur (DTRP : Dynamic Traveling Repair Problem) introduit par [Psaraftis 1988] est considéré comme une version simplifiée du problème VRP dynamique. Il a été traité plus récemment par Larsen dans son travail de thèse [Larsen 2001]. Ce problème ne comporte qu'un seul véhicule (celui du réparateur) dans sa version de base ; Ce réparateur doit visiter chaque client qui demande un service de dépannage au cours de la journée. Dans ce problème, nous n'avons ni contrainte de capacité du véhicule, ni contrainte de fenêtre de temps. L'objectif de ce problème est de minimiser la distance totale parcourue par le réparateur et

de minimiser le temps d'attente des clients. Ce temps correspond au temps écoulé entre l'instant où la demande du client est formulée et l'instant où celui-ci commence à être servi.

3.1.1 Formulation du DTRP

[Bertsimas et al. 1991] ont défini le DTRP comme suit :

- Un réparateur (ou un système véhicule/serveur) voyage avec une vitesse unitaire dans une région convexe A .
- Toutes les demandes sont dynamiques et arrivent au cours du temps selon une loi de Poisson de paramètre λ . Les localisations des demandes sont indépendantes et uniformément distribuées dans A .
- Chaque demande exige une quantité indépendamment et identiquement distribuée de temps de service sur site avec une durée moyenne s_m et un moment s_m^2 . Le temps passé par le serveur (réparateur) sur le site à satisfaire la demande est noté ρ . Pour les systèmes stables, $\rho = \lambda \times s_m$. Ce temps de service, qui suit une loi de distribution log-normale, n'est pas connu par le répartiteur tant que le service chez le client n'a pas été terminé.
- Le temps de système T_i , de la demande i est défini comme l'écart entre le temps d'arrivée de la demande i et le temps où le serveur termine le service associé à la demande. Un temps de système équilibré noté T est défini par $T = \lim_{i \rightarrow \infty} E[T_i]$ où E est l'espérance mathématique.
- Le temps d'attente W_i de la demande i est défini comme étant le temps écoulé entre l'arrivée de la demande jusqu'à ce que le service démarre $T_i = W_i + s_i$. Le temps d'attente à l'équilibre W est : $W = T - s_m$.

Le problème est donc de concevoir une stratégie d'élaboration des tournées qui minimise T . La valeur optimale de T est notée T^* . Selon [Bertsimas et al. 1991], bien que le problème du DTRP s'apparente à un réseau de files d'attente, cette théorie ne peut être appliquée car le temps de système T inclut les temps de trajet qui ne peuvent pas être vus comme des variables stochastiques indépendantes.

3.1.2 Formulation du PDTRP

Dans le DTRP introduit par [Bertsimas et al. 1991], toutes les demandes sont considérées comme dynamiques. Le PDTRP est une variante de ce problème impliquant à la fois des demandes connues à l'avance et des demandes dynamiques.

Le problème du PDTRP est défini comme suit :

- Un réparateur ou un ensemble véhicule/serveur, voyage à une vitesse constante, dans une région A ;
- Les demandes se décomposent en deux sous-ensembles :
 - Le sous-ensemble des demandes dynamiques (ou immédiates). Elles arrivent selon une loi de Poisson de paramètre λ . Les lieux de demande sont indépendamment et uniformément distribués dans A ;
 - Le sous-ensemble des demandes statiques connues avant le départ du véhicule.

- Chaque demande requiert un temps de service sur site indépendamment et identiquement distribué qui est connu une fois que le service est terminé ;
- L'itinéraire est mis à jour seulement à chaque arrivée chez le client. Ainsi, un véhicule ne peut pas changer de destination en cours de route ;
- L'objectif est de minimiser le coût de tournée du réparateur (normalement, la distance de trajet).

Le PDTRP diffère du DTRP du fait que les lieux géographiques des clients statiques à traiter sont connus à l'avance avant que le réparateur ne quitte le dépôt. Il est à noter que les temps de service ne sont pas connus tant que le service chez un client n'a pas été accompli. Ces temps de service suivent une loi de distribution Log-normale.

3.1.3 Les applications de problème DTRP

Dans le problème d'élaboration de tournée dynamique du réparateur, le temps d'attente est souvent un critère plus important que le coût du voyage. Nous avons choisi d'étudier le problème de tournée dynamique du réparateur car c'est un problème qui d'une part permet d'aborder un problème de tournée dynamique pouvant se rapporter à de nombreuses applications réelles, clairement spécifié, d'une complexité raisonnable et d'autre part il a été généralisé au problème du ramassage et de la livraison simultané dynamique (DPDP : Dynamic Pick-up and Delivery Problem) dans les travaux de [Swihart et al. 1999] en considérant un véhicule à capacité unique ou multiple.

Le DTRP s'applique à de nombreuses situations concrètes dans les cas de systèmes réels de distribution où les offres et les demandes arrivent aléatoirement au cours du temps et dans lesquelles l'affectation des véhicules aux demandes des clients pour élaborer les tournées est un traitement continu sur un horizon déterminé. De plus, ce type d'applications accorde une importance particulière au temps d'attente des clients.

Habituellement, les sociétés de service (électricité, gaz, eau [Madsen et al. 1995]) mettent en place un service de réparation chargé de répondre aux demandes d'entretien ou de réparation à domicile de l'équipement de leurs clients. Dans ce problème, l'objectif est de déterminer, pour un niveau donné de service et sur un horizon de planification donné (un jour, une semaine), une stratégie optimale de conception de tournées pour réduire au minimum la distance totale parcourue.

L'aspect dynamique du problème vient du besoin de dire rapidement au client si la demande peut être acceptée et d'indiquer dans l'affirmative, une fenêtre de temps dans laquelle l'équipe de service devra arriver chez le client. L'algorithme traite les nouvelles demandes une à une. Pour chaque demande, un sous-ensemble d'itinéraires est choisi, basé sur des mesures de proximité. Puis, la position d'insertion du nouveau client qui minimise le détour pour ce sous-ensemble de tournées est déterminée. En conclusion, une fenêtre de temps est affectée au client avec l'objectif d'équilibrer la charge pour chaque tranche de la journée.

Cette application est connue sous le nom de Problème Dynamique du Représentant de Commerce (DTSP) [Psaraftis 1988] et du Problème Dynamique de la Tournée du Réparateur (DTRP) [Bertsimas et al. 1991] où les demandes de service doivent être satisfaites dès que possible après leur apparition. D'ailleurs, ces deux problèmes sont traités par des approches similaires qui se distinguent essentiellement par le critère temporel et leur domaine application.

Dans [Bertsimas et al. 1991] est proposé, une analyse mathématique de différentes stratégies d'élaboration de tournées ayant pour but de minimiser le temps moyen d'un client passé dans

le système, c'est-à-dire le temps d'attente augmenté du temps de service client. En conséquence, aucune tournée n'est construite en tant que telle, mais c'est une stratégie spécifiée à l'avance qui permettra d'identifier le prochain client à servir parmi une file d'attente de demandes en attente. Ces stratégies sont utilisées après que le véhicule ait accompli le service du client courant. Ce travail est fortement inspiré de la théorie des files d'attente : un véhicule est considéré comme un serveur mobile, avec des demandes apparaissant dans le temps selon une loi de Poisson et avec des temps de service sur site stochastiques.

Les auteurs considèrent l'élaboration de tournées par l'analyse d'une stratégie adaptée où les demandes de service sont cumulées afin de former des ensembles de taille fixe.

Puisque le véhicule est affecté à ces ensembles, et non pas à des demandes individuelles, les demandes de ces ensembles peuvent être ordonnancées. Une prolongation de ces travaux pour le cas de plusieurs véhicules avec des contraintes de capacité est analysée dans [Bertsimas et al. 1993a].

Dans ce cas, un véhicule peut servir au plus q clients avant de retourner au dépôt. Ainsi, les clients sont groupés en sous-ensemble de taille q avant qu'un nouveau véhicule ne soit lancé.

3.2 Approches de résolution du DTRP

3.2.1 Approche de Bertsimas et Van Ryzin

[Bertsimas et al. 1991] a analysé un éventail de stratégies naïves pour élaborer la tournée du DTRP. Elles sont décrites ci-dessous :

- *FCFS (First Come First Served)* Les demandes sont servies dans l'ordre dans lequel elles sont reçues par le répartiteur ;
- *FCFS-SQM (Stochastic Queue Median)* La politique FCFS-SQM est une modification de la politique FCFS. Les véhicules sont initialement placés à la médiane de la région et y retournent une fois les services clients terminés par la stratégie FCFS ;
- *NN (Nearest Neighbour)* Une fois le service complété sur un lieu, le serveur se dirige vers la demande voisine la plus proche ;
- *TSP (Travelling Salesman Problem)* Les demandes sont rassemblées dans des ensembles de taille n . A chaque fois qu'un nouvel ensemble de demandes est rassemblé, un algorithme de type TSP est résolu. Les demandes sont alors servies selon un TSP optimal pour des ensembles de taille réduite. Si plus d'un ensemble apparaît en même temps, les ensembles sont alors servis suivant une stratégie de type FCFS ;
- *SFC (Space Filling Curve)* Les demandes sont servies au fur et à mesure de leur apparition en suivant l'ordre de rotation des aiguilles d'une montre. Cette stratégie a été définie par [Platzman et al. 1983] ;
- *PART (Partitioning Policy)* La région de service A est divisée en m^2 partitions - $m \times m$ car on découpe suivant les deux directions d'un plan. Dans une partition les demandes sont servies en utilisant la stratégie FCFS. Quand toutes les demandes à l'intérieur d'une sous-région sont satisfaites, le serveur se dirige vers la prochaine sous-région.

Les heuristiques de résolution du PDTRP proposées par [Bertsimas et al. 1991] sont les suivantes : FCFS, SQM, NN, PART.

3.2.2 Approche de Kilby, Prosser et Shaw

Les méthodes employées par [Kilby et al. 1998b] pour résoudre ce problème dynamique sont basées sur une méthode d'insertion et d'amélioration (insert and improve). Chaque nouvelle

demande est insérée au meilleur emplacement de la tournée courante. La méthode employée pour trouver une solution consiste d'abord à trouver une première séquence qui incorpore tous les clients connus a priori. Chaque demande dynamique est ensuite ajoutée à cette séquence en l'insérant dans une position optimale.

Il emploie une méthode itérative [Caseau et al. 1999] qui insère un certain nombre de visites et tend à optimiser la fonction d'objectif avant de continuer à insérer les clients suivants.

Pour les problèmes présentés, [Kilby et al. 1998b] propose d'insérer un certain nombre fixé de visites à chaque itération, et d'améliorer la solution en utilisant une recherche locale telle que le 2-Opt, Or-Opt et d'autres opérateurs tels que le remplacement [Savelsbergh 1988], l'échange [Taillard 1994] et le croisement [Potvin et al. 1995].

[Kilby et al. 1998b] a constaté que l'ordre de l'insertion ainsi que les heuristiques d'insertion employées avaient une grande influence sur l'efficacité de la résolution du problème.

Au départ une tournée qui ne tient pas compte de toutes les contraintes mais couvrant toutes les visites avec un seul véhicule est déterminée en utilisant la distance de trajet comme fonction objectif. Cette tournée est ensuite améliorée grâce au 2-Opt.

Des améliorations de ce modèle ont été présentées. Le concept de « pas de temps » a été proposé. Il permet de résoudre l'algorithme d'amélioration en se limitant à un pas de temps déterminé. Toutes les améliorations qui ont été trouvées pendant ce temps sont alors incorporées dans la séquence. Toute nouvelle demande qui est apparue au cours du processus d'amélioration est ajoutée dans la position de moindre coût.

3.2.3 Approche de Irani, Lu, Regan

[Irani et al. 2004] propose deux algorithmes de résolution du DTRP.

Le premier est l'algorithme en lots ou BATCH. On adjoint à chaque client une fenêtre de temps uniforme et de longueur 1. La limite inférieure de cette fenêtre correspond à la date de demande du client. On découpe l'horizon en pas de temps constants de longueur $1/2$. Au commencement de chaque intervalle, les demandes qui sont arrivées à l'intervalle précédent sont prises en compte. L'algorithme détermine le nombre maximum de demandes qui pourront être servies à partir de la position courante du véhicule dans le prochain intervalle de temps. Ainsi les requêtes considérées dans un intervalle de temps sont celles qui sont apparues au cours de l'intervalle de temps précédent et qui pourront être servies dans l'intervalle de temps suivant.

Le second algorithme proposé est l'algorithme dit à « double gain ». Il s'applique plus généralement dans le cas où les fenêtres de temps ne sont pas uniformes pour toutes les demandes. L'état de cet algorithme est défini par la position courante et un ensemble de demandes en attente. A chaque fin de route, l'algorithme doit définir une nouvelle route permettant de satisfaire au mieux les demandes en attente qui sont encore d'actualité.

3.3 Proposition d'un modèle de résolution de DTRP avec temps de trajet dépendant du temps

Aucun des travaux cités ci-dessus n'ont pris en considération la variabilité des temps de trajets. Nous pensons que celle-ci est d'autant plus critique dans les gestions dynamiques de tournées de véhicules. Dans cette partie nous voulons introduire la notion de temps de trajet

dépendant du temps que nous avons décrite dans le chapitre 2 pour résoudre de manière plus réaliste le problème dynamique de la tournée du réparateur. Pour cela le modèle de calcul de temps de trajet d'Ichoua a été repris [Ichoua et al. 2003] pour l'intégrer dans une plateforme que nous avons développée.

3.3.1 Un modèle de vitesse de trajet dépendant du temps

Les heuristiques de la littérature mentionnées dans la section 3.2.1 considèrent toutes des temps de trajet constants. Mon travail a consisté à étudier pour ces problèmes DTRP et PDTRP, des modèles de vitesse basés sur des trajets variables au cours du temps de manière à se rapprocher des conditions réelles d'exécution des tournées.

Les différentes approches possibles pour évaluer et modéliser des temps de trajet ont été présentées au chapitre 2. En effet, les temps de trajets peuvent fluctuer suivant la distance à parcourir, la période de temps de la journée considérée et le type de route utilisée dans l'espace à traverser. Nous utilisons dans cette étude sur le modèle de vitesses dépendant de temps d'Ichoua [Ichoua et al. 2003] (voir la section 2.3.2).

Considérons à titre d'exemple, deux villes i et j éloignées de 200km. Les vitesses de trajet sont décrites par des profils de vitesses qui sont donnés en figure 3.1 ; ces profils sont constitués de plusieurs périodes de temps sur lesquelles la vitesse est supposée constante. On suppose que l'heure de départ de la ville est midi (12H). Lorsque le véhicule arrive à la fin de la première période (13H), une heure s'est écoulée, on a parcouru 60km en une heure (vitesse de 60km/h) et la distance restante est donc le 140km. Ensuite, lorsque le véhicule arrive à la fin de la deuxième période (15H) deux heures se sont écoulées, la distance parcourue à la vitesse de 40km/h est donc de 80 km et la distance restante est alors de 60km. Il faut donc encore 0,75 heures pour terminer le parcours et arriver à la ville j . Donc le temps total de trajet est : $1+2+0,75=3,75H$. Le processus de calcul est présenté en figure 3.2.

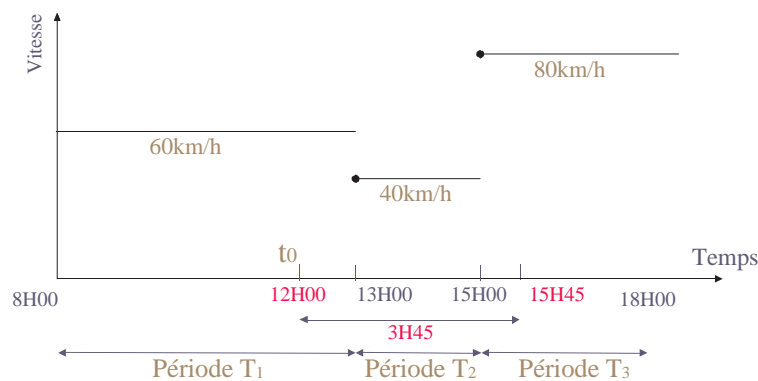


FIG. 3.1 – Profil de vitesse

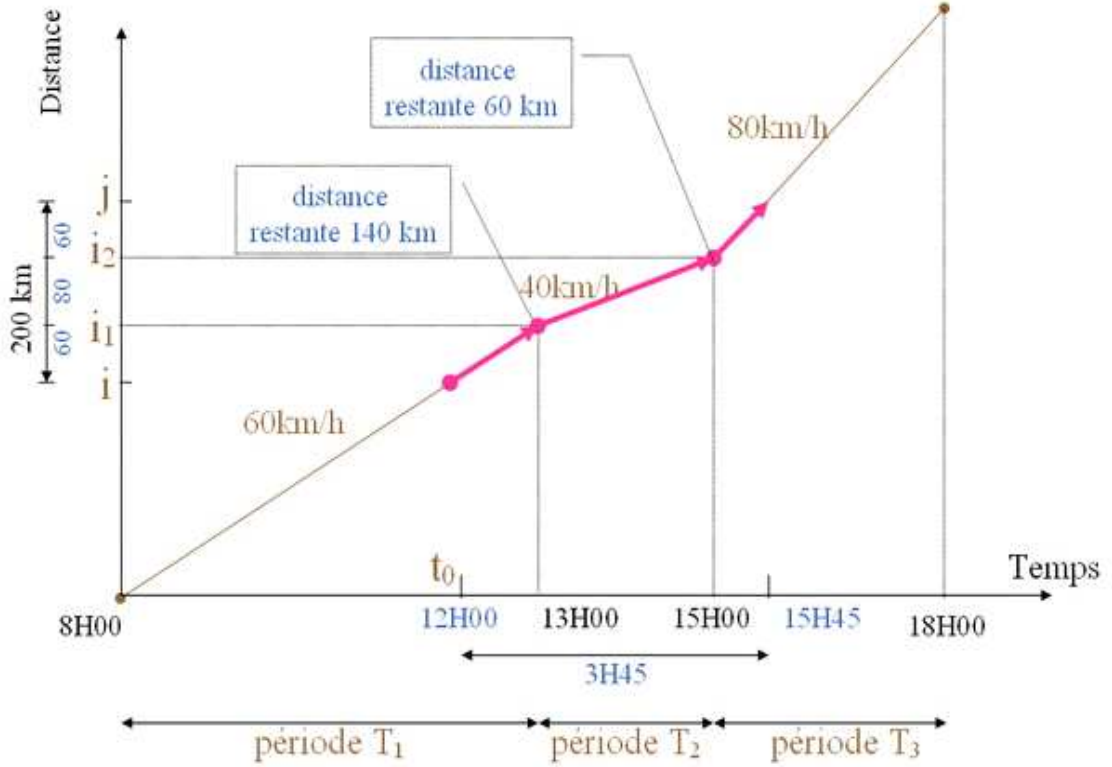


FIG. 3.2 – Exemple de calcul de temps de trajet

La figure 3.3 présente six autres exemples pour le calcul des temps de trajet en prenant des instants de départ situés dans chacune des périodes T_1 , T_2 et T_3 et des distances à parcourir différentes. Cette figure est basée sur le profil de vitesses de la figure 3.2. Par exemple, si la distance à parcourir est 100Km et l'heure de départ est 12H (située en période T_1), le temps total de trajet est de 2 heures.

Dans la figure 3.3, v représente la vitesse de chaque période, b représente les instants de début et de fin de chaque période, t_0 l'instant de départ du véhicule, d_{ij} la distance entre i et j , t le temps total de trajet (résultat final).

3.3.2 Matrice des profils de vitesse

Afin de respecter une condition de vitesse moyenne identique à celle employée par [Ichoua et al. 2003], le profil de vitesse traduit la variation de vitesse sous la forme d'une fonction constante par morceaux selon la période du jour considéré. Dans notre problème il y a 3 périodes qui ont la même durée. Les première et troisième périodes représentent les heures de pointe du matin et du soir. La deuxième période correspond au milieu de journée, quand la densité du trafic est moindre. Les vitesses de trajet dans les heures de pointe correspondent aux vitesses de trajet en milieu de journée divisées par un facteur de ralentissement α .

Un problème constitué de n nœuds peut donc être représenté par une matrice symétrique de profil de vitesse modélisant le graphe complet du réseau. Par exemple, la figure 3.4, contient 4 clients (A, B, C, D) et 3 profils distincts : le profil '1' est associé à la liaison entre les nœuds A et B ; le profil '2' est associé à la liaison entre les nœuds B et C etc. Nous obtenons la matrice

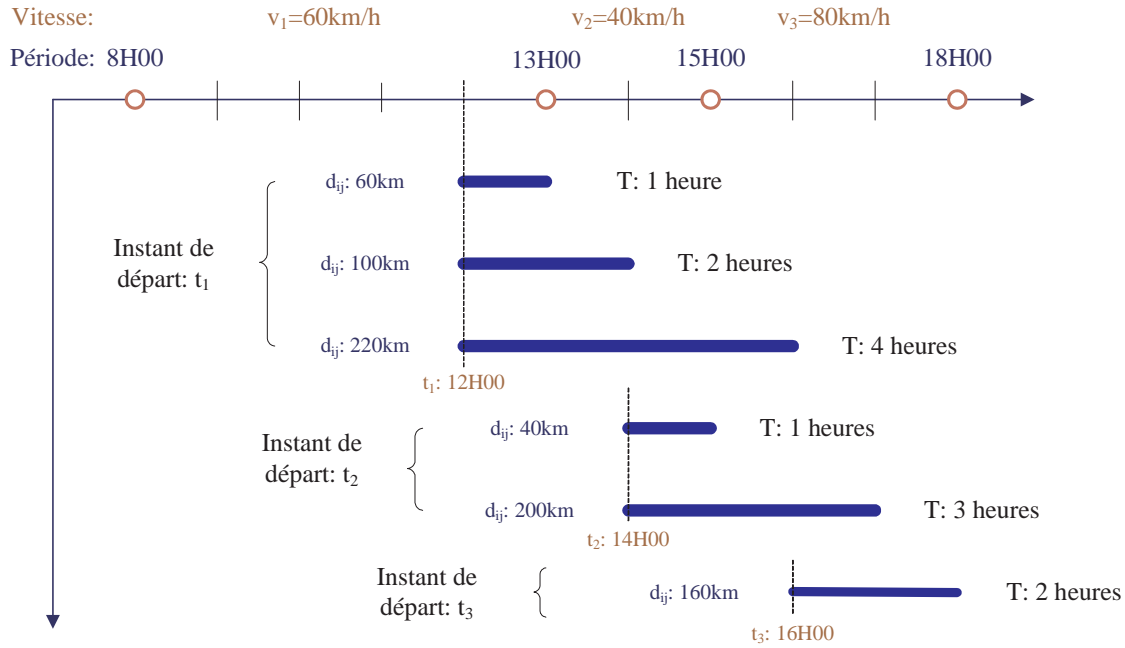


FIG. 3.3 – Exemples de temps de trajet

des profils des temps de trajet représentée par le tableau 3.1. Il n'y a pas de profil entre chaque nœud et lui-même. Donc la matrice contient des '0' dans la diagonale. L'affectation des types de profils sur les arcs du réseau suit une loi aléatoire uniforme.

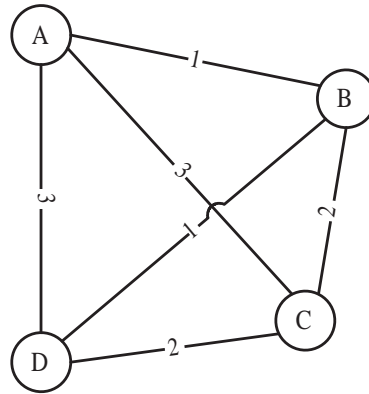


FIG. 3.4 – Exemple de graphe avec des profils de routes différents

Nous avons créé 3 profils différents qui correspondent à 3 types différents de routes dont les vitesses moyennes varient de lent à rapide.

Par ailleurs, nous avons élaboré 3 types de scénarios pour faire varier le facteur de ralentissement α . Dans les scénarios 1, 2 et 3, α vaut respectivement 1,5, 2 et 4. La vitesse moyenne dans chaque scénario est supposée égale à 40km/h comme dans le cas des problèmes à vitesse constante.

Le tableau 3.2 présente les valeurs correspondantes de ces trois scénarios. Dans ce tableau,

	A	B	C	D
A	0	1	3	3
B	1	0	2	1
C	3	2	0	2
D	3	1	2	0

TABLEAU 3.1 – Matrice des profils de vitesse

chaque ligne correspond à un profil de vitesse pour un arc donné et chaque colonne à une période de temps.

	Profil	T1(08H00-10H40)	T2(10H40-13H20)	T3(13H20-16H00)
Scénario 1 $\alpha=1,5$	1	21,6	32,4	21,6
	2	32,4	48,6	32,4
	3	48,6	72,9	48,6
Scénario 2 $\alpha=2$	1	13,2	26,8	13,2
	2	26,8	53,2	26,8
	3	53,2	106,8	53,2
Scénario 3 $\alpha=4$	1	4,8	18,4	4,8
	2	18,4	76,8	18,4
	3	38,4	153,6	38,4

TABLEAU 3.2 – Profils de vitesse de trajet pour les scénarios 1 à 3

3.3.3 Les stratégies modifiées

Dans le contexte étudié qui met en œuvre un modèle de vitesses dépendant du temps, il est nécessaire de faire évoluer les stratégies présentées ci-dessus (la section 3.2.1) afin de prendre en compte cette caractéristique. A titre d'exemple, la stratégie NN du plus proche voisin basée sur la distance à parcourir, a été remplacée par une stratégie de plus proche voisin basée sur le temps de trajet.

Les stratégies modifiées pour prendre en compte le nouveau modèle de calcul de temps de trajet dépendant des temps sont décrites de manière simplifiée ci-dessous :

Algorithme 3.1 Stratégie NN basée sur le temps de trajet

1. Le temps de trajet est initialement vide (pas de nœud visité) ;
 2. Calculer les temps de trajet entre le dépôt et les nœuds voisins selon le profil de vitesse correspondant ;
 3. Ajouter au dernier nœud du trajet en cours d'élaboration, le temps de déplacement vers le voisin le plus proche dans le temps de trajet. Si le nœud est déjà dans la liste des nœuds visités, on prend le deuxième nœud le plus proche, le troisième si besoin, etc ;
 4. Répéter 2 et 3 jusqu'à ce qu'il n'y ait plus de nœud disponible ;
 5. Relier le dernier nœud au premier nœud.
-

Algorithme 3.2 Stratégie PartNN2 basée sur le temps de trajet

1. Découper l'espace en 4 partitions égales ;
 2. Le premier client choisi est le client dont le temps de trajet depuis le dépôt est le plus court. Dans la partition de ce client on adopte une stratégie NN basée sur le plus court temps ;
 3. Lorsque tous les clients dans une sous-région sont servis, le véhicule voyage à la prochaine sous-région ;
 4. Lorsque l'on change de sous-région, on va dans la sous-région comportant le plus de requêtes.
-

3.4 Plate-forme et résultats

3.4.1 La plate-forme développée

Nous avons développé une plateforme Java pour ce problème. Cette plate-forme intègre les différentes stratégies FCFS, NN, SFSC, TSP, GA, Part, PartNN et PartNN2. Seules les deux stratégies NN et PartNN2 prennent en compte le modèle de temps de trajet variable.

Nous avons utilisé le benchmark proposé par [Larsen 2001] pour examiner le comportement de notre approche pour la résolution de ce type de problème. Spécifiquement, nous nous sommes concentrés sur l'impact du degré de dynamisme, sur la distance parcourue et le temps d'attente des clients. Larsen a généré un ensemble de benchmarks dynamiques pour ce problème. Il a tout d'abord choisi des positions géographiques des clients uniformément distribués dans une région $[0, 10000] \times [0, 10000]$, avec le dépôt situé au centre en position (5000, 5000). La vitesse des véhicules a été fixée à 40km/heure. Il a employé une échelle de graduation de 100 unités par kilomètre ; la région de service est donc équivalente à $[10kilomtres \times 10kilomtres]$. Les temps de service ont été produits selon une distribution logarithmique normale. Ceci signifie que le temps moyen de service était de 3 minutes, tandis que la variance était de 5 minutes. Ces valeurs ont été choisies pour ressembler aux temps de ramassage et de livraison dans les services de courrier longue distance.

Deux groupes d'instances dynamiques de PDTRP ont été exploités. Ces deux groupes sont constitués de 30 clients et de 50 clients. Chaque groupe contient des problèmes ayant des degrés de dynamisme différents (10%, 20%, ..., 100%), soit 10 sous groupes au total. Chacun de ces sous groupes possèdent 100 problèmes différents. Donc chaque groupe contient au total $10 \times 100 = 1000$ problèmes. La journée de service commence à 8h et se termine à 16h : le temps d'ouverture de la journée est donc égal à 8 heures. Pour chaque problème, on a lancé 5 exécutions (runs) différents pour tenir compte du caractère stochastique des stratégies utilisées.

Les stratégies NN et PartNN2 ont été testées sur le benchmark de 30 clients et de 50 clients. Un exemple de benchmark est présenté comme le tableau A.1 dans l'annexe A.1. Les résultats (indicateurs de performances) fournis sont le temps de trajet moyen de véhicule et le temps d'attente moyen des clients.

Les essais réalisés utilisent les scénarios S1, S2, S3 décrits dans la section 3.3.2.

Les résultats (indicateurs de performances) fournis sont le temps de trajet moyen et le temps d'attente moyen.

Le caractère dynamique du problème ne permet pas d'utiliser des méthodes de résolution

exactes. Pour évaluer la qualité des solutions fournies par notre approche, nous avons cherché à comparer la robustesse des solutions fournies par les deux modes de résolution suivants [Ichoua et al. 2003] :

- Résolution 1 : la résolution prend en compte un environnement avec un temps de trajet variable selon un profil de vitesse déterminé, soit S la solution obtenue. Cette solution fournit les indicateurs (temps de trajet et temps d'attente moyens) R1 ;
- Résolution 2 : la résolution ne prend pas en compte la variabilité de la vitesse. L'évaluation de la qualité de la solution doit alors se faire en 2 étapes. a. Le problème est résolu en supposant que le temps de trajet est constant : soit S' la solution obtenue ; b. Le parcours S' , ainsi obtenu, est évalué par simulation en le « replongeant » dans un environnement temps de trajet variable respectant les profils de vitesse précédents. Cette solution fournit les indicateurs R2.

La comparaison des indicateurs R1 et R2 obtenus par ces 2 modes de solution permet d'évaluer l'apport de la prise en compte d'un modèle temps de trajet variable ou la dégradation observée sur un modèle à vitesse constante.

Le processus de comparaison des deux résolutions est présenté sur la figure 3.5.

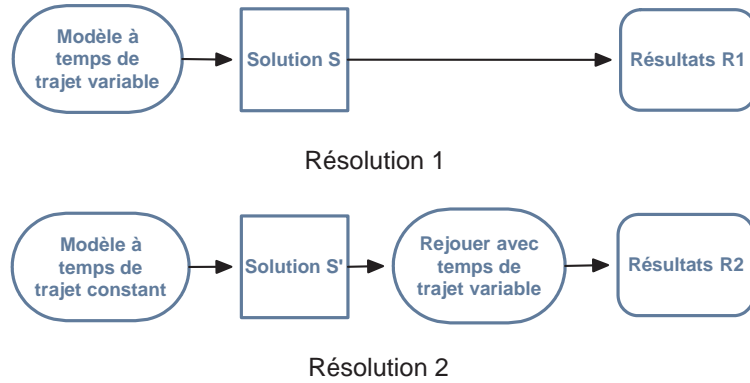


FIG. 3.5 – Deux modes de résolution

3.4.2 Résultats et analyses de la simulation

Une présentation partielle des résultats (Jeu de 100 problèmes à 30 clients pour un dod 50% et pour la stratégie NN) est donnée comme le tableau 3.3. Les résultats complets peuvent être trouvés en l'annexe A.2.

En ce qui concerne le jeu de 100 problèmes à 30 clients pour un taux de dynamisme de 50% et la stratégie NN (tableau 3.3), il apparaît clairement que la résolution 1 basée sur une planification utilisant le modèle de temps variables donne en moyenne de meilleurs résultats que la résolution 2 basée sur la planification utilisant le modèle de temps non variables.

Pour évaluer plus finement cet apport, nous utilisons de plus un indice de « Gain » pour décrire le niveau d'amélioration des résultats qui est défini comme suit :

$$\text{Gain} = \frac{M_{R2} - M_{R1}}{M_{R1}} \quad (3.1)$$

Où :

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
50%	S1	TT-R1 ^a	23409,87	3993,15	0,14
		TT-R2 ^b	23443,09	4045,97	
		TA-R1 ^c	1059,26	167,16	0,30
		TA-R2 ^d	1062,42	166,90	
	S2	TT-R1	23588,63	4205,28	0,80
		TT-R2	23777,97	4388,95	
		TA-R1	1106,40	168,51	1,12
		TA-R2	1118,79	170,06	
	S3	TT-R1	24698,54	4827,02	7,65
		TT-R2	26588,35	7135,40	
		TA-R1	1446,69	201,57	11,73
		TA-R2	1616,39	306,60	
	Moyenne	TT-R1	23899,01	4341,82	2,95
		TT-R2	24603,14	5190,11	
		TA-R1	1204,12	179,08	5,13
		TA-R2	1265,86	214,52	

^a Temps de trajet pour la résolution 1

^b Temps de trajet pour la résolution 2

^c Temps d'attente pour la résolution 1

^d Temps d'attente pour la résolution 2

TABLEAU 3.3 – Jeu de 100 problèmes à 30 clients pour un dod 50% : stratégie NN

- M_{R1} : Valeur moyenne de R1 ;
- M_{R2} : Valeur moyenne de R2 ;

Les figures 3.6, 3.7 présentent respectivement les valeurs de « Gain » de temps de trajet et temps d'attente pour les 3 scénarios en fonction du degré de dynamisme (dod) des problèmes à 30 clients et en utilisant la stratégie NN.

En ce qui concerne les figures 3.6 et 3.7, on remarque que la stratégie NN donne de très bons résultats avec les 3 scénarios. Pour le scénario 3, le Gain est plus important que pour les autres deux scénarios. Ce résultat met en évidence que plus les profils de vitesses variables ont un facteur de ralentissement important, plus le niveau d'amélioration du modèle avec temps de trajet dépendant du temps est significatif.

Pour les autres stratégies et pour le cas comportant 50 clients, les gains sont présentés dans les figures 3.8, 3.9, 3.10, 3.11, 3.12 et 3.13. Les mêmes conclusions que précédemment peuvent être effectuées.

3.5 Conclusion

Les résultats obtenus montrent donc que le modèle à vitesse variable basé sur des profils historiques présentent un intérêt légitime pour prendre en compte les fluctuations de la vitesse d'un véhicule. La prise en compte de cette connaissance est donc significative, les solutions

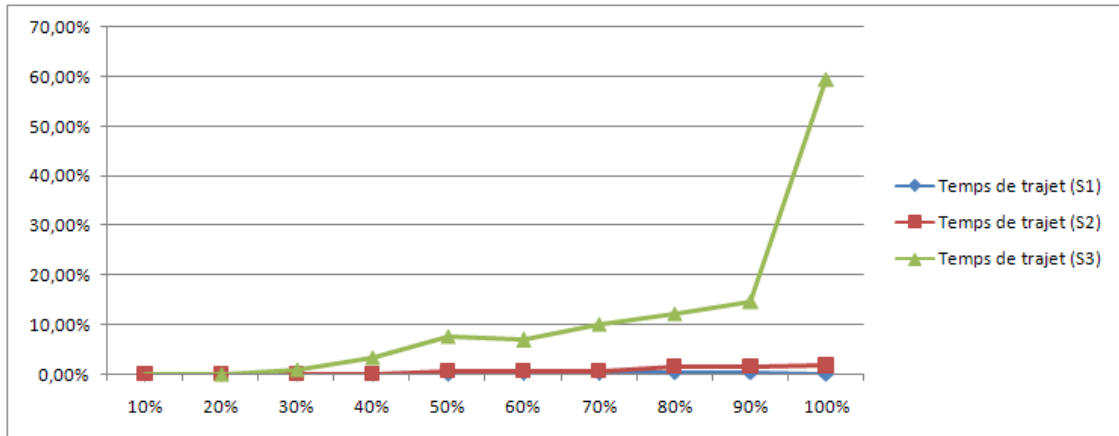


FIG. 3.6 – Gain de « temps de trajet » pour la stratégie NN, 30 clients

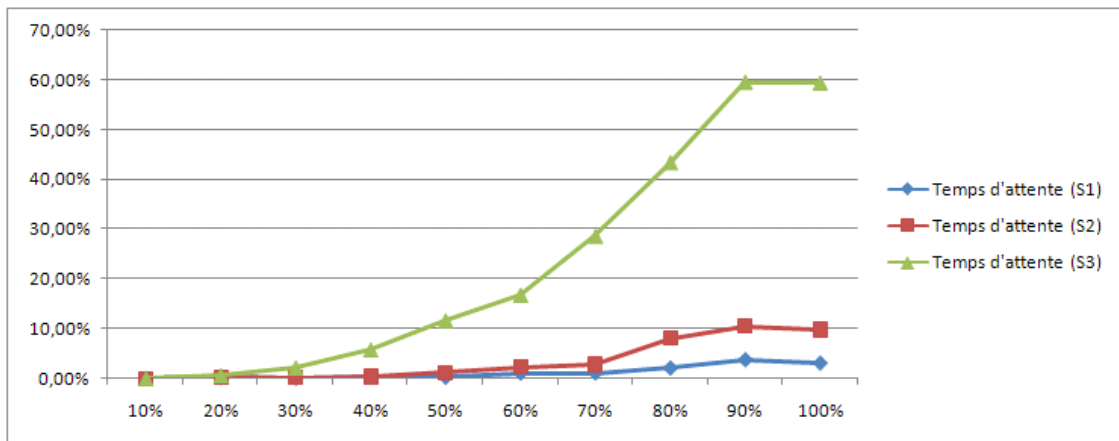


FIG. 3.7 – Gain de « temps d'attente » pour la stratégie NN, 30 clients

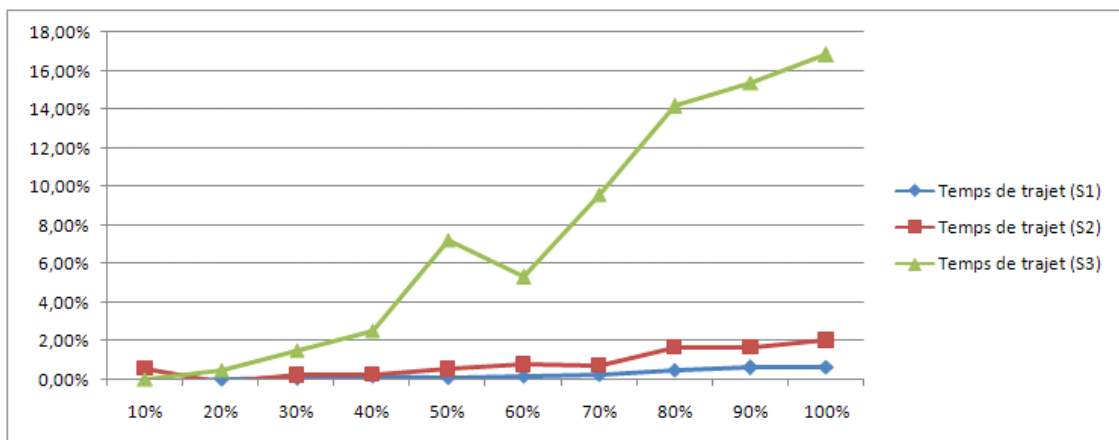


FIG. 3.8 – Gain de « temps de trajet » pour la stratégie PartNN2, 30 clients

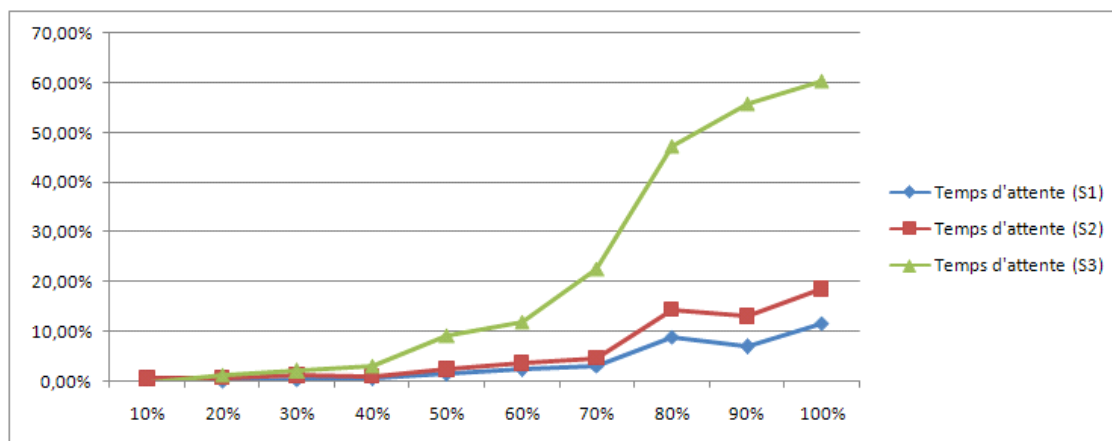


FIG. 3.9 – Gain de « temps d'attente » pour la stratégie PartNN2, 30 clients

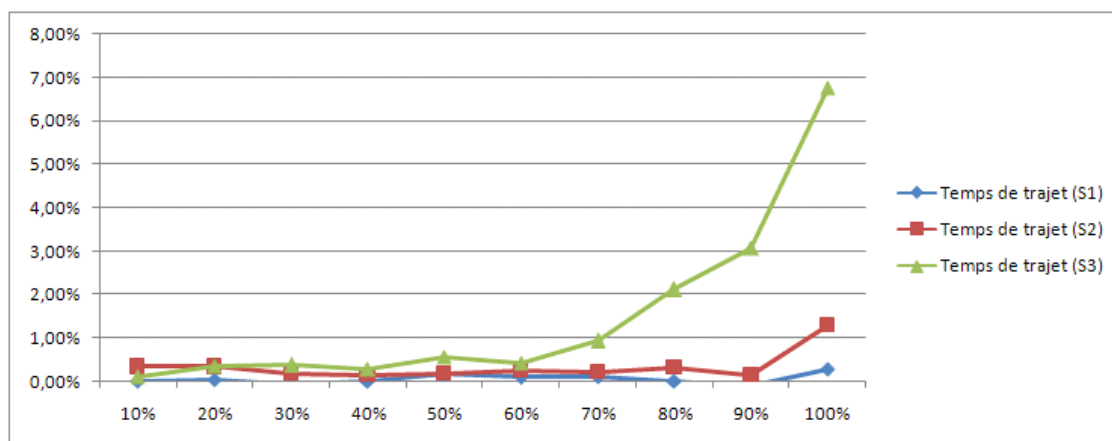


FIG. 3.10 – Gain de « temps de trajet » pour la stratégie NN, 50 clients

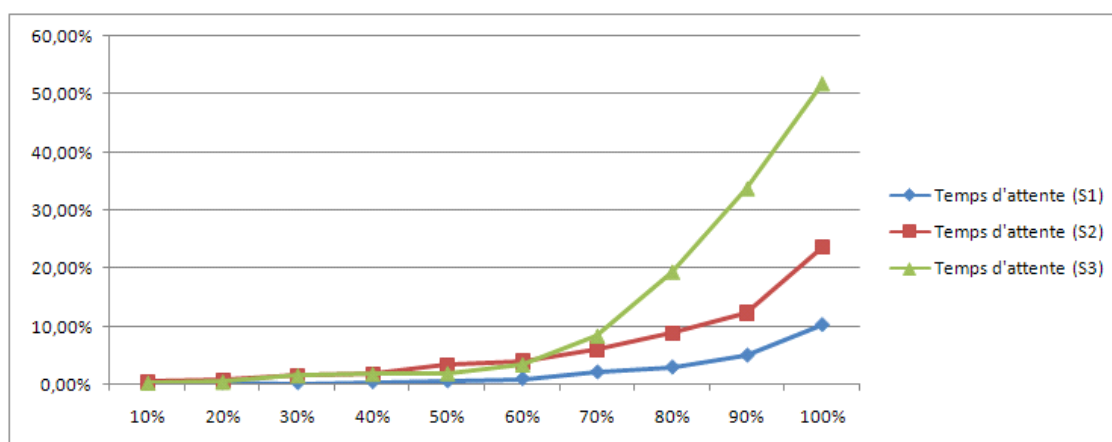


FIG. 3.11 – Gain de « temps d'attente » pour la stratégie NN, 50 clients

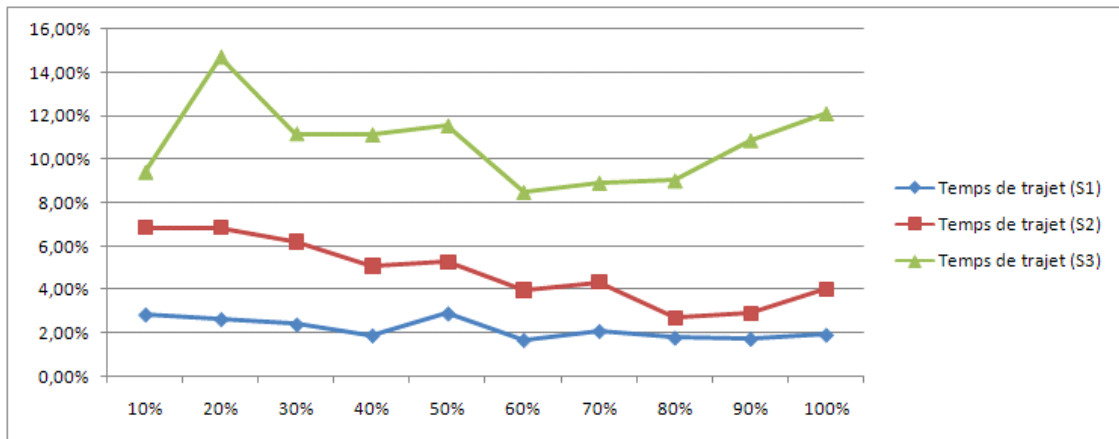


FIG. 3.12 – Gain de « temps de trajet » pour la stratégie PartNN2, 50 clients

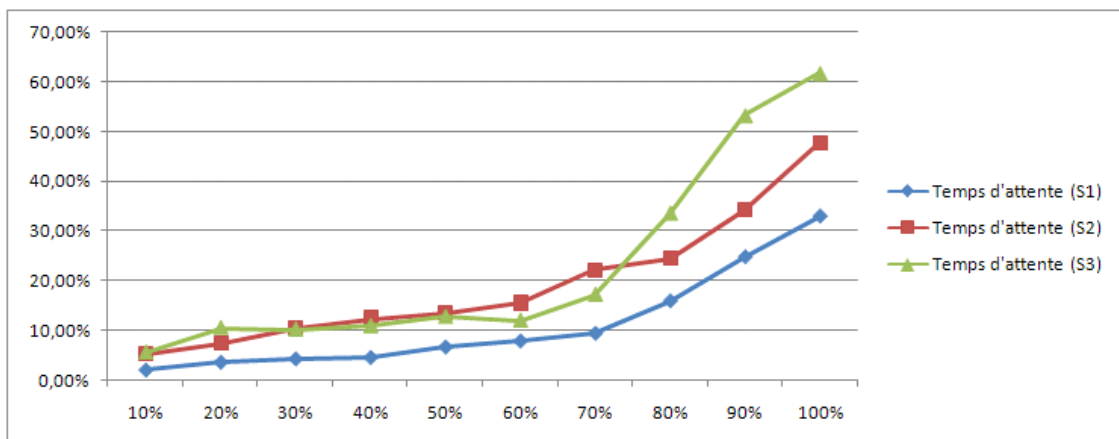


FIG. 3.13 – Gain de « temps d'attente » pour la stratégie PartNN2, 50 clients

obtenues sont plus robustes si on les compare avec celles fournies avec un modèle à vitesse variable. Le gain est d'autant plus important que le degré de dynamisme est élevé.

En ce qui concerne les performances relatives des différentes stratégies testées, les stratégies NN et PartNN2 ont permis d'obtenir des résultats meilleurs que ceux obtenus avec un modèle à temps de trajet constant. Ce travail permet donc une modélisation plus fine des problèmes de routage de véhicules de type tournée du réparateur qui s'approche davantage de la réalité du terrain.

Chapitre 4

Le problème de la gestion dynamique de tournées de véhicules avec fenêtres de temps et temps de trajet dépendant du temps

Sommaire

4.1	Introduction	64
4.2	Le problème VRPTW	64
4.2.1	Formulation du VRPTW	65
4.2.2	Revue de littérature sur le VRPTW	66
4.3	Le problème DVRPTW avec temps de trajet dépendant du temps	69
4.4	Algorithme Génétique pour le problème DVRPTW	69
4.4.1	Introduction	69
4.4.2	Le codage de la solution	70
4.4.3	Fonction d'évaluation	71
4.4.4	Décodage d'un chromosome	72
4.4.5	Population initiale	74
4.4.6	Opérateurs de Sélection	74
4.4.7	Opérateurs de croisement (Crossover)	75
4.4.8	Opérateurs de mutation	78
4.4.9	Méthode de remplacement par l'élitisme	80
4.5	Plate-forme de Simulation Dynamique	80
4.5.1	Le composant Algorithme Génétique (AG)	81
4.5.2	Le composant Gestion d'Evénements (GE)	81
4.5.3	Gestion du temps	82
4.5.4	Planification des tournées	82
4.5.5	L'évènement «fin de service»	84
4.5.6	L'évènement «arrivée d'un nouveau client»	84
4.6	Résolution du DVRPTW avec temps de trajet dépendant du temps	87
4.6.1	Les paramètres de l'algorithme génétique	87
4.6.2	Les profils de vitesse dépendant du temps	87
4.6.3	Benchmark utilisé	88

4.6.4	Méthode de validation	90
4.6.5	Les résultats de simulation	90
4.7	Conclusion	92

4.1 Introduction

Dans ce chapitre, nous allons d'abord présenter le problème de la gestion dynamique de tournées de véhicules avec fenêtres de temps (DVRPTW) dans la section 4.2. Le problème DVRPTW utilisant un modèle de temps de trajet dépendant du temps qui est traité dans le cadre de notre travail sera présenté dans la section 4.3. L'approche développée qui se base sur un algorithme génétique intégrant ce modèle de temps de trajet est décrite dans la section 4.4. Puis une architecture de simulation est présentée dans la section 4.5. Les solutions obtenues pour le problème à traiter seront évaluées dans la section 4.6. Dans cette section, nous allons aussi présenter les jeux de tests statiques et dynamiques (benchmarks) que nous avons utilisés pour évaluer nos modèles. Nous terminons ce chapitre par une conclusion dans la section 4.7.

4.2 Le problème VRPTW

Le Problème de Tournées de Véhicules avec Fenêtre de Temps (Vehicle Routing Problem with Time Windows - VRPTW, [Solomon 1987], [Duhamel 2001]), constitue une généralisation du VRP dans la mesure où nous introduisons en plus une contrainte temporelle sur le service demandé. Chaque client dispose d'une fenêtre de temps à l'intérieur de laquelle il désire être servi. Le dépôt central possède également une fenêtre de temps que nous désignons couramment comme horizon de service ou temps d'ouverture de la journée. Son rôle est de fixer une plage horaire durant laquelle les véhicules peuvent effectuer leur tournée. Ces contraintes temporelles vont rendre nécessaire l'utilisation de plusieurs véhicules pour satisfaire l'ensemble des clients sur l'horizon de service. On peut vouloir borner le nombre de véhicules à utiliser et dans ce cas des clients risquent de ne pas être servis.

Bien que simple en apparence, ce problème est notablement très difficile à résoudre. En fait, il a été montré que le problème VRP classique était NP-difficile, et que ce résultat pouvait être étendu au VRPTW. Ainsi, si il est tout à fait possible de déterminer une solution optimale pour des instances de petite taille, cela devient rapidement irréalisable pour des instances de moyenne ou de grande taille. Or la plupart des problèmes de la vie réelle se situe justement dans cette seconde catégorie. En dépit du caractère fortement restrictif de sa définition, le problème VRPTW conserve un pouvoir descriptif important. Il permet de modéliser un panel étendu d'applications réelles [Duhamel 2001] :

- **service postal** : il s'agit d'établir des tournées pour les facteurs de manière à distribuer le courrier le plus efficacement possible, mais on peut aussi réaliser des tournées pour effectuer le ramassage de courrier dans les boîtes aux lettres publiques.
- **coopératives agricoles** : il s'agit de mettre en place des tournées de véhicules pour collecter la production de différents exploitants agricoles. Ceci peut concerner le lait, les produits céréaliers, les produits maraîchers ou encore le ramassage d'animaux par exemple.
- **service bancaire** : il s'agit de déterminer des routes pour les convoyeurs de fond de manière à réaliser la collecte des recettes des commerçants et des grandes surfaces.
- **ramassage scolaire** : il s'agit d'établir les itinéraires de bus pour amener les écoliers des points de collecte vers l'école le matin et inversement le soir.

– etc.

Ces quelques exemples permettent d'illustrer la profonde implication de ce problème dans la vie courante.

4.2.1 Formulation du VRPTW

Nous formulons le problème VRPTW en utilisant la formulation de VRP classique déjà exprimée et en y ajoutant les variables, les constantes et les équations suivantes [LeBouthillier 2000] :

Variables à déterminer :

1. a_i = instant d'arrivée chez le client $i \in C$.
2. b_i = instant de début de service chez le client $i \in C$.
3. b_0^v = instant auquel le véhicule v quitte le dépôt.
4. $b_{n_c+1}^v$ = instant auquel le véhicule v retourne au dépôt.
5. w_i = temps d'attente chez le client $i \in C$.

Constantes connues :

1. e_i = borne inférieure de la fenêtre de temps du client $i \in C$.
2. l_i = borne supérieure de la fenêtre de temps du client $i \in C$.
3. $c_{i,j}$ = coût du déplacement de i à j : $i, j \in C$.
4. $t_{i,j}$ = le temps de parcours entre les deux clients i et j , $i, j \in C$.
5. s_i = temps de service chez le client $i \in C$.

L'attente est permise lorsqu'un véhicule arrive trop tôt chez le client j après que le service soit fini chez le client $i \in C$, autrement dit, avant e_j . Le temps de début de service chez le client $j \in C$ se définit comme étant $b_j = \max\{e_j, a_j\}$ où $a_j = \{b_i + s_i + t_{i,j}\}$ et le temps d'attente chez le client j comme étant $w_j = b_j - a_j$. Il est alors possible d'écrire les contraintes supplémentaires de la formulation de VRP pour formuler le problème VRPTW :

$$x_{i,j}^v = 1 \Rightarrow b_i + s_i + t_{i,j} \leq b_j, \forall i, j \in C, v \in V \quad (4.1)$$

$$x_{0,j}^v = 1 \Rightarrow b_0^v + t_{0,j} \leq b_j, \forall j \in C, v \in V \quad (4.2)$$

$$x_{i,n+1}^v = 1 \Rightarrow b_i + s_i + t_{i,n+1} \leq b_{n+1}^v, \forall i \in C, v \in V \quad (4.3)$$

$$e_i \leq b_i \leq l_i, \forall i \in C \quad (4.4)$$

$$e_0 \leq b_0^v \leq l_0, \forall v \in V \quad (4.5)$$

$$e_{n+1} \leq b_{n+1}^v \leq l_{n+1}, \forall v \in V \quad (4.6)$$

Le temps d'utilisation réel des véhicules dans la solution (X) est :

$$\text{temps horaire}(X) = \sum_{v \in V} (b_{n_c+1}^v - b_0^v) \quad (4.7)$$

Les contraintes (4.1 - 4.6) définissent les contraintes temporelles. La mesure (4.7) permet d'identifier le temps total d'utilisation des véhicules.

Dans un VRP, la définition de distance entre deux clients qui nous vient naturellement à l'esprit est la distance que requiert le parcours direct entre ces deux clients. Cette distance est la même dans les deux sens, autrement dit, la distance entre le client i et j est la même entre j et i , donc la matrice des distances entre les clients est symétrique. Dans les algorithmes utilisés pour le VRPTW, cette même métrique de distance est généralement utilisée.

4.2.2 Revue de littérature sur le VRPTW

Dans la littérature, nous constatons que le problème VRPTW est le plus étudié parmi les extensions du VRP. Comme pour le problème VRP, nous avons deux catégories de méthodes de résolution qui traitent le VRPTW :

4.2.2.1 Méthodes exactes

Les approches basées sur des *méthodes exactes* ont tardé à apparaître. Une des raisons principales tient à la difficulté de trouver un modèle suffisamment performant pour exprimer le VRPTW. En particulier la prise en compte des contraintes de capacité et de fenêtre temps reposent sur un mécanisme récursif : l'application des contraintes sur un client dépend du résultat sur le client précédent. Nous trouverons dans l'article de [Cordeau et al. 2002] une excellente revue de littérature sur le sujet. Nous pouvons classer les différentes approches en quatre grandes catégories :

- **Programmation dynamique** : [Kolen et al. 1987] reprend le principe de la méthode de relaxation mise au point par [Christofides et al. 1981] pour le problème VRP. Les décisions de branchement portent sur l'affectation des clients aux tournées. Cette approche leur a permis à l'époque de résoudre optimalement des problèmes comptant jusqu'à 15 clients. Notons que c'est l'un des travaux les plus anciens sur le VRPTW.
- **Génération de colonnes** : [Desrochers et al. 1992] propose l'application de la technique de génération de colonnes sur la relaxation du VRPTW exprimé comme problème de set covering. Cette approche, très performante, leur autorise la résolution d'instances comprenant jusqu'à 100 clients.
- **Décomposition Lagrangienne** : cette approche repose sur un schéma de partage des variables. Ceci permet d'obtenir des bornes inférieures pour la solution, que l'on peut ensuite intégrer dans un mécanisme de Branch and Bound. [Jörnsten et al. 1986], [Madsen 1990], et [Fisher et al. 1997] proposent chacun un moyen différent de réaliser la décomposition Lagrangienne. Des instances de 100 clients ont ainsi pu être résolues.
- **Relaxation Lagrangienne** : elle a été utilisée par Kohl et Madsen [Kohl et al. 1997] ainsi que par Fisher [Fisher et al. 1997]. Cette approche permet de résoudre quelques problèmes à 100 clients.

Actuellement, aucune approche basée sur le Branch and Cut ne semble avoir été tentée pour le VRPTW. La raison principale réside certainement dans la très grande complexité à pouvoir exhiber des facettes du polyèdre des contraintes.

Notons que les trois premières catégories font appel à un sous-problème de «Plus Court Chemin avec Capacité et Fenêtre de Temps». Il est utilisé soit pour la génération des colonnes, soit pour mettre à jour la relaxation Lagrangienne. Bien qu'il soit reconnu NP-complet, ce sous-problème peut être aisément résolu par des algorithmes pseudo-polynômiaux à base de programmation dynamique.

4.2.2.2 Méthodes approchées

Du fait de sa complexité intrinsèque, le VRPTW constitue un champ d'investigation particulièrement intéressant et fécond pour les méthodes approchées. [Desrosiers et al. 1995], ainsi que [Fisher 1995] présentent des états de l'art précis sur le sujet. Nous nous contenterons de relever les contributions les plus significatives dans chaque catégorie :

1. Heuristiques de **construction de tournées** : reprenant les travaux précurseurs de Clarke et Wright [Clarke et al. 1964] sur le problème VRP, Solomon [Solomon 1987] décrit quelques méthodes de construction de tournées reposant sur l'insertion séquentielle des clients. [Potvin et al. 1993], puis [Russell 1995] proposent des méthodes de construction de tournées en parallèle. Notons qu'une analyse dans le pire des cas a été développée par Solomon [Solomon 1986] pour certaines heuristiques simples.
2. Heuristiques **d'amélioration de tournées** : ces méthodes partent d'une solution initiale obtenue par une heuristique du type précédent et l'améliorent progressivement en procédant à des échanges d'arcs. Les différents opérateurs proposés sont presque tous basés sur le principe du voisinage $\lambda - Opt$ [Lin 1965] qui consiste à échanger deux suites de clients de taille λ , tout en intégrant les contraintes de fenêtre de temps et de capacité.
3. Heuristiques **composites** : à la différence des heuristiques d'amélioration de tournées, qui nécessitent la connaissance d'une solution initiale, les heuristiques composites combinent une phase de construction et une phase d'amélioration des tournées. Elles sont d'inspiration relativement récente bien qu'on puisse remonter aux travaux de [Gillett et al. 1974]. Le lecteur pourra se référer aux travaux de [Derigs et al. 1993], [Kontoravdis et al. 1992], ou encore [Russell 1995] pour plus de détails.
4. Heuristiques **basées sur un principe exact** : elles forment une classe à part dans la mesure où elles sont déduites de méthodes exactes. Elles reprennent en fait un schéma d'optimisation exact, avec quelques aménagements, en particulier au niveau des critères décisionnels. Ainsi [Koskosidis et al. 1992] utilise des critères heuristiques pour réaliser l'assignation des clients aux tournées. Après décomposition, ces dernières sont ensuite obtenues via une méthode exacte de routage.
5. **Métaheuristiques** : elles constituent la branche la plus récente dans le domaine des méthodes approchées et sont réputées pour leur excellent niveau de performances. [Potvin et al. 1996a] comme [Rochat et al. 1995] ont proposé une résolution par recherche tabou. [Russell 1995] a eu recours au recuit simulé tandis que [Kontoravdis et al. 1995] a appliqué GRASP⁵ (Greedy Randomized Adaptive Search Procedure). De nombreux travaux ont aussi porté sur l'utilisation d'algorithmes génétiques pour résoudre ce problème tels que ceux de [Blanton et al. 1993], [Potvin et al. 1996b] ainsi que ceux de [Thangiah 1995].

⁵le principe de GRASP consiste simplement à l'utilisation alternée de deux mécanismes, la construction d'une solution faisable et la recherche locale pour trouver l'optimum local

Notons également le travail de [Bramel et al. 1993]. En se basant sur une analyse probabiliste du problème VRPTW, ils ont mis au point des heuristiques convergeant asymptotiquement vers la solution optimale. L'intérêt théorique de cette approche est indéniable puisque l'on dispose d'une garantie sur la qualité du résultat. Malheureusement, à l'instar du théorème de [Hajek 1988] pour le recuit simulé, il y a peu de chances pour que ce résultat puisse déboucher sur un algorithme combinant les avantages des méthodes exactes et approchées.

De ces différentes approches, il ressort que les métaheuristiques sont celles qui tirent le mieux leur épingle du jeu. Elles offrent une qualité de résultats bien supérieure aux autres approches, au prix il est vrai de temps de calculs parfois conséquents.

Enfin, nous remarquons que dans la littérature, les travaux traitant le problème VRPTW n'ont pas utilisé la même fonction objectif ; dans la table (4.1) nous présentons quelques références sur le problème VRPTW avec leurs fonctions objectifs ordonnées par priorité.

Référence	Fonctions objectifs
[Solomon 1987]	1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue, 3 - Minimiser le temps d'attente des clients.
[Desrochers et al. 1992]	1 - Minimiser la distance totale parcourue.
[Potvin et al. 1993]	1 - Minimiser le nombre de véhicules, 2 - Minimiser le temps de parcours.
[Thangiah et al. 1994]	1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue.
[Russell 1995]	1 - Minimiser le nombre de véhicules,
[Rochat et al. 1995]	1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue.
[Potvin et al. 1996a]	1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue.
[Potvin et al. 1996b]	1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue.
[Chiang et al. 1996]	1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue.
[Badeau et al. 1997]	1 - Minimiser la distance totale parcourue.
[Kohl et al. 1997]	1 - Minimiser la distance totale parcourue.
[Taillard et al. 1997]	1 - Minimiser la distance totale parcourue, 2 - Minimiser le retard total chez les clients.
[Kilby et al. 1998b]	1 - Minimiser la distance totale parcourue.
[Homberger et al. 1999]	1 - Minimiser la distance totale parcourue, 2 - Minimiser le nombre de véhicules.
[Sushil et al. 1999]	1 - Minimiser la distance totale parcourue, 2 - Minimiser le retard total chez les clients, 3 - Minimiser le retard au dépôt.
[Bräysy et al. 2000]	1 - Minimiser la distance totale parcourue.
[Tan et al. 2001]	1 - Minimiser la distance totale parcourue.

TABLEAU 4.1 – Les objectifs choisis pour la résolution du VRPTW

4.3 Le problème DVRPTW avec temps de trajet dépendant du temps

Nous considérons dans ce travail le problème VRPTW en mode dynamique (DVRPTW).

Le problème que nous traitons considère une extension particulière du VRPTW dans laquelle les temps de trajets sont supposés variables dans le temps. Ce problème, comme celui étudié au chapitre 3, consiste à se rapprocher de la réalité en considérant que les trajets ne sont plus effectués à vitesse constante sur toute la journée, mais seulement sur des plages horaires prédéfinies (i.e. vitesse constante par morceaux). Ces plages correspondent aux différentes périodes d'une journée (heures de pointes ou heures creuses).

Plus précisément, le problème DVRPTW avec temps de trajet dépendant du temps peut être défini comme suit :

1. Les véhicules se déplacent à vitesse variable pour effectuer un ensemble d'interventions chez des clients répartis géographiquement ;
2. Les demandes émanant des clients peuvent être connues à l'avance, par exemple la veille au soir pour être planifiées le jour suivant (demandes statiques) ; Sinon elles peuvent arriver au cours de la journée pendant l'exécution des tournées (i.e. demandes dynamiques) ;
3. Dans ce dernier cas, elles seront, si possible, intégrées dans une des tournées existantes sinon elles seront traitées le jour suivant comme une demande statique ;
4. L'objectif du problème est de minimiser le temps de tournée des véhicules et les retards clients ; en effet dans notre problème, les contraintes associées aux fenêtres de temps sont des contraintes dites « souples » car elles peuvent ne pas être respectées, ce qui engendre un retard. La difficulté sera alors de limiter ces retards qui seront intégrés comme des pénalités dans la fonction objectif à optimiser.

Dans ce chapitre, nous réutilisons le modèle de temps de trajet dépendant du temps qui est présenté au chapitre 3.

4.4 Algorithme Génétique pour le problème DVRPTW

La résolution du problème dynamique de tournées de véhicules avec temps de trajet dépendant du temps que nous abordons s'appuie sur une plate-forme de simulation développée au laboratoire LIG2A dans le cadre de la thèse de [Housroum 2005]. Mon travail a consisté à intégrer un modèle de temps de trajet dépendant du temps dans cette plate-forme de résolution développée en Java. Cette plate-forme est basée sur la mise en œuvre d'un Algorithme Génétique fonctionnant dans un environnement dynamique. La comparaison de cette approche AG avec une heuristique simple et avec une méthode Tabu a montré que l'algorithme génétique fournit de meilleurs résultats que l'heuristique et des résultats comparables et dans certains cas meilleurs que la méthode Tabu. Par conséquent, l'algorithme génétique est une méthode alternative qui est efficace et souple pour résoudre ce type de problème dynamique. Nous reprenons ci-dessous les éléments de l'algorithme génétique qui ont été impactés par cette intégration ainsi que les éléments nécessaires à la compréhension générale du modèle.

4.4.1 Introduction

C'est Holland qui a formulé le premier, le principe des Algorithmes Génétiques en 1975 [Holland 1975]. [Goldberg 1989] apporta également beaucoup à la création de ces algorithmes.

Les algorithmes génétiques font partie des algorithmes dits « *évolutionnistes* ». Ces algorithmes sont classés dans les méthodes d'optimisation stochastiques et peuvent opérer dans des espaces de recherche extrêmement vastes. Le terme évolutionniste fait référence aux techniques basées sur les mécanismes de l'évolution naturelle.

Un algorithme génétique est capable de fournir de bonnes solutions aux problèmes pour lesquels on ne connaît pas de méthode de résolution ou dont la solution exacte est trop complexe pour être calculée en un temps raisonnable. On utilise ce genre d'outil pour planifier des tournées de véhicules (**VRP**), constituer des équipes de travail, implanter de manière optimale des points de vente dans une région, gérer des portefeuilles financiers et même pour trouver la meilleure forme à donner à la turbine d'un réacteur nucléaire, etc.

4.4.2 Le codage de la solution

Chaque individu de la population représente une solution au problème à optimiser. Un individu est représenté par un chromosome. Ce chromosome est constitué de gènes qui peuvent prendre des valeurs appelées « allèles ».

A chaque individu est associée une évaluation (appelée aussi *fitness*) qui mesure la qualité de la solution. L'évaluation représente la performance de l'individu vis-à-vis du problème. Le « *génotype* » représente l'ensemble des valeurs prises par les gènes du chromosome alors que le « *phénotype* » représente la solution réelle qui en découle. De la pertinence du codage va dépendre le choix des opérateurs de reproduction et l'efficacité globale de l'algorithme.

Il existe deux cas de figures pour coder un problème. Le premier utilise un codage « **direct** » dans lequel toute l'information de la solution est présente dans le chromosome. Le deuxième reporte la difficulté de représentation d'une solution du problème à l'extérieur du codage, en utilisant un codage « **indirect** ».

1. **Direct** : Dans un codage direct, le gène doit représenter une solution complète. Ce dernier doit contenir toutes les informations utiles à la solution [Tavares et al. 2003]. La représentation chromosomique du problème VRPTW qui utilise ce type de codage peut être tout simplement un ensemble de vecteurs de taille différente. Chaque vecteur de cet ensemble représente une tournée d'un véhicule et contient une suite des clients à visiter par ce véhicule. Ces vecteurs ont des tailles variées selon le nombre de clients à visiter pour chaque tournée (voir la figure 4.1).
2. **Indirect** : Dans ce type de codage, le génotype ne représente pas directement la solution. Il est nécessaire de passer par une étape complémentaire pour obtenir la solution [Sushil et al. 1999], [Prins 2004]. En utilisant ce codage pour représenter la solution du problème VRPTW, le chromosome vecteur va contenir une suite ordonnée de clients à servir. Dans cette représentation, nous ne pouvons pas distinguer les tournées de véhicules avant que le chromosome ne passe par une étape de décodage qui va construire les tournées en appliquant par exemple l'heuristique **PFIH** (Push Forward Insertion Heuristic) [Solomon 1987][Zhu 2000] (voir la figure 4.2).

Chaque type de codage a ses avantages et inconvénients :

- Dans le codage direct on n'a pas besoin d'une étape complémentaire pour construire les tournées. Cela signifie qu'avec ce codage nous gagnerons du temps de calcul.

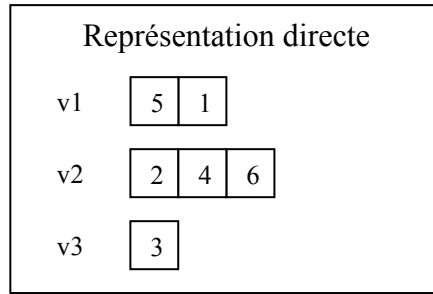


FIG. 4.1 – Représentation directe

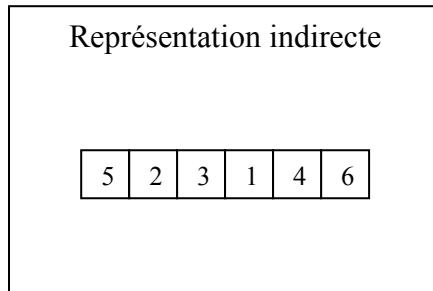


FIG. 4.2 – Représentation indirecte

- Il est plus simple d'appliquer les opérateurs génétiques sur un chromosome représenté par un codage indirect. Certaines contraintes du problème peuvent en effet être reportées dans la phase de décodage.

Dans notre problème d'élaboration dynamique de tournées de véhicules avec fenêtres de temps, nous avons préféré utiliser le codage indirect des solutions. D'une part ce codage est bien adapté pour traiter les événements dynamiques et d'autre part nous avons utilisé une heuristique rapide pour réaliser le décodage dans un environnement « temps réel ».

Ce codage indirect repose sur l'utilisation couplée de deux structures de données : d'une part un vecteur « chromosome » qui est dédié à la planification prévisionnelle ou future du routage et d'autre part « un ensemble de vecteurs » permettant de mémoriser la partie réalisée des tournées. Plus précisément, à chaque instant t de notre cadre dynamique, chaque chromosome va contenir dans un vecteur tous les clients restant à servir, c'est-à-dire tous ceux qui n'ont pas encore été servis, par la flotte de véhicules de la journée en cours (voir figure 4.3). De plus, les clients servis ou sur le point de l'être par les véhicules disponibles seront mémorisés à part dans une seconde structure de type matriciel (voir la figure 4.4). Chaque vecteur de cet ensemble représente les clients rangés dans l'ordre où ils ont été servis par un véhicule donné (i.e. une tournée).

4.4.3 Fonction d'évaluation

Notre problème est de type multi-objectifs, puisque nous cherchons par exemple à minimiser le temps total de trajet par les véhicules, à minimiser le retard total chez les clients, à minimiser le retard total des véhicules pour retourner au dépôt, etc. Plusieurs méthodes existent pour transformer ce problème en un problème uni-objectif. Parmi celles-ci, la méthode d'agrégation consiste à affecter des poids (représentant des préférences) aux différents objectifs du problème.

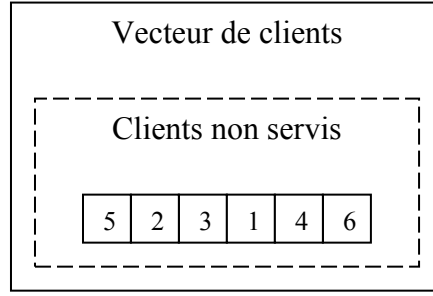


FIG. 4.3 – Le chromosome représentant les clients qui restent à servir à l’instant t

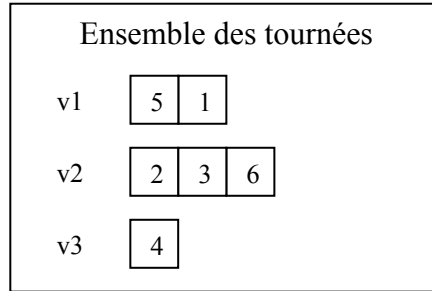


FIG. 4.4 – L’ensemble des tournées représentant les clients servis ou en cours de service à l’instant t

Ces poids sont déterminés en général par un expert du domaine.

Dans notre problème, nous avons pris la fonction objectif du problème VRPTW donnée par [Ichoua et al. 2003] ; cette fonction est la somme pondérée de tous les objectifs de ce problème :

$$F = w_1.T + w_2.R_1 \quad (4.8)$$

Où :

1. T : est le temps total de trajet de l’ensemble des véhicules.
2. R_1 : est le retard total chez les clients.
3. w_i : sont des poids (préférences) associés à chaque critère.

De plus dans notre problème, w_1 et w_2 sont fixés à 1.

4.4.4 Décodage d’un chromosome

Pour évaluer la qualité d’une solution, c’est-à-dire calculer le fitness d’un chromosome, il faut au préalable décoder le chromosome pour obtenir la solution correspondante.

Le décodage d’un chromosome pour obtenir la solution correspondante, utilise une heuristique d’insertion simple. Cette heuristique est décrite ci-dessous (algorithme 4.1) :

L’insertion d’un client se fait toujours en fin d’une tournée existante (voir la figure 4.5). Le chromosome vecteur de clients représente en fait, une liste ordonnée de préférence décroissante pour l’heuristique d’insertion. Un client i , placé avant un autre client j dans le chromosome et

Algorithme 4.1 Algorithme d'insertion (décodage d'un chromosome)

Entrée :

Liste : la liste des clients issue du chromosome à décoder
Tournees : l'ensemble des tournées contenant les clients déjà servis
NbVehicules : le nombre des véhicules

Sortie :

Solution : L'ensemble des tournées prévisionnelles

Début :

Solution = *Tournees*

// Les clients déjà servis sont intégrés dans la solution

TantQue *Liste* n'est pas vide **Répéter**

v = 0

v_{min} = *v*

c = le premier client dans *Liste*

f_{min} = le sous fitness issu de l'insertion du client *c* à la fin de la tournée *v* dans *Solution*

// Le sous fitness est la valeur de la fonction objectif de la partie de *Solution* faite

// jusqu'à l'insertion du *c*

v ++

TantQue *v* < *NbVehicules* **Répéter**

f = le sous fitness issu de l'insertion du client *c* à la fin de la tournée *v* dans *Solution*

Si (*f* < *f_{min}*)

f_{min} = *f*

v_{min} = *v*

FinSi

v ++

FinTantQue

Insérer le client *c* à la fin de la tournée numéro *v_{min}* dans *Solution*

Supprimer le client *c* de *Liste*

FinTantQue

Fin.

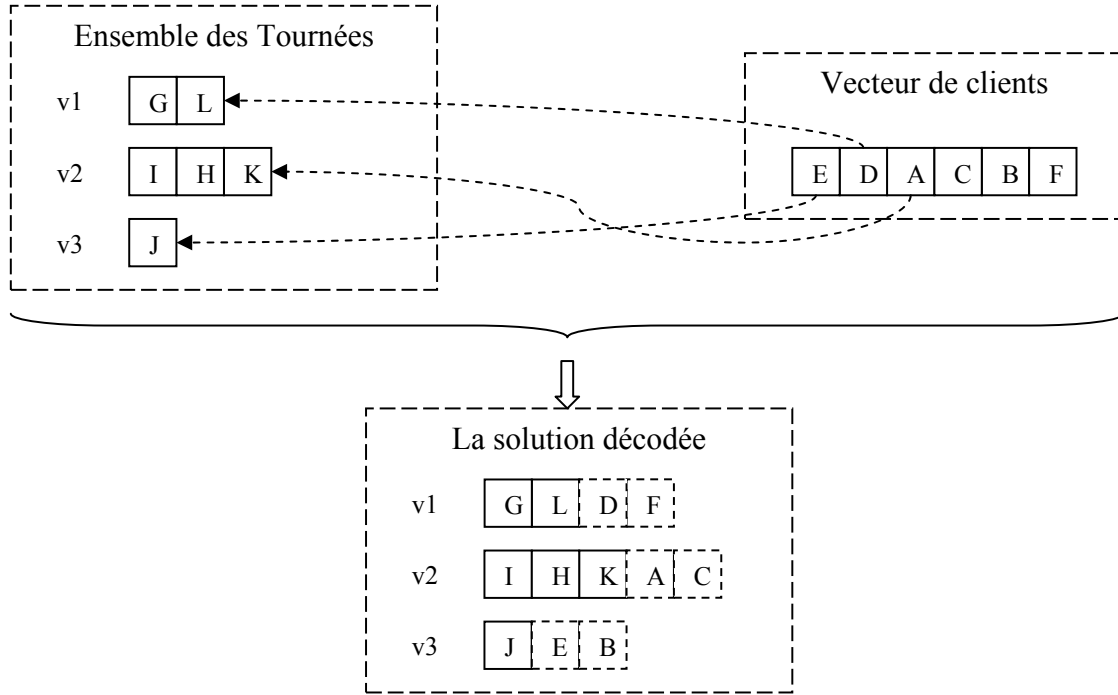


FIG. 4.5 – Le décodage d'un chromosome

affecté au même véhicule sera toujours servi en premier dans la tournée correspondante. L'ordre indiqué dans le chromosome est donc important pour la qualité des solutions décodées ; en effet, nous avons utilisé une heuristique simple pour réduire le temps de décodage parce que nous traitons un problème dynamique où le temps joue un grand rôle.

La complexité de cette procédure est en $O(n * m)$, où n est le nombre des clients à servir et m est le nombre des véhicules.

4.4.5 Population initiale

La construction d'une population initiale représente l'étape de départ de l'algorithme génétique. En général, cette population est créée de façon aléatoire. Il est possible de placer les individus initiaux de façon uniforme dans l'espace de recherche. Une partie de ces individus peut être construite en utilisant des méthodes heuristiques simples pour obtenir des individus de bonne qualité. Par exemple l'heuristique de **PFIH** (Push Forward Insertion Heuristic) a été utilisée par [Zhu 2000] ou l'heuristique de *sweep* a été utilisée par [Baker et al. 2003]. Cette dernière méthode permet d'accroître la vitesse de convergence, mais empêche l'algorithme de se focaliser sur une autre partie de l'espace de recherche qui pourrait contenir l'optimum. Dans notre approche nous avons choisi de générer aléatoirement la population initiale.

4.4.6 Opérateurs de Sélection

A chaque génération, l'opérateur de sélection sert à former une population intermédiaire à la population de la génération courante i . Ces individus seront ensuite croisés et mutés pour former la population de la génération suivante $i + 1$. Les chromosomes sélectionnés représentent

les individus les mieux adaptés. Il existe de nombreuses techniques de sélection, nous allons présenter les plus connues :

1. La plus simple, connue sous le nom de « *ranking* » consiste à classer les n chromosomes de la population par ordre croissant de leur évaluation respective (ou décroissant selon l'objectif). Les m premiers individus sont ensuite sélectionnés. Ainsi, seuls les meilleurs individus sont conservés.
2. La sélection par roulette ou « *roulette wheel* » : elle consiste à associer à chaque chromosome un segment dont la longueur est proportionnelle à son fitness. Ces segments sont ensuite concaténés sur un axe gradué que l'on normalise entre 0 et 1. On tire alors un nombre aléatoire de distribution uniforme entre 0 et 1, puis on identifie le segment sélectionné et le chromosome correspondant. Avec cette technique, les bons chromosomes seront plus souvent sélectionnés que les mauvais, et un même chromosome pourra avec cette méthode être sélectionné plusieurs fois. Néanmoins, sur des populations de petite taille, il est difficile d'obtenir exactement l'espérance mathématique de sélection à cause du faible nombre de tirages. On aura donc un biais de sélection plus ou moins fort suivant la dimension de la population.
3. *Sélection par tournoi* : Cette technique tire au hasard deux ou plusieurs individus de la population et le plus fort est sélectionné, c'est-à-dire celui ayant le fitness le plus intéressant.
4. La sélection *aléatoire* : comme son nom l'indique, ce type de sélection choisit le chromosome selon une distribution uniforme.

Dans cette thèse, nous avons choisi la sélection par **tournoi** qui est utilisée le plus souvent par de nombreux auteurs [Goldberg et al. 1992], [Deb et al. 1999].

4.4.7 Opérateurs de croisement (Crossover)

Le croisement permet d'accoupler deux parents pour former deux enfants. L'idée est que les enfants garderont les meilleures caractéristiques de leurs parents. Prenons l'exemple de la fonction $f(x)$ décrite par la figure 4.6. Le principe du croisement consiste à recombinaison les bonnes parties des chromosomes des deux parents P_1 et P_2 pour engendrer deux enfants E_1 et E_2 de meilleure qualité. La difficulté réside dans le fait que la bonne partie du chromosome n'est pas connue et qu'il est alors possible de créer des enfants de plus mauvaise qualité que celles des parents initiaux. L'enfant E_3 est un exemple de cette difficulté dans le cas d'une minimisation de la fonction $f(x)$. Cette difficulté est d'autant plus importante lorsqu'il s'agit d'un codage de type indirect puisque les bonnes parties des solutions parentes sont « masquées » par celui-ci.

Il existe de nombreuses techniques de croisement, nous allons présenter celles utilisées pour la résolution du problème VRPTW.

PMX (Partially Mapped Crossover) : Le PMX fut proposé par [Goldberg et al. 1985] dans le cadre de problèmes d'ordonnancement. Le but était de construire un enfant par le choix de sous séquences ordonnancées de l'un des parents et de préserver l'ordre et la position d'autant de sous séquences que possible des autres parents. La sous-séquence de l'ordonnancement est sélectionnée par le choix de deux points de coupure aléatoire, lesquels servent de frontière pour l'opération de substitution. Considérons par exemple les deux parents :

$$\begin{aligned} P_1 &= 123 \mid 4567 \mid 89 \\ P_2 &= 452 \mid 1876 \mid 93 \end{aligned}$$

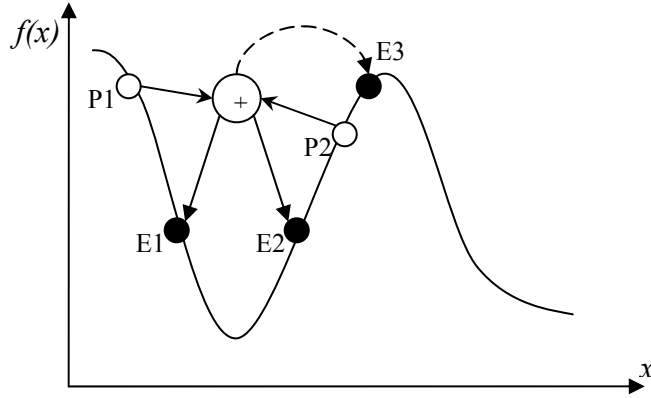


FIG. 4.6 – Principe du croisement

- **Etape 1** : ces deux parents vont produire deux enfants. Dans cette première étape, les segments compris entre les points de coupures sont échangés :

$$E_1 = xxx \mid 1876 \mid xx$$

$$E_2 = xxx \mid 4567 \mid xx$$

Notons qu'en vue de procéder à l'étape 3 de résolution des conflits, il est possible d'établir entre les deux points de coupures une table de correspondances entre les allèles ; cette table est définie comme suit dans notre exemple :

$$1 \leftrightarrow 4, 8 \leftrightarrow 5, 7 \leftrightarrow 6, 6 \leftrightarrow 7$$

- **Etape 2** : les chromosomes « *Enfants* » sont complétés en transmettant les allèles non conflictuelles issus du second parent.

$$E_1 = x23 \mid 1876 \mid x9$$

$$E_2 = xx2 \mid 4567 \mid 93$$

- **Etape 3** : cette dernière étape consiste à terminer l'élaboration des descendants en résolvant les conflits à l'aide de la table de correspondance :

$$E_1 = 423 \mid 1876 \mid 59$$

$$E_2 = 182 \mid 4567 \mid 93$$

MX1T/D : Ce croisement a été proposé par [Sushil et al. 1999]. Il est composé de deux parties : le croisement heuristique (**MX1D**) et le croisement de fusion (**MX1T**). Le premier (MX1D) est basé sur les distances entre les clients et le deuxième (MX1T) est basé sur la borne inférieure des fenêtres de temps des clients. Dans ce croisement, deux parents produisent un seul descendant avec le croisement heuristique. Le deuxième descendant est obtenu à partir des mêmes parents avec le croisement de fusion.

1. **MX1T** : Nous allons expliquer ce type de croisement en utilisant l'exemple suivant :

$$P_1 = \underline{2} \ 5 \ 6 \ 1 \ 0 \ 7 \ 3 \ 8 \ 4 \ 9$$

$$P_2 = \underline{4} \ 1 \ 6 \ 9 \ 3 \ 8 \ 2 \ 0 \ 5 \ 7$$

Cet opérateur utilise un vecteur global de précédence. A titre d'exemple, considérons le vecteur suivant :

$$0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9$$

Ce vecteur est basé sur la borne inférieure de la fenêtre de temps (e_i) de chaque client. Les clients dans ce vecteur, sont triés en ordre croissant selon leur borne (e_i). Dans ce vecteur, le client 1 peut être servi avant les clients 2, 3 et ainsi de suite. Les clients à l'étude sont soulignés. Le premier client de P_1 s'avère être placé dans le vecteur global de précédence plus tôt que le premier client de P_2 . Ce client est placé comme premier client dans l'enfant E_1 . Pour maintenir la faisabilité de la suite de l'élaboration du chromosome, le premier client de P_2 est permuté. C'est le client 4 qui est permuté avec le client 2 (accentués en gras) :

$$\begin{aligned} P_1 &= 2 \ \underline{5} \ 6 \ 1 \ 0 \ 7 \ 3 \ 8 \ 4 \ 9 \\ P_2 &= \mathbf{2} \ \underline{1} \ 6 \ 9 \ 3 \ 8 \ 4 \ 0 \ 5 \ 7 \\ E_1 &= 2 \ x \ x \ x \ x \ x \ x \ x \ x \end{aligned}$$

Le deuxième client de P_2 apparaît dans le vecteur de précédence avant celui de P_1 , ainsi nous le plaçons en deuxième client de E_1 . Nous permutons le deuxième client de P_1 pour maintenir la faisabilité, et nous continuons de cette façon jusqu'à la fin des chromosomes :

$$\begin{aligned} P_1 &= 2 \ \mathbf{1} \ 6 \ \mathbf{5} \ 0 \ 7 \ 3 \ 8 \ 4 \ 9 \\ P_2 &= 2 \ 1 \ 6 \ 9 \ 3 \ 8 \ 4 \ 0 \ 5 \ 7 \\ E_1 &= 2 \ 1 \ x \ x \ x \ x \ x \ x \ x \end{aligned}$$

Finalement, l'enfant obtenu est comme suit :

$$E_1 = 2 \ 1 \ 6 \ 5 \ 0 \ 7 \ 3 \ 4 \ 8 \ 9$$

2. **MX1D** : Cet opérateur exploite les distances entre les clients. Par exemple, nous avons les deux parents suivants :

$$\begin{aligned} P_1 &= \underline{2} \ 5 \ 6 \ 1 \ 0 \ 7 \ 3 \ 8 \ 4 \ 9 \\ P_2 &= \underline{4} \ 1 \ 6 \ 9 \ 3 \ 8 \ 2 \ 0 \ 5 \ 7 \end{aligned}$$

D'abord, nous choisissons aléatoirement l'un des deux premiers clients des parents (par exemple, le client 4 de P_2), et nous initialisons l'enfant E_2 avec ce client :

$$\begin{aligned} P_1 &= \underline{4} \ \underline{5} \ 6 \ 1 \ 0 \ 7 \ 3 \ 8 \ 2 \ 9 \\ P_2 &= \underline{4} \ \underline{1} \ 6 \ 9 \ 3 \ 8 \ 2 \ 0 \ 5 \ 7 \\ E_2 &= 4 \ x \ x \ x \ x \ x \ x \ x \ x \end{aligned}$$

Maintenant nous comparons la distance entre le client 4 et le client 5, et la distance entre le client 4 et le client 1. Si $d_{4,5} > d_{4,1}$, nous choisissons alors 1 pour être le prochain client de E_2 . Pour les mêmes raisons que précédemment (faisabilité de la poursuite), on doit

permuter le client 1 et le client 5 dans P_1 . On continue de la même manière jusqu'à ce qu'un nouveau chromosome soit entièrement produit E_2 .

$$P_1 = 4 \underline{16} 5 0 7 3 8 2 9$$

$$P_2 = 4 \underline{16} 9 3 8 2 0 5 7$$

$$E_2 = 4 1 x x x x x x x$$

...

$$E_2 = 4 1 6 5 3 7 2 8 0 9$$

Les parents choisis par l'opérateur de sélection ne seront pas tous croisés. Une probabilité P_{Cross} permet de décider si les parents seront croisés entre eux ou s'ils seront tout simplement copiés. Le croisement est davantage basé sur l'exploitation du voisinage des parents que sur l'exploration de l'espace de recherche.

4.4.8 Opérateurs de mutation

La mutation permet d'explorer l'espace de recherche en évitant à l'algorithme de converger trop rapidement vers un optimum local. La figure 4.7 présente un exemple de fonction $f(x)$ à optimiser. Les jetons noirs représentent les chromosomes de la population qui ont tous convergé vers un optimum local. Le principe de la mutation est donc de prendre un de ces chromosomes et de le modifier afin d'explorer une autre partie de l'espace de recherche que l'on espère plus intéressante.

La mutation est traditionnellement considérée comme un opérateur marginal bien qu'elle confère en quelque sorte aux algorithmes génétiques la propriété d'ergodicité (i.e. tous les points de l'espace de recherche peuvent être atteints). Cet opérateur est donc d'une grande importance. Il a de fait un double rôle :

- celui d'effectuer une recherche locale
- et/ou de sortir d'une vallée.

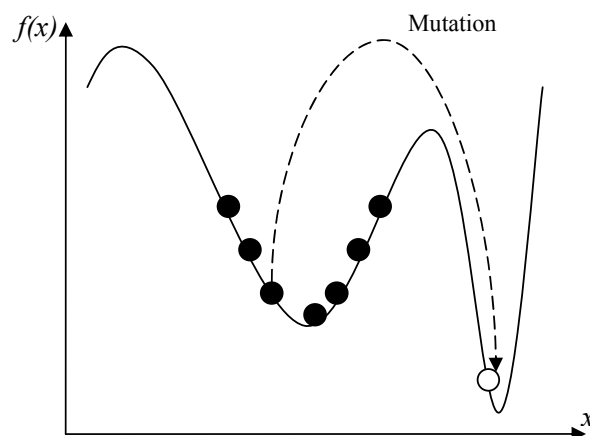


FIG. 4.7 – Principe de la mutation

Il existe de nombreuses façons de réaliser la mutation sur un chromosome. Cet opérateur fait

appel à la notion de voisinage. Nous allons rappeler trois techniques basiques de mutation ou de voisinage :

1. **1-Opt** (nommé aussi *swap*) : Cet opérateur est inspiré de technique de mutation utilisée pour la résolution de problèmes de voyageur de commerce (TSP) [Grefenstette et al. 1985]. Cet opérateur est le plus répandu. Il consiste à prendre au hasard deux gènes (2 clients) du chromosome, D et B dans l'exemple (voir la figure 4.8) et à les inverser.
2. **2-Opt** : Dans la littérature, cet opérateur appartient à la famille des opérateurs $\lambda - Opt$ ($\lambda = 2, 3, 4, \dots, n$) proposée par [Lin 1965], il s'agit de remplacer λ arcs existants (l'arc est la liaison entre deux clients consécutifs dans une tournée) par λ nouveaux arcs. Dans notre cas, il est impossible d'utiliser directement cette famille parce que le chromosome ne permet pas de distinguer clairement les tournées dans le codage indirect que nous utilisons. Donc, nous avons du adapter l'opérateur $2 - Opt$. Celui-ci choisit aléatoirement deux suites de clients dont la taille est égale à deux, puis les échange.
3. **Or-Opt** : Opère de la même manière, mais les tailles des suites choisies peuvent varier entre 0 et 3. Cet opérateur échange deux suites ou il déplace une suite vers une autre position si la taille de l'une des suites est égale à zéro (voir la figure 4.9). L'opérateur *CROSS-exchange* proposé dans [Taillard et al. 1997] est une généralisation de cet opérateur.

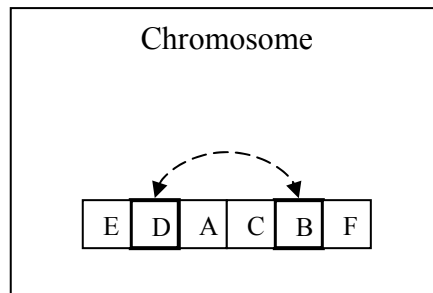


FIG. 4.8 – L'opérateur de mutation 1-Opt

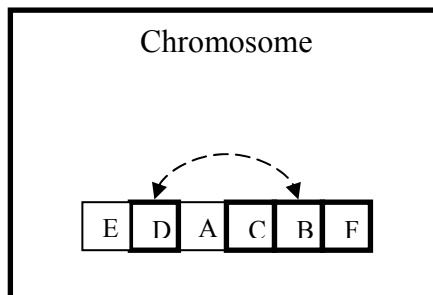


FIG. 4.9 – L'opérateur de mutation Or-Opt

L'opérateur de mutation ne s'applique pas à tous les chromosomes. Chacun a une probabilité P_{Mut} , d'être muté. Cette probabilité est en général très faible. Une fois les chromosomes croisés et mutés, il faut les réévaluer afin de pouvoir sélectionner la prochaine population pour l'itération

suivante. Cette évaluation se réalise à l'aide de la fonction d'évaluation, c'est-à-dire la fonction objectif du problème traité.

4.4.9 Méthode de remplacement par l'élitisme

A la création d'une nouvelle population, il y a de grandes chances pour que les meilleurs chromosomes soient perdus après les opérations de croisement et de mutation. Pour éviter cela, on utilise une méthode d'élitisme. Elle consiste à copier un ou plusieurs des meilleurs chromosomes dans la nouvelle génération. Ensuite, on génère le reste de la population selon l'algorithme de reproduction usuel. Cette méthode améliore en général considérablement les algorithmes génétiques, car elle permet de ne pas perdre les meilleures solutions trouvées au cours de l'évolution [Michalewicz 1994].

Dans notre approche, les enfants générés par les opérateurs génétiques, sont copiés dans une population temporaire. Nous obtiendrons donc deux populations ; la première est la population courante contenant les chromosomes parents et la deuxième est la population contenant les chromosomes enfants. Ensuite, nous trions les deux populations en ordre croissant des fitness des chromosomes. Puis, nous construisons la nouvelle population pour la prochaine génération de la manière suivante : nous remplissons un pourcentage de la nouvelle population avec les meilleurs chromosomes de la population triée tandis que l'autre partie de la nouvelle population est remplie par des chromosomes générés aléatoirement (voir la figure 4.10) afin d'introduire de la diversité.

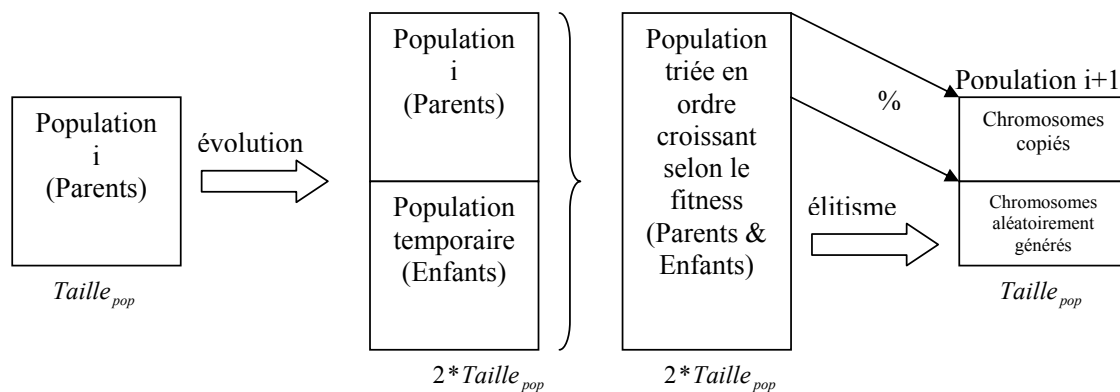


FIG. 4.10 – L'évolution générale de la population

A ce stade, nous avons présenté les différentes étapes de notre approche qui concerne l'adaptation de l'algorithme génétique au problème VRPTW. Nous allons continuer avec la partie de notre approche qui traite les événements dynamiques présents dans le problème DVRPTW. Pour prendre en compte ce caractère, un gestionnaire événementiel a été couplé à notre algorithme génétique. Pour tester celui-ci en situation, un environnement de simulation a été développé.

4.5 Plate-forme de Simulation Dynamique

Pour traiter le problème VRPTW dynamique, nous avons développé un simulateur ayant la capacité de simuler une journée de service [Goncalves et al. 2007]. Le simulateur s'appuie sur deux composants principaux qui coopèrent (voir la figure 4.11). Un composant **Algorithme Génétique** qui cherche en tâche de fond à améliorer la population courante selon l'algorithme génétique présenté ci-dessus. Un composant **Gestion d'Événements** qui attend l'arrivée

d'un évènement et déclenche les actions appropriées à cet évènement. L'Algorithme Génétique peut continuer ainsi d'améliorer les solutions courantes en tenant compte de ces changements [Housroum 2005]. Elle utilise une heuristique d'insertion pour ajouter le nouveau client dans la meilleure position de chaque chromosome.

Enfin le composant **Monde réel** permet de générer les évènements extérieurs au planificateur. Deux évènements sont à considérer. L'« *arrivée d'un nouveau client* » permet de simuler l'arrivée d'une nouvelle demande d'un client en cours de journée. L'arrivée d'un client peut être ainsi généré par la lecture d'un fichier de données (Benchmark). La « *fin de service d'un client* » simule la fin du service donnée chez le client. pour calculer ce temps, il est nécessaire de calculer le temps d'arrivée chez le client et le temps d'attente chez celui ci avant que ne commence le service.

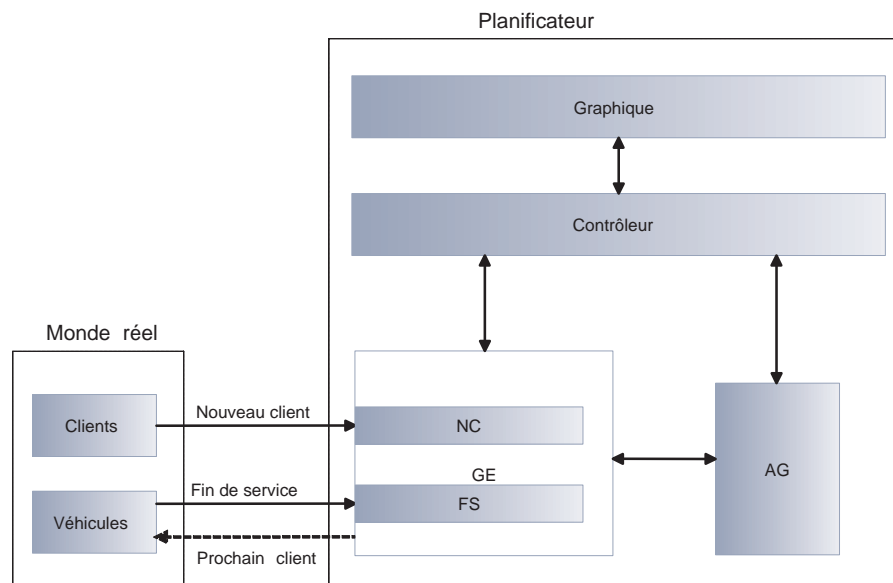


FIG. 4.11 – Simulateur pour le DVRPTW

4.5.1 Le composant Algorithme Génétique (AG)

Cette partie de notre planificateur s'occupe de tout ce qui concerne les opérations génétiques sur la population courante. Comme nous l'avons déjà vu dans la section 4.4, l'évaluation d'un chromosome passe par une phase de décodage qui utilise une heuristique d'insertion.

4.5.2 Le composant Gestion d'Événements (GE)

Ce composant traite les deux types d'évènements (« *arrivée d'un nouveau client* », « *fin de service d'un client* ») que l'on cherche à générer par notre simulateur. L'évènement « *arrivée d'un nouveau client* » vient en fait du Benchmark (fichier de données) qui décrit le problème dynamique à résoudre. Une procédure appropriée à l'évènement est appliquée sur la population courante. L'évènement *fin de service d'un client* consiste à fournir le prochain client à servir au véhicule qui se libère et ce, en utilisant la meilleure solution actuelle.

4.5.3 Gestion du temps

Le temps réel d'ouverture d'une journée de service $[0, T]$ est simulé pendant une durée $[0, T_{sim}]$. Cette durée est divisée en petites tranches de temps ayant des tailles variables selon la durée d'une génération (G_k) de l'algorithme génétique. Après chaque génération nous devons calculer le temps écoulé pour vérifier l'apparition éventuelle de nouveaux évènements (figure 4.12). Le temps T_{sim} (30 min) est beaucoup plus petit que la durée réelle T (8 heures) d'une journée et permet d'avoir des résultats satisfaisants sur des périodes de simulation courtes.

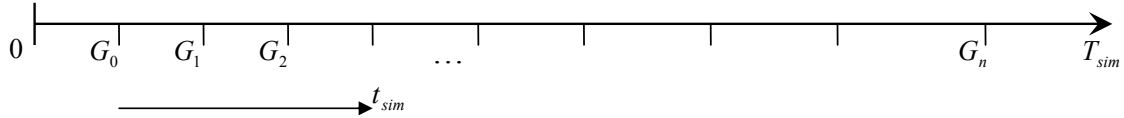


FIG. 4.12 – Durée de la simulation

En l'absence d'évènement, l'algorithme génétique continue à faire évoluer la population au cours du temps. Dans le cas contraire, de nouveaux évènements sont à prendre en compte. Examinons (figure 4.13) l'évolution de la situation quand trois nouvelles demandes apparaissent au cours du temps. Nous avons pris comme hypothèse dans cette figure que la durée prise par l'AG pour produire une nouvelle population est de quatre unités de temps. La prise en compte d'un nouveau client C_i au temps G_{j+3} entraîne une augmentation du temps nécessaire à l'AG pour produire la prochaine population. En effet l'apparition d'un nouveau client accroît la taille de chaque chromosome de la population courante qui dépend du nombre de clients à servir. Le temps de calcul de l'AG passe alors par exemple à cinq unités de temps. L'application des opérateurs génétiques prend plus de temps. Ce phénomène peut se reproduire à chaque apparition d'un nouveau client (par exemple : C_{i+1} , C_{i+2} , etc).

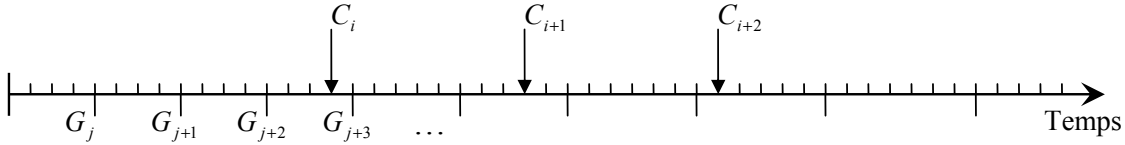


FIG. 4.13 – La simulation de la journée de service

4.5.4 Planification des tournées

Le planificateur fonctionne en deux étapes décrites comme suit :

1. Dans la première étape, le planificateur utilise l'algorithme génétique pour résoudre le problème VRPTW statique. Dans cette étape nous cherchons à trouver les meilleures solutions pour les clients statiques, ceux-ci représentent les clients connus à l'avance (i.e. avant le début de la journée). A la fin de cette étape nous obtenons une population initiale de solutions pour le problème statique.
2. La deuxième étape de notre planificateur s'occupe de la partie dynamique du problème DVRPTW. Cette étape fonctionne comme suit :
 - (a) Initialiser le temps de simulation t_{sim} à zéro.

- (b) Faire évoluer la population courante et calculer la nouvelle valeur du temps de simulation t_{sim} .
- (c) Vérifier si un ou plusieurs évènements se sont produits pendant la dernière génération (Figure 4.14).
 - Si oui, démarrer le GE correspondant à chaque évènement.
- (d) Retourner en (2b) si le temps écoulé n'est pas arrivé à la fin de la simulation.

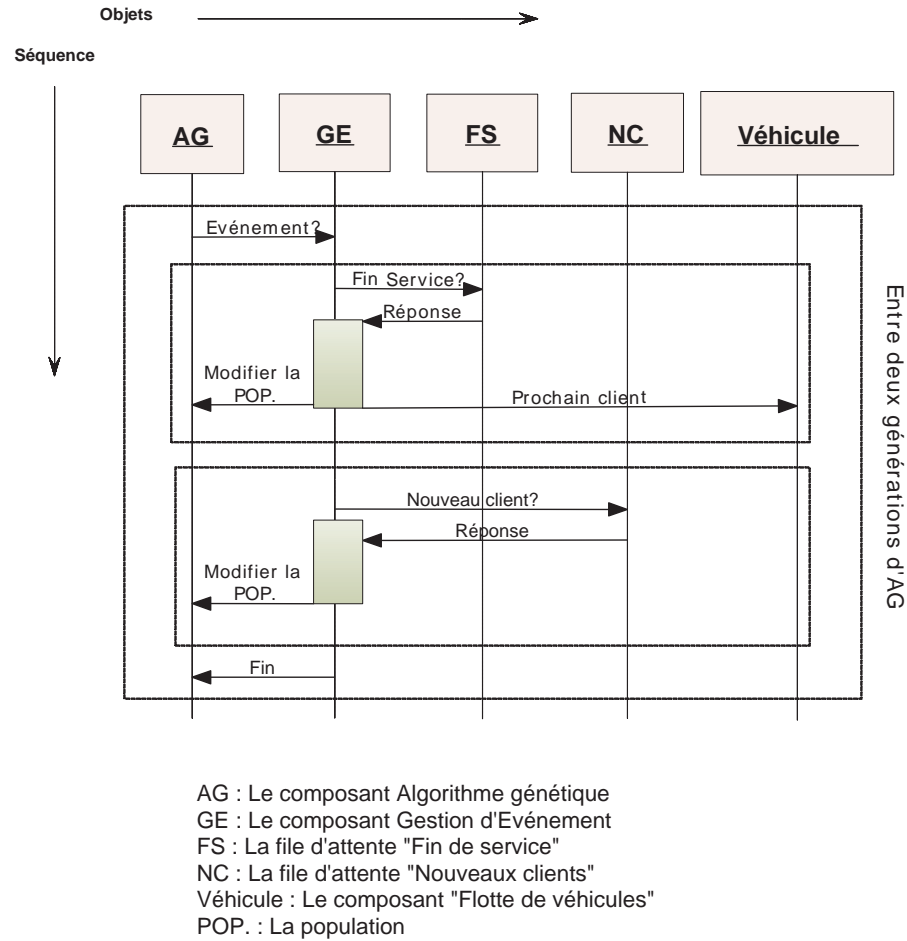


FIG. 4.14 – Diagramme UML montrant la séquence d'interactions entre les composants AG et GE, à la fin d'une génération

La figure 4.14 représente les interactions entre les deux composants AG et GE du planificateur ; cette interaction se fait après chaque génération de l'AG. Elle commence par l'envoi d'un message d'interrogation « Événement ? ». Ce message permet de vérifier l'existence de nouveaux évènements. L'interaction se termine par le message « Fin ». Entre ces deux messages, le composant *GE* vérifie l'apparition de nouveaux évènements ; il vérifie d'abord l'existence de nouveaux évènements « Fin de service », ensuite il vérifie les nouveaux évènements « Nouveaux clients » et il déclenche les services correspondants.

Comme nous l'avons énoncé, au cours de la simulation, la taille des chromosomes varie (i.e. elle correspond aux clients restant à servir par les véhicules) mais pour une population fixée, tous

les chromosomes ont une taille identique. En fin de simulation (i.e. fin de la journée de service) si chaque chromosome a une taille égale à zéro, cela signifie que tous les clients du problème ont été servis. Dans le cas contraire (i.e. taille différente de zéro), il existe des clients en fin de journée qui n'ont pu être servis par les véhicules. Ces clients pourront être pris en compte la journée suivante et seront considérés dans ce cas comme des clients statiques.

Nous avons déjà vu que pour chaque événement, nous appliquons une procédure correspondante à cet événement ; en conséquence nous avons donc dû construire deux procédures répondant aux deux types d'événements, ce sont :

1. L'évènement « fin de service ».
2. L'évènement « arrivée d'un nouveau client ».

4.5.5 L'évènement « fin de service »

Une procédure est conçue pour traiter l'évènement fin de service d'un client (algorithme 4.2). Le planificateur doit fournir au conducteur du véhicule correspondant, le prochain client à servir. Dans le cas où il n'existe plus de clients à servir par ce véhicule, alors celui-ci retourne au dépôt.

Algorithme 4.2 Algorithme de traitement de l'évènement « fin de service »

Entrée :

Vehicule : Le véhicule qui a généré l'évènement.

Chrom : Meilleur chromosome de la population courante.

Tournees : L'ensemble des tournées contenant les clients déjà servis ou en cours de service.

Sortie :

Tournees : L'ensemble des tournées contenant les clients déjà servis ou en cours de service.

Nouveaucient : Le nouveau client associé au Vehicule.

Début :

Solution = Décoder le chromosome *Chrom*.

En déduire le nouveau client à servir par le *Vehicule* (premier *Client* non encore servi du *Vehicule*).

Ajouter ce nouveau client à la tournée correspondante dans *Tournees*.

Supprimer le client *Client* de tous les chromosomes de la population.

Fin.

Par exemple, la figure 4.15 montre le déroulement de cette procédure au moment où le véhicule *v3* a fini de servir le client *J*. Elle commence par la récupération du meilleur chromosome de la population courante (celui ayant le meilleur fitness). Elle décode ensuite celui-ci pour obtenir la solution correspondante. Puis, le planificateur peut simplement décider quel est le prochain client à servir (i.e. le client *E* dans notre exemple). Finalement, on élimine le client choisi de l'ensemble des chromosomes (vecteur des clients) de la population courante et on ajoute celui-ci dans l'ensemble des tournées de véhicules.

4.5.6 L'évènement « arrivée d'un nouveau client »

Une procédure est utilisée pour chaque événement « arrivée d'un nouveau client » (algorithme 4.3). Elle utilise une heuristique d'insertion pour ajouter le nouveau client dans la meilleure

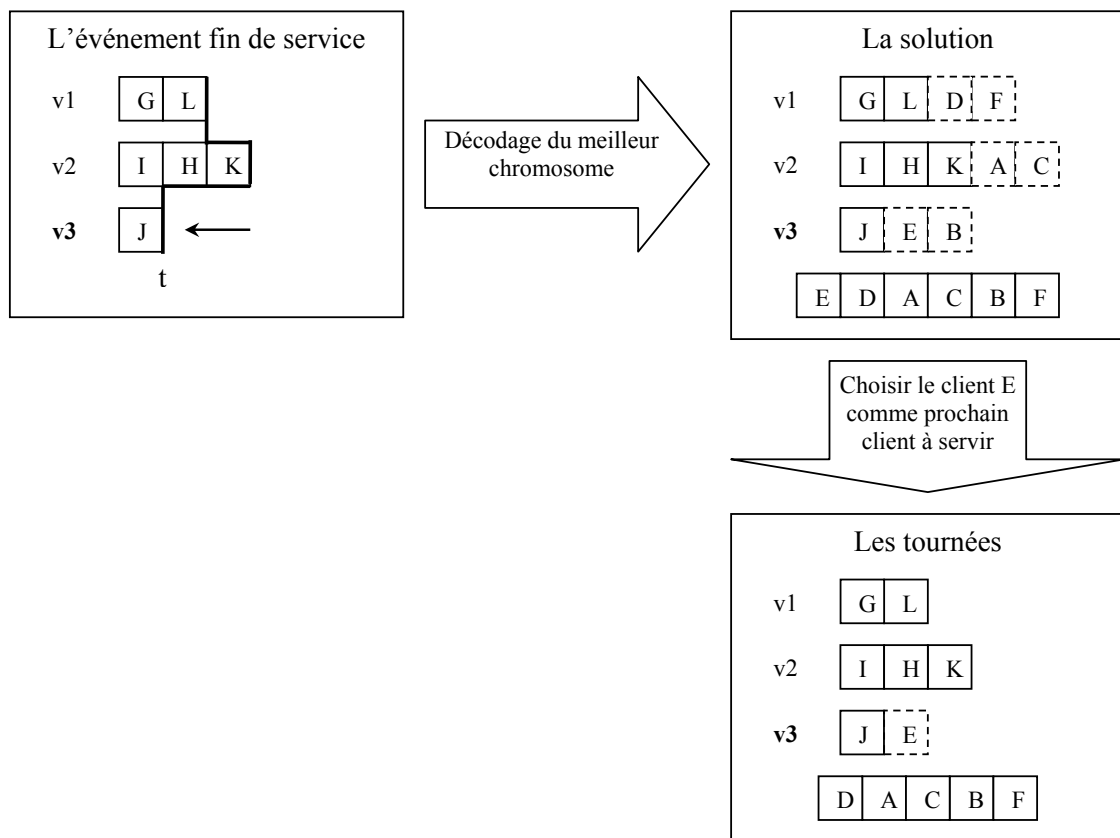


FIG. 4.15 – Procédure d'évènement fin de service

position de chaque chromosome.

Algorithme 4.3 Algorithme de traitement de l'évènement « nouveau client »

Entrée :

Client : le nouveau client

Chrom : le chromosome de taille N

Sortie :

Chrom : le chromosome de taille $N + 1$

Début :

$pos = 0$

f_{min} = le fitness de *Chrom* prenant en compte le client *Client* à la position pos

$pos++$

TantQue ($pos \leq$ la taille de *Chrom*) **Répéter**

f = le fitness de *Chrom* prenant en compte le client *Client* à la position pos

Si ($f < f_{min}$)

$f_{min} = f$

$p_{min} = pos$

FinSi

$pos++$

FinTantQue

Insérer le client *Client* à la position p_{min} dans *Chrom*

Fin.

La Figure (4.16) montre un exemple d'évènement « arrivée d'un nouveau client » au moment où le client *G* demande un service. La procédure cherche à insérer ce client dans tous les chromosomes de la population courante. L'opération d'insertion au niveau d'un chromosome consiste à tester l'insertion du client *G* dans toutes les positions possibles du chromosome considéré. A chaque essai, nous évaluons le fitness du chromosome obtenu, puis nous pouvons choisir la position d'insertion du client *G* qui donne le meilleur fitness. Cette procédure est simple et sa complexité est en $O(n^2)$ où n représente le nombre de clients.

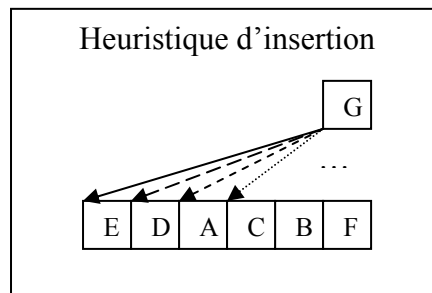


FIG. 4.16 – Procédure d'évènement nouveau client

Finalement, il faut rappeler que le planificateur fonctionne *en ligne* ; c'est-à-dire qu'il réagit directement après chaque évènement. Comme nous l'avons évoqué, il continue à faire évoluer la population en arrière plan tant qu'il n'y a pas de nouveaux évènements à traiter.

4.6 Résolution du DVRPTW avec temps de trajet dépendant du temps

4.6.1 Les paramètres de l'algorithme génétique

L'efficacité de l'algorithme génétique dépend fortement de la valeur des paramètres suivants :

1. la taille de la population(Taille) ;
2. le type de croisement(Cr) ;
3. la probabilité de croisement(TCr) ;
4. le type de mutation (Mt) ;
5. la probabilité de mutation(TMt) ;
6. le taux d'élitisme(TElt).

Le réglage des paramètres reste généralement très empirique. [Housroum 2005] a utilisé les plans d'expériences pour régler les paramètres de l'algorithme génétique dédié à la résolution du problème VRPTW. Pour le problème DVRPTW avec temps de trajet dépendant du temps que nous traitons, nous avons gardé les réglages optimaux obtenus par [Housroum 2005] afin de faciliter la convergence rapide de l'algorithme génétique vers une bonne solution. Les réglages utilisés pour chacun des problèmes statiques se trouvent dans le tableau 4.2.

Problème	Taille	Cr	TCr	Mt	TMt	TElt(%)
C1	30	MX1T/D	0,8	Or-Opt	0,05	70
R1	30	MX1T/D	0,7	Or-Opt	0,2	50
C2	30	MX1T/D	0,7	Or-Opt	0,2	50
R2	30	MX1T/D	0,7	Or-Opt	0,2	50
RC1	30	MX1T/D	0,7	Or-Opt	0,2	50
RC2	30	MX1T/D	0,7	Or-Opt	0,2	50

TABLEAU 4.2 – Le réglage de chaque groupe des benchmarks statiques

Les réglages utilisés pour chacun des problèmes dynamiques se trouvent dans les tableaux B.1 et B.2 dans l'annexe B.1. Les problèmes utilisés sont issus du Benchmark de Solomon [Solomon 1987] et ont été modifiés par [Gendreau et al. 1999] (voir section 4.6.3).

4.6.2 Les profils de vitesse dépendant du temps

Comme nous l'avons présenté dans la section 3.3.2, le profil de vitesse traduit la variation de vitesse sous la forme d'une fonction constante par morceau selon la période du jour considérée. Dans ce problème DVRPTW avec temps de trajet dépendant du temps, on considère trois périodes ayant la même durée. Les première et troisième périodes représentent les heures de pointe correspondantes à la circulation difficile du matin et du soir. La deuxième période correspond au milieu de journée, quand la densité du trafic est moindre. Les vitesses de trajet dans les heures de pointe correspondent aux vitesses de trajet en milieu de journée divisées par un facteur de ralentissement α .

Nous avons créé 3 profils différents qui correspondent à 3 types différents de routes (route, voie rapide, autoroute) dont les vitesses moyennes varient de lent à rapide.

Par ailleurs, nous avons élaboré 3 types de scénarios pour faire varier le facteur de ralentissement α . Dans les scénarios 1, 2 et 3, α vaut respectivement 1.5, 2 et 4. La vitesse moyenne dans chaque scénario est supposée unitaire comme dans le cas à vitesse constante.

Le tableau 4.3 présente les valeurs correspondantes de ces trois scénarios. Dans ce tableau, chaque ligne correspond à un profil de vitesse et chaque colonne à une période de temps. Un profil donné est affecté à chaque arête du graphe origine-destination représentant le réseau de transport ; les trois types de profils sont répartis de façon aléatoire sur l'ensemble des arêtes. La vitesse moyenne des véhicules pour chaque scénario (comprenant 9 valeurs de vitesse) est égale à 1.0.

	Profil	T1	T2	T3
Scénario 1 $\alpha=1,5$	1	0,54	0,81	0,54
	2	0,81	1,22	0,81
	3	1,22	1,82	1,22
Scénario 2 $\alpha=2$	1	0,33	0,67	0,33
	2	0,67	1,33	0,67
	3	1,33	2,67	1,33
Scénario 3 $\alpha=4$	1	0,12	0,46	0,12
	2	0,46	1,92	0,46
	3	0,96	3,84	0,96

TABLEAU 4.3 – Profils de vitesse pour les scénarios 1 à 3

4.6.3 Benchmark utilisé

Nous avons choisi d'utiliser le jeu de problèmes fournis par Solomon [Solomon 1987] et modifié par [Gendreau et al. 1999]. Il présente en effet plusieurs avantages. Le plus important est qu'il constitue implicitement une base de référence pour l'étalonnage des performances de la plupart des méthodes proposées pour le VRPTW.

Cette librairie comprend 56 problèmes au total. Chaque problème contient 100 clients distribués géographiquement dans un carré de 100 sur 100. Le dépôt est situé au centre du carré. On suppose dans notre application que tous les véhicules sont identiques (i.e. flotte homogène) et que ceux-ci se déplacent à une même vitesse constante. De ce fait, le temps de trajet entre deux clients est directement proportionnel à la distance euclidienne qui sépare ces deux clients. La vitesse des véhicules est supposée égale à 1.0 (unité de distance/unité de temps).

Les problèmes de ce benchmark peuvent être regroupés en 6 catégories en fonction de :

1. la loi de distribution des clients dans l'espace (voir la figure 4.17)
 - (a) le type **R** correspond à une distribution uniforme des clients dans l'espace.
 - (b) le type **C** correspond à une distribution des clients sous la forme de regroupements (i.e. clusters) deux à deux disjoints.
 - (c) le type **RC** est une combinaison des deux modes de répartition précédents. Une partie des clients est distribuée uniformément dans le carré tandis que l'autre partie est distribuée dans des regroupements (clusters).

2. la taille de la fenêtre de temps sur chaque client

- (a) le type 1 correspond à une fenêtre de temps étroite pour chaque client. Ceci implique un horizon de service court (i.e. fenêtre de temps réduite du dépôt central) ainsi que des tournées comportant peu de clients. La capacité des véhicules est également réduite.
- (b) le type 2 correspond à des fenêtres de temps larges. L'horizon de service étant plus étendu, les tournées comprendront en principe plus de clients. De plus la capacité des véhicules est assez large.

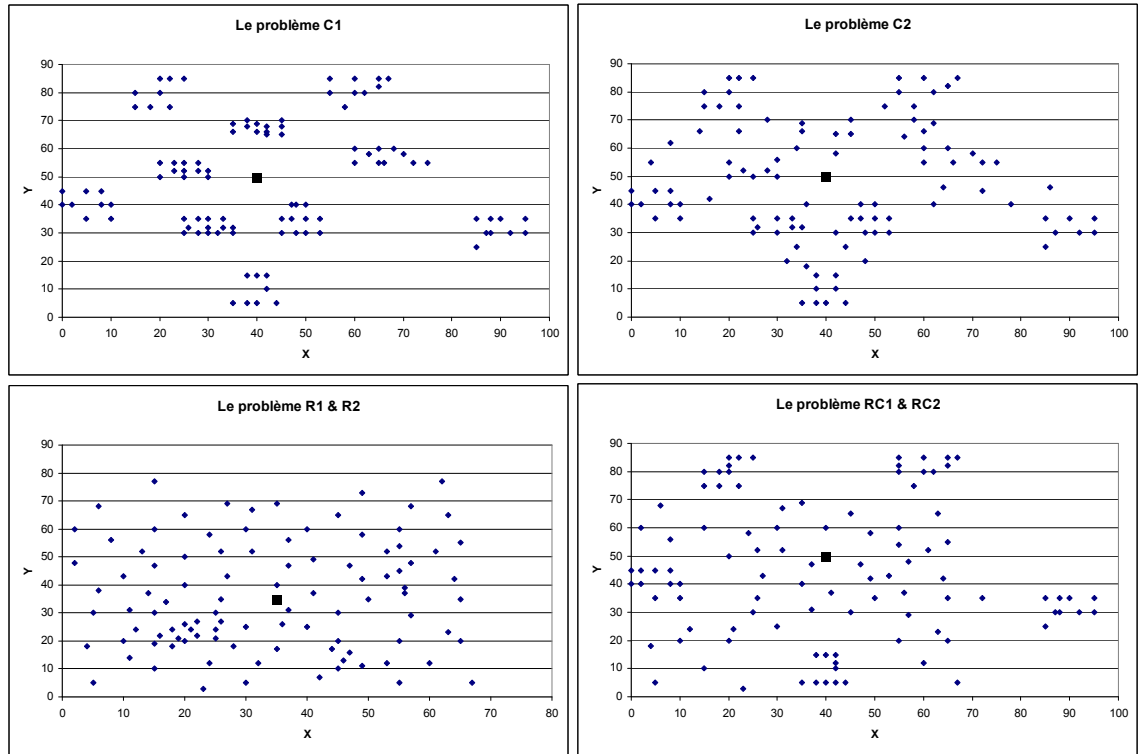


FIG. 4.17 – Les jeux de tests de Solomon pour le VRPTW

Dans ce benchmark, chaque fichier associé à un problème test contient trois types de données :

1. La partie qui concerne les attributs des clients :
 - (x, y) : est la position de client.
 - d : est la quantité de service demandée par le client.
 - s : est la durée de service chez le client.
 - (e, l) : est la fenêtre de temps de client, dans laquelle il souhaite être servi.
2. La partie qui concerne le dépôt :
 - (x, y) : détermine sa position.
 - (e, l) : sont les horaires d'ouverture et de fermeture.
3. La dernière partie concerne les véhicules disponibles :
 - c : la capacité du véhicule.
 - v : la vitesse du véhicule.

Le benchmark de Solomon a été modifié par [Gendreau et al. 1999] pour l'adapter au problème VRPTW dynamique. Pour cela, il a ajouté un nouvel attribut à chaque client. Cet attribut, noté (t) , est l'heure d'apparition de la demande. Si cet attribut t vaut 0 pour un client, ce client est déclaré statique c'est-à-dire connu en avance (avant le début d'ouverture de la journée). Dans le cas contraire, il est déclaré dynamique et la valeur de (t) donne l'heure où la demande apparaît dans la journée. Ces clients dynamiques sont aléatoirement choisis parmi l'ensemble des clients.

Il est à noter que dans ce benchmark modifié, le nombre de véhicules disponibles n'est pas illimité comme dans la version statique de ce benchmark. En fait, il a été fixé au nombre optimal de véhicules obtenu en résolvant la version statique du problème VRPTW correspondant. Par ailleurs dans notre cas, nous avons fixé à 30 *min* le temps de simulation nécessaire pour simuler une journée d'ouverture. Par ailleurs l'hypothèse sur la capacité des véhicules faite par [Gendreau et al. 1999] a été conservée : cette capacité a été modifiée par rapport aux problèmes originaux de Solomon afin que la contrainte soit toujours satisfaite.

4.6.4 Méthode de validation

La méthode de validation est identique à celle du chapitre 3. Elle est basée sur la comparaison des performances des solutions fournies par deux modes de résolution suivants [Ichoua et al. 2003] :

- Résolution1 : la résolution prend en compte un environnement avec des temps de trajet variables définis par des profils de vitesse déterminés ; soit S la solution obtenue. Cette solution fournit les indicateurs (équation 4.8) R1 ;
- Résolution2 : la résolution ne prend pas en compte la variabilité de la vitesse. L'évaluation de la qualité de la solution doit se faire alors en 2 étapes. a. Le problème est résolu en supposant que le temps de parcours est constant : soit S' la solution obtenue ; b. Le parcours S' , ainsi obtenu, est évalué par simulation en le « replongeant » dans un environnement temps de trajet variable respectant les profils de vitesse précédents, Cette solution « rejouée » fournit alors les indicateurs R2.

La comparaison des indicateurs R1 et R2 obtenus par ces deux modes de résolution permet d'évaluer l'apport d'une prise en compte d'un modèle de résolution avec temps de trajet variable par rapport à une solution obtenue par un modèle à vitesse constante « replongée » en situation à vitesse variable.

4.6.5 Les résultats de simulation

Le tableau 4.4 montre les résultats obtenus par les deux modes de résolutions pour les trois scénarios envisagés sur des problèmes statiques. Dans ce cas, toutes les demandes de clients sont connues avant le départ des véhicules. Ils correspondent aux six groupes de problèmes du benchmark de [Solomon 1987] Chaque résultat est la moyenne de chaque groupe qui comporte de 8 à 12 problèmes. Ces six groupes de problèmes sont notés respectivement : C1, C2, R1, R2, RC1 et RC2. Les valeurs du tableau correspondent à la valeur fournie par la fonction objectif du problème. Il apparaît que la résolution1 basée sur une planification utilisant le modèle de vitesses variables donne presque toujours de meilleurs résultats que la résolution2 basée sur la planification utilisant le modèle de vitesse constante.

A partir de ces résultats, nous calculons un indice de gain pour évaluer le niveau d'amélioration relatif des deux modèles comme cela a été fait dans la section 3.4.2 dans le chapitre 3.

4.6. Résolution du DVRPTW avec temps de trajet dépendant du temps

Ensemble de problème		Scénario1		Scénario2		Scénario3	
		R1	R2	R1	R2	R1	R2
C1 9 problèmes	TT ^a	765,88	771,97	746,50	769,31	1013,66	1001,73
	TA ^b	287,54	260,64	384,10	262,08	169,73	226,97
	TR ^c	13,03	7,07	11,29	21,78	130,08	174,15
	VO ^d	778,26	779,03	757,79	791,08	1143,56	1324,56
	Gain(%)		0,10		4,21		13,66
R1 12 problèmes	TT	1112,29	1114,55	1007,72	1037,06	1091,62	1146,56
	TA	337,08	344,59	367,70	380,25	178,25	328,32
	TR	63,83	64,73	39,98	64,01	125,55	164,80
	VO	1175,29	1179,28	1047,70	1101,08	1216,86	1311,36
	Gain(%)		0,34		4,85		7,21
RC1 8 problèmes	TT	1135,83	1146,93	1028,60	1051,27	1195,97	1202,72
	TA	286,95	274,31	301,43	418,99	248,31	243,17
	TR	32,47	26,78	26,43	39,61	86,11	131,97
	VO	1168,30	1173,72	1055,04	1090,88	1282,09	1334,69
	Gain(%)		0,46		3,29		3,94
C2 8 problèmes	TT	701,03	720,71	769,54	787,42	1004,42	852,10
	TA	47,48	42,93	65,85	67,84	31,24	49,87
	TR	71,94	95,78	97,14	127,44	713,84	344,95
	VO	763,33	816,49	866,68	914,86	1057,53	1197,05
	Gain(%)		6,51		5,27		11,66
R2 11 problèmes	TT	1105,43	1101,92	1216,53	1073,13	1239,99	1248,90
	TA	440,09	446,14	330,09	464,84	330,61	400,55
	TR	70,23	99,70	409,40	257,83	479,52	553,61
	VO	1175,66	1201,62	1237,93	1330,96	1719,51	1802,51
	Gain(%)		2,16		6,99		4,60
RC2 8 problèmes	TT	1186,66	1237,14	1119,60	1157,69	1390,21	1409,68
	TA	636,06	618,24	670,11	686,79	534,05	592,30
	TR	24,64	12,77	39,32	44,12	180,62	351,90
	VO	1211,30	1249,91	1158,92	1201,81	1570,83	1761,58
	Gain(%)		3,09		3,57		10,83
Tous les problèmes	TT	1001,19	1015,54	981,41	979,31	1155,98	1143,62
	TA	339,20	331,14	353,21	380,13	248,70	306,87
	TR	46,02	51,14	103,93	92,47	285,95	286,90
	VO	1045,36	1066,67	1020,68	1071,78	1331,73	1455,29
	Gain(%)		2,00		4,77		8,49

^a Temps de trajet

^b Temps d'attente

^c Temps retard

^d Valeur d'objectif (= valeur de temps de trajet + valeur de temps retard)

TABEAU 4.4 – Résultats pour le benchmark statique

En ce qui concerne le tableau 4.4, nous remarquons que pour le scénario 3, le gain est plus grand que les deux autres deux scénarios. Ce résultat s'explique par le fait que les vitesses varient dans une plus grande proportion (i.e. facteur de ralentissement α plus important) donc les résultats fournis par le modèle à vitesse constante sont beaucoup plus imprécis.

Nous avons également obtenu les résultats pour les problèmes dynamiques. Ceux-ci sont présentés dans le tableau 4.5 pour les trois types de scénarios. Nous avons utilisé le benchmark de Solomon modifié par [Gendreau et al. 1999] pour incorporer des demandes dynamiques. Ces résultats conduisent aux mêmes conclusions qu'avec le modèle statique. Les gains sont en moyenne plus importants pour les problèmes dynamiques (voir dernière ligne de chaque tableau).

4.7 Conclusion

D'après les résultats obtenus, nous remarquons que le modèle à vitesse variable donne des résultats meilleurs que le modèle à vitesse constante lorsque l'on «replonge» les solutions de celui-ci dans un environnement à vitesse variable. Ce qui montre bien l'intérêt de planifier des tournées avec un modèle de calcul de temps de trajet prévisionnel plus réaliste.

En ce qui concerne l'influence du facteur de ralentissement, les performances montrent que plus ce facteur est grand, meilleurs sont les résultats obtenus. Nos travaux présentent donc une modélisation plus fine des problèmes de tournées de véhicules qui s'approche davantage de la réalité du terrain.

Ensemble de problème		Scénario1		Scénario2		Scénario3	
		R1	R2	R1	R2	R1	R2
C1 9 problèmes	TT	779,01	784,48	774,25	772,82	1160,44	1142,32
	TA	337,79	435,00	383,40	447,65	375,59	416,44
	TR	60,31	123,42	71,42	83,99	193,27	482,54
	VO	839,32	907,90	845,67	856,82	1353,71	1624,86
	Gain(%)		7,55		1,30		16,69
R1 12 problèmes	TT	1077,58	1067,11	951,20	1068,12	1066,04	1039,55
	TA	457,48	463,99	529,77	427,67	494,20	531,74
	TR	100,55	136,77	123,69	173,53	217,43	339,66
	VO	1178,14	1203,88	1074,89	1241,65	1283,47	1379,21
	Gain(%)		2,14		13,43		6,94
RC1 8 problèmes	TT	1124,55	1160,58	991,40	1116,27	1210,76	1251,09
	TA	395,49	335,78	453,81	384,07	368,20	298,19
	TR	103,64	96,49	79,15	67,03	235,92	430,16
	VO	1217,99	1257,06	1070,54	1183,30	1446,68	1681,25
	Gain(%)		3,11		9,53		13,95
C2 8 problèmes	TT	710,49	708,67	735,89	741,90	1024,13	1103,44
	TA	97,93	114,67	167,90	186,47	180,54	160,73
	TR	125,48	145,59	146,40	214,70	900,43	1464,20
	VO	835,97	854,26	882,29	956,60	1924,56	2567,63
	Gain(%)		2,14		7,77		25,05
R2 11 problèmes	TT	1089,92	1090,84	1039,04	1039,04	1186,89	1209,33
	TA	437,79	456,60	524,44	524,44	416,74	456,88
	TR	137,94	145,36	210,38	340,18	709,37	894,42
	VO	1227,85	1236,20	1249,41	1379,21	1896,26	2103,75
	Gain(%)		0,67		9,41		9,86
RC2 8 problèmes	TT	1179,95	1207,36	1139,69	1152,27	1271,99	1350,86
	TA	724,07	700,93	765,98	794,81	708,22	676,56
	TR	74,28	78,18	146,87	150,61	305,64	465,81
	VO	1254,20	1285,55	1286,56	1302,88	1577,63	1816,67
	Gain(%)		2,44		1,25		13,16
Tous les problèmes	TT	993,58	1003,17	938,58	981,74	1153,37	1182,77
	TA	408,43	417,83	470,88	460,85	423,92	423,42
	TR	100,37	120,97	129,65	171,67	427,01	679,46
	VO	1092,25	1124,14	1068,23	1153,41	1580,38	1862,23
	Gain(%)		2,84		7,39		15,13

TABLEAU 4.5 – Résultats pour le benchmark dynamique

Chapitre 5

Le problème DVRPTW temps réel avec temps de trajet dépendant du temps

Sommaire

5.1	Introduction	95
5.2	Le problème DVRPTW temps réel avec temps de trajet dépendant du temps	96
5.2.1	Technologies temps réel	96
5.2.2	Revue de la littérature	97
5.3	Approche de résolution du DVRPTW temps réel avec temps de trajet dépendant du temps	99
5.3.1	Méthode de calcul de temps de trajet réel	100
5.3.2	Le modèle de prévision	100
5.3.3	Calcul des temps de trajet intégrant la prévision	102
5.4	Environnement de simulation	103
5.4.1	Fichiers de vitesses	103
5.4.2	Synchronisation du planificateur	103
5.4.3	Architecture du simulateur	104
5.4.4	Gestion d'Événements	105
5.4.5	Déroulement d'une simulation	107
5.5	Expériences et résultats	109
5.5.1	Génération des vitesses réelles	109
5.5.2	Réglage du facteur de lissage β	111
5.5.3	Méthode d'évaluation de la qualité des solutions	112
5.5.4	Résultats	114
5.6	Conclusions	119

5.1 Introduction

Ce chapitre propose une extension au contexte temps réel du travail effectué dans le chapitre précédent dans lequel nous avons présenté une approche de résolution dynamique du problème VRPTW avec des temps de trajet dépendant du temps. En d'autres termes, nous voulons traiter ce problème en utilisant la plate-forme de résolution basée sur un algorithme génétique qui a été

décrite au chapitre 4 en y intégrant un suivi périodique de la variation des vitesses des véhicules en temps réel.

Ce chapitre est divisé en cinq parties : la partie 5.2 est consacrée à l'introduction du contexte temps réel dans la gestion des tournées de véhicules et à la présentation de la littérature afférente. Une méthode pour résoudre le problème DVRPTW temps réel avec temps de trajet dépendant du temps, ainsi que le modèle de prévision intégré à l'algorithme génétique pour la planification des tournées sont proposés en partie 5.3. La partie 5.4 est consacrée à la description de la plate-forme logicielle et notamment de la simulation des aspects temps réels. Les résultats des simulations sur des problèmes tests (benchmark) sont présentés et analysés dans le paragraphe 5.5 et les conclusions de ce chapitre sont énoncées en dernière partie.

5.2 Le problème DVRPTW temps réel avec temps de trajet dépendant du temps

5.2.1 Technologies temps réel

Les problèmes de décision en temps réel jouent un rôle de plus en plus important dans plusieurs secteurs ou domaines d'application car les technologies de la communication et de l'information permettent aujourd'hui d'obtenir et de traiter rapidement l'information en temps réel.

Parmi les avancées technologiques adaptées à notre problématique, il convient de citer les systèmes de positionnement global (GPS), les systèmes d'information géographique (SIG), la localisation automatique de véhicules (AVL), qui permettent d'avoir des informations en temps réel pour diverses applications dans de nombreux domaines tels que la défense (missiles d'orientation), l'aéronautique, le transport, etc. Ces avancées permettent d'obtenir des informations pertinentes représentant l'état réel d'un système. Mais, cet état change rapidement et l'ensemble des informations que nous avons à l'instant t peuvent devenir erronées à l'instant $(t + n)$. Par conséquent, ces applications doivent traiter rapidement les informations disponibles en temps réel pour apporter au système le maximum d'efficacité.

Le transport est l'un des domaines les plus intéressés par la disponibilité des informations en temps réel. En général, le but d'une entreprise de transport est de déplacer un ensemble de véhicules sur un réseau de transport pour y accomplir certains services ou demandes. Dans ce domaine, il est utile d'avoir des informations sur l'état de tous les acteurs et composantes qui interviennent dans ce problème de transport. Par exemple, nous avons besoin de connaître la localisation, la destination du véhicule, l'état de trafic, les conditions météorologiques dans le réseau de transport, etc. Ces éléments d'information doivent être exacts et pertinents. Fournir des données et des informations en telle qualité est effectivement possible maintenant en utilisant les technologies de communication.

La majorité des problèmes de transport en temps réel se concentrent sur une catégorie de problèmes dynamiques de tournées de véhicules dans laquelle les demandes ne sont pas complètement connues à l'avance mais révélées en temps réel au cours de la journée. Nous avons présenté le principe et les modèles de résolutions de ce type de problème dans le premier chapitre de ce document. Cependant en situation réelle, la variabilité des conditions de trafic routier influence fortement les résultats et elle ne peut donc être ignorée si on veut obtenir une optimisation des routes de bonne qualité. Par conséquent, une extension de ces problèmes concerne la prise en

compte en temps réel des conditions de trafic routier et notamment, le changement des vitesses ou des temps de trajet associés.

5.2.2 Revue de la littérature

La capacité (ou type) de la route, l'état des routes, les conditions météorologiques et l'état du trafic peuvent influencer fortement la vitesse des véhicules et par voie de conséquence les temps de trajet. Dans notre travail nous nous sommes focalisés sur la résolution du problème DVRPTW temps réel intégrant des temps de trajet dépendant du temps car peu de travaux de recherche ont été effectués dans ce domaine. Nous présentons ici les principaux modèles qui à notre connaissance, existent sur ce sujet. Ces modèles prennent en compte les congestions du réseau routier.

5.2.2.1 Modèle à base de profils de vitesses

[Woensel et al. 2008] et [Woensel et al. 2002] (voir la section 2.3.5 dans le chapitre 2) ont développé et traité un problème dynamique de tournées de véhicules avec temps de trajet variable intégrant les congestions potentielles du trafic. L'approche développée introduit la notion de congestion du trafic en se basant sur la théorie des files d'attente [Woensel et al. 2002].

Les principales contributions de ces travaux sont les suivantes :

1. Les temps de trajet dépendant du temps sont modélisés par une approche basée sur les files d'attente et sont intégrés dans un modèle de résolution de problèmes de tournées de véhicules. Plus spécifiquement, la nature stochastique et dynamique du temps de trajet est explicitement prise en compte par les modèles analytiques de files d'attente. Un avantage majeur de ces modèles est que les caractéristiques physiques réelles du réseau routier (densité de trafic, flux de trafic, etc.) peuvent être intégrées sous la forme de paramètres du modèle de files d'attente. En outre, les temps de trajet variables en temps réel peuvent être explicitement pris en compte par les modèles analytiques de files d'attente. Cette approche s'avère meilleure par rapport aux approches de modélisation avec vitesses dépendantes du temps mais non « temps réel » et présente des gains importants (Les gains varient entre 2,87% et 40,12%).

2. L'effet de la discrétisation du temps sur la qualité de la solution et sur le temps de calcul est évalué de manière très détaillée. Cette approche permet de définir un certain nombre d'intervalles dans lesquels le temps est discrétisé. [Woensel et al. 2008] compare son approche comportant 144 intervalles de temps de 10 minutes avec d'autres approches utilisant uniquement trois intervalles de temps telles que celles présentées aux chapitres 3 et 4 qui se basent sur les travaux de [Ichoua et al. 2003] ; il compare également son approche avec un modèle ne comportant qu'un seul intervalle de temps (la vitesse est alors constante pour la journée de service). Les trois types de discrétisation de la vitesse sont décrits dans la figure 5.1. Conjointement à cette étude de l'influence de la granularité du découpage temporel, il analyse également les effets de différents types de route sur la qualité de la solution obtenue (un, deux ou trois types de route [Ichoua et al. 2003]). Il a constaté que lorsque le nombre d'intervalles de temps augmente, la qualité de la solution est améliorée. En outre, cet effet est encore plus significatif selon le type de route considéré (27,42% dans le cas trois types de route). Un compromis doit être alors trouvé entre qualité de solution désirée et temps de calcul nécessaire.

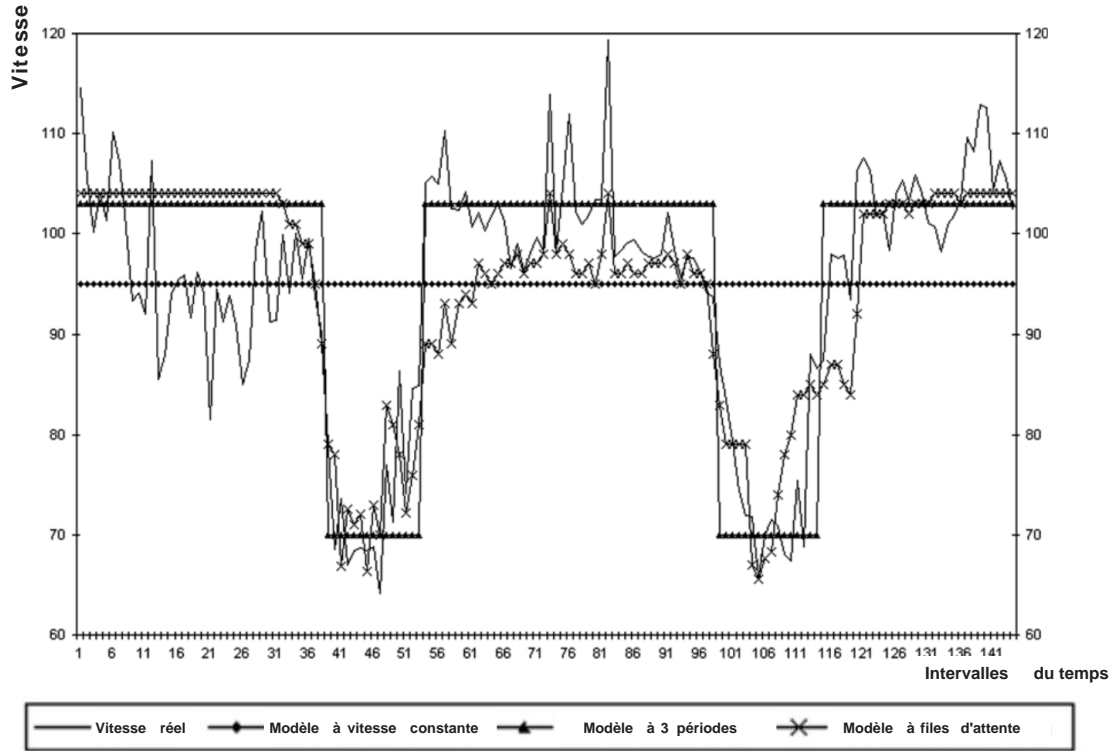


FIG. 5.1 – Les différents types de discrétisation de la vitesse

5.2.2.2 Modèle à base de profils de temps de trajet

La caractéristique principale des modèles cités précédemment est d'être basée sur des profils de temps de trajet statiques qui n'évoluent donc pas en temps réel. Pour se rapprocher des problématiques réelles, il faut étendre ces modèles afin que les temps de trajet prennent en compte l'apparition de nouvelles situations lorsque des événements non-prévisibles de trafic apparaissent. [Chen et al. 2006] a créé un profil de temps de trajet dynamique qui traduit les variations de l'état du trafic. Par rapport au modèle utilisé par [Ichoua et al. 2003], les profils de temps de trajet utilisés dans cet article ne sont donc plus statiques. Un exemple de profil de ce type est décrit dans la figure 5.2.

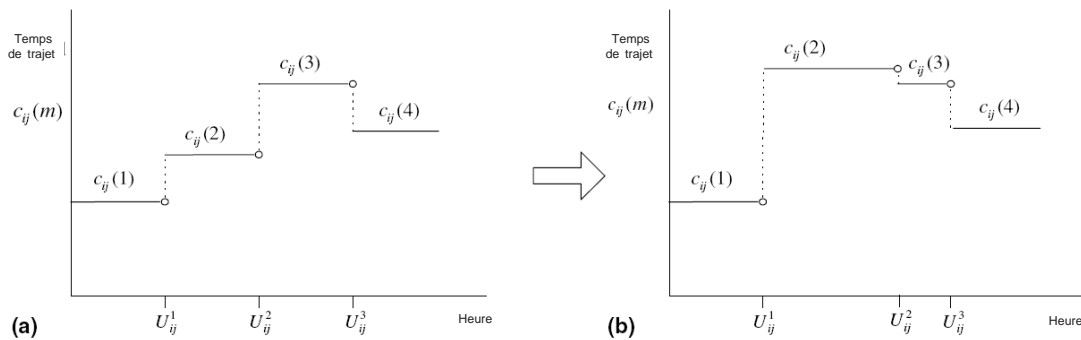


FIG. 5.2 – Modification en temps réel des profils de temps de trajet

Le modèle de temps de trajet dépendant du temps considéré est caractérisé par une fonction par morceaux. La figure 5.2(a) représente des temps de trajet prédictifs (c_{ij} entre deux clients i et j) sur les différents intervalles de temps (U_{ij}^T entre deux client i et j). Lorsqu'un incident se produit (i.e. événement a priori non prévisible), le profil doit être mis à jour en temps réel, comme montré dans la figure 5.2(b). En conséquence, le routage planifié des véhicules doit être mis à jour. L'apparition d'une congestion aléatoire peut engendrer une modification du nombre d'intervalles de temps et des temps de trajet correspondants.

Il convient de noter que le profil de temps de trajet dans ce modèle n'est pas une fonction continue à la différence du modèle de [Ichoua et al. 2003] dans lequel c'est la fonction de vitesse qui est constante par morceaux et non le temps de trajet. Donc l'hypothèse First-In-First-Out (FIFO) n'est pas respectée (voir le paragraphe 2.2.1 dans le chapitre 2).

Afin de prendre en compte les caractéristiques dynamiques spécifiques du problème DVRPTW, [Chen et al. 2006] a utilisé le concept d'horizon de temps glissant et par conséquent, a adopté une série de sous-modèles de programmation mixte en nombres entiers pour formaliser le problème. Chaque sous-modèle représente un problème VRPTW à l'instant où le temps de trajet change et/ou des nouvelles demandes apparaissent. Chacun de ces problèmes, nécessite que les véhicules partent des clients actuels (appelés nœuds critiques, et non pas dépôt) et retournent au dépôt réel, car ces véhicules ont déjà effectués une partie de leur itinéraire lorsque le temps et/ou des demandes changent. L'objectif associé à ce problème consiste à trouver, à tout instant, un ensemble de tournées de véhicules de coût minimal, du problème formé des nœuds critiques N_c , de l'ensemble des nœuds non-servis N_u et du dépôt. Le coût des tournées de véhicule se compose des temps de trajet, des temps d'attente avant service et des temps d'attente avant départ.

Le nœud critique est une caractéristique importante du problème DVRPTW en temps réel qui est décrite plus en détail ci-dessous :

Une fois qu'un client a été servi, celui-ci est supprimé de la planification prévue actuelle. Seuls les clients qui ne sont pas servis seront considérés dans la planification suivante des tournées de véhicules. Pour que la tournée suivante puisse être effectuée, les nœuds critiques doivent être identifiés immédiatement dès qu'une demande apparaît en temps réel ou qu'un temps de trajet change.

Les nœuds critiques peuvent être identifiés par la méthode suivante : on suppose que deux véhicules effectuent leur tournée et sont situés à un instant donné aux positions représentées en figure 5.3. Un véhicule se dirige vers le nœud 4, et l'autre véhicule est arrêté au nœud 8. Donc les nœuds critiques sont respectivement les nœuds 4 et 8 et l'ensemble des nœuds non servis sont 5, 9 et 10.

5.3 Approche de résolution du DVRPTW temps réel avec temps de trajet dépendant du temps

Les modèles de résolution du problème avec des temps de trajet dépendant du temps décrits ci-dessus considèrent les modifications du trafic routier en temps réel mais ne prennent pas en compte la prévision sur l'état futur de ce trafic. Les tournées de véhicules sont planifiées et se basent sur l'état actuel du trafic en faisant l'hypothèse que celui-ci restera identique dans le futur. Dans certains cas, les tournées des véhicules seront effectuées dans des conditions de trafic futur qui ne correspondent pas aux conditions de trafic utilisées lors de la planification. C'est

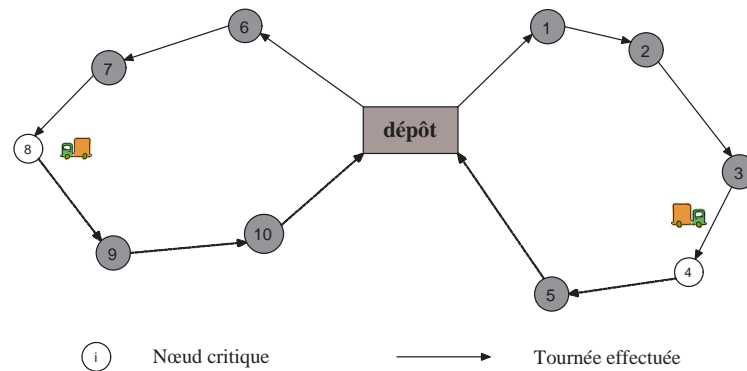


FIG. 5.3 – Tournées de véhicules et nœuds critiques

pourquoi nous présentons un modèle de prévision de vitesse destiné à fournir des projections sur l'état futur du trafic qui seront intégrées lors de la planification afin d'obtenir un modèle plus proche de la réalité.

5.3.1 Méthode de calcul de temps de trajet réel

La variation des vitesses réelles observées est une fonction continue irrégulière qui rend très difficile l'obtention de temps de trajet précis. Pour élaborer notre modèle temps réel de temps de trajet dépendant du temps et calculer les temps de trajet réels, cette variation de vitesse réelle est modélisée sous la forme d'une fonction par morceaux. L'horizon de la planification (i.e. journée) est divisé en plusieurs périodes et la vitesse est variable sur la durée de la journée mais constante dans chaque période.

5.3.2 Le modèle de prévision

Dans le chapitre précédent, nous avons présenté le principe d’une méthode basée sur un algorithme génétique pour résoudre le problème DVRPTW avec des temps de trajets dépendant du temps. Cette approche recherche des tournées de bonne qualité en utilisant un modèle simplifié de calcul de temps de trajet. Dans notre problème DVRPTW avec temps de trajet dépendant du temps en temps réel, les variations des vitesses réelles sont remontées en temps réel au système de planification. C’est pourquoi il serait utile de pouvoir exploiter ces informations pour déterminer une vitesse prévisionnelle pour les prochaines périodes de temps. Par conséquent, il faut définir un modèle de prévision pour prévoir la vitesse que nous allons utiliser dans cette approche. Nous adoptons un modèle de lissage exponentiel pour établir cette prévision ([Gourieroux et al. 1995]) car il est peu coûteux en temps de calcul et s’adapte donc bien à un contexte temps réel.

Soit une série chronologique $Y_1, Y_2, Y_3, \dots, Y_{T-2}, Y_{T-1}, Y_T$. La valeur de Y_T , étant connue à l'instant T , le problème consiste à prédire Y_{T+h} où h est l'horizon de prévision. Pour ce faire, on fera appel à une méthode que l'on appelle le lissage exponentiel simple.

Cette méthode se base sur le fait que plus les observations sont éloignées de la période T , plus leur influence sur la prévision sera faible. On considère que cette influence décroît de façon exponentielle. La formule de prévision est définie comme suit :

$$\hat{Y}_{T+h} = \beta \sum_{j=0}^{T-1} (1-\beta)^j Y_{T-j} \quad (5.1)$$

Selon la formule 5.1, la valeur prévisionnelle de Y_{T+h} qui est notée \hat{Y}_{T+h} ne dépend pas de l'horizon h de prévision, on peut donc simplifier l'écriture en posant $\hat{Y}_{T+h} \equiv \hat{Y}_T$. \hat{Y}_T désigne la prévision que l'on fait à l'instant T pour la période suivante c'est-à-dire $T + h$. La formule 5.1 exprime le fait que \hat{Y}_T est une moyenne des observations passées où le poids de chaque observation décroît de façon exponentielle avec le temps. Le paramètre β ($0 < \beta < 1$) se nomme coefficient de lissage. Plus β tend vers 0, plus la prévision sera influencée par les observations éloignées dans le temps. D'ailleurs, on dit que la prévision est plus rigide à mesure que β tend vers 0 dans la mesure où la prévision n'est pas sensible aux fluctuations à court terme. Plus β tend vers 1, plus la prévision est influencée par les observations récentes. Considérons à titre d'exemple le cas dans lequel $\beta = 0.90$ et $T = 3$:

$$\hat{Y}_{3+h} = 0.90Y_3 + 0.90 \times 0.10Y_2 + 0.90 \times 0.10^2Y_1 \quad (5.2)$$

$$\hat{Y}_{3+h} = 0.90Y_3 + 0.09Y_2 + 0.009Y_1 \quad (5.3)$$

Dans l'estimation de Y_{3+h} , on remarque que lorsque β est grand (0,90), l'observation la plus récente Y_3 aura beaucoup d'influence (un poids de 0,90) sur Y_3 .

On peut également réécrire 5.1 de la manière suivante :

$$(1 - \beta)\hat{Y}_{T-1} = \beta \sum_{j=1}^{T-1} (1 - \beta)^j Y_{T-j} \quad (5.4)$$

$$\hat{Y}_T - (1 - \beta)\hat{Y}_{T-1} = \beta \left[\sum_{j=0}^{T-1} (1 - \beta)^j Y_{T-j} - \sum_{j=1}^{T-1} (1 - \beta)^j Y_{T-j} \right] \quad (5.5)$$

$$\hat{Y}_T - (1 - \beta)\hat{Y}_{T-1} = \beta Y_T \quad (5.6)$$

$$\hat{Y}_T = \beta Y_T + (1 - \beta)\hat{Y}_{T-1} \quad (5.7)$$

À partir de 5.7, on peut interpréter \hat{Y}_T comme étant une moyenne pondérée entre la valeur estimée \hat{Y}_{T-1} faite en période $(T - 1)$ et la dernière observation de la série. On pose $\hat{Y}_1 = Y_1$ et on initialise le processus de lissage à $T = 1$. L'avantage de 5.7 est qu'on n'a pas à lisser de nouveau tout le processus lorsqu'une nouvelle observation s'ajoute à la série.

Dans notre problème, nous choisissons l'équation 5.7 pour prévoir la vitesse à la période suivante. Plus β tend vers 1, plus la prévision sera influencée par la vitesse réelle mesurée récemment. Plus β tend vers 0, plus la prévision est influencée par la vitesse prévisionnelle précédente.

Les fonctions « vitesse mesurée » et « vitesse prévisionnelle » sont évaluées sur plusieurs périodes de temps ; chaque période a une durée identique dans laquelle la vitesse est considérée comme constante. Pour chaque période T et pour un arc donné joignant deux clients, une vitesse mesurée est fournie et remontée au système de planification en temps réel. Elle permet ainsi de calculer la vitesse prévue pour la prochaine période $T + 1$. Cette vitesse prévisionnelle (période $T + 1$) et l'historique des vitesses des autres périodes restantes (période $T + 2$, $T + 3$, etc.) permet de former ensemble un nouveau profil de vitesse pour cet arc. Ces nouveaux profils de vitesse seront ensuite utilisés pour calculer le temps de trajet pour chaque paire de clients (matrice Origine-Destination). Cette matrice O-D est utilisée pour déterminer les tournées avec

l'algorithme génétique décrit dans le chapitre 4. Les vitesses dans le profil de planification intègrent les deux vitesses prévisionnelles et historiques. Elles sont déterminées comme mentionné ci-dessous :

$$\hat{V}_{T+h} = \begin{cases} \beta V_T + (1 - \beta) \hat{V}_T, \text{ pour } h = 1 \\ V_{his(T+h)}, \text{ pour } h \geq 2 \end{cases} \quad (5.8)$$

La figure 5.4 décrit un exemple d'élaboration d'un nouveau profil de planification. Nous supposons que la journée de service est divisée en neuf périodes de temps. Le profil historique de vitesse est une fonction par morceaux constituée de trois parties ; les première, seconde et troisième parties correspondent respectivement aux périodes comprises en T et $T + 3$, $T + 3$ et $T + 6$ et $T + 6$ et $T + 9$. Le nombre des périodes de temps utilisées pour mesurer et prévoir la vitesse est neuf également. A l'instant $T + 1$, on mesure la vitesse dans l'intervalle entre T et $T + 1$ (soit v_m cette vitesse), puis on prévoit la vitesse dans l'intervalle entre $T + 1$ et $T + 2$ (soit v_p cette vitesse). De cette façon, nous obtenons le nouveau profil de vitesse à utiliser pour planifier les tournées est construit en utilisant l'équation 5.9. Nous appelons ce nouveau profil le « profil de planification ».

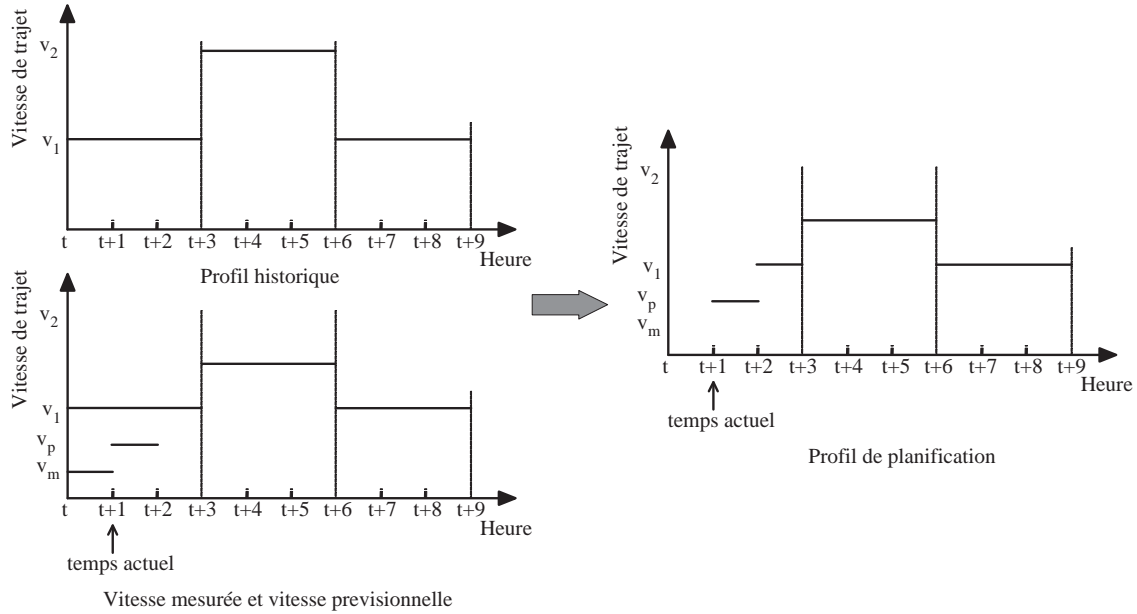


FIG. 5.4 – Principe d'élaboration du profil utilisé pour la planification de tournées

5.3.3 Calcul des temps de trajet intégrant la prévision

Nous complétons donc l'algorithme de calcul des temps de trajet présenté dans le chapitre 2 pour prendre en considération la prévision, comme l'algorithme 5.1 ci-dessous :

Dans cet algorithme, les notations suivantes sont utilisées :

- t : temps courant
- t_0 : instant de départ. $t_0 \in [\bar{t}_k, \bar{t}_k]$ et $t_0 \in [\bar{t}'_k, \bar{t}'_k]$
- d : distance restante
- d_{ij} : distance entre les clients i et j
- t' : temps d'arrivée au client

Algorithme 5.1 Calcul du temps de trajet intégrant la prévision*Données matrices des profils d'historique et des profils de prévision*

```

 $t \leftarrow t_0$  // Temps courant initialisé avec temps de départ
 $d \leftarrow d_{ij}$  // Distance restante initialisée avec la distance totale
 $t' \leftarrow t + (d/V_p)$  // Estimation du temps restant nécessaire de parcours
Si ( $t' > \bar{t}_k$ ) Alors // Si temps restant dépasse la borne supérieure de la période
     $d \leftarrow d - V_p(\bar{t}_k - t)$  // de prévision, effectuer un changement de période vitesse
     $t \leftarrow \bar{t}_k$ 
     $t' \leftarrow t + (d/V_k)$ 
    Tant Que ( $t' > \bar{t}_k$ ) Faire // Tant que temps restant dépasse la borne supérieure
         $d \leftarrow d - V_k(\bar{t}_k - t)$  // de la période, effectuer un changement de période vitesse
         $t \leftarrow \bar{t}_k$ 
         $t' \leftarrow t + (d/V_{k+1})$ 
         $k \leftarrow k + 1$ 
    Fin Tant Que
Fin Si
Retourner ( $t' - t_0$ ) // Retourner le temps de trajet total

```

- V_p : vitesse prévisionnelle à la période de prévision
- \bar{t}_k : temps de la borne supérieure à la période de prévision
- V_k : vitesse à la période k
- \bar{t}_k : temps de la borne supérieure à la période k

5.4 Environnement de simulation

5.4.1 Fichiers de vitesses

Pour faciliter la simulation, nous avons utilisé trois types de fichiers qui stockent respectivement les vitesses historiques, les vitesses mesurées et les vitesses prévisionnelles. Les deux premiers fichiers sont élaborés au départ de la simulation. Seul le dernier fichier contenant les vitesses prévisionnelles sera construit pendant la phase de simulation. Ces vitesses prévisionnelles sont calculées par le modèle de prévision (voir la section 5.3.1) en utilisant les vitesses historiques et mesurées. La valeur de cette vitesse prévisionnelle va remplacer la valeur de la vitesse correspondante à la même période dans le profil historique. Le simulateur utilise désormais ce nouveau « profil de planification » pour planifier les tournées.

5.4.2 Synchronisation du planificateur

Dans un contexte réel, les vitesses prévisionnelles utilisées pour planifier les tournées ne sont pas égales aux vitesses réelles des véhicules qui effectuent les tournées. Il est nécessaire de synchroniser les arrivées des véhicules chez les différents clients visités. En général, le véhicule n'arrivera pas à l'heure prévue, il faut donc tenir compte de ce décalage temporel pour planifier la suite des clients à visiter. Par conséquent, il faut étendre les fonctionnalités du simulateur présenté au chapitre 4. Le fonctionnement de ce nouveau simulateur est présenté dans la figure 5.5 :

La figure 5.5 considère deux clients consécutifs i et j dans une tournée de véhicule. Le temps estimé d'arrivée au client i est le point ' a ' sur l'axe temporel et le temps estimé de fin de service

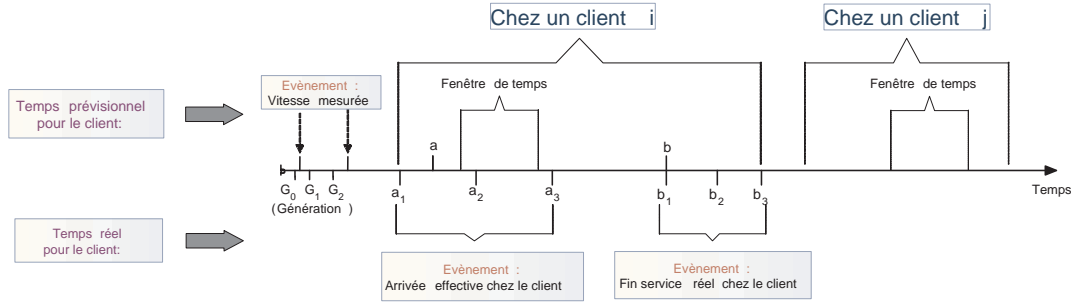


FIG. 5.5 – Synchronisation des temps d'arrivée

est le point ' b' '. Ces temps estimés par le modèle sont basés sur l'instant de départ chez le client précédent dans la tournée du véhicule, le profil de planification et l'algorithme de calcul de temps de trajet avec prévision (vus précédemment). Dans la réalité, des événements imprévisibles (congestion du réseau routier, etc.) vont venir perturber ces estimations. Le véhicule peut arriver chez le client i à l'instant ' a_1 ', ' a_2 ' ou ' a_3 '.

Les instants ' a_1 ', ' a_2 ' et ' a_3 ' correspondent respectivement aux trois possibilités d'arrivée chez le client i : en avance par rapport à la borne inférieure de la fenêtre de temps, ou dans la fenêtre elle-même ou encore après la fenêtre de temps du client i . Pour synchroniser la planification des véhicules, il est donc nécessaire de mettre à jour les temps réels d'arrivée, d'attente, de retard, ainsi que les temps réels de début et de fin de service pour ce client i à l'instant correspondant à l'événement « arrivée effective chez le client ». Dès lors, l'instant de départ de ce client i vers le client suivant j , et le temps réel de fin de service peuvent être recalculés. A titre d'exemple le temps réel de fin de service est le point ' b_1 ' si le véhicule est arrivé en ' a_1 ' ; il correspond aux points ' b_2 ' ou ' b_3 ' si il est arrivé en ' a_2 ' ou ' a_3 '. La suite de la tournée peut alors être planifiée par l'algorithme génétique en utilisant le « profil de planification » entre le client i et les autres clients.

Il y a quatre événements principaux gérés par le simulateur : « Vitesse mesurée » ; « Arrivée effective chez un client » ; « Fin de service d'un client » et « Arrivée d'un nouveau client ». Nous allons les présenter plus en détail à la section suivante.

5.4.3 Architecture du simulateur

Pour traiter le problème VRPTW dynamique avec prise en compte de l'information de trafic en temps réel, nous avons étendu les fonctionnalités du simulateur décrit au paragraphe 4.5 du chapitre 4 qui simule une journée de service pour le problème VRPTW dynamique.

Le simulateur s'appuie principalement sur deux composants qui coopèrent : l'« Algorithme Génétique » et la « Gestion d'Événements ».

Le composant « Algorithme Génétique » utilise maintenant le « profil de planification » qui est obtenu par le modèle de prévision décrit au paragraphe 5.3.2. Ce profil est mis à jour périodiquement avec une fréquence qui est fixée au début de la simulation.

Le composant « Gestion d'Événements » traite les quatre types d'événements : « Vitesse mesurée » ; « Arrivée effective chez un client » ; « Fin de service d'un client » et « Arrivée d'un nouveau client ».

L'environnement de simulation est présenté sur la figure 5.6. On distingue d'une part le planificateur qui produit les solutions et d'autre part le composant qui simule les événements du monde réel. Ce dernier est lui-même composé des trois parties suivantes : « Vitesses réelles », « Clients » et « Véhicules ».

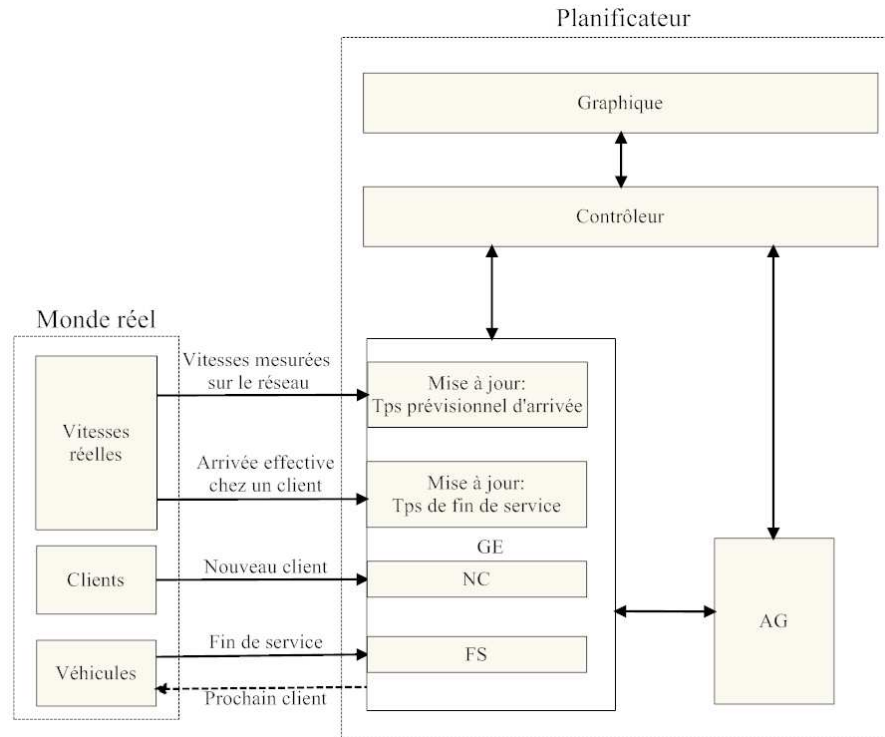


FIG. 5.6 – Simulateur pour le DVRPTW temps réel avec de temps de trajet dépendant du temps

Le diagramme de séquence de la figure 5.7 ci-dessous représente les interactions entre les deux composants AG et GE du planificateur. Il s'agit d'une extension du modèle d'interactions présenté au chapitre 4 concernant la résolution du problème DVRPTW dynamique. Après chaque génération de l'AG, l'examen des interactions survenues pendant la dernière génération de l'AG est déclenchée. L'examen commence par l'envoi d'un message d'interrogation « Événement ? ». Ce message permet de vérifier l'existence de nouveaux événements à traiter. Ces interactions se terminent par le message « Fin ». Entre ces deux messages, le composant GE vérifie l'existence des événements « Vitesse mesurée », « Arrivée chez un client », « Fin de service » et « Nouveaux clients » et il déclenche les services correspondants.

5.4.4 Gestion d'Événements

Pour chaque occurrence d'évènement, nous appliquons une procédure correspondant à cet évènement ; en conséquence nous avons donc dû définir quatre procédures répondant aux quatre types d'évènements présentés ci-dessous :

5.4.4.1 L'évènement « Vitesse mesurée »

La vitesse effective mesurée des véhicules est remontée à intervalle régulier au système de planification. A chaque période et pour chaque arc du réseau le profil de planification est mis à

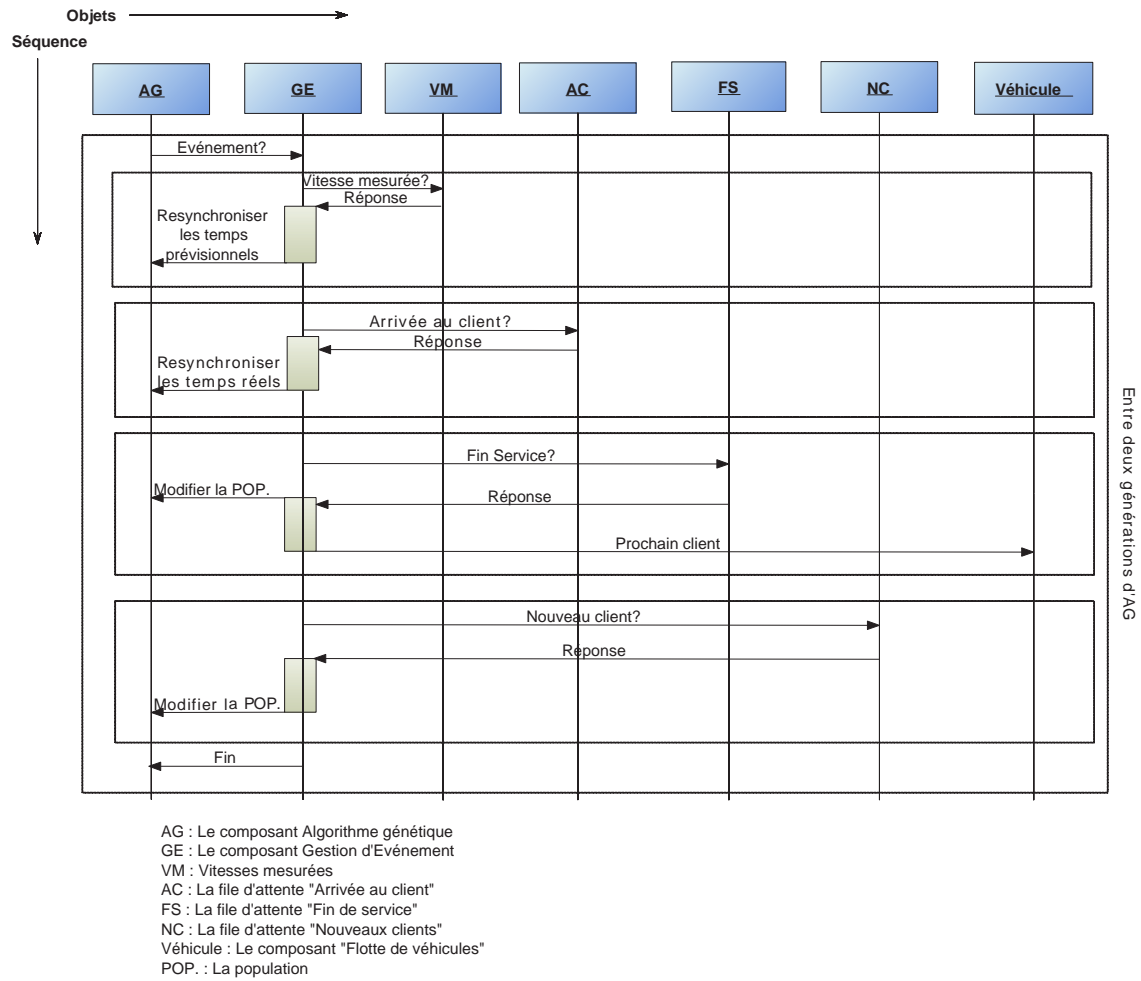


FIG. 5.7 – Diagramme de séquence entre les composants AG et GE du simulateur

jour. Une nouvelle vitesse prévisionnelle est calculée pour période $T + h$ et le temps prévisionnel d'arrivée chez le client en cours pour chacun des véhicules. Dans les conditions réelles, si la vitesse est mesurée à l'intervalle de temps (i.e. période) h , il faut alors prévoir la vitesse de la période de temps suivante. Cette vitesse sera plus proche de la vitesse mesurée que de la vitesse historique. Cette mise à jour se produit après chaque période h (algorithme 5.2).

Algorithme 5.2 Algorithme de traitement de l'évènement « Vitesse mesurée »

Entrée :*Profil* : Profils de vitesse de planification.*t* : Le numéro de période correspondant au temps courant.**Sortie :***Profil* : Profils modifiés de vitesse.*Temps* : Ensemble des temps prévisionnel d'arrivée chez le client en cours de chaque véhicule.**Début :**

Pour chaque route :

Calcul de la vitesse prévisionnelle ;

Mise à jour du profil de planification.

Pour chaque véhicule :

Calcul du temps prévisionnel d'arrivée.

Fin.

Ce type d'évènement est généré automatiquement par le simulateur à chaque période h .

5.4.4.2 L'évènement « Arrivée effective chez un client »

Cet évènement permet de remonter au système de planification le temps d'arrivée effectif chez le client. Il est généralement différent du temps d'arrivée prévisionnel. Il faut donc mettre à jour (i.e. synchroniser) les temps de trajet, les temps d'attente, les temps de retard et les temps de fin de service de ce client afin de correspondre à la situation réelle (algorithme 5.3).

5.4.4.3 L'évènement « Fin de service »

Cet algorithme est proche de l'algorithme défini au chapitre 4 mais il faut calculer le temps réel de trajet entre le client servi et le prochain client à servir (algorithme 5.4). On peut ainsi générer un évènement « Arrivée effective chez un client ».

5.4.4.4 L'évènement « Arrivée d'un nouveau client »

L'algorithme associé à cet évènement est inchangé par rapport à l'algorithme 4.3 décrit au paragraphe 4.5.6 du chapitre 4. Il s'agit d'une nouvelle demande de client à prendre en compte. Cet évènement est généré par le simulateur à l'aide des données du fichier benchmark à exécuter.

5.4.5 Déroulement d'une simulation

Le principe général de résolution de notre problème peut être divisé en deux phases : la première est une phase initiale qui correspond à la période précédant le départ des véhicules

Algorithme 5.3 Algorithme de traitement de l'évènement « Arrivée effective chez un client »

Entrée :

Client : Le client sur lequel arrive le véhicule.
Vehicule : Le véhicule qui a généré l'évènement.
Temps d'arrivée réel.
Fenêtre de temps.
Durée de service de ce client.

Sortie :

Temps réel d'attente.
Temps réel de retard.
Temps réel de fin service.

Début :

Calculer le temps réel d'attente de ce client (basé sur la fenêtre de temps).
Calculer le temps réel de retard de ce client (basé sur la fenêtre de temps).
Calculer le temps réel fin de service de ce client (basé sur la durée de service de ce client)
pour l'évènement « Fin de service » correspondant à ce client.

Fin.

Algorithme 5.4 Algorithme de traitement de l'évènement « fin de service »

Entrée :

Vehicule : Le véhicule qui a généré l'évènement.
Chrom : Meilleur chromosome de la population courante.
Tournees : L'ensemble des tournées contenant les clients déjà servis ou en cours de service.
Client : Client qui vient d'être servi.

Sortie :

Tournees : L'ensemble des tournées contenant les clients déjà servis ou en cours de service.
Nouveaucient : Le nouveau client associé au *Vehicule*.
Temps réel d'arrivée au *Nouveaucient*.

Début :

Solution = Décoder le chromosome *Chrom* (par l'algorithme 4.1).
En déduire le nouveau client à servir par la *Vehicule*
(premier Client non encore servi du *Vehicule*).
Ajouter ce nouveau client à la tournée correspondante dans *Tournees*.
Calculer le temps de trajet réel entre le client servi et ce nouveau client.
En déduire temps d'arrivée réel chez le nouveau client à servir pour générer
l'évènement « Arrivée effective chez un client ».
Supprimer le *Client* servi de tous les chromosomes de la population.

Fin.

(phase statique), la seconde est la planification des tournées lorsque les véhicules ont démarrés (phase dynamique).

Pour évaluer la qualité de la solution obtenue on va rejouer la solution dans un contexte basé sur les vitesses réelles (phase réelle). Ces trois phases utilisent séparément trois types de profils de vitesses : le profil historique, le profil de planification et le profil réel. Ces profils sont mémorisés dans les fichiers de vitesse (section 5.4.1).

5.4.5.1 Phase statique basée sur les profils historiques

Dans cette étape les véhicules ne sont pas encore partis et les données de vitesse réelle et prévisionnelle ne sont pas disponibles. Par conséquent, les véhicules vont recevoir les tournées planifiées par l'algorithme génétique en utilisant uniquement les profils historiques. Ces profils viennent de données statistiques et ils sont modélisés par trois périodes de vitesses différentes (matin, midi et après midi) de la journée.

5.4.5.2 Phase dynamique basée sur les profils de planification

Une fois le départ des véhicules effectué, les données concernant les vitesses mesurées sont remontées périodiquement au système de planification et deviennent ainsi disponibles. Elles permettent de calculer les vitesses prévisionnelles et les temps de trajet prévisionnels. Les tournées sont alors planifiées dans leur intégralité. La vitesse réelle est mesurée pour chaque intervalle de temps. Dans notre problème cet intervalle de temps est égal au $1/24$ du temps d'ouverture de la journée de service.

5.4.5.3 Phase réelle basée sur les profils de vitesses réelles

Les tournées ont été élaborées en utilisant le profil de planification pour les vitesses. Cependant, les véhicules circulent sur les routes avec des vitesses réelles qui seront différentes. Pour apprécier la qualité des solutions obtenues par le modèle de prévision, on va rejouer la solution obtenue dans un contexte où l'on dispose des vitesses réelles. Dans cette phase nous recalculons les temps de trajet réels en utilisant les profils de vitesses réelles. La vitesse sur ces profils demeure constante sur des intervalles de courte durée. Dans notre problème cet intervalle est égal au $1/96$ du temps d'ouverture de la journée de service.

5.5 Expériences et résultats

5.5.1 Génération des vitesses réelles

Afin de simuler l'évolution réelle du parcours des véhicules et analyser notre modèle de résolution du problème DVRPTW temps réel avec temps de trajet dépendant du temps, il est nécessaire de générer des vitesses réelles. Cette génération est basée principalement sur les profils de vitesses historiques. Plus précisément les vitesses réelles seront également composées de phénomènes de perturbation et de congestion dont la simulation est décrite ci-dessous.

Une perturbation est définie comme une variation assez faible de la vitesse, une congestion représente une variation importante de la vitesse.

Pour simuler les perturbations, nous divisons d'abord le temps d'ouverture de la journée en 12 périodes ; le profil des vitesses historiques étant quant à lui décomposé en 3 périodes. La vitesse perturbée dans la période i est définie comme suit : $V'_i = V_i \times (1 + n)$ où n est un nombre

aléatoire tel que $n \in [-0.1, +0.1]$ et V_i est la vitesse historique dans la période i . On obtient ainsi un profil de vitesses perturbées. En raison de la variabilité importante de ce profil et pour pouvoir calculer rapidement les temps de trajet correspondants, nous transformons cette fonction continue de vitesse en une fonction par morceaux avec des périodes de temps de courte durée. Supposons que le temps réel d'ouverture d'une journée de service $[0, T]$ soit de 8 heures, alors la vitesse réelle est mise à jour toutes les 5 minutes. Donc le nombre de périodes est de 96. Le profil de vitesses avec perturbations est présenté sur la figure 5.8 :

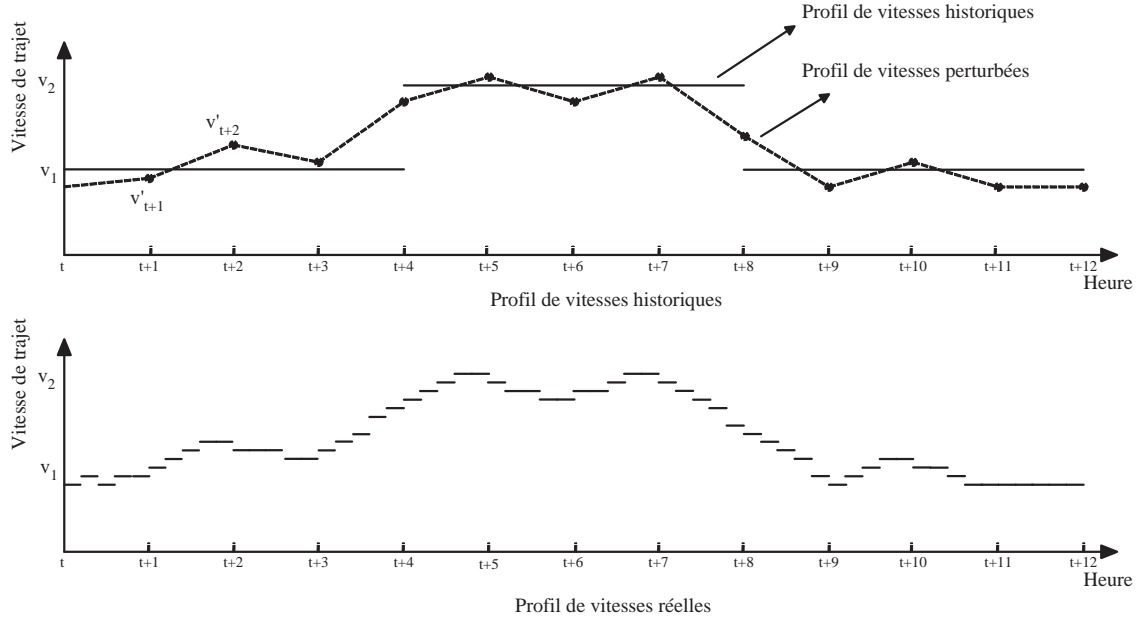


FIG. 5.8 – Profil de vitesses réelles avec perturbations

Pour simuler une congestion, nous supposons que la vitesse baisse avec une tendance linéaire jusqu'à un seuil bas (par exemple de 0,54 à 0,1) et puis augmente avec une tendance inversée jusqu'à la valeur initiale. Cette congestion peut être positionnée sur quelques périodes de l'horizon d'ouverture ; pour les autres périodes, le profil de vitesse est inchangé et ne comportera que les perturbations légères que nous avons présentées ci-dessus.

Par exemple, si une congestion se produit dans la première période de temps, le profil réel de vitesse est présenté dans la figure 5.9 :

Pour réaliser nos simulations, nous supposons que ces congestions peuvent se produire indépendamment dans les trois périodes de temps du profil de vitesses historiques. En conséquence, huit distributions différentes de congestions ont été élaborées ; ces distributions référencées de 1 à 8 dans la suite de ce document sont représentées sur la figure 5.10 :

Notons que dans cette figure 5.10, nous avons simplifié les distributions de vitesses en ne montrant que les congestions. Par exemple sur la distribution 1, il n'y a aucune congestion sur ce profil de vitesses. Sur la distribution 8, on observe une congestion sur chaque période de la journée. Entre ces deux cas extrêmes, on observera des congestions sur une ou deux périodes de la journée (distribution 2 à 7).

Nous avons par ailleurs considéré dans notre réseau trois types de route (route, voie rapide, autoroute). Pour cela nous avons créé des profils différents de vitesses réelles pour les trois types

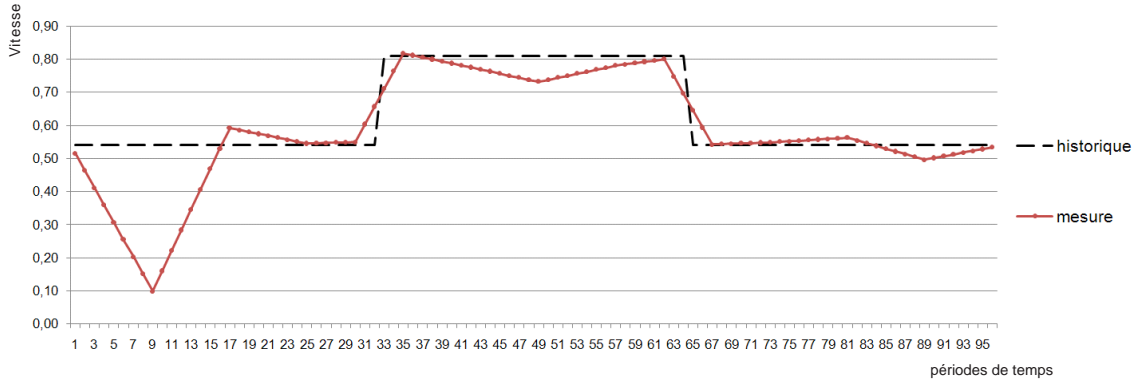


FIG. 5.9 – Un exemple de profil de vitesses avec une congestion et des perturbations

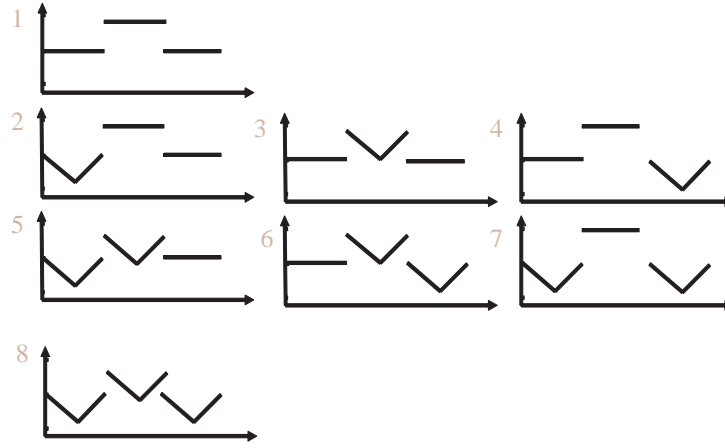


FIG. 5.10 – Les différentes distributions de congestion

de route comme mentionnés dans [Ichoua et al. 2003]. Chaque type de route est associé à une distribution différente dans un scénario donné.

5.5.2 Réglage du facteur de lissage β

Dans notre simulation, on supposera recevoir l'information trafic (i.e vitesse réelle mesurée) avec une période correspondante au $1/24$ du temps d'ouverture de la journée de service. Pour un horizon d'ouverture de 8 heures, cela revient à recevoir la vitesse réelle mesurée toutes les 20 minutes.

La vitesse prévisionnelle utilisée pour planifier les tournées sera également mise à jour avec cette même périodicité. Pour trouver la meilleure valeur du paramètre β de notre modèle de prévision, nous calculons d'une part les valeurs de pourcentage d'erreurs entre les vitesses historiques et les vitesses réelles (PH) et d'autre part entre les vitesses prévisionnelles et les vitesses réelles (PP). Ces pourcentages sont calculés pour chaque valeur possible du couple formé par le paramètre β (variant de 0,1, 0,2, 0,5 à 0,8) et du type de distributions de congestion (de 1 à 8). Ce pourcentage d'erreur est défini comme suit :

$$P = \sqrt{\frac{\sum (d_i - m_i)^2}{n}} / \bar{Y} \quad (5.9)$$

où :

- P : Pourcentage d'erreur,
- d_i : Valeur de la période i pour la série vitesses réelles,
- m_i : Valeur de la période i pour la série vitesses historiques ou vitesses prévisionnelles,
- n : Nombre de périodes,
- \bar{Y} : Valeur moyenne de la série vitesses réelles.

Les pourcentages d'erreur pour les trois types de profil de vitesse et pour le scénario 1 (voir le tableau du chapitre 4), sont mentionnés dans le tableau 5.1.

Nous pouvons remarquer que lorsque le paramètre β est égal à 0,8, le pourcentage d'erreur est plus petit que pour les autres valeurs de β . Donc nous choisissons la valeur 0,8 pour le paramètre β de notre modèle de prévision. En examinant les résultats on observe que le modèle basé sur les vitesses historiques est davantage influencé par les variations des vitesses réelles mesurées. Le taux d'erreur observé pour le modèle basé sur les prévisions est nettement inférieur. Le modèle se veut donc plus réactif aux dernières observations mesurées.

5.5.3 Méthode d'évaluation de la qualité des solutions

De même que dans le chapitre 4, nous avons choisi d'utiliser le jeu de problèmes tests fourni par Solomon [Solomon 1987] et modifié par Gendreau [Gendreau et al. 1999]. Le réglage des paramètres est identique aux réglages utilisés pour la résolution du problème DVRPTW présenté au chapitre 4 ; Ces paramètres figurent dans les tableaux B.1 et B.2 de l'annexe B.1. Dans ce problème, nous avons fixé à 60 minutes le temps nécessaire pour simuler le temps d'ouverture d'une journée de services puisque le nombre de périodes de temps à considérer est grand (96 périodes dans le profil de vitesses réelles) ; à titre d'information la durée de simulation du temps d'ouverture était de 30 minutes dans les expérimentations précédentes.

La méthode d'évaluation utilisée est basée sur la comparaison des performances des solutions fournies par les deux modes de résolution décrits ci dessous. Pour chacun des modes on procède en deux étapes notées a et b :

- Résolution1 : a. Le problème est résolu en utilisant le « profil de planification » : soit S la solution obtenue ; b. Les tournées correspondantes à S sont ensuite réévaluées par simulation en jouant les parcours avec le « profil de vitesses réelles » généré comme indiqué dans la section précédente. Cette solution fournit les indicateurs (équation 4.8) R1 ;
- Résolution2 : a. Le problème est résolu en utilisant le « profil historique » de vitesse : soit S' la solution obtenue ; b. Les parcours correspondants à S' sont réévalués par simulation comme dans le cas précédent avec le « profil des vitesses réelles ». Cette solution fournit les indicateurs R2.

La comparaison des indicateurs R1 et R2 obtenus par ces deux modes de résolution permet d'évaluer l'apport d'une prise en compte d'un modèle temps réel de temps de trajet variable. Nous utilisons également un indicateur « gain » dans ce paragraphe pour évaluer le niveau d'amélioration de l'approche proposée.

Le processus de comparaison des deux modes de résolution est présenté sur la figure 5.11.

β	Distribution	Profil 1		Profil 2		Profil 3	
		PH ^a (%)	PP ^b (%)	PH (%)	PP (%)	PH (%)	PP (%)
0,8	1	6,22	3,41	6,25	3,45	6,18	3,37
	2	18,65	5,95	19,99	6,27	19,99	6,27
	3	30,91	8,21	32,54	8,62	32,54	8,62
	4	18,95	5,96	20,30	6,29	20,30	6,29
	5	38,14	10,18	40,65	10,82	40,65	10,82
	6	38,52	10,24	41,06	10,88	41,06	10,88
	7	27,46	8,10	29,69	8,66	29,69	8,66
	8	46,15	12,31	49,69	13,22	49,69	13,22
	Moyenne	28,13	8,05	30,02	8,52	30,01	8,51
0,5	1	6,22	4,86	6,25	4,91	6,18	4,79
	2	18,65	8,52	19,99	8,99	19,99	8,99
	3	30,91	11,62	32,54	12,20	32,54	12,20
	4	18,95	8,73	20,30	9,20	20,30	9,20
	5	38,14	14,47	40,65	15,38	40,65	15,38
	6	38,52	14,68	41,06	15,61	41,06	15,61
	7	27,46	11,78	29,69	12,60	29,69	12,60
	8	46,15	17,67	49,69	18,98	49,69	18,98
	Moyenne	28,13	11,54	30,02	12,23	30,01	12,22
0,2	1	6,22	8,40	6,25	8,49	6,18	8,28
	2	18,65	14,48	19,99	15,27	19,99	15,27
	3	30,91	18,44	32,54	19,39	32,54	19,39
	4	18,95	15,76	20,30	16,61	20,30	16,61
	5	38,14	23,31	40,65	24,82	40,65	24,82
	6	38,52	24,38	41,06	25,94	41,06	25,94
	7	27,46	20,74	29,69	22,15	29,69	22,15
	8	46,15	29,38	49,69	31,59	49,69	31,59
	Moyenne	28,13	19,36	30,02	20,53	30,01	20,51
0,1	1	6,22	11,70	6,25	11,83	6,18	11,52
	2	18,65	19,20	19,99	20,19	19,99	20,19
	3	30,91	21,52	32,54	22,64	32,54	22,64
	4	18,95	21,06	20,30	22,13	20,30	22,13
	5	38,14	27,64	40,65	29,45	40,65	29,45
	6	38,52	29,19	41,06	31,06	41,06	31,06
	7	27,46	27,33	29,69	29,10	29,69	29,10
	8	46,15	35,38	49,69	38,04	49,69	38,04
	Moyenne	28,13	24,13	30,02	25,56	30,01	25,52

^a Pourcentage d'erreur entre les vitesses historiques et les vitesses réelles

^b Pourcentage d'erreur entre les vitesses prévisionnelles et les vitesses réelles

TABLEAU 5.1 – Variation du pourcentage d'erreur en fonction du paramètre β et du profil de route

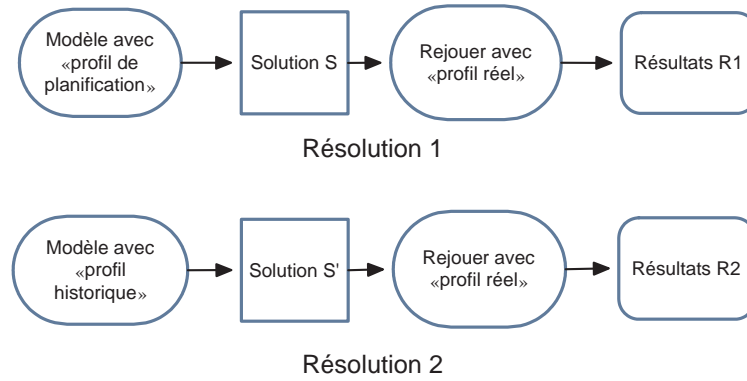


FIG. 5.11 – Deux modes de résolution

Comme mentionné dans le paragraphe 4.6.2 du chapitre 4, dans ce problème DVRPTW temps réel avec temps de trajet dépendant du temps, le profil historique est composé de 3 périodes qui ont la même durée. Ces périodes sont données dans le tableau 4.3 du chapitre 4.

5.5.4 Résultats

Deux séries d'expérimentation ont été menées pour évaluer l'approche de résolution proposée : la première considère différents scénarios de congestions affectés sur les différents types de route ; la seconde concerne l'influence du facteur de ralentissement pour les différents types de route.

Une dernière série de mesures est effectuée sur l'ensemble du benchmark mais en fixant les paramètres de congestion et le facteur de ralentissement.

5.5.4.1 Expérimentations basées sur des scénarios de congestions

Quatre scénarios ont été élaborés en affectant les différentes distributions de congestion sur les trois types de routes existants. Ces scénarios sont montrés dans le tableau 5.2. Il est à noter que pour ces essais, le facteur de ralentissement α de ce scénario est 1,5 (premier scénario du tableau 4.3 dans le chapitre 4). Le numéro de distribution mentionné à l'intérieur du tableau 5.2 correspond au type de distribution référencée en figure 5.10. Par exemple, pour le scénario D2, la congestion se produit à la première période dans le profil historique de vitesse (début de journée) pour le premier type de route ; à la deuxième période dans le profil historique de vitesse (milieu de journée) pour le deuxième type de route et à la troisième période dans le profil historique de vitesse (fin de journée) pour le troisième type de route. Le nombre de congestions de ces quatre scénarios varie de 2 à 9.

Type de route	Scénarios			
	D1	D2	D3	D4
1	2	2	5	8
2	2	3	6	8
3	1	4	7	8
Nombre de congestions	2	3	6	9

TABLEAU 5.2 – Quatre scénarios d'affectation des congestions sur les 3 types de routes

Nous choisissons les neufs problèmes tests de la catégorie C1 du benchmark modifié par [Gendreau et al. 1999]. Pour chaque problème 5 exécutions (runs) ont été lancées et on a adopté la valeur moyenne. Les résultats obtenus pour le scénario D1 sont présentés dans le tableau 5.3 :

Mode de résolution		Résultats issus de la planification (étape a)				Résultats en rejouant la solution (étape b)			
	Problème	TT ^a	TA ^b	TR ^c	VO ^d	TT	TA	TR	VO
R1	C101	740,29	45,18	0,54	740,83	757,87	55,04	0,00	757,87
	C102	1229,30	74,83	1337,62	2566,92	817,47	396,44	143,81	961,28
	C103	1127,22	934,44	342,57	1469,79	1450,40	891,47	1086,80	2537,20
	C104	811,77	804,57	0,00	811,77	943,52	323,08	12,84	956,36
	C105	775,68	245,05	254,66	1030,34	751,41	300,65	264,51	1015,92
	C106	727,90	211,34	21,32	749,22	1122,95	404,43	600,72	1723,68
	C107	745,66	132,01	15,12	760,78	879,06	240,61	98,34	977,40
	C108	845,03	217,88	65,93	910,96	879,42	284,44	78,21	957,63
	C109	758,99	149,02	30,95	789,94	975,08	337,16	0,00	975,08
	Moyenne	862,43	312,70	229,86	1092,28	953,02	359,26	253,92	1206,94
	Ecart								10,50
R2	C101	718,48	50,43	0,00	718,48	1132,03	0,00	832,80	1964,82
	C102	847,46	346,30	18,05	865,51	1339,01	241,61	552,05	1891,07
	C103	897,93	958,00	0,00	897,93	1068,69	725,34	339,09	1407,79
	C104	738,90	422,56	14,98	753,88	1078,28	764,68	49,94	1128,22
	C105	723,11	260,31	246,85	969,96	1122,69	213,27	557,34	1680,03
	C106	708,27	218,56	0,72	708,99	1050,93	152,71	441,96	1492,90
	C107	718,36	225,22	36,31	754,66	1089,22	91,54	230,38	1319,60
	C108	759,33	134,31	0,00	759,33	1123,70	161,17	366,90	1490,60
	C109	727,42	137,50	20,85	748,27	1048,90	197,19	34,68	1083,58
	Moyenne	759,92	305,91	37,53	797,45	1117,05	283,06	378,35	1495,40
	Ecart								87,52
Gain(%)									19,29

^a Temps de trajet

^b Temps d'attente

^c Temps retard

^d Valeur d'objectif (= valeur de temps de trajet + valeur de temps retard)

TABLEAU 5.3 – Résultats du problème C1 pour le scénario D1

Nous remarquons dans le tableau 5.3 qu'en moyenne les résultats basés sur notre méthode intégrant la prévision des vitesses (1206,94) sont meilleurs que celles basées sur les profils historiques (1495,40). Le niveau d'amélioration (gain, calculé par l'équation 5.10) est de 19,29%. Ce résultat confirme que la planification intégrant la variation en temps réel d'information trafic s'approche davantage du contexte réel et résiste ainsi mieux aux aléas qui perturbent le réseau.

$$Gain = \frac{M_{R2b} - M_{R1b}}{M_{R1b}} \quad (5.10)$$

Où :

- M_{R1b} : Valeur moyenne d'étape b pour la résolution 1 ;
- M_{R2b} : Valeur moyenne d'étape b pour la résolution 2 ;

Les écarts de performance entre les résultats obtenus lors de la planification et les résultats obtenus en rejouant la planification sont plus petits pour le mode de résolution R1 (10,50%) que pour le mode de résolution R2 (87,52%).

De plus, nous pouvons observer à partir des résultats de planification (étape a) des problèmes de type C1 que la valeur moyenne de l'objectif de la résolution 1 (1092,28) est plus grande que celle de la résolution 2 (797,45). Ceci est dû au fait que la résolution 1 utilise des valeurs faibles de vitesse prévisionnelle basées sur l'influence des congestions lors de la planification des tournées. Par contre, la résolution 2 utilise toujours les valeurs nominales de vitesses historiques et ne considère donc pas l'influence de ces congestions.

Le tableau 5.4 montre les résultats moyens pour les neufs problèmes C1 et pour l'ensemble des quatre scénarios D1, D2, D3 et D4. Les résultats complets sont présentés dans l'annexe C.1.

	Scénario D1		Scénario D2		Scénario D3		Scénario D4	
	R1	R2	R1	R2	R1	R2	R1	R2
TT	953,02	1117,05	1301,08	1202,31	1536,17	3015,78	1781,37	1859,60
TA	359,26	283,06	348,87	505,62	312,16	172,01	564,70	614,69
TR	253,92	378,35	335,48	925,26	1505,23	2095,84	2967,51	3234,81
VO	1206,94	1495,40	1636,56	2127,57	3041,40	5111,62	4748,88	5094,41
Gain(%)	19,29		23,08		40,50		6,78	
STAT	3,24		5,52		13,62		2,27	

TABLEAU 5.4 – Résultats du problème C1 pour les 4 scénarios de distribution des congestions

Nous remarquons dans le tableau 5.4 que les résultats de la résolution 1 pour ces quatre scénarios sont toujours meilleurs que ceux de la résolution 2.

De plus, les niveaux d'amélioration augmentent de 19,29% à 40,50% pour les scénarios D1, D2 et D3. Ceci est justifié car le nombre de congestions augmente du scénario D1 au scénario D3 (voir la figure 5.10). Notre méthode de résolution basée sur un profil de planification intégrant des prévisions va chercher à éviter les routes sur lesquelles une congestion se produit. Par contre la méthode basée uniquement sur les profils historiques n'a pas connaissance des perturbations et va effectuer une planification des tournées sur des routes qui semblent a priori plus intéressantes. En réalité, les congestions qui vont apparaître sur ces tournées peuvent dégrader fortement les performances de celles-ci.

Dans certains cas le niveau d'amélioration pour le scénario D4 n'est pas très élevé (6,78%). En effet les congestions pour ce scénario se produisent à la fois sur les trois périodes de la journée et pour les trois types de route (voir la figure 5.10). Par conséquent, notre méthode de planification ne peut plus éviter les routes congestionnées et donc ne peut plus améliorer fortement les résultats.

Nous contrôlons en même temps un indice statistique STAT (Test d'Aspin-Welch de comparaison de moyennes) sur les résultats pour vérifier si les valeurs sont significatives ou simplement dues à des phénomènes de distribution aléatoire. Il est défini comme ci-dessous :

$$STAT = \frac{M_{R2b} - M_{R1b}}{\sqrt{\frac{E_{R2b}^2}{n_1} + \frac{E_{R1b}^2}{n_2}}} \quad (5.11)$$

Où :

- M_{R1b} : Valeur moyenne de l'étape b pour la résolution 1 ;
- M_{R2b} : Valeur moyenne de l'étape b pour la résolution 2 ;
- E_{R1b} : Ecart type étape b pour la résolution 1 ;
- E_{R2b} : Ecart type étape b pour la résolution 2 ;
- n_i : Nombre d'exécutions ;

Au seuil de 2% , les quatre valeurs d'indice statistique sont toujours supérieures à 2 (valeur du quantile d'ordre 2% de la loi de Gauss centrée réduite). Donc les résultats sont significatifs.

5.5.4.2 Expérimentations basées sur la variation du facteur de ralentissement

Ce type de comparaison est basé sur la variation du facteur de ralentissement des vitesses de chaque période comme le montre le tableau 5.5. Les facteurs de ralentissement α valent respectivement 1.5, 2 et 4. Nous choisissons le problème test C1 et le scénario D2 dans lequel les distributions de type 2, 3, 4 sont respectivement affectées aux types de route 1, 2, 3.

	Profils	T1	T2	T3
Scénario V1 $\alpha=1,5$	1	0,54	0,81	0,54
	2	0,81	1,22	0,81
	3	1,22	1,82	1,22
Scénario V2 $\alpha=2$	1	0,33	0,67	0,33
	2	0,67	1,33	0,67
	3	1,33	2,67	1,33
Scénario V3 $\alpha=4$	1	0,12	0,46	0,12
	2	0,46	1,92	0,46
	3	0,96	3,84	0,96

TABLEAU 5.5 – Profils de vitesse pour les scénarios V1 à V3

Les résultats globaux pour ces trois scénarios sont présentés dans le tableau 5.6. Les résultats complets sont dans l'annexe C.2.

	Scénario V1		Scénario V2		Scénario V3	
	R1	R2	R1	R2	R1	R2
TT	1301,08	1202,31	1239,63	1315,71	1658,40	1682,92
TA	348,87	505,62	447,49	601,08	531,39	712,79
TR	335,48	925,26	991,12	1654,38	1268,48	2495,05
VO	1636,56	2127,57	2230,75	2970,09	2926,88	4177,97
Gain(%)	23,08		24,89		29,94	
STAT	5,52		4,86		8,23	

TABLEAU 5.6 – Résultats du problème C1 pour les 3 scénarios de variation de vitesses de chaque période

Nous remarquons dans le tableau 5.6 que les résultats de la résolution 1 pour ces trois scénarios sont toujours meilleurs que ceux de la résolution 2. Les niveaux d'amélioration varient entre 23,08% et 29,94%.

De plus, ces niveaux d'amélioration augmentent pour les scénarios V1, V2 et V3. Ceci est justifié car les variations des différentes vitesses augmentent du scénario V1 au scénario V3. Notre méthode de planification tend à rejeter l'utilisation des routes à faible vitesse.

Un test statistique montre que le gain observé est également significatif au seuil de 2% .

5.5.4.3 Résultats sur l'ensemble des problèmes tests du benchmark

Enfin pour terminer l'évaluation de notre approche, nous donnons dans cette partie, les résultats globaux obtenus pour toutes les catégories de problèmes tests C1, R1, RC1, C2, R2 et RC2 en fixant le scénario de distribution à D2 (le nombre de congestions (3) suffisamment important) et le scénario de ralentissement à V1. Les résultats sont présentés dans le tableau 5.7. Les résultats obtenus ci-dessus confirment que les résultats de la résolution 1 pour toutes les catégories de problèmes de test sont toujours meilleurs que ceux de la résolution 2. Le niveau d'amélioration varie entre 5,48% et 23,08%. Les résultats complets sont présentés dans l'annexe C.3. Par contre pour les jeu R1 et RC1, le test statistique montre que le gain observé n'est pas vérifié au seuil de 2% .

	C1		R1		RC1	
	R1	R2	R1	R2	R1	R2
TT	1301,08	1202,31	1364,63	1367,85	1413,37	1433,21
TA	348,87	505,62	442,93	454,33	374,43	393,10
TR	335,48	925,26	744,31	912,51	644,37	743,73
VO	1636,56	2127,57	2108,94	2280,36	2057,74	2176,94
Gain(%)	23,08		7,52		5,48	
STAT	5,52		1,13		1,34	
	C2		R2		RC2	
	R1	R2	R1	R2	R1	R2
TT	866,53	866,44	1403,42	1510,08	1477,25	1470,73
TA	132,04	128,00	369,48	299,19	603,25	596,56
TR	229,33	497,45	1758,60	2232,45	594,55	896,95
VO	1095,86	1363,90	3162,02	3742,53	2071,80	2367,68
Gain(%)	19,65		15,51		12,50	
STAT	3,01		3,82		3,33	

TABLEAU 5.7 – Résultats des 6 catégories de problèmes pour les scénarios D2 et V1

5.5.4.4 Analyse des résultats selon le degré de dynamisme

Nous avons analysé l'influence de degré de dynamisme sur le gain obtenu par notre approche. Il apparaît qu'il n'y a pas de relation simple (i.e. régression linéaire, ...) entre le degré de dynamisme (dod) et le niveau d'amélioration (gain). Le degré de dynamisme n'est pas le seul paramètre qui influe sur le gain. D'autres paramètres importants sont également les caractéristiques de fenêtre de temps (i.e. large ou étroite).

5.6 Conclusions

Dans ce chapitre nous avons présenté une méthode de planification basée sur un modèle de prévision pour résoudre le problème DVRPTW temps réel avec temps de trajet dépendant du temps. Nous avons présenté un modèle de lissage exponentiel et étendu l'algorithme de calcul des temps de trajet afin d'intégrer l'information historique (profil historique) et l'information de trafic en temps réel (remontée des vitesses réelles) pour obtenir les résultats qui se rapprochent d'un contexte réel.

Nous avons également développé l'environnement de simulation afin de traiter les nouveaux événements qui se produisent dans ce contexte temps réel.

Nous avons effectué deux types d'expérimentations pour valider notre méthode de résolution qui intègre un modèle de temps de trajet dépendant du temps en contexte temps réel : la première considère différents scénarios d'affectation des distributions de congestion sur les différents types de routes ; la seconde porte sur des scénarios dont le facteur de ralentissement varie. Les analyses des résultats de ces expérimentations montrent que notre méthode temps réel avec modèle de prévision peut fournir une bonne performance par rapport à un simple modèle avec temps de trajet dépendant du temps.

Conclusions générales et perspectives

L'objectif du travail de cette thèse était d'étudier la possibilité de traiter le problème de la gestion dynamique des tournées de véhicules en considérant que les temps de trajet sont variables et dépendants du temps.

Pour atteindre ce but, nous avons utilisé un profil de vitesse variable modélisé sous la forme d'une fonction constante par morceau. Un algorithme de calcul de temps de trajet [Ichoua et al. 2003] a été adapté pour tenir compte de ce profil. On s'est focalisé sur deux types de problème DVRP : Problème dynamique de la tournée du réparateur (DTRP) et le problème dynamique des tournées de véhicules avec fenêtres de temps (DVRPTW).

Pour le problème dynamique de la tournée du réparateur (DTRP), Nous avons modifié les stratégies classiques NN et Part pour les adapter aux nouveaux objectifs qui consistent à minimiser le temps de trajet total de véhicule et minimiser le temps d'attente des clients.. Nous avons créé 3 profils différents qui correspondent à 3 types différents de routes dont les vitesses moyennes varient de lent à rapide. Pour ce type de problème on a considéré que la journée de service était composée de 3 périodes de durée identique. Les première et troisième périodes représentent les heures de pointe du matin et du soir. La deuxième période correspond au milieu de journée, quand la densité du trafic est moindre. Les vitesses de trajet dans les heures de pointe correspondent aux vitesses de trajet en milieu de journée divisée par un facteur de ralentissement α . Nous avons élaboré 3 types de scénarios pour faire varier le facteur de ralentissement α . Dans les scénarios 1, 2 et 3, α vaut respectivement 1,5, 2 et 4. La vitesse moyenne dans chaque scénario est supposée égale à 40 km/h comme dans le cas à vitesse constante.

Pour comparer la qualité des solutions fournies par notre approche, nous avons cherché à comparer la robustesse des solutions fournies par les deux modes de résolution suivants [Ichoua et al. 2003] :

- Résolution1 : la résolution prend en compte un environnement avec un temps de trajet variable selon un profil de vitesse déterminé, soit S la solution obtenue. Cette solution fournit les indicateurs (temps de parcours et temps d'attente moyens) R1 ;
- Résolution2 : la résolution ne prend pas en compte la variabilité de la vitesse. L'évaluation de la qualité de la solution doit alors se faire en 2 étapes. Etape a. Le problème est résolu en supposant que le temps de parcours est constant : soit S' la solution obtenue ; Etape b. Le parcours S' , ainsi obtenu, est évalué par simulation en le « replongeant » dans un environnement temps de trajet variable respectant les profils de vitesse précédents. Cette solution fournit les indicateurs R2.

Nous avons utilisé le benchmark proposé par [Larsen 2001] pour examiner le comportement de notre approche pour la résolution de ce type de problème. La comparaison des indicateurs R1 et R2 obtenus par ces 2 modes de solution permet d'évaluer l'apport de la prise en compte d'un modèle temps de trajet variable ou la dégradation observée sur la résolution 2.

Pour évaluer plus finement cet apport, nous utilisons de plus un indice de « Gain » pour décrire le niveau d'amélioration des résultats. Les résultats obtenus montrent donc que le modèle à vitesse variable basé sur des profils historiques présente un intérêt légitime pour prendre en compte les fluctuations de la vitesse d'un véhicule. La prise en compte de cette connaissance est donc significative, les solutions obtenues sont plus robustes si on les compare avec celles fournies avec un modèle à vitesse constante. Le gain est d'autant plus important que le degré de dynamisme est élevé.

Pour le problème dynamique des tournées de véhicules avec fenêtres de temps (DVRPTW), nous adoptons une approche évolutionniste qui utilise l'algorithme génétique comme méthode de résolution métaheuristique. Cette approche est développée pour intégrer le modèle de temps de trajet dépendant du temps ; des extensions de cet algorithme ont été proposées pour qu'il puisse traiter les deux types d'événements issus de ce problème dynamique : L'évènement « **arrivée d'un nouveau client** » et l'évènement « **fin de service d'un client** ». Comme pour le problème dynamique de la tournée du réparateur, trois profils différents de routes et trois périodes de temps de même durée ont été considérés. Les trois types de profils sont répartis de façon aléatoire sur l'ensemble des liaisons inter clients. La vitesse moyenne des véhicules pour chaque scénario (9 valeurs de vitesse) est égale à 1.0. Nous avons utilisé aussi les deux modes de résolution présentés ci-dessus et le jeu de problèmes fourni par Solomon [Solomon 1987] pour évaluer notre méthode. Ce benchmark de Solomon a été modifié par [Gendreau et al. 1999] pour l'adapter au problème VRPTW dynamique. D'après les résultats obtenus, nous remarquons que le modèle à vitesse variable donne des résultats meilleurs que le modèle à vitesse constante lorsque l'on *replonge* les solutions de celui-ci dans un environnement à vitesse variable. Ce qui montre bien l'intérêt de planifier des tournées avec un modèle de calcul de temps de trajet prévisionnel plus réaliste.

En raison de la variabilité des conditions de trafic routier qui influencent fortement les performances des tournées planifiées, nous avons étendu les travaux précédents pour y intégrer une prise en compte des changements de trafic en temps réel. Pour cela, nous proposons un modèle de prévision pour planifier les tournées qui utilise cette remontée d'information conjointement avec les profils historiques pour construire un nouveau profil de planification. Nous avons étendu la plateforme de simulation pour prendre en compte ces nouveaux aspects : l'évènement « **vitesse mesurée** », l'évènement « **arrivée effective chez un client** ». Les deux modes de résolution présentés ci-dessus ont développés pour évaluer l'approche :

- Résolution 1 : a. Le problème est résolu en utilisant le « profil à planifier » : soit S la solution obtenue ; b. Le parcours S , ainsi obtenu, est évalué par simulation en utilisant le profil réel de vitesse. Cette solution fournit les indicateurs R1 ;
- Résolution 2 : a. Le problème est résolu en utilisant toujours le profil historique de vitesse : soit S' la solution obtenue ; b. Le parcours S' , ainsi obtenu, est évalué par simulation en utilisant le profil réel de vitesse. Cette solution fournit les indicateurs R2.

Nous avons réalisé deux types d'expériences pour analyser les résultats : différentes distributions de congestion et différents facteurs de ralentissement ont été considérés sur les différentes périodes de la journée. Le benchmark de Salomon modifié a été aussi utilisé. Les analyses des résultats montrent que la méthode de prévision proposée peut fournir de bonnes performances par rapport aux méthodes avec temps de trajet dépendant du temps mais sans aspect temps réel.

Les perspectives que nous pouvons donner à ces travaux sont de plusieurs ordres. Sur le plan des applications d'autres problèmes peuvent être facilement adaptés à notre outil de planification

comme le problème du bus à la demande, le problème DVRPTW avec une flotte hétérogène de véhicules ou le routage de véhicules d'urgences (i.e. ambulance).

Enfin un modèle de prévision plus sophistiqué peut être envisagé pour améliorer le taux d'erreur sur les prévisions de vitesse. Celui-ci pourrait considérer un effet tendanciel du trafic routier et donner des vitesses prévisionnelles sur un horizon de prévision de plusieurs périodes (2 ou 3 périodes). Si la congestion dure dans le temps, les prévisions obtenues seraient de meilleure qualité. Le profil de planification qui en découlerait, serait ainsi moins influencé par le profil historique disponible.

Annexes

Annexe A

Résultats des problèmes de PDTRP avec temps de trajet dépendant du temps

A.1 Exemple : Le benchmark utilisé

TABLEAU A.1: Le benchmark de 50 clients et 30% de *dod*

No.client	Heure de la demande	Coordonnée X	Coordonnée Y	Temps de service
1	08 :00 :00	6824.70	1123.23	00 :04 :07
2	08 :00 :00	9414.11	6258.86	00 :03 :51
3	08 :00 :00	1908.70	9389.91	00 :01 :27
4	08 :00 :00	9864.03	8104.92	00 :03 :32
5	08 :00 :00	3.83	3453.09	00 :02 :57
6	08 :00 :00	7018.33	3777.35	00 :00 :24
7	08 :00 :00	9398.16	5650.27	00 :02 :42
8	08 :00 :00	8752.95	3797.16	00 :03 :59
9	08 :00 :00	38.61	9570.85	00 :04 :22
10	08 :00 :00	6341.75	6797.05	00 :05 :42
11	08 :00 :00	6315.96	1563.85	00 :05 :14
12	08 :00 :00	7032.52	3986.66	00 :04 :34
13	08 :00 :00	1433.00	1129.27	00 :01 :38
14	08 :00 :00	1315.99	368.41	00 :01 :37
15	08 :00 :00	1950.96	6603.18	00 :02 :45
16	08 :00 :00	8116.84	8161.92	00 :03 :12
17	08 :00 :00	8267.79	9158.15	00 :02 :59
18	08 :00 :00	325.24	7787.72	00 :03 :49
19	08 :00 :00	502.47	3814.40	00 :02 :54
20	08 :00 :00	3839.44	5715.85	00 :06 :00
21	08 :00 :00	9542.69	893.32	00 :03 :03
22	08 :00 :00	5604.31	6502.87	00 :00 :48
23	08 :00 :00	9859.01	5143.94	00 :01 :15
24	08 :00 :00	7.66	8420.40	00 :02 :07

A suivre ⇒

no.client	Heure de la demande	coordonnée X	coordonnée Y	temps de service
25	08 :00 :00	4096.79	2634.17	00 :07 :32
26	08 :00 :00	8435.73	8221.37	00 :01 :34
27	08 :00 :00	9427.92	7422.55	00 :01 :35
28	08 :00 :00	3791.14	794.88	00 :02 :32
29	08 :00 :00	2177.47	1012.40	00 :01 :02
30	08 :00 :00	340.98	3728.37	00 :01 :48
31	08 :00 :00	336.24	7639.45	00 :00 :54
32	08 :00 :00	9389.76	5125.76	00 :01 :19
33	08 :00 :00	3228.26	7554.79	00 :01 :59
34	08 :00 :00	2961.41	5422.74	00 :02 :05
35	08 :00 :00	3823.08	8639.48	00 :00 :51
36	08 :07 :00	926.30	395.81	00 :00 :47
37	08 :18 :46	3922.89	1824.00	00 :01 :42
38	08 :35 :10	6143.85	8876.05	00 :01 :27
39	09 :22 :22	6880.30	7316.56	00 :01 :50
40	09 :25 :04	1687.31	840.95	00 :00 :49
41	09 :27 :16	2352.63	9910.23	00 :02 :04
42	09 :41 :52	1282.72	1195.22	00 :01 :54
43	10 :24 :48	5961.44	1460.95	00 :05 :54
44	11 :06 :49	8804.04	4865.79	00 :03 :02
45	11 :15 :07	7886.24	6807.97	00 :03 :26
46	11 :18 :31	1259.57	5322.09	00 :02 :13
47	11 :36 :37	6741.70	219.67	00 :02 :29
48	12 :15 :17	9383.13	3991.29	00 :00 :50
49	12 :24 :47	2842.75	7698.68	00 :00 :30
50	13 :08 :48	2295.45	6912.30	00 :01 :45

A.2 Résultats complets

TABLEAU A.2: Jeu de 100 fichiers de 30 clients pour la stratégie NN

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
10%	S1	TT-R1	18721,37	5812,26	0,04
		TT-R2	18728,00	5809,28	
		TA-R1	2108,63	209,93	0,07
		TA-R2	2110,04	209,79	
	S2	TT-R1	18686,87	5765,37	0,17
		TT-R2	18718,78	5729,96	
		TA-R1	2114,45	199,97	0,04
		TA-R2	2115,27	200,92	
	S3	TT-R1	18944,23	5538,32	-0,06
		TT-R2	18932,79	5552,11	
		TA-R1	2489,35	207,11	0,08
		TA-R2	2491,37	205,84	

A suivre ⇒

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
20%	Moyenne	TT-R1	18784,16	5705,32	0,05
		TT-R2	18793,19	5697,11	
		TA-R1	2237,48	205,67	0,06
		TA-R2	2238,89	205,52	
	S1	TT-R1	21674,46	5195,41	0,02
		TT-R2	21678,15	5200,47	
		TA-R1	1823,17	219,97	0,24
		TA-R2	1827,54	222,42	
	S2	TT-R1	21652,33	5209,11	0,05
		TT-R2	21663,63	5205,68	
		TA-R1	1874,42	236,01	0,24
		TA-R2	1878,86	236,50	
	S3	TT-R1	21945,46	5296,02	0,19
		TT-R2	21986,22	5491,54	
		TA-R1	2240,61	248,70	0,73
		TA-R2	2256,93	257,54	
	Moyenne	TT-R1	21757,42	5233,51	0,09
		TT-R2	21776,00	5299,23	
		TA-R1	1979,40	234,89	0,42
		TA-R2	1987,78	238,82	
30%	S1	TT-R1	22083,06	4741,38	0,05
		TT-R2	22093,10	4750,84	
		TA-R1	1558,43	233,43	0,11
		TA-R2	1560,13	233,78	
	S2	TT-R1	22173,17	4836,12	0,05
		TT-R2	22184,24	4841,74	
		TA-R1	1595,91	214,00	0,19
		TA-R2	1598,94	217,70	
	S3	TT-R1	23073,36	5453,61	1,06
		TT-R2	23317,98	5940,45	
		TA-R1	1967,27	237,11	2,26
		TA-R2	2011,68	252,61	
	Moyenne	TT-R1	22443,20	5010,37	0,39
		TT-R2	22531,77	5177,68	
		TA-R1	1707,20	228,18	0,96
		TA-R2	1723,58	234,70	
40%	S1	TT-R1	23903,12	4026,24	0,09
		TT-R2	23924,13	4041,57	
		TA-R1	1295,43	187,75	0,37
		TA-R2	1300,27	187,51	
	S2	TT-R1	23981,88	4097,48	0,06
		TT-R2	23995,78	4126,06	
		TA-R1	1335,20	180,08	0,47
		TA-R2	1341,44	179,96	
	S3	TT-R1	25041,03	4582,66	3,37
		TT-R2	25885,94	5738,49	

A suivre \Rightarrow

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
	Moyenne	TA-R1	1709,79	216,09	5,86
		TA-R2	1809,96	263,94	
		TT-R1	24308,68	4235,46	1,21
		TT-R2	24601,95	4635,37	
		TA-R1	1446,80	194,64	2,56
		TA-R2	1483,89	210,47	
50%	S1	TT-R1	23409,87	3993,15	0,14
		TT-R2	23443,09	4045,97	
		TA-R1	1059,26	167,16	0,30
		TA-R2	1062,42	166,90	
	S2	TT-R1	23588,63	4205,28	0,80
		TT-R2	23777,97	4388,95	
		TA-R1	1106,40	168,51	1,12
		TA-R2	1118,79	170,06	
	S3	TT-R1	24698,54	4827,02	7,65
		TT-R2	26588,35	7135,40	
		TA-R1	1446,69	201,57	11,73
		TA-R2	1616,39	306,60	
	Moyenne	TT-R1	23899,01	4341,82	2,95
		TT-R2	24603,14	5190,11	
		TA-R1	1204,12	179,08	5,13
		TA-R2	1265,86	214,52	
60%	S1	TT-R1	24890,66	3589,78	0,24
		TT-R2	24949,51	3623,53	
		TA-R1	857,16	139,81	1,12
		TA-R2	866,73	143,96	
	S2	TT-R1	25143,99	3663,64	0,73
		TT-R2	25326,69	3799,49	
		TA-R1	917,31	141,35	2,32
		TA-R2	938,63	147,88	
	S3	TT-R1	26842,65	4288,94	6,97
		TT-R2	28713,65	6031,78	
		TA-R1	1261,79	173,77	16,85
		TA-R2	1474,38	249,06	
	Moyenne	TT-R1	25625,77	3847,45	2,75
		TT-R2	26329,95	4484,93	
		TA-R1	1012,09	151,64	8,02
		TA-R2	1093,25	180,30	
70%	S1	TT-R1	25131,14	3392,88	0,23
		TT-R2	25188,85	3447,16	
		TA-R1	704,38	134,19	1,12
		TA-R2	712,30	134,81	
	S2	TT-R1	25466,32	3511,68	0,71
		TT-R2	25646,58	3742,39	
		TA-R1	768,22	137,08	2,82
		TA-R2	789,90	140,66	

A suivre ⇒

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
	S3	TT-R1	27258,44	4129,66	10,15
		TT-R2	30026,47	6365,17	
		TA-R1	1080,32	188,78	28,65
		TA-R2	1389,82	280,08	
	Moyenne	TT-R1	25951,97	3678,08	3,86
		TT-R2	26953,97	4518,24	
		TA-R1	850,97	153,35	13,28
		TA-R2	964,01	185,18	
80%	S1	TT-R1	25700,92	3078,45	0,37
		TT-R2	25796,28	3207,26	
		TA-R1	579,94	117,85	2,13
		TA-R2	592,28	126,23	
	S2	TT-R1	26042,24	3100,48	1,56
		TT-R2	26447,44	3623,40	
		TA-R1	634,94	123,71	8,05
		TA-R2	686,03	151,43	
	S3	TT-R1	27968,43	3573,81	12,29
		TT-R2	31406,81	6488,25	
		TA-R1	957,83	200,28	43,44
		TA-R2	1373,86	408,96	
	Moyenne	TT-R1	26570,53	3250,91	4,94
		TT-R2	27883,51	4439,64	
		TA-R1	724,24	147,28	22,07
		TA-R2	884,05	228,87	
90%	S1	TT-R1	26004,11	2610,27	0,40
		TT-R2	26108,91	2692,81	
		TA-R1	494,75	100,33	3,73
		TA-R2	513,19	106,82	
	S2	TT-R1	26455,54	2713,88	1,58
		TT-R2	26874,67	3092,19	
		TA-R1	590,09	126,70	10,47
		TA-R2	651,90	166,30	
	S3	TT-R1	28373,97	2819,68	14,67
		TT-R2	32537,28	6039,80	
		TA-R1	973,85	254,13	59,57
		TA-R2	1553,95	533,59	
	Moyenne	TT-R1	26944,54	2714,61	5,80
		TT-R2	28506,95	3941,60	
		TA-R1	686,23	160,39	32,08
		TA-R2	906,35	268,90	
	S1	TT-R1	26506,00	2512,92	0,20
		TT-R2	26559,47	2654,17	
		TA-R1	471,45	107,81	3,08
		TA-R2	485,98	117,19	
	S2	TT-R1	26895,88	2537,53	1,80
		TT-R2	27378,84	3073,02	

A suivre \Rightarrow

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
100%		TA-R1	555,99	131,60	9,77
		TA-R2	610,31	157,20	
	S3	TT-R1	28621,46	3044,27	16,70
		TT-R2	33401,05	6278,39	
		TA-R1	940,00	251,91	59,42
		TA-R2	1498,52	520,85	
	Moyenne	TT-R1	27341,11	2698,24	6,48
		TT-R2	29113,12	4001,86	
		TA-R1	655,81	163,77	31,89
		TA-R2	864,94	265,08	

TT-R1 : Temps de trajet pour la résolution 1 ;

TT-R2 : Temps de trajet pour la résolution 2 ;

TA-R1 : Temps d'attente pour la résolution 1 ;

TA-R2 : Temps d'attente pour la résolution 2.

TABEAU A.3: Jeu de 100 fichiers de 30 clients pour la stratégie PartNN2

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
10%	S1	TT-R1	19548,93	4877,51	-0,16
		TT-R2	19517,53	4903,20	
		TA-R1	2821,08	288,82	-0,07
		TA-R2	2819,21	287,11	
	S2	TT-R1	19374,80	4993,51	0,58
		TT-R2	19486,30	4897,20	
		TA-R1	2857,85	291,62	0,54
		TA-R2	2873,36	294,29	
	S3	TT-R1	19866,20	4659,89	0,01
		TT-R2	19867,48	4616,87	
		TA-R1	3403,17	415,60	-0,11
		TA-R2	3399,36	414,95	
	Moyenne	TT-R1	19596,64	4843,63	0,14
		TT-R2	19623,77	4805,76	
		TA-R1	3027,37	332,01	0,11
		TA-R2	3030,64	332,12	
20%	S1	TT-R1	21917,56	4833,78	0,01
		TT-R2	21919,95	4820,94	
		TA-R1	2428,28	324,77	0,02
		TA-R2	2428,80	323,98	
	S2	TT-R1	21933,46	4804,95	-0,12
		TT-R2	21906,40	4830,94	
		TA-R1	2479,39	335,31	0,66
		TA-R2	2495,77	334,29	
	S3	TT-R1	22251,14	4851,25	0,48
		TT-R2	22358,70	5056,81	
		TA-R1	3002,86	381,75	1,35
		TA-R2	3043,47	410,12	

A suivre ⇒

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
30%	Moyenne	TT-R1	22034,05	4829,99	0,13
		TT-R2	22061,68	4902,90	
		TA-R1	2636,84	347,28	0,73
		TA-R2	2656,02	356,13	
	S1	TT-R1	22268,02	4362,72	0,06
		TT-R2	22281,88	4387,02	
		TA-R1	2066,58	347,65	0,35
		TA-R2	2073,85	346,14	
	S2	TT-R1	22351,15	4495,38	0,22
		TT-R2	22401,16	4449,43	
		TA-R1	2129,08	324,29	1,21
		TA-R2	2154,82	352,97	
	S3	TT-R1	23381,42	5074,20	1,50
		TT-R2	23731,08	5513,13	
		TA-R1	2636,48	430,44	2,05
		TA-R2	2690,64	442,91	
	Moyenne	TT-R1	22666,86	4644,10	0,61
		TT-R2	22804,71	4783,20	
		TA-R1	2277,38	367,46	1,28
		TA-R2	2306,44	380,67	
40%	S1	TT-R1	23954,03	3919,07	0,17
		TT-R2	23994,89	3894,08	
		TA-R1	1761,44	287,56	0,57
		TA-R2	1771,52	298,37	
	S2	TT-R1	24052,11	3989,91	0,28
		TT-R2	24118,91	3914,65	
		TA-R1	1833,52	288,40	1,10
		TA-R2	1853,77	312,59	
	S3	TT-R1	25373,46	4570,06	2,52
		TT-R2	26012,30	5555,20	
		TA-R1	2362,62	360,13	3,22
		TA-R2	2438,68	362,36	
	Moyenne	TT-R1	24459,87	4159,68	1,02
		TT-R2	24708,70	4454,64	
		TA-R1	1985,86	312,03	1,79
		TA-R2	2021,32	324,44	
50%	S1	TT-R1	23440,46	3966,85	0,13
		TT-R2	23469,96	4019,96	
		TA-R1	1377,73	272,44	1,49
		TA-R2	1398,23	278,02	
	S2	TT-R1	23684,53	4220,63	0,54
		TT-R2	23813,61	4346,77	
		TA-R1	1459,73	255,90	2,42
		TA-R2	1494,98	284,70	
	S3	TT-R1	25036,53	4949,94	7,25
		TT-R2	26851,42	7077,81	

A suivre \Rightarrow

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
	Moyenne	TA-R1	1943,62	318,49	9,21
		TA-R2	2122,58	424,61	
		TT-R1	24053,84	4379,14	2,73
		TT-R2	24711,66	5148,18	
		TA-R1	1593,69	282,28	4,91
		TA-R2	1671,93	329,11	
60%	S1	TT-R1	24953,03	3467,86	0,19
		TT-R2	25001,59	3554,26	
		TA-R1	1122,78	260,23	2,36
		TA-R2	1149,28	271,27	
	S2	TT-R1	25244,44	3634,77	0,76
		TT-R2	25436,36	3610,50	
		TA-R1	1203,09	246,27	3,64
		TA-R2	1246,91	267,54	
	S3	TT-R1	27603,91	4324,56	5,32
		TT-R2	29071,11	5733,53	
		TA-R1	1731,78	348,04	12,08
		TA-R2	1940,94	378,47	
	Moyenne	TT-R1	25933,79	3809,06	2,19
		TT-R2	26503,02	4299,43	
		TA-R1	1352,55	284,85	6,89
		TA-R2	1445,71	305,76	
70%	S1	TT-R1	25148,08	3417,68	0,25
		TT-R2	25210,13	3478,30	
		TA-R1	877,45	208,62	3,21
		TA-R2	905,58	226,44	
	S2	TT-R1	25521,76	3544,98	0,73
		TT-R2	25707,20	3771,50	
		TA-R1	966,62	215,52	4,63
		TA-R2	1011,39	219,38	
	S3	TT-R1	27722,43	4208,90	9,56
		TT-R2	30373,56	6315,85	
		TA-R1	1429,36	306,97	22,68
		TA-R2	1753,56	366,62	
	Moyenne	TT-R1	26130,76	3723,85	3,70
		TT-R2	27096,96	4521,88	
		TA-R1	1091,15	243,70	12,13
		TA-R2	1223,51	270,81	
80%	S1	TT-R1	25738,89	3098,06	0,51
		TT-R2	25869,23	3193,47	
		TA-R1	698,79	184,16	8,97
		TA-R2	761,47	218,53	
	S2	TT-R1	26168,40	3203,26	1,66
		TT-R2	26603,75	3582,87	
		TA-R1	821,96	230,08	14,32
		TA-R2	939,69	270,87	

A suivre ⇒

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
	S3	TT-R1	28453,11	3579,47	14,15
		TT-R2	32478,99	6341,11	
		TA-R1	1284,47	350,67	47,22
		TA-R2	1890,94	608,22	
	Moyenne	TT-R1	26786,80	3293,59	5,71
		TT-R2	28317,32	4372,49	
		TA-R1	935,07	254,97	28,05
		TA-R2	1197,37	365,87	
90%	S1	TT-R1	26123,61	2593,91	0,62
		TT-R2	26285,45	2653,37	
		TA-R1	559,65	135,89	6,92
		TA-R2	598,36	172,87	
	S2	TT-R1	26685,46	2638,89	1,66
		TT-R2	27128,72	3102,66	
		TA-R1	685,61	186,17	13,10
		TA-R2	775,41	242,33	
	S3	TT-R1	29146,50	3013,22	15,39
		TT-R2	33633,44	5890,64	
		TA-R1	1172,38	332,51	55,73
		TA-R2	1825,79	628,15	
	Moyenne	TT-R1	27318,52	2748,67	6,21
		TT-R2	29015,87	3882,22	
		TA-R1	805,88	218,19	32,34
		TA-R2	1066,52	347,78	
100%	S1	TT-R1	26666,93	2474,40	0,64
		TT-R2	26838,65	2597,74	
		TA-R1	522,11	139,75	11,72
		TA-R2	583,28	197,57	
	S2	TT-R1	27235,88	2401,83	2,02
		TT-R2	27786,10	2851,44	
		TA-R1	616,41	166,23	18,58
		TA-R2	730,95	230,08	
	S3	TT-R1	29430,24	2641,04	16,81
		TT-R2	34377,69	5524,89	
		TA-R1	1059,24	320,67	60,48
		TA-R2	1699,87	545,61	
	Moyenne	TT-R1	27777,68	2505,75	6,80
		TT-R2	29667,48	3658,02	
		TA-R1	732,59	208,89	37,14
		TA-R2	1004,70	324,42	

TABLEAU A.4: Jeu de 100 fichiers de 50 clients pour la stratégie NN

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
	S1	TT-R1	21883,02	4571,52	0,01
		TT-R2	21885,47	4541,48	

A suivre ⇒

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
10%		TA-R1	5307,82	504,15	0,11
		TA-R2	5313,67	506,92	
	S2	TT-R1	21773,50	4624,70	0,35
		TT-R2	21849,20	4519,28	
		TA-R1	5315,54	491,12	0,59
		TA-R2	5346,65	505,81	
	S3	TT-R1	21965,66	4444,88	0,11
		TT-R2	21990,36	4361,60	
		TA-R1	5913,24	462,82	0,30
		TA-R2	5931,16	460,94	
	Moyenne	TT-R1	21874,06	4547,03	0,16
		TT-R2	21908,34	4474,12	
		TA-R1	5512,20	486,03	0,33
		TA-R2	5530,49	491,22	
20%	S1	TT-R1	23377,80	4164,30	0,05
		TT-R2	23388,37	4144,34	
		TA-R1	4714,10	458,20	0,26
		TA-R2	4726,12	458,18	
	S2	TT-R1	23297,84	4214,22	0,34
		TT-R2	23377,53	4081,89	
		TA-R1	4737,10	442,49	0,80
		TA-R2	4774,94	456,04	
	S3	TT-R1	23455,22	4199,21	0,37
		TT-R2	23541,36	4079,11	
		TA-R1	5363,20	401,88	0,49
		TA-R2	5389,70	391,54	
	Moyenne	TT-R1	23376,95	4192,58	0,25
		TT-R2	23435,75	4101,78	
		TA-R1	4938,13	434,19	0,52
		TA-R2	4963,59	435,25	
30%	S1	TT-R1	24285,86	3859,98	-0,05
		TT-R2	24273,55	3852,22	
		TA-R1	4062,58	463,80	0,21
		TA-R2	4071,30	467,21	
	S2	TT-R1	24211,76	3897,59	0,20
		TT-R2	24259,12	3786,86	
		TA-R1	4074,46	414,55	1,57
		TA-R2	4138,30	432,10	
	S3	TT-R1	24325,62	3986,94	0,37
		TT-R2	24416,78	3854,19	
		TA-R1	4691,80	428,79	1,57
		TA-R2	4765,47	445,43	
	Moyenne	TT-R1	24274,41	3914,84	0,17
		TT-R2	24316,48	3831,09	
		TA-R1	4276,28	435,71	

A suivre ⇒

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
		TA-R2	4325,02	448,25	1,14
40%	S1	TT-R1	25082,56	3112,96	
		TT-R2	25086,59	3087,38	0,02
		TA-R1	3335,04	442,46	
		TA-R2	3348,24	445,53	0,40
	S2	TT-R1	24984,34	3173,52	
		TT-R2	25024,59	3102,54	0,16
		TA-R1	3339,16	450,12	
		TA-R2	3402,00	485,00	1,88
	S3	TT-R1	25199,07	3286,02	
		TT-R2	25274,44	3159,49	0,30
		TA-R1	3907,27	438,97	
		TA-R2	3978,12	472,58	1,81
	Moyenne	TT-R1	25088,66	3190,83	
		TT-R2	25128,54	3116,47	0,16
		TA-R1	3527,16	443,85	
		TA-R2	3576,12	467,70	1,39
50%	S1	TT-R1	25086,97	3303,17	
		TT-R2	25130,70	3244,46	0,17
		TA-R1	2819,35	435,85	
		TA-R2	2834,57	426,85	0,54
	S2	TT-R1	25008,56	3362,94	
		TT-R2	25053,67	3266,03	0,18
		TA-R1	2818,37	441,99	
		TA-R2	2915,75	489,88	3,46
	S3	TT-R1	25222,22	3433,73	
		TT-R2	25363,91	3251,18	0,56
		TA-R1	3364,67	476,12	
		TA-R2	3429,74	498,25	1,93
	Moyenne	TT-R1	25105,92	3366,61	
		TT-R2	25182,76	3253,89	0,31
		TA-R1	3000,80	451,32	
		TA-R2	3060,02	471,66	1,97
60%	S1	TT-R1	25619,92	2792,53	
		TT-R2	25644,78	2732,37	0,10
		TA-R1	2174,42	457,29	
		TA-R2	2194,63	461,47	0,93
	S2	TT-R1	25509,76	2843,48	
		TT-R2	25573,16	2722,74	0,25
		TA-R1	2149,30	451,17	
		TA-R2	2235,85	486,83	4,03
	S3	TT-R1	25868,88	2891,75	
		TT-R2	25983,12	2728,99	0,44
		TA-R1	2636,82	484,61	
		TA-R2	2728,68	501,07	3,48
		TT-R1	25666,19	2842,59	

A suivre \Rightarrow

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
70%	Moyenne	TT-R2	25733,69	2728,03	0,26
		TA-R1	2320,18	464,36	
		TA-R2	2386,39	483,13	2,85
	S1	TT-R1	26045,72	2281,04	
		TT-R2	26075,20	2210,63	0,11
		TA-R1	1729,79	347,66	
		TA-R2	1765,90	350,97	2,09
	S2	TT-R1	25948,69	2305,01	
		TT-R2	26008,18	2192,23	0,23
		TA-R1	1701,38	324,51	
		TA-R2	1805,92	388,21	6,14
	S3	TT-R1	26351,65	2332,32	
		TT-R2	26597,34	2248,16	0,93
		TA-R1	2177,09	384,28	
		TA-R2	2359,43	437,48	8,38
	Moyenne	TT-R1	26115,35	2306,12	
		TT-R2	26226,91	2217,01	0,43
		TA-R1	1869,42	352,15	
		TA-R2	1977,08	392,22	5,76
80%	S1	TT-R1	26774,95	2209,54	
		TT-R2	26779,08	2093,67	0,02
		TA-R1	1343,79	256,41	
		TA-R2	1383,21	268,63	2,93
	S2	TT-R1	26639,88	2284,55	
		TT-R2	26728,89	2101,55	0,33
		TA-R1	1317,86	235,58	
		TA-R2	1434,30	289,14	8,84
	S3	TT-R1	27096,34	2303,94	
		TT-R2	27670,68	2880,69	2,12
		TA-R1	1730,88	314,23	
		TA-R2	2066,31	639,62	19,38
	Moyenne	TT-R1	26837,06	2266,01	
		TT-R2	27059,55	2358,64	0,83
		TA-R1	1464,18	268,74	
		TA-R2	1627,94	399,13	11,18
90%	S1	TT-R1	27143,21	1968,44	
		TT-R2	27119,69	1831,47	-0,09
		TA-R1	1153,03	245,35	
		TA-R2	1210,67	278,69	5,00
	S2	TT-R1	26998,40	1975,69	
		TT-R2	27041,32	1786,65	0,16
		TA-R1	1144,19	217,04	
		TA-R2	1284,52	288,63	12,27
	S3	TT-R1	27513,00	1923,57	
		TT-R2	28359,02	2559,75	3,07
		TA-R1	1526,79	277,47	

A suivre ⇒

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
	Moyenne	TA-R2	2040,97	650,03	33,68
		TT-R1	27218,20	1955,90	1,06
		TT-R2	27506,68	2059,29	
		TA-R1	1274,67	246,62	18,62
		TA-R2	1512,05	405,78	
100%	S1	TT-R1	27187,69	2062,26	0,28
		TT-R2	27263,63	1862,98	
		TA-R1	1102,48	268,54	10,30
		TA-R2	1216,03	328,12	
	S2	TT-R1	27053,27	2135,44	1,31
		TT-R2	27407,91	1807,88	
		TA-R1	1115,62	252,29	23,52
		TA-R2	1377,97	383,88	
	S3	TT-R1	27549,99	2122,22	6,77
		TT-R2	29414,44	3048,66	
		TA-R1	1703,61	399,89	51,71
		TA-R2	2584,60	804,17	
	Moyenne	TT-R1	27263,65	2106,64	2,81
		TT-R2	28028,66	2239,84	
		TA-R1	1307,24	306,91	32,05
		TA-R2	1726,20	505,39	

TABLEAU A.5: Jeu de 100 fichiers de 50 clients pour la stratégie PartNN2

% dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
10%	S1	TT-R1	23648,46	2972,15	2,85
		TT-R2	24322,70	2743,19	
		TA-R1	7172,00	738,79	2,12
		TA-R2	7323,81	755,22	
	S2	TT-R1	23336,87	3225,42	6,83
		TT-R2	24929,89	2917,03	
		TA-R1	7097,80	658,73	5,33
		TA-R2	7476,33	824,37	
	S3	TT-R1	25316,28	3853,68	9,42
		TT-R2	27700,48	4670,45	
		TA-R1	7964,92	799,49	5,61
		TA-R2	8412,11	906,17	
	Moyenne	TT-R1	24100,54	3350,42	6,43
		TT-R2	25651,02	3443,55	
		TA-R1	7411,58	732,34	4,40
		TA-R2	7737,42	828,59	
	S1	TT-R1	24733,50	2728,91	2,64
		TT-R2	25387,01	2488,48	
		TA-R1	6509,64	806,51	3,67
		TA-R2	6748,85	886,89	

A suivre \Rightarrow

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
20%	S2	TT-R1	24445,90	3089,09	6,83
		TT-R2	26114,66	2428,56	
		TA-R1	6504,34	799,00	7,53
		TA-R2	6993,99	853,66	
	S3	TT-R1	26018,59	3834,15	14,73
		TT-R2	29850,25	5013,51	
		TA-R1	7330,73	835,23	10,61
		TA-R2	8108,20	1083,63	
	Moyenne	TT-R1	25066,00	3217,38	8,18
		TT-R2	27117,31	3310,18	
		TA-R1	6781,57	813,58	7,40
		TA-R2	7283,68	941,39	
30%	S1	TT-R1	25201,00	2828,28	2,40
		TT-R2	25805,46	2435,46	
		TA-R1	5694,76	822,77	4,36
		TA-R2	5943,04	893,73	
	S2	TT-R1	25150,47	2826,85	6,18
		TT-R2	26704,91	2440,26	
		TA-R1	5714,66	771,19	10,56
		TA-R2	6318,32	985,78	
	S3	TT-R1	26648,80	3147,74	11,18
		TT-R2	29629,39	4553,97	
		TA-R1	6637,64	872,29	10,27
		TA-R2	7319,53	1124,71	
	Moyenne	TT-R1	25666,76	2934,29	6,67
		TT-R2	27379,92	3143,23	
		TA-R1	6015,69	822,08	8,50
		TA-R2	6526,96	1001,41	
40%	S1	TT-R1	25742,68	2262,13	1,90
		TT-R2	26230,51	2082,01	
		TA-R1	4854,26	762,97	4,66
		TA-R2	5080,34	810,95	
	S2	TT-R1	25500,38	2574,90	5,07
		TT-R2	26792,44	2027,18	
		TA-R1	4841,37	710,70	12,42
		TA-R2	5442,79	937,55	
	S3	TT-R1	26661,34	2786,67	11,14
		TT-R2	29632,20	4035,48	
		TA-R1	5779,35	808,03	10,96
		TA-R2	6412,51	1114,35	
	Moyenne	TT-R1	25968,13	2541,23	6,10
		TT-R2	27551,72	2714,89	
		TA-R1	5158,32	760,57	9,44
		TA-R2	5645,22	954,28	
	S1	TT-R1	25703,71	2655,86	2,90
		TT-R2	26449,66	2355,54	

A suivre ⇒

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
50%		TA-R1	4254,43	818,57	6,73
		TA-R2	4540,77	956,09	
	S2	TT-R1	25614,91	2752,37	5,27
		TT-R2	26963,85	2225,29	
		TA-R1	4268,29	820,17	13,64
		TA-R2	4850,56	952,47	
	S3	TT-R1	26889,49	2933,25	11,55
		TT-R2	29995,04	4000,75	
		TA-R1	5132,72	911,85	12,80
		TA-R2	5789,94	1162,83	
60%		TT-R1	26069,37	2780,49	6,65
		TT-R2	27802,85	2860,53	
	Moyenne	TA-R1	4551,82	850,19	11,17
		TA-R2	5060,42	1023,80	
	S1	TT-R1	26125,44	2117,10	1,68
		TT-R2	26564,81	1872,10	
		TA-R1	3268,42	754,97	7,95
		TA-R2	3528,13	891,05	
	S2	TT-R1	25980,91	2276,15	3,98
		TT-R2	27014,38	2021,90	
70%		TA-R1	3304,26	806,28	15,56
		TA-R2	3818,57	1100,86	
	S3	TT-R1	27132,32	2578,80	8,49
		TT-R2	29434,60	3758,10	
		TA-R1	4180,02	959,63	12,07
		TA-R2	4684,75	1147,58	
	Moyenne	TT-R1	26412,89	2324,02	4,76
		TT-R2	27671,26	2550,70	
		TA-R1	3584,23	840,29	11,89
		TA-R2	4010,48	1046,49	
70%	S1	TT-R1	26367,09	2051,75	2,10
		TT-R2	26919,84	1699,08	
		TA-R1	2687,37	663,43	9,49
		TA-R2	2942,34	676,94	
	S2	TT-R1	26233,98	2093,85	4,33
		TT-R2	27371,12	1614,48	
		TA-R1	2695,16	669,18	22,14
		TA-R2	3291,84	859,59	
	S3	TT-R1	27387,51	2031,41	8,91
		TT-R2	29828,44	3150,78	
		TA-R1	3635,92	939,57	17,27
		TA-R2	4264,02	1053,34	
	Moyenne	TT-R1	26662,86	2059,00	5,16
		TT-R2	28039,80	2154,78	
		TA-R1	3006,15	757,39	

A suivre \Rightarrow

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
		TA-R2	3499,40	863,29	16,41
80%	S1	TT-R1	26983,21	1998,64	
		TT-R2	27465,63	1629,76	1,79
		TA-R1	1983,53	565,06	
		TA-R2	2300,86	635,13	16,00
	S2	TT-R1	26836,60	2097,74	
		TT-R2	27560,58	1731,75	2,70
		TA-R1	1971,85	495,01	
		TA-R2	2455,99	639,22	24,55
	S3	TT-R1	27759,44	1967,22	
		TT-R2	30264,20	3582,44	9,02
		TA-R1	2734,07	702,48	
		TA-R2	3654,51	1042,10	33,67
	Moyenne	TT-R1	27193,08	2021,20	
		TT-R2	28430,14	2314,65	4,55
		TA-R1	2229,82	587,52	
		TA-R2	2803,78	772,15	25,74
90%	S1	TT-R1	27478,53	1683,96	
		TT-R2	27954,91	1400,31	1,73
		TA-R1	1646,80	550,63	
		TA-R2	2055,52	658,64	24,82
	S2	TT-R1	27290,66	1820,59	
		TT-R2	28080,11	1320,41	2,89
		TA-R1	1693,15	566,43	
		TA-R2	2272,49	703,30	34,22
	S3	TT-R1	28292,07	1637,63	
		TT-R2	31367,60	3722,96	10,87
		TA-R1	2427,31	753,08	
		TA-R2	3720,09	1410,87	53,26
	Moyenne	TT-R1	27687,09	1714,06	
		TT-R2	29134,21	2147,89	5,23
		TA-R1	1922,42	623,38	
		TA-R2	2682,70	924,27	39,55
100%	S1	TT-R1	27868,76	1651,86	
		TT-R2	28404,26	1322,06	1,92
		TA-R1	1499,76	579,64	
		TA-R2	1994,04	659,60	32,96
	S2	TT-R1	27685,89	1816,86	
		TT-R2	28796,59	1364,31	4,01
		TA-R1	1524,65	576,74	
		TA-R2	2252,13	742,87	47,71
	S3	TT-R1	28684,94	1448,43	
		TT-R2	32158,33	3510,92	12,11
		TA-R1	2320,32	760,80	
		TA-R2	3754,34	1166,57	61,80
		TT-R1	28079,86	1639,05	

A suivre ⇒

%dod	Scénario	Critère	Moyenne	Ecart Type	Gain (%)
	Moyenne	TT-R2	29786,39	2065,76	6,08
		TA-R1	1781,58	639,06	
		TA-R2	2666,84	856,35	49,69

Annexe B

Résultats des problèmes de DVRPTW avec temps de trajet dépendant du temps

B.1 Réglages des paramètres de l'algorithme génétique

TABLEAU B.1: Réglages des paramètres de l'algorithme génétique pour le type 1 de problèmes dynamiques

Problème	Taille	Cr	TCr	Mt	TMt	TElt(%)
C101	30	MX1T/D	0.5	Or-Opt	0.2	70
C102	30	MX1T/D	0.5	Or-Opt	0.2	70
C103	30	PMX	0.5	Or-Opt	0.2	50
C104	30	PMX	0.5	Or-Opt	0.2	50
C105	30	MX1T/D	0.5	Or-Opt	0.2	70
C106	30	MX1T/D	0.5	Or-Opt	0.2	70
C107	30	MX1T/D	0.5	Or-Opt	0.2	70
C108	30	MX1T/D	0.5	Or-Opt	0.2	70
C109	30	MX1T/D	0.5	Or-Opt	0.2	70
R101	50	MX1T/D	0.7	Or-Opt	0.5	50
R102	30	PMX	0.5	1-Opt	0.2	50
R103	30	PMX	0.5	1-Opt	0.2	50
R104	30	PMX	0.5	1-Opt	0.2	50
R105	50	MX1T/D	0.7	Or-Opt	0.5	50
R106	30	PMX	0.5	1-Opt	0.2	50
R107	30	PMX	0.5	1-Opt	0.2	50
R108	30	PMX	0.5	1-Opt	0.2	50
R109	50	MX1T/D	0.7	Or-Opt	0.5	50
R110	50	MX1T/D	0.7	Or-Opt	0.5	50
R111	50	MX1T/D	0.7	Or-Opt	0.5	50
R112	50	MX1T/D	0.7	Or-Opt	0.5	50
RC101	30	MX1T/D	0.7	Or-Opt	0.2	70

A suivre \Rightarrow

Problème	Taille	Cr	TCr	Mt	TMt	TElt(%)
RC102	30	MX1T/D	0.7	Or-Opt	0.2	70
RC103	50	MX1T/D	0.5	Or-Opt	0.5	50
RC104	50	MX1T/D	0.5	Or-Opt	0.5	50
RC105	30	MX1T/D	0.7	Or-Opt	0.2	70
RC106	30	MX1T/D	0.7	Or-Opt	0.2	70
RC107	30	MX1T/D	0.7	Or-Opt	0.2	70
RC108	30	MX1T/D	0.7	Or-Opt	0.2	70

TABLEAU B.2: Réglages des paramètres de l'algorithme génétique pour le type 2 de problèmes dynamiques

Problème	Taille	Cr	TCr	Mt	TMt	TElt(%)
C201	50	MX1T/D	0.7	Or-Opt	0.2	50
C202	50	MX1T/D	0.7	Or-Opt	0.2	50
C203	50	PMX	0.5	Or-Opt	0.2	50
C204	50	PMX	0.5	Or-Opt	0.2	50
C205	50	MX1T/D	0.7	Or-Opt	0.2	50
C206	50	MX1T/D	0.7	Or-Opt	0.2	50
C207	50	MX1T/D	0.7	Or-Opt	0.2	50
C208	50	MX1T/D	0.7	Or-Opt	0.2	50
R201	50	MX1T/D	0.5	Or-Opt	0.5	50
R202	50	MX1T/D	0.5	Or-Opt	0.5	50
R203	50	PMX	0.7	Or-Opt	0.5	70
R204	50	PMX	0.7	Or-Opt	0.5	70
R205	50	MX1T/D	0.5	Or-Opt	0.5	50
R206	50	MX1T/D	0.5	Or-Opt	0.5	50
R207	50	PMX	0.7	Or-Opt	0.5	70
R208	50	PMX	0.7	Or-Opt	0.5	70
R209	50	MX1T/D	0.5	Or-Opt	0.5	50
R210	50	MX1T/D	0.5	Or-Opt	0.5	50
R211	50	MX1T/D	0.5	Or-Opt	0.5	50
RC201	50	MX1T/D	0.7	1-Opt	0.5	70
RC202	50	MX1T/D	0.7	1-Opt	0.5	70
RC203	30	PMX	0.7	Or-Opt	0.5	50
RC204	30	PMX	0.7	Or-Opt	0.5	50
RC205	50	MX1T/D	0.7	1-Opt	0.5	70
RC206	50	MX1T/D	0.7	1-Opt	0.5	70
RC207	50	MX1T/D	0.7	1-Opt	0.5	70
RC208	50	MX1T/D	0.7	1-Opt	0.5	70

B.2 Exemple : Le problème C101 dynamique

Le problème C101 dynamique est divisé en 3 parties :

1. Les clients : Chaque client est représenté par son identification (la colonne Id), sa position (les colonnes X,Y), les deux limites de temps (les colonnes E,L), le temps de service (la colonne S) et le temps de réception (la colonne Tr).
2. Les véhicules : Les véhicules disponibles (V) sont homogènes et caractérisés par leur nombre et leur capacité.
3. Le dépôt : Il est défini par son identification (Id), sa position (X, Y) et son horaire d'ouverture (E, L).

TABLEAU B.3: Fichier de données du problème dynamique C101Dyn (Format texte)

1. Les clients							
Id	X	Y	D	E	L	S	Tr
0	42	66	10	65	146	90	0
1	42	68	10	727	782	90	0
2	42	65	10	15	67	90	0
3	38	68	20	255	324	90	0
4	38	70	10	534	605	90	0
5	35	66	10	357	410	90	0
6	35	69	10	448	505	90	0
7	22	75	30	30	92	90	0
8	20	80	40	384	429	90	0
9	20	85	40	475	528	90	0
10	15	80	10	278	345	90	0
11	30	50	10	10	73	90	0
12	30	52	20	914	965	90	0
13	28	55	10	732	777	90	0
14	25	50	10	65	144	90	0
15	25	55	10	622	701	90	0
16	23	52	10	261	316	90	0
17	20	50	10	358	405	90	0
18	20	55	10	449	504	90	0
19	10	35	20	200	237	90	0
20	10	40	30	31	100	90	0
21	8	40	40	87	158	90	0
22	5	35	10	283	344	90	0
23	5	45	10	665	716	90	0
24	2	40	20	383	434	90	0
25	0	40	30	479	522	90	0
26	0	45	20	567	624	90	0
27	35	30	10	264	321	90	0
28	33	35	10	16	80	90	0

A suivre \Rightarrow

Id	X	Y	D	E	L	S	Tr
29	30	32	30	448	509	90	0
30	28	30	10	632	693	90	0
31	26	32	10	815	880	90	0
32	42	10	40	186	257	90	0
33	40	5	30	385	436	90	0
34	40	15	40	35	87	90	0
35	38	5	30	471	534	90	0
36	50	40	50	171	218	90	0
37	48	30	10	632	693	90	0
38	48	40	10	76	129	90	0
39	47	35	10	826	875	90	0
40	47	40	10	12	77	90	0
41	45	30	10	734	777	90	0
42	45	35	10	916	969	90	0
43	95	35	20	293	360	90	0
44	92	30	10	478	551	90	0
45	53	35	50	353	412	90	0
46	85	35	30	47	124	90	0
47	75	55	20	369	420	90	0
48	72	55	10	265	338	90	0
49	66	55	10	173	238	90	0
50	65	55	20	85	144	90	0
51	65	60	30	645	708	90	0
52	60	55	10	20	84	90	0
53	60	60	10	836	889	90	0
54	65	85	40	475	518	90	0
55	60	80	10	95	156	90	0
56	60	85	30	561	622	90	0
57	58	75	20	30	84	90	0
58	55	80	10	743	820	90	0
59	55	85	20	647	726	90	0
60	45	68	10	912	967	90	217,05
61	45	70	30	825	870	90	357,92
62	40	69	20	621	702	90	369,09
63	40	66	20	170	225	90	0,03
64	25	85	20	652	721	90	62,89
65	22	85	10	567	620	90	411,22
66	18	75	20	99	148	90	73,84
67	15	75	20	179	254	90	87,17
68	28	52	20	812	883	90	144,01
69	25	52	40	169	224	90	69,34
70	23	55	20	546	593	90	66,95
71	8	45	20	751	816	90	342,84
72	35	32	10	166	235	90	64,08
73	33	32	20	68	149	90	44,64
74	32	30	10	359	412	90	164,05

A suivre \Rightarrow

Id	X	Y	D	E	L	S	Tr
75	30	30	10	541	600	90	192,81
76	30	35	10	1054	1127	90	124,72
77	28	35	10	1001	1066	90	240
78	25	30	10	725	786	90	615,99
79	25	35	10	912	969	90	315,89
80	44	5	20	286	347	90	44,19
81	42	15	10	95	158	90	8,72
82	38	15	10	651	740	90	534,18
83	35	5	20	562	629	90	96,41
84	50	30	10	531	610	90	482,17
85	50	35	20	262	317	90	118,58
86	95	30	30	387	456	90	258,24
87	53	30	10	450	505	90	117,07
88	45	65	20	997	1068	90	731,08
89	90	35	10	203	260	90	21,92
90	88	30	10	574	643	90	121,2
91	88	35	20	109	170	90	26,75
92	87	30	10	668	731	90	481,6
93	85	25	10	769	820	90	536,54
94	70	58	20	458	523	90	269,76
95	68	60	30	555	612	90	269,66
96	63	58	10	737	802	90	94,25
97	67	85	20	368	441	90	59,15
98	65	82	10	285	336	90	147,94
99	62	80	30	196	239	90	106,62

2. Les véhicules

Id	Nombre	Capacité
V	10	2000

3. Le dépôt

Id	X	Y	E	L
D	40	50	0	1236

Annexe C

Résultats des problèmes de DVRPTW temps réel avec temps de trajet dépendant du temps

C.1 Résultats pour les neuf problèmes C1 et pour les scénarios D2, D3 et D4

TABLEAU C.1 – Résultats du problème C1 pour le scénario D2

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C101	965,50	46,41	10,64	976,13	1163,65	0,00	15,37	1179,01
	C102	1079,89	496,68	176,82	1256,71	1314,82	427,15	236,38	1551,19
	C103	1394,06	1144,90	381,37	1775,43	1669,89	1053,81	668,32	2338,20
	C104	1175,83	586,16	81,78	1257,60	1374,34	533,91	132,16	1506,50
	C105	1035,91	219,88	321,96	1357,86	1223,19	167,90	349,12	1572,31
	C106	1123,34	393,53	399,72	1523,06	1501,34	346,59	1209,85	2711,19
	C107	988,84	146,78	209,49	1198,34	1069,70	136,63	231,99	1301,69
	C108	1017,91	216,29	35,92	1053,82	1242,67	163,95	137,29	1379,96
	C109	1018,40	318,40	47,77	1066,17	1150,17	309,92	38,83	1188,99
	Moyenne	1088,85	396,56	185,05	1273,90	1301,08	348,87	335,48	1636,56
R2	C101	835,34	219,40	805,90	1641,24	1055,35	193,06	824,39	1879,73
	C102	1083,88	147,49	1072,50	2156,38	1594,37	318,15	2990,93	4585,31
	C103	1155,65	1318,89	615,33	1770,98	1478,66	1131,89	810,11	2288,78
	C104	853,51	975,37	0,00	853,51	1096,65	928,11	25,04	1121,70
	C105	723,11	260,31	246,85	969,96	1174,96	289,25	585,63	1760,59
	C106	775,72	330,82	96,81	872,53	983,78	312,35	100,87	1084,65
	C107	715,65	219,89	26,90	742,55	1032,44	167,36	125,99	1158,43
	C108	829,74	346,05	178,70	1008,44	1278,09	985,93	2701,07	3979,17
	C109	785,13	249,95	8,74	793,87	1126,48	224,44	163,31	1289,79
	Moyenne	861,97	452,02	339,08	1201,05	1202,31	505,62	925,26	2127,57
Gain(%)									23,08

TABLEAU C.2 – Résultats du problème C1 pour le scénario D3

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C101	1023,21	373,40	1621,91	2645,11	1244,91	336,52	1921,59	3166,50
	C102	1467,18	81,85	2214,37	3681,55	1688,04	76,40	2841,41	4529,45
	C103	1430,71	601,04	1821,37	3252,08	1711,58	523,43	2015,38	3726,96
	C104	1322,81	564,44	206,36	1529,17	1597,21	487,06	331,79	1929,00
	C105	1113,55	243,28	837,34	1950,89	1307,16	218,57	1364,32	2671,48
	C106	1363,76	419,92	1401,42	2765,18	1698,03	400,35	2303,24	4001,27
	C107	1317,03	221,12	615,15	1932,18	1564,75	211,18	960,64	2525,38
	C108	1525,15	424,76	1311,61	2836,77	1769,04	348,76	1695,18	3464,22
	C109	1003,20	216,94	72,61	1075,81	1244,84	207,14	113,52	1358,36
	Moyenne	1285,18	349,64	1122,46	2407,64	1536,17	312,16	1505,23	3041,40
R2	C101	718,48	50,43	0,00	718,48	2747,68	0,00	2286,19	5033,86
	C102	1006,53	350,62	1070,30	2076,83	3245,63	97,19	3667,95	6913,58
	C103	1136,20	694,84	282,66	1418,85	3215,58	415,43	2180,96	5396,54
	C104	802,52	637,49	30,90	833,42	2955,42	370,75	454,13	3409,55
	C105	723,61	297,82	246,85	970,46	2741,10	136,60	1160,20	3901,30
	C106	708,27	218,56	0,72	708,99	2880,15	89,31	1929,88	4810,03
	C107	858,41	151,11	154,36	1012,77	3172,53	75,07	3439,53	6612,06
	C108	906,05	357,96	139,73	1045,78	3242,22	231,93	3129,13	6371,35
	C109	779,49	236,42	18,35	797,84	2941,69	131,82	614,62	3556,30
	Moyenne	848,84	332,80	215,99	1064,82	3015,78	172,01	2095,84	5111,62
Gain(%)									40,50

TABLEAU C.3 – Résultats du problème C1 pour le scénario D4

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C101	1269,91	162,76	1587,32	2857,23	1612,83	145,16	2392,45	4005,28
	C102	1327,61	506,47	3808,44	5136,05	2057,67	98,14	4293,90	6351,57
	C103	1692,85	1828,67	3726,40	5419,25	2005,50	1756,75	4749,20	6754,70
	C104	1166,07	513,35	64,09	1230,16	1370,69	453,37	185,00	1555,70
	C105	935,97	411,38	1238,13	2174,10	1446,23	346,05	1408,00	2854,23
	C106	1062,98	520,02	2866,63	3929,61	1705,86	485,58	3137,10	4842,97
	C107	1097,37	413,63	3985,66	5083,03	1988,43	691,27	4094,68	6083,12
	C108	1213,36	989,18	4413,61	5626,97	2396,93	911,71	6191,04	8587,97
	C109	963,96	478,53	112,08	1076,04	1448,14	194,30	256,21	1704,35
	Moyenne	1192,23	647,11	2422,49	3614,72	1781,37	564,70	2967,51	4748,88
R2	C101	980,23	308,97	1589,94	2570,17	1699,14	226,95	3205,53	4904,67
	C102	1345,10	511,14	2670,33	4015,43	2186,94	341,28	5396,76	7583,70
	C103	1475,33	1658,02	3769,42	5244,75	2249,06	1417,58	5476,15	7725,21
	C104	1055,97	1233,14	73,74	1129,71	1789,83	1119,57	597,56	2387,38
	C105	1038,20	399,23	1072,37	2110,57	1543,24	233,84	2425,51	3968,76
	C106	1089,79	392,63	798,49	1888,28	1771,56	349,14	2845,04	4616,60
	C107	1038,58	513,04	990,42	2029,00	1780,64	472,53	3218,42	4999,07
	C108	1204,03	902,11	2804,42	4008,45	1916,37	694,11	4052,66	5969,03
	C109	1075,71	781,75	701,46	1777,18	1799,65	677,23	1895,61	3695,26
	Moyenne	1144,77	744,45	1607,84	2752,61	1859,60	614,69	3234,81	5094,41
Gain(%)									6,78

C.2 Résultats pour les neuf problèmes C1 et pour les trois scénarios V1, V2 et V3

TABLEAU C.4 – Résultats du problème C1 pour le scénario V1

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C101	965,50	46,41	10,64	976,13	1163,65	0,00	15,37	1179,01
	C102	1079,89	496,68	176,82	1256,71	1314,82	427,15	236,38	1551,19
	C103	1394,06	1144,90	381,37	1775,43	1669,89	1053,81	668,32	2338,20
	C104	1175,83	586,16	81,78	1257,60	1374,34	533,91	132,16	1506,50
	C105	1035,91	219,88	321,96	1357,86	1223,19	167,90	349,12	1572,31
	C106	1123,34	393,53	399,72	1523,06	1501,34	346,59	1209,85	2711,19
	C107	988,84	146,78	209,49	1198,34	1069,70	136,63	231,99	1301,69
	C108	1017,91	216,29	35,92	1053,82	1242,67	163,95	137,29	1379,96
	C109	1018,40	318,40	47,77	1066,17	1150,17	309,92	38,83	1188,99
	Moyenne	1088,85	396,56	185,05	1273,90	1301,08	348,87	335,48	1636,56
R2	C101	835,34	219,40	805,90	1641,24	1055,35	193,06	824,39	1879,73
	C102	1083,88	147,49	1072,50	2156,38	1594,37	318,15	2990,93	4585,31
	C103	1155,65	1318,89	615,33	1770,98	1478,66	1131,89	810,11	2288,78
	C104	853,51	975,37	0,00	853,51	1096,65	928,11	25,04	1121,70
	C105	723,11	260,31	246,85	969,96	1174,96	289,25	585,63	1760,59
	C106	775,72	330,82	96,81	872,53	983,78	312,35	100,87	1084,65
	C107	715,65	219,89	26,90	742,55	1032,44	167,36	125,99	1158,43
	C108	829,74	346,05	178,70	1008,44	1278,09	985,93	2701,07	3979,17
	C109	785,13	249,95	8,74	793,87	1126,48	224,44	163,31	1289,79
	Moyenne	861,97	452,02	339,08	1201,05	1202,31	505,62	925,26	2127,57
Gain(%)									23,08

TABLEAU C.5 – Résultats du problème C1 pour le scénario V2

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C101	1168,65	301,64	1887,26	3055,91	1296,49	297,37	1962,72	3259,21
	C102	1192,31	341,74	1046,37	2238,68	1348,61	332,54	1069,47	2418,08
	C103	1509,97	627,38	1058,01	2567,97	1593,60	626,71	1176,78	2770,38
	C104	0,00	0,00	0,00	919,47	1060,26	777,69	44,02	1104,28
	C105	1066,57	281,27	515,73	1582,30	1168,90	273,85	512,39	1681,29
	C106	1033,93	329,04	959,07	1992,99	1252,00	313,52	1060,49	2312,49
	C107	988,84	146,78	209,49	1198,34	1069,70	136,63	231,99	1301,69
	C108	1192,98	968,37	2806,94	3999,93	1216,93	959,13	2823,43	4040,35
	C109	1018,40	318,40	47,77	1066,17	1150,17	309,92	38,83	1188,99
	Moyenne	1019,07	368,29	947,85	2069,08	1239,63	447,49	991,12	2230,75
R2	C101	983,10	400,32	2293,05	3276,15	1333,51	383,02	2732,67	4066,17
	C102	1285,33	80,83	1614,09	2899,41	1416,41	654,34	2468,96	3885,36
	C103	1004,65	834,97	101,55	1106,20	1624,24	1388,72	2706,19	4330,43
	C104	957,75	779,10	25,55	983,30	1230,61	751,51	137,58	1368,19
	C105	831,87	459,88	1705,52	2537,39	1008,11	417,23	1756,27	2764,38
	C106	891,96	328,99	795,03	1686,98	1245,83	306,92	1208,27	2454,10
	C107	994,52	510,65	890,05	1884,57	1290,40	506,82	1120,42	2410,82
	C108	950,81	824,43	793,21	1744,03	1421,14	731,07	2515,06	3936,21
	C109	954,62	280,90	242,50	1197,12	1271,14	270,14	243,98	1515,12
	Moyenne	983,84	500,01	940,06	1923,91	1315,71	601,08	1654,38	2970,09
Gain(%)									24,89

TABLEAU C.6 – Résultats du problème C1 pour le scénario V3

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C101	1561,68	240,52	2318,82	3880,50	1614,98	232,47	2334,82	3949,81
	C102	1801,01	485,58	2235,98	4036,99	1931,28	453,40	2619,70	4550,98
	C103	1645,30	1084,52	1376,40	3021,70	1687,66	1080,71	1390,87	3078,54
	C104	1373,01	1163,02	69,88	1442,89	1442,52	1160,56	75,36	1517,87
	C105	1423,87	91,60	401,32	1825,19	1638,59	87,24	431,19	2069,78
	C106	1493,10	256,69	948,06	2441,16	1595,15	245,79	950,15	2545,30
	C107	1497,17	252,99	771,23	2268,41	1570,45	253,96	782,66	2353,12
	C108	1555,10	733,14	2485,99	4041,09	1666,84	728,67	2563,85	4230,69
	C109	1677,18	544,47	234,69	1911,87	1778,14	539,75	267,73	2045,86
	Moyenne	1558,60	539,17	1204,71	2763,31	1658,40	531,39	1268,48	2926,88
R2	C101	1361,33	76,90	1846,87	3208,20	1333,51	383,02	2732,67	4066,17
	C102	1537,52	238,62	1554,97	3092,50	1594,37	318,15	2990,93	4585,31
	C103	1576,23	1544,66	1945,02	3521,24	1836,75	1784,80	3515,07	5351,82
	C104	1125,59	822,44	13,99	1139,58	1594,16	762,31	709,74	2303,90
	C105	1400,32	377,55	1747,18	3147,50	1810,34	367,27	2367,78	4178,12
	C106	1447,64	356,48	2047,22	3494,86	1728,19	743,83	3424,98	5153,17
	C107	1137,33	255,34	539,73	1677,07	1583,20	247,18	1118,15	2701,35
	C108	1140,49	363,83	202,16	1342,65	1952,74	972,29	4869,90	6822,63
	C109	1363,86	840,62	272,45	1636,31	1712,98	836,25	726,24	2439,22
	Moyenne	1343,37	541,83	1129,95	2473,32	1682,92	712,79	2495,05	4177,97
Gain(%)									29,94

C.3 Résultats pour toutes les catégories de problèmes de test avec la distribution D2 et le scénario V1

TABEAU C.7 – Résultats du problème C1

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C101	965,50	46,41	10,64	976,13	1163,65	0,00	15,37	1179,01
	C102	1079,89	496,68	176,82	1256,71	1314,82	427,15	236,38	1551,19
	C103	1394,06	1144,90	381,37	1775,43	1669,89	1053,81	668,32	2338,20
	C104	1175,83	586,16	81,78	1257,60	1374,34	533,91	132,16	1506,50
	C105	1035,91	219,88	321,96	1357,86	1223,19	167,90	349,12	1572,31
	C106	1123,34	393,53	399,72	1523,06	1501,34	346,59	1209,85	2711,19
	C107	988,84	146,78	209,49	1198,34	1069,70	136,63	231,99	1301,69
	C108	1017,91	216,29	35,92	1053,82	1242,67	163,95	137,29	1379,96
	C109	1018,40	318,40	47,77	1066,17	1150,17	309,92	38,83	1188,99
	Moyenne	1088,85	396,56	185,05	1273,90	1301,08	348,87	335,48	1636,56
R2	C101	835,34	219,40	805,90	1641,24	1055,35	193,06	824,39	1879,73
	C102	1083,88	147,49	1072,50	2156,38	1594,37	318,15	2990,93	4585,31
	C103	1155,65	1318,89	615,33	1770,98	1478,66	1131,89	810,11	2288,78
	C104	853,51	975,37	0,00	853,51	1096,65	928,11	25,04	1121,70
	C105	723,11	260,31	246,85	969,96	1174,96	289,25	585,63	1760,59
	C106	775,72	330,82	96,81	872,53	983,78	312,35	100,87	1084,65
	C107	715,65	219,89	26,90	742,55	1032,44	167,36	125,99	1158,43
	C108	829,74	346,05	178,70	1008,44	1278,09	985,93	2701,07	3979,17
	C109	785,13	249,95	8,74	793,87	1126,48	224,44	163,31	1289,79
	Moyenne	861,97	452,02	339,08	1201,05	1202,31	505,62	925,26	2127,57
Gain(%)									23,08

TABLEAU C.8 – Résultats du problème R1

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	R101	1424,90	1284,73	292,88	1717,78	1603,16	1241,92	314,52	1917,68
	R102	1332,43	1079,52	332,49	1664,92	1499,67	1035,81	406,35	1906,02
	R103	1230,91	527,08	218,32	1449,24	1373,11	521,83	354,49	1727,60
	R104	999,05	197,90	63,03	1062,08	1107,53	183,63	127,22	1234,75
	R105	1362,82	557,59	441,58	1804,40	1468,70	538,62	510,79	1979,49
	R106	1330,95	493,04	1081,45	2412,40	1447,44	476,97	1326,95	2774,39
	R107	1170,36	141,31	542,59	1712,94	1272,31	142,18	756,34	2028,65
	R108	1161,15	69,54	569,18	1730,33	1344,35	68,94	1201,79	2546,14
	R109	1243,63	305,26	839,84	2083,47	1373,97	304,48	1138,90	2512,87
	R110	1256,14	183,63	441,89	1698,04	1370,29	179,75	682,98	2053,27
	R111	1225,76	271,23	1067,04	2292,79	1339,51	264,17	1328,89	2668,40
	R112	1081,67	360,47	532,09	1613,75	1175,52	356,87	782,45	1957,97
	Moyenne	1234,98	455,94	535,20	1770,18	1364,63	442,93	744,31	2108,94
R2	R101	1384,56	1116,57	563,36	1947,92	1613,14	1042,79	680,64	2293,78
	R102	1301,81	1060,42	377,90	1679,72	1545,98	1019,62	495,86	2041,85
	R103	1126,09	627,72	285,96	1412,05	1280,15	578,30	344,37	1624,52
	R104	1010,30	123,74	56,84	1067,14	1235,10	99,04	276,55	1511,65
	R105	1259,83	523,91	1037,20	2297,03	1441,78	514,26	1410,34	2852,12
	R106	1197,55	387,93	613,97	1811,52	1409,24	378,92	1088,85	2498,08
	R107	1087,73	280,17	618,32	1706,05	1261,71	277,37	1181,92	2443,63
	R108	1173,34	387,16	889,24	2062,59	1369,00	381,39	1409,42	2778,42
	R109	1230,54	297,99	769,55	2000,09	1376,37	293,87	1072,58	2448,95
	R110	1130,00	309,40	461,05	1591,05	1291,81	300,07	819,87	2111,67
	R111	1209,44	305,23	1045,68	2255,12	1367,87	301,07	1530,63	2898,49
	R112	1040,49	270,00	233,77	1274,26	1222,11	265,21	639,09	1861,20
	Moyenne	1179,31	474,19	579,40	1758,71	1367,85	454,33	912,51	2280,36
Gain(%)									7,52

TABLEAU C.9 – Résultats du problème RC1

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	RC101	1408,06	603,21	439,26	1847,32	1554,02	591,37	565,89	2119,91
	RC102	1314,29	129,60	262,77	1577,06	1464,58	130,76	576,99	2041,57
	RC103	1337,81	284,87	560,32	1898,12	1408,49	284,28	687,49	2095,97
	RC104	1128,68	197,47	106,72	1235,41	1253,90	198,27	306,26	1560,16
	RC105	1430,38	722,53	613,09	2043,47	1590,69	720,86	810,39	2401,07
	RC106	1311,02	523,92	463,58	1774,60	1412,17	504,97	605,19	2017,36
	RC107	1217,94	252,94	610,12	1828,07	1330,57	245,97	706,58	2037,15
	RC108	1180,99	331,02	624,49	1805,48	1292,54	318,98	896,17	2188,71
	Moyenne	1291,15	380,70	460,04	1751,19	1413,37	374,43	644,37	2057,74
R2	RC101	1403,33	587,35	536,48	1939,81	1685,70	531,65	914,89	2600,59
	RC102	1205,17	493,05	230,74	1435,92	1409,84	464,48	413,15	1822,99
	RC103	1257,22	348,81	566,01	1823,23	1419,77	338,13	900,84	2320,61
	RC104	1105,37	125,20	68,19	1173,56	1292,28	109,91	322,64	1614,92
	RC105	1298,12	810,80	744,61	2042,73	1500,32	766,81	958,78	2459,11
	RC106	1214,92	465,62	376,35	1591,26	1434,67	453,57	824,73	2259,40
	RC107	1227,40	250,52	564,07	1791,47	1374,40	243,63	769,34	2143,74
	RC108	1193,22	246,15	453,06	1646,27	1348,65	236,63	845,50	2194,15
	Moyenne	1238,09	415,94	442,44	1680,53	1433,21	393,10	743,73	2176,94
Gain(%)									5,48

TABLEAU C.10 – Résultats du problème C2

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	C201	847,14	16,45	229,47	1076,60	790,45	15,08	8,51	798,96
	C202	827,10	293,32	176,37	1003,47	809,53	291,93	128,57	938,10
	C203	1066,17	100,38	456,72	1522,89	1045,97	98,65	491,59	1537,56
	C204	1244,62	219,02	764,42	2009,04	1216,81	225,16	729,76	1946,57
	C205	732,89	42,61	0,00	732,89	697,48	41,72	0,00	697,48
	C206	779,16	48,60	0,00	779,16	758,26	47,76	0,00	758,26
	C207	735,01	93,76	0,00	735,01	705,94	94,96	0,00	705,94
	C208	905,20	248,44	474,97	1380,17	907,79	241,04	476,22	1384,01
	Moyenne	892,16	132,82	262,74	1154,90	866,53	132,04	229,33	1095,86
R2	C201	594,15	16,45	0,00	594,15	760,85	15,08	67,09	827,94
	C202	785,45	10,96	86,92	872,37	930,22	2,27	313,89	1244,11
	C203	1160,03	470,79	2477,18	3637,21	1275,44	469,17	2918,28	4193,72
	C204	902,91	352,54	101,02	1003,93	1095,21	351,31	577,42	1672,64
	C205	526,07	67,66	0,00	526,07	681,49	60,34	49,08	730,57
	C206	593,50	67,57	0,00	593,50	751,85	13,37	13,33	765,19
	C207	569,45	104,01	0,28	569,74	693,98	94,96	0,00	693,98
	C208	580,39	47,19	0,00	580,39	742,49	17,50	40,54	783,03
	Moyenne	713,99	142,15	333,18	1047,17	866,44	128,00	497,45	1363,90
Gain(%)									19,65

TABLEAU C.11 – Résultats du problème R2

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	R201	1450,47	1075,01	102,11	1552,58	1530,75	993,78	245,73	1776,47
	R202	1394,91	344,34	520,72	1915,64	1479,30	293,79	792,73	2272,03
	R203	1431,02	540,68	688,39	2119,40	1484,37	489,44	663,01	2147,38
	R204	1057,91	128,79	1104,15	2162,07	1098,23	133,36	1411,24	2509,47
	R205	1325,82	277,64	174,68	1500,49	1399,39	252,41	217,48	1616,87
	R206	1453,18	389,02	1692,66	3145,84	1586,19	343,96	2373,71	3959,90
	R207	1306,74	402,65	588,10	1894,84	1423,16	363,43	813,54	2236,70
	R208	1171,43	45,21	819,27	1990,70	1187,63	44,58	926,98	2114,61
	R209	1289,39	489,11	1088,71	2378,10	1421,67	484,87	1761,62	3183,29
	R210	1422,56	427,53	1166,73	2589,29	1583,27	377,49	1932,91	3516,19
	R211	1248,01	288,09	8339,64	9587,65	1243,66	287,13	8205,67	9449,33
	Moyenne	1322,86	400,73	1480,47	2803,33	1403,42	369,48	1758,60	3162,02
R2	R201	1310,89	1160,84	22,55	1333,43	1713,83	865,23	1054,23	2768,06
	R202	1256,40	371,49	376,26	1632,67	1445,81	311,52	1765,23	3211,04
	R203	1265,39	534,32	691,38	1956,77	1693,80	408,44	3253,75	4947,55
	R204	1015,85	3,94	240,61	1256,46	1197,50	30,40	1642,59	2840,09
	R205	1162,03	365,84	163,02	1325,05	1528,15	259,62	2783,68	4311,84
	R206	1067,40	464,02	100,23	1167,63	1214,36	263,15	308,11	1522,47
	R207	1244,93	598,21	953,07	2197,99	1776,44	479,16	3007,15	4783,59
	R208	964,99	62,56	87,02	1052,00	1295,06	0,00	1624,10	2919,16
	R209	1078,86	365,67	131,37	1210,23	1559,92	267,97	1825,14	3385,06
	R210	1234,35	474,89	302,81	1537,16	1515,80	405,59	1485,82	3001,61
	R211	1232,04	7,14	2418,10	3650,14	1670,22	0,00	5807,10	7477,32
	Moyenne	1166,65	400,81	498,77	1665,41	1510,08	299,19	2232,45	3742,53
Gain(%)									15,51

TABLEAU C.12 – Résultats du problème RC2

Mode de résolution	Problème	Résultats issus de la planification (étape 1)				Résultats en rejouant la solution (étape 2)			
		TT	TA	TR	VO	TT	TA	TR	VO
R1	RC201	1635,41	824,75	398,03	2033,43	1661,94	811,21	403,41	2065,35
	RC202	1438,45	1073,47	92,30	1530,75	1521,94	1047,34	167,23	1689,17
	RC203	1438,41	403,56	375,21	1813,61	1545,07	340,77	551,07	2096,14
	RC204	1124,03	458,10	6,82	1130,85	1176,56	445,62	31,41	1207,97
	RC205	1321,35	1188,73	322,36	1643,71	1360,99	1149,51	353,84	1714,83
	RC206	1444,63	262,00	503,88	1948,51	1575,50	262,67	929,78	2505,28
	RC207	1523,47	289,90	871,46	2394,93	1559,33	285,30	991,32	2550,65
	RC208	1345,35	485,35	1072,61	2417,97	1416,64	483,55	1328,35	2744,99
	Moyenne	1408,89	623,23	455,33	1864,22	1477,25	603,25	594,55	2071,80
R2	RC201	1602,61	828,57	264,16	1866,77	1794,24	727,93	385,64	2179,87
	RC202	1275,29	1143,40	67,85	1343,14	1557,23	956,26	379,16	1936,39
	RC203	1214,00	497,80	655,32	1869,32	1523,41	384,58	993,76	2517,17
	RC204	1033,18	394,02	47,84	1081,03	1233,96	268,72	82,14	1316,10
	RC205	1369,59	1096,61	1022,06	2391,65	1683,35	877,43	1921,05	3604,40
	RC206	1097,78	445,37	162,98	1260,76	1274,29	423,79	293,86	1568,15
	RC207	1167,76	543,56	1374,61	2542,37	1373,18	549,03	2426,52	3799,70
	RC208	1153,26	585,26	259,55	1412,82	1326,14	584,75	693,48	2019,62
	Moyenne	1239,18	691,82	481,80	1720,98	1470,73	596,56	896,95	2367,68
Gain(%)									12,50

Bibliographie

- [Archetti et al. 2006] C. Archetti , A. Hertz , and M. G. Speranza . « A Tabu Search Algorithm for the Split Delivery Vehicle Routing Problem ». *Transportation science*, 40(1), pages 64–73, 2006.
- [Aronson 1996] L.D. Aronson . « Algorithms for vehicle routing - A survey ». Technical Report 21, Delft University of Technology, 1996.
- [Badeau et al. 1997] P. Badeau , M. Gendreau , F. Guertin , J.-Y. Potvin , and E. Taillard . « A Parallel Tabu Search Heuristic for the Vehicle Routing Problem with Time Windows ». *Transportation Research - C* 5, pages 109–122, 1997.
- [Baker et al. 2003] B. M. Baker and M.A. Ayechev . « A genetic algorithm for the vehicle routing problem ». *Computers and Operations Research* 30, pages 787–800, 2003.
- [Beasley 1983] J.E. Beasley . « Route first - cluster second methods for vehicle routing ». *Omega* 11(4), pages 403–408, 1983.
- [Bent et al. 2003] R. Bent and P. Van Hentenryck . « Dynamic Vehicle Routing with Stochastic Requests ». *International Joint Conference on Artificial Intelligence*, pages 1362–1363, 2003.
- [Berger et al. 2004] J. Berger and M. Barkaoui . « A new hybrid genetic algorithm for the capacitated vehicle routing problem ». *Journal of the Operational Research Society*, 54, pages 1254–1262, 2004.
- [Bertsimas et al. 1991] D.J. Bertsimas and G.V. Ryzin . « A stochastic and dynamic vehicle routing problem in the euclidean plane ». *Operations Research* 39 (4), pages 601–615, 1991.
- [Bertsimas et al. 1993a] D.J. Bertsimas and G.V. Ryzin . « A stochastic and dynamic vehicle routing in the euclidean plane with multiple capacitated vehicles ». *Operations Research* 41 (1), pages 60–76, 1993.
- [Bertsimas et al. 1993b] D.J. Bertsimas and G.V. Ryzin . « A stochastic and dynamic vehicle routing with general demand interarrival time distributions ». *Advanced Applied Probability* 25, pages 947–978, 1993.
- [Bianchi 2000] Leonora Bianchi . « Notes on dynamic vehicle routing - the state of the art - ». Technical Report, IDSIA-05-01, December 2000.
- [Blanton et al. 1993] J.L. Blanton and R.L. Wainwright . « Multiple Vehicle Routing with Time and Capacity Constraints using Genetic Algorithms ». *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 452–459, 1993.

- [Braca et al. 1997] J. Braca , J. Bramel , B. Posner , and Simchi-Levi D . « A Computerized Approach to the New York City School Bus Routing Problem ». *IIE transactions (IIE trans.)*, 29(8), pages 693–702, 1997.
- [Bramel et al. 1993] J. Bramel , C.L. LI , and D. Simchi-Levi . « Probabilistic analysis of a Vehicle Routing Problem with Time Windows ». *American Journal of Mathematical and Management Sciences* 13, pages 267–322, 1993.
- [Breedam 1996] A. Van Breedam . « An Analysis of the Effect of Local Improvement Operators in Genetic Algorithms and Simulated Annealing for the Vehicle Routing Problem ». *RUCA, Faculty of Applied Economics, University of Antwerp, Antwerp, Belgium*, page Working Paper 96/14, 1996.
- [Bräysy et al. 2000] O. Bräysy , J. Berger , and M. Barkaoui . « A New Hybrid Evolutionary Algorithm for the Vehicle Routing Problem with Time Windows ». *Presented at the Route 2000 workshop, Danemark*, 2000.
- [Bräysy et al. 2001] O. Bräysy and M. Gendreau . « Metaheuristics for the Vehicle Routing Problem with Time Windows ». *SINTEF Applied Mathematics, Research Council of Norway*, 2001.
- [Caricato et al. 2003] P. Caricato , G. Ghiani , A. Grieco , and E. Guerriero . « Parallel Tabu Search For A Pickup And Delivery Problem Under Track Contention ». *Parallel Computing* 29 (5), pages 631–639, 2003.
- [Caseau et al. 1999] Y. Caseau and F. Laburthe . « Heuristics for Large Constrained Vehicle Routing Problems ». *Journal of Heuristics*, 5(3), pages 281–303, 1999.
- [Chen et al. 2006] Huey-Kuo Chen , Che-Fu Hsueh , and Mei-Shiang Chang . « The real-time time-dependent vehicle routing problem ». *Transportation Research Part E : Logistics and Transportation Review*, 42, pages 383–408, September 2006.
- [Chiang et al. 1996] W. Chiang and R.A. Russell . « Simulated Annealing Metaheuristics for the Vehicle Routing Problem with Time Windows ». *Annals of Operations Research*, 63, pages 3–27, 1996.
- [Christofides et al. 1979] N. Christofides , A. Mingozzi , and P. Toth . « The vehicle Routing Problem ». *Combinatorial Optimization. Wiley, Chichester*, pages 315–338, 1979.
- [Christofides et al. 1981] N. Christofides , S. A. Mingozzi , and P. Toth . « State Space Relaxation Procedures for the Computation of Bounds to Routing Problems ». *Networks* 11, pages 145–164, 1981.
- [Clarke et al. 1964] G. Clarke and J.W. Wright . « Sheduling of vehicules from a central depot to a number of delivery ponts ». *Operations Research* 12, pages 568–581, 1964.
- [Colorni et al. 1992] A. Colorni , M. Dorigo , and V. Maniezzo . « Distributed optimization by ant coloniess ». *proceedings of the first European Conference on Artificial Life (ECAL 91)*, pages 134–142, 1992.
- [Cordeau et al. 1997] J.-F. Cordeau , M. Gendreau , and G. Laporte . « A tabu search heuristic for periodic and multi-depot vehicle routing problems ». *Networks*, 30, pages 105–119, 1997.

-
- [Cordeau et al. 2001] J.-F. Cordeau , G. Laporte , and A. Mercier . « A unified tabu search heuristic for vehicle routing problems with time windows ». *Journal of the Operational Research Society* 52, pages 928–936, 2001.
- [Cordeau et al. 2002] J.F. Cordeau , G. Desaulniers , J. Desrosiers , M. Solomon , and F. Soumis . « The VRP with Time Windows ». *SIAM Monographs on Discrete Mathematics and Applications*, 9, P. Toth and D. Vigo (eds.), SIAM, Philadelphia, PA, pages 157–193, 2002.
- [Cordeau et al. 2004] J.-F. Cordeau , M. Gendreau , A. Hertz , G. Laporte , and J.-S. Sormany . « New Heuristics for the Vehicle Routing Problem ». Technical Report, G-2004-33, Université de Montréal, Canada, April 2004.
- [Davis 1991] L. Davis . *Handbook of genetic algorithms*. Van Nostrand Reinhold Company, 1991.
- [Deb et al. 1999] K. Deb and S. Agrawal . « Understanding interactions among genetic algorithm parameters ». *Foundation of Genetic Algorithms*, pages 265–286, 1999.
- [Derigs et al. 1993] U. Derigs and G. Grabenbauer . « A new heuristic approach to the Vehicle Routing Problem with Time Windows with a Bakery Fleet Case ». *American Journal of Mathematical and Management Sciences* 13, pages 249–366, 1993.
- [Desrochers et al. 1992] M. Desrochers , J. Desrosiers , and M. Solomon . « A new Optimization Algorithm for the Vehicle Routing Problem with Time Windows ». *Operations Research* 40, pages 342–354, 1992.
- [Desrosiers et al. 1995] J. Desrosiers , Y. Dumas , M. Solomon , and F. Soumis . « Time Constrained Routing and Scheduling ». *Handbooks in Operations Research and Management Science* 8, North Holland, Amsterdam, pages 35–139, 1995.
- [Duhamel 2001] C. Duhamel . « *Un Cadre Formel pour les Méthodes par Amélioration Itérative - Application à deux problèmes d’Optimisation dans les Réseaux* - ». PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, Mars 22 2001.
- [Eglese et al. 2005] R.W. Eglese , Z. Fu , and L. Li . « A tabu search heuristic for the open vehicle routing problem ». *Journal of the Operational Research Society* 56 (3), pages 267–274, 2005.
- [Eilon et al. 1971] Samuel Eilon , C. D. T. Watson-Gandy , and Nicos Christofides . *Distribution Management : Mathematical Modelling and Practical Analysis*. Hafner, New York, 1971.
- [Fischetti et al. 1999] M. Fischetti , A. Lodi , and P. Toth . « A Branch-and-Cut Algorithm for the Multiple Depot Vehicle Scheduling Problem ». *Dipartimento di Elettronica e Informatica, Università di Padova, Italy*, 1999.
- [Fisher et al. 1997] M.L. Fisher , K. J Rnsten , and O.B.G. Madsen . « Vehicle Routing with Time Windows : two optimization algorithms ». *Operations Research* 45 (3), pages 488–492, 1997.

- [Fisher 1995] M.L. Fisher . « Vehicle Routing ». *Handbooks in Operations Research and Management Science 8, North Holland, Amsterdam*, pages 1–33, 1995.
- [Gambardella et al. 2003] L.M. Gambardella , A. E. Rizzoli , F. Oliverio , N. Casagrande , A. V. Donati , R. Montemanni , and E. Lucibello . « Ant Colony Optimization for Vehicle Routing in Advanced Logistics Systems ». In *In Proceedings of the International Workshop on Modelling and Applied Simulation (MAS 2003)*, pages 3–9, 2003.
- [Garey et al. 1979] Michael R. Garey and David S. Johnson . *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co. New York, NY, USA, 1979.
- [Gendreau et al. 1998] M. Gendreau and J. Potvin . « Dynamic vehicle routing and dispatching ». *Fleet Management and Logistics*, pages 115–126, 1998.
- [Gendreau et al. 1999] Michel Gendreau , Francois Guertin , Jean-Yves Potvin , and Eric Taillard . « Parallel Tabu Search for Real-Time Vehicle Routing and Dispatching ». *Transportation Science*, 33, pages 381–390, 1999.
- [Ghiani et al. 2003] G. Ghiani , F. Guerriero , G. Laporte , and R. Musmanno . « Real-Time Vehicle Routing : Solution Concepts, Algorithms and Parallel Computing Strategies ». *European Journal of Operational Research 151*, pages 1–11, 2003.
- [Gillett et al. 1974] B.E. Gillett and L.R. Miller . « A heuristic algorithm for the Vehicle Dispatch Problem ». *Operations Research 22*, pages 340–349, 1974.
- [Glover 1986] F. Glover . « Future paths for Integer Programming and Links to Artificial Intelligence ». *Computers and Operations Research*, 5, pages 533–549, 1986.
- [Goldberg et al. 1985] D.E. Goldberg and R. Lingle . « Alleles, loci, and the tsp ». In *Proceedings of the First International Conference on Genetic Algorithms*, pages 154–159, 1985.
- [Goldberg et al. 1992] D. E. Goldberg , K. Deb , and J. H. Clark . « Genetic algorithms, noise, and the sizing of populations ». *Complex Systems*, 6, pages 333–362, 1992.
- [Goldberg 1989] D.E. Goldberg . « Genetic algorithms in search ». *Optimisation and Machine Learning*, 1989.
- [Goncalves et al. 2007] G. Goncalves , T. Hsu , R. Dupas , and H. Housroum . « Une plate-forme de simulation pour la gestion dynamique de tournées de véhicules ». *Journal Européen des Systèmes Automatisés*, 41(5), pages 515–539, 2007.
- [Goss et al. 1989] S. Goss , S. Aron , J.-L. Deneubourg , and J.M. Pasteels . « Self organized shortcuts in the argentine ant ». *Naturwissenschaften*, Vol. 76, pages 579–581, 1989.
- [Gourieroux et al. 1995] Christian Gourieroux and Alain Monfort . *Séries temporelles et modèles dynamiques*. Economica, 1995.
- [Grefenstette et al. 1985] J. Grefenstette , R. Gopal , R. Rosmaita , and D. Gucht . « Genetic algorithms for the traveling salesman problem ». In *Proceedings of the*

-
- Second International Conference on Genetic Algorithms. Lawrence Erlbaum Associates, Mahwah, NJ, pages 160–165, 1985.*
- [Gulay et al. 1999] B. Gulay and D. Ozgur . « A tabu search algorithm for the vehicle routing problem ». *Computers and Operations Research* 26 (3), pages 255–270, 1999.
- [Guntsch et al. 2000] M. Guntsch , J. Branke , and S. Hartmut . « ACO strategies for dynamic TSP ». *Second International Workshop on Ant Algorithms*, pages 59–62, 2000.
- [Haghani et al. 2005] Ali Haghani and Soojung Jung . « A dynamic vehicle routing problem with time-dependent travel times ». *Computers & Operations Research*, 32, pages 2959–2986, November 2005.
- [Haines et al. 1982] G. H. Haines and J. Wolff . « Alternative Approaches to Demand Responsive Scheduling Algorithms ». *Transportation Research Part A*, 16(1), pages 43–54, 1982.
- [Hajek 1988] B. Hajek . « Cooling schedules for optimal annealing ». *MOR* 13, pages 311–329, 1988.
- [Heidemann et al. 1996] D. Heidemann and Jean-Baptiste Lesort . « A queueing theory approach to speed-flow-density relationships ». In ROYAUME-UNIETATS-UNIS (1996) (Monographie) Pergamon Press , Oxford, editor, *Internaional symposium on transportation and traffic theory No13, Lyon, FRANCE*, pages 103–118, Lyon, 1996. Internaional symposium on transportation and traffic theory No13, Lyon , FRANCE.
- [Hill et al. 1992] Arthur V Hill and W. C Benton . « Modeling intra-city time-dependent travel speeds for vehicle scheduling problems ». *Journal of Operational Research Society*, 43(4), pages 343–351, April 1992.
- [Holland 1975] J. Holland . « Adaptation in natural and artificial systems ». *University of Michigan Press, Ann Arbor, Canberra ACT 2601, Australia*, 1975.
- [Homberger et al. 1999] J. Homberger and H. Gehring . « Two Evolutionary Meta-heuristics for the Vehicle Routing Problem with Time Windows ». *INFOR* 37, pages 297–318, 1999.
- [Housroum 2005] Haiyan Housroum . « *Une approche génétique pour la résolution du problème VRPTW dynamique* ». Phd thesis, Université d’Artois, May 2005.
- [Ichoua et al. 2003] Soumia Ichoua , Michel Gendreau , and Jean-Yves Potvin . « Vehicle dispatching with time-dependent travel times ». *European Journal of Operational Research*, 144, pages 379–396, 2003.
- [Irani et al. 2004] Sandy Irani , Xiangwen Lu , and Amelia Regan . « On-Line Algorithms for the Dynamic Traveling Repair Problem ». *Journal of Scheduling*, 7(3), pages 243–258, 2004.
- [Jacobs-Blecha et al. 1993] C. Jacobs-Blecha and M. Goetschalckx . « The Vehicle Routing Problem with Backhauls : An Optimization Based Approach ». *Proceedings of the 2nd Industrial Engineering Research Conference*, pages 504–509, 1993.

- [Jörnsten et al. 1986] K. Jörnsten , O. Madsen , and B. Sorensen . « Exact solution of the Vehicle Routing and Scheduling Problem with Time Windows by Variable Splitting ». *Research Report 5/1986, IMSOR, The Technical University of Denmark, Lyngby, Denmark*, 1986.
- [Jung et al. 2001] Soojung Jung and Ali Haghani . « Genetic Algorithm for the Time-Dependent Vehicle Routing Problem ». *Transportation Research Record*, 1771, pages 164–171, 2001.
- [Kerbache et al. 2005] Laoucine Kerbache and Tom Van Woensel . « Planning and Scheduling Transportation Vehicle fleet in a Congested Traffic Environment ». *Supply Chain Management - European Perspectives*, (803), pages 262–284, December 2005.
- [Kilby et al. 1998a] P. Kilby , P. Prosser , and P. Shaw . « Dynamic VRPs : A Study of Scenarios ». *CSIRO, Canberra ACT 2601, Australia*, September 1998.
- [Kilby et al. 1998b] P. Kilby , P. Prosser , and P. Shaw . « Guided Local Search for the Vehicle Routing Problem with Time Windows ». *META-HEURISTICS Advances and Trends in Local Search Paradigms for Optimization, Kluwer Academic Publishers, Boston*, pages 473–486, 1998.
- [Kohl et al. 1997] N. Kohl and O. Madsen . « An optimization algorithm for the Vehicle Routing Problem with Time Windows based on Lagrangean Relaxation ». *Operations Research 45 (3)*, pages 395–406, 1997.
- [Kolen et al. 1987] A. Kolen , H. Rinoy Kan , and H. Trienekens . « Vehicle Routing with Time Windows ». *Operations Research 35*, pages 266–273, 1987.
- [Kolesar 1975] Peter Kolesar . « A Model for Predicting Average Fire Engine Travel Times ». *Operations Research*, 23(4), pages 603–613, 1975.
- [Kontoravdis et al. 1992] G. Kontoravdis and J. Bard . « Improved heuristics for the Vehicle Routing Problem with Time Windows ». *Working paper, Operations Research Group, University of Texas at Austin, Austin*, 1992.
- [Kontoravdis et al. 1995] G. Kontoravdis and J. Bard . « A GRASP for the Vehicle Routing Problem with Windows ». *ORSA Journal on Computing 7*, pages 10–23, 1995.
- [Koskosidis et al. 1992] Y.A. Koskosidis , W.B. Powell , and M. Solomon . « An optimization-based heuristic for Vehicle Routing and Scheduling with soft Time Window Constraints ». *Transportation Science 26*, pages 69–85, 1992.
- [Krumke et al. 2003] S.O. Krumke , W.E. de Paepe , D. Poensgen , and L. Stougie . « News from the Online Traveling Repairman ». *Theoretical Computer Science 295*, pages 279–294, 2003.
- [Laporte et al. 1986] G. Laporte , H. Mercure , and Y. Nobert . « A exact algorithm for the asymmetrical capacitated vehicle routing problem ». *Networks, 16*, pages 33–46, 1986.
- [Laporte et al. 1989] G. Laporte , M. Dror , and P. Trudeau . « Vehicle routing with stochastic demands : properties and solution frameworks ». *Transportation Science, 23*, pages 166–176, 1989.

-
- [Larsen et al. 2004] A. Larsen , O. Madsen , and M. Solomon . « The A-priori Dynamic Traveling Salesman Problem with Time Windows ». *Transportation Science*, 38 (4), pages 459–472, 2004.
- [Larsen 2001] Allan Larsen . « *The Dynamic Vehicle Routing Problem* ». Thèse, Informatics and Mathematical Modelling, Technical University of Denmark, IMM-PHD-2000-73, Lyngby, 2001.
- [Larson et al. 1981] Richard C. Larson and Amedeo R. Odoni . *Urban Operations Research*. Dynamic Ideas, 1981.
- [Lau et al. 2003] H.C. Lau , M. Sim , and K.M. Teo . « Vehicle routing problem with time windows and a limited number of vehicles ». *European Journal of Operational Research* 148, pages 559–569, 2003.
- [LeBouthillier 2000] A. LeBouthillier . « Modélisation UML pour une architecture coopérative appliquée au problème de tournées de véhicules avec fenêtres de temps ». Technical Report, Département d’informatique et de recherche opérationnelle, Faculté des arts et des sciences, Université de Montréal, Canada, Avril 2000.
- [Li et al. 2004] F. Li , B.L. Golden , and E.A. Wasil . « Very large-scale vehicle routing : New test problems, algorithms, and results ». *Computers & Operations Research* 32, pages 1165–1179, 2004.
- [Lin 1965] S. Lin . « Computer solutions of the traveling salseman problem ». *Bell System Technical Journal* 44, pages 2245–2269, 1965.
- [Lund et al. 1996] K. Lund , O. Madsen , and J. M. Rygaard . « Vehicle Routing Problems with Varying Degrees of Dynamism ». Technical Report, IMM, The Department of Mathematical Modelling, Technical University of Denmark, 1996.
- [Madsen et al. 1995] Oli B.G. Madsen , K. Tosti , and J. Vælds . « A Heuristic Method for Dispatching Repair Men ». *Operations Research*, 61, pages 213–226, 1995.
- [Madsen 1990] O.B.G. Madsen . « Lagrangean Relaxation and Vehicle Routing ». *Working paper. IMSOR, The Technical University of Denmark, Lyngby, Denmark*, 1990.
- [Malandraki et al. 1992] C Malandraki and MS Daskin . « Time dependent vehicle routing problems : formulations, properties and heuristic algorithms ». *Transportation Science*, 26, pages 185–200, 1992.
- [Malandraki et al. 1996] Chryssi Malandraki and Robert B. Dial . « A restricted dynamic programming heuristic algorithm for the time dependent traveling salesman problem ». *European Journal of Operational Research*, 90, pages 45–55, April 1996.
- [Mechti 1995] R. Mechti . « Tournées de véhicules à la demande : problèmes et méthodes ». Technical Report, Laboratoire PRiSM, Université de Versailles-St. Quentin, France, 1995.
- [Mester et al. 2005] D. Mester and O. Bräysy . « Active guided evolution strategies for the large scale vehicle routing problems with time windows ». *Computers & Operations Research* 32, pages 1593–1614, 2005.

- [Michalewicz 1994] Z. Michalewicz . *Genetic Algorithms + Data Structures = Evolution Programs*. Berlin : Springer-Verlag, seconde édition, 1994.
- [Miller 1995] D.L Miller . « A matching based exact algorithm for capacitated vehicle routing problems ». *ORSA Journal on Computing*, 7, pages 1–9, 1995.
- [Mitrovic-Minic et al. 2004] S. Mitrovic-Minic and G. Laporte . « Waiting Strategies for the Dynamic Pickup and Delivery Problem with Time Windows ». *Transportation Research Part B*, 38, pages 635–655, 2004.
- [Mole et al. 1976] R.H. Mole and S.R. Jameson . « A sequential route-building algorithm employing a generalized saving criterion ». *Operational Research* 27, pages 503–511, 1976.
- [Montamenni et al. 2003] R. Montamenni , L.M. Gambardella , A.E. Rizzoli , and A.V. Donati . « A new algorithm for a Dynamic Vehicle Routing Problem based on Ant Colony System ». *IDSIA-23-02, presented at ODYSSEUS 2003*, 2003.
- [Özlem Ergun et al. 2006] Özlem Ergun , James B. Orlin , and Abran Steele-Feldman . « Creating very large scale neighborhoods out of smaller ones by compounding moves ». *Journal of Heuristics*, 12(1-2), pages 115–140, 2006.
- [Park et al. 1997] Y.-B. Park and S.-H. Song . « Vehicle Scheduling Problems with Time-Varying Speed ». *Computers & Industrial Engineering* 33, pages 853–856, 1997.
- [Pereira et al. 2002] F. Pereira , J. Tavares , P. Machado , and E. Costa . « GVR : a New Genetic Representation for the Vehicle Routing Problem ». *Artificial Intelligence and Cognitive Science : 13th, Irish Conference Proceedings*, pages 95–102, 2002.
- [Platzman et al. 1983] L.K. Platzman and J.J. Bartholdi . « Spacefilling Curves and the Planar Travelling Salesman Problem ». Technical Report, PDRC Technical Report 83-02, Georgia Institute of Technology, 1983.
- [Potvin et al. 1993] J.-Y. Potvin and J.-M. Rousseau . « A parallel route building algorithm for the Vehicle Routing and Scheduling Problem with Time Windows ». *European Journal of Operational Research* 66, pages 331–340, 1993.
- [Potvin et al. 1995] J-Y Potvin and J-M. Rousseau . « An exchange heuristic for routing problems with time windows ». *Journal of the Operational Research Society*, 46, pages 1433–1446, 1995.
- [Potvin et al. 1996a] J.-Y. Potvin and S. Bengio . « The Vehicle Routing Problem with Time Windows - part 2 : genetic search ». *INFORMS Journal on Computing* 8, pages 165–172, 1996.
- [Potvin et al. 1996b] J.-Y. Potvin , T. Kervahut , B.L. Garcia , and J.-M. Rousseau . « The Vehicle Routing Problem with Time Windows - part 1 : tabu search ». *ORSA Journal on Computing* 8 (2), pages 158–164, 1996.
- [Potvin et al. 2006] Jean-Yves Potvin , Ying Xu , and Ilham Benyahia . « Vehicle routing and scheduling with dynamic travel times ». *Computers & Operations Research*, 33, pages 1129–1137, April 2006.

-
- [Powell et al. 1995] W.B. Powell , P. Jaillet , and A. Odoni . « *Operations Research and Management Science, Vol 8, Network Routing* », Chapitre Stochastic and Dynamic Networks and Routing, pages 141–295. North-Holland, North-Holland : Amsterdam, 1995.
- [Prins 2002] C. Prins . « Efficient heuristics for the heterogeneous fleet multitrip vehicle routing problem ». *Journal of Mathematical Modelling and Algorithms*, 1(2), pages 135–150, 2002.
- [Prins 2004] C. Prins . « A simple and effective evolutionary algorithm for the vehicle routing problem ». *Computers and Operations Research*, 31(12), pages 1985–2002, 2004.
- [Psaraftis 1988] Harilaos N Psaraftis . « *Vehicle Routing : Methods and Studies* », Chapitre Dynamic Vehicle Routing Problem, pages 223–248. Elsevier Science Publishers B.V, 1988.
- [Psaraftis 1995] Harilaos N. Psaraftis . « Dynamic vehicle routing : status and prospects ». *Annals of Operations Research*, 61(1), pages 143–164, 12 1995.
- [Ratliff et al. 1999] H Donald Ratliff and Xinglong Zhang . « Estimating traveling time/speed ». *Journal of Business Logistics*, 20(2), pages 121–139, 1999.
- [Rego et al. 1994] C. Rego and C. Roucairol . « Le Problème de Tournées de Véhicules : Etude et Résolution Approchée ». Technical Report, INRIA, Février 1994.
- [Rego 2001] C. Rego . « Node ejection chains for the Vehicle Routing Problem : Sequential and Parallel Algorithms ». *Parallel Computing* 27 (3), pages 201–222, 2001.
- [Reimann et al. 2004] M. Reimann , K. Doerner , and R.F. Hartl . « D-Ants : Savings based ants divide and conquer the vehicle routing problem ». *Computers & Operations Research*, 31(4), pages 563–591, 2004.
- [Rochat et al. 1995] Y. Rochat and E. Taillard . « Probabilistic diversification and intensification in local search for Vehicle Routing ». *Journal of Heuristics* 1, pages 147–167, 1995.
- [Russell 1995] R.A. Russell . « Hybrid heuristics for Vehicle Routing Problem with Time Windows ». *Transportation Science* 29, pages 156–166, 1995.
- [Savelsbergh 1988] M.W.P. Savelsbergh . « *Computer Aided Routing* ». PhD thesis, Centre for Mathematics and Computer Science, Amsterdam, 1988.
- [Savlesberg et al. 1991] M. Savlesberg and M. Sol. . « DRIVE : Dynamic routing of independent vehicles ». *Operations Research*, 46(4), pages 474–490, 1991.
- [Séguin et al. 1997] René Séguin , Jean-Yves Potvin , Michel Gendreau , Teodor G. Crainic , and Patrice Marcotte . « Real-Time Decision Problems : an Operations Research Perspective ». *Journal of the Operational Research Society*, 48, pages 162–174, February 1997.
- [Solomon 1986] M.M. Solomon . « On the worst-case performance of some heuristics for the Vehicle Routing and Scheduling Problem with Time Window Constraints ». *Networks* 16, pages 161–174, 1986.

- [Solomon 1987] M.M. Solomon . « Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints ». *Operations Research* 35, pages 254–265, 1987.
- [Sushil et al. 1999] J. L. Sushil , X. Yin , and Z. Y. Yuan . « Multiple Vehicle Routing with Time Windows Using Genetic Algorithms ». *In Proceedings of IEEE Congress on Evolutionary Computation. Piscataway : IEEE Press*, pages 1804–1808, 29 January 1999.
- [Swihart et al. 1999] Michael R. Swihart and Jason D. Papastavrou . « A stochastic and dynamic model for the single-vehicle pick-up and delivery problem ». *European Journal of Operational Research*, 114(3), pages 447–464, May 1999.
- [Taillard et al. 1997] E. Taillard , P. Badeau , M. Gendreau , F. Guertin , and J.-Y. Potvin . « A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows ». *Transportation Science* 31, pages 170–186, 1997.
- [Taillard 1994] E. Taillard . « Parallel iterative search methods for vehicle routing problems ». *NETWORKS* 23(8), pages 661–673, 1994.
- [Taillard 1999] E. Taillard . « A heuristic column generation method for the heterogeneous fleet VRP ». *RAIRO-Operations Research*, 33, pages 1–14, 1999.
- [Tan et al. 2001] K.C. Tan , L.H. Lee , and K. Ou . « Hybrid Genetic Algorithms in Solving Vehicle Routing Problems with Time Window Constraints ». *Asia-Pacific Journal of Operational Research* 18, pages 121–130, 2001.
- [Tarantilis et al. 2002] C.-D. Tarantilis and C.T. Kiranoudis . « Bone Route : An adaptive memory-based method for effective fleet management ». *Annals of Operations Research*, 115, pages 227–241, 2002.
- [Tavares et al. 2003] J. Tavares , F. B. Pereira , P. Machado , and E. Costa . « Crossover and Diversity : A Study about GVR ». *In Proceedings of the Analysis and Design of Representations and Operators (ADoRo'2003) a bird-of-a-feather workshop at the 2003 Genetic and Evolutionary Computation Conference (GECCO-2003), Chicago, Illinois, USA, 12-16 July 2003*.
- [Thangiah et al. 1994] S.R. Thangiah , I. Osman , and T. Sun . « Hybrid Genetic Algorithm, Simulated Annealing and Tabu Search Methods for Vehicle Routing Problems with Time Windows ». *Working Paper UKC/IMS/OR94/4, Institute of Mathematics and Statistics, University of Kent, Canterbury*, 1994.
- [Thangiah 1995] S.R. Thangiah . « Vehicle Routing with Time Windows using genetic algorithms ». *Handbook of Genetic Algorithms : new frontiers, Vol. 2*, pages 253–277, 1995.
- [Toth et al. 2002] P. Toth and D. Vigo . *The Vehicle Routing Problem*. SIAM - Monographs on Discrete Mathematics and Applications, 2002.
- [Toth et al. 2003] P. Toth and D. Vigo . « The granular tabu search and its application to the vehicle routing problem ». *INFORMS Journal on Computing*, 15, pages 333–346, 2003.

-
- [Vandaele et al. 2000] Nico Vandaele , Tom Van Woensel , and Aviel Verbruggen . « A queueing based traffic flow model ». *Transportation Research Part D : Transport and Environment*, 5, pages 121–135, March 2000.
- [Woensel et al. 2001] T. Van Woensel , Ruth Creten , and Nico Vandaele . « Managing the environmental externalities of traffic logistics : the issue of emissions ». *Production and Operations Management*, Summer, pages 207–223, 2001.
- [Woensel et al. 2002] T. Van Woensel , H. Peremans , and N. Vandaele . « A Vehicle Routing Problem with Stochastic Travel Times ». In *Forthcoming Proceedings of the 4th Aegean International Conference on Design and Analysis of Manufacturing Systems*, Samos, Greece, June 2002.
- [Woensel et al. 2008] T. Van Woensel , L. Kerbacheb , H. Peremansc , and N. Vandaeled . « Vehicle routing with dynamic travel times : A queueing approach ». *European Journal of Operational Research*, 186(3), pages 990–1007, may 2008.
- [Zhu 2000] K. Q. Zhu . « A new genetic algorithm for VRPTW ». *International Conference on Artificial Intelligence, Las Vegas, USA*, April 13 2000.

Publications personnelles

- [Zhao et al. 2007a] Xin ZHAO, Rémy DUPAS, Tienté HSU and Gilles GONCALVES. « Tournées du réparateur avec temps de trajet dépendant du temps ». *Conférence conjointe FRANCORO V/ROADEF 2007*, pages 447–448, 2007.
- [Zhao et al. 2007b] Xin ZHAO, Rémy DUPAS, Tienté HSU and Gilles GONCALVES. « Repairman routing problem with time-dependent travel times ». *International Conference on Artificial Intelligence and Systems*, Vol. 4, pages 28–34, 2007.
- [Zhao et al. 2008a] Xin ZHAO, Gilles GONCALVES, Tienté HSU and Rémy DUPAS. « Un problème de tournées de véhicules avec temps de trajet dépendant du temps ». *7ème Conférence Internationale de Modélisation et Simulation, MOSIM'08, Vol. 2*, pages 1067–1076, 2008.
- [Zhao et al. 2008b] Xin ZHAO, Gilles GONCALVES and Rémy DUPAS. « A Genetic Approach to Solving the Vehicle Routing Problem with Time-Dependent Travel Times ». *16th IEEE Mediterranean Conference on Control and Automation, MED'08*, pages 413–418, 2008.
- [Zhao et al. 2008c] Xin ZHAO, Gilles GONCALVES and Rémy DUPAS. « A genetic algorithm to solving the dynamic vehicle routing problem with time windows and real-time travel times ». *The 2nd International Conference on Metaheuristics and Nature Inspired Computing, META'08*, 2008.
- [Zhao et al. 2009] Xin ZHAO, Gilles GONCALVES and Rémy DUPAS. « Un problème de tournées de véhicules temps réel avec temps de parcours variables ». *La dixième conférence de la Société Française de Recherche Opérationnelle et d'Aide à la Décision, ROADEF 2009*, 2009.

Index

- évènements, 86
- 1-Opt, 80
- 2-Opt, 80
- aléatoire, 77
- Algorithmes génétiques, 18
- arrivée d'un nouveau client, 88
- Branch (&) Cut, 15
- Branch & Bound, 14
- Charles Darwin, 70
- chromosome, 70, 73
- Clarke (&) Wright, 15
- Classification, 27
- colonie de fourmis, 23
- composant AG, 82
- composant GE, 83
- contraintes, 65
- croisement, 20, 77
- déterministe, 28
- degré de dynamisme, 32
- direct, 70, 72
- DVRP, 11, 26
- dynamique, 28
- Elitisme, 20, 81
- fin de service, 86
- fitness, 70
- Fonction, 73
- fonctionnement du simulateur, 83
- Formulation, 8, 65
- Heuristiques, 15
- indirect, 70, 72
- insertion, 74, 88
- Insertion séquentielle, 16
- interactions, 84
- m-VRP, 12
- Métaheuristiques, 17
- méthodes approchées, 67
- méthodes de résolution, 12
- méthodes exactes, 66
- MDVRP, 11
- mutation, 20, 80
- MX1T/D, 78
- Or-Opt, 81
- OVRP, 12
- PFIH, 76
- PMX, 77
- population, 70, 76
- Programmation Dynamique, 15
- PVRP, 11
- ranking, 76
- Recherche locale, 17
- Recherche Tabou, 22
- Recuit simulé, 21
- roulette, 76
- Sélection, 20, 76
- Simulation, 82
- solution, 70
- statique, 28
- stochastique, 11, 28
- SVRP, 11
- swap, 80
- tournée, 73
- tournoi, 77
- TRP, 11
- TSP, 11
- Variantes, 10
- VRPB, 11
- VRPHF, 12
- VRPPD, 12
- VRPSD, 12
- VRPTW, 10

Résumé

Le problème de tournées de véhicules (PTV ou VRP pour Vehicle Routing Problem) est un problème rencontré en logistique qui se situe à un niveau important : la planification opérationnelle qui concerne l'optimisation d'une flotte de véhicules. C'est un nom générique pour désigner une classe de problèmes dans laquelle un certain nombre de clients doit être visité par un certain nombre de véhicules. Les tournées de véhicules doivent être organisées de façon à équilibrer ou à optimiser un certain nombre de critères tels que la distance totale parcourue, le nombre de véhicules utilisés et les temps d'attente des clients.

Le PTV est l'un des problèmes les plus populaires et donc les plus étudiés de la communauté Recherche Opérationnelle. Il y a de nombreuses variantes de ce problème intégrant une ou plusieurs des contraintes suivantes : capacité des véhicules, fenêtres de temps pour visiter les clients, etc. Le VRP permet de modéliser de nombreux types d'applications. Les exemples peuvent concerner la conception, la reconfiguration d'un réseau de transport, ainsi que la gestion quotidienne de la collecte et de la livraison de marchandises, organisation des services de messageries ou de soins à domicile.

La majorité des études effectuées sur le VRP suppose que les données sont parfaitement connues et disponibles au moment où l'on effectue la planification des tournées. Dans la réalité, nous avons beaucoup de causes externes qui peuvent remettre en question cette hypothèse : congestions du réseau routier, pannes de véhicule, arrivées de nouveaux clients, etc.

Dans nos travaux, nous nous sommes placés dans un cadre dit dynamique c'est-à-dire que nous considérons que des informations nouvelles vont apparaître au cours du temps et influencer sur la planification en cours. Les problèmes de décision en temps réel jouent un rôle de plus en plus important car les technologies de communication et d'information permettent aujourd'hui d'obtenir et de traiter rapidement l'information en temps réel. Par conséquent, nous traitons plus précisément le problème de l'élaboration dynamique de tournées de véhicules avec fenêtres de temps (DVRPTW) et le problème de la tournée du réparateur partiellement dynamique (PDTRP) où la prise en compte de nouveaux clients en cours de l'exécution des tournées est possible. De plus, dans une résolution classique de ce problème, les valeurs moyennes issues d'historiques sont généralement considérées pour déterminer le temps de trajet entre deux clients. On suppose généralement qu'elles ne sont pas sujettes à des variations stochastiques, la vitesse est considérée constante pour toute la journée.

Dans une première approche, nous considérons des profils de vitesse basés sur des temps de trajet moyens qui varient selon la période de temps (matin, midi, après midi) pour tenir compte de l'évolution du trafic routier. En fonction du type de connexion entre les deux clients, plusieurs profils peuvent être considérés. Dans une seconde approche, nous intégrons des informations trafic obtenues en temps réel pour modifier le profil de vitesse par rapport aux aléas du réseau routier (congestions, etc.). Nous adoptons un lissage exponentiel amélioré pour prévoir le temps de trajet en fonction des vitesses mesurées sur le parcours considéré.

Mots clés : Problème de tournées de véhicules, PTV, VRPTW, PDTRP, Algorithme génétique, simulations, temps réel, temps de trajets dépendant du temps, profil de vitesse

Abstract

We treat the dynamic vehicle routing problem with time windows (DVRPTW) and the partially dynamic travelling repairman problem (PDTRP) where the inclusion of new customers during the routing execution is possible. In a first approach, we consider speed profiles based on average travel time depending on the time period (morning, noon, afternoon) to reflect the evolution of road traffic. Depending on the connection type between two customers, several profiles can be considered (feeder road, arterial road and freeway). In a second approach, we integrate traffic information obtained in real time to change the speed profile according to the accidents of the road network (congestion, etc.). The analysis of the results of these experiments shows that our method with real-time traffic information provides a good performance, a better robustness against a simple model with time dependent travel time.

Keywords : Vehicle routing problem, VRP ; VRPTW, PDTRP, Genetic algorithm, Simulation, Real time, Time-dependent travel time, Speed profile

