



**HAL**  
open science

# Metaheuristics approaches to solve some variants of the Heterogeneous Fixed fleet Vehicle Routing Problems

Jalel Euchi

► **To cite this version:**

Jalel Euchi. Metaheuristics approaches to solve some variants of the Heterogeneous Fixed fleet Vehicle Routing Problems. Artificial Intelligence [cs.AI]. Université du Havre, 2011. English. NNT : . tel-01273816

**HAL Id: tel-01273816**

**<https://theses.hal.science/tel-01273816>**

Submitted on 14 Feb 2016

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

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



# THESE DE DOCTORAT



## Metaheuristics approaches to solve some variants of the Heterogeneous Fixed fleet Vehicle Routing Problems

Présentée et soutenue publiquement le 30 mai 2011 par :

**Jalel EUCHI**

En vue de l'obtention du grade de :

**Docteur de l'Université de Sfax, Spécialité « Méthodes Quantitatives »**

et

**Docteur de l'Université du Havre, Spécialité « Mathématiques Appliquées et Informatique »**

Sous la direction de :

**Monsieur Habib CHABCHOUB** Professeur à l'université de Sfax  
**Monsieur Adnan YASSINE** Professeur à l'université du Havre

Thèse présentée devant le jury composé de :

*Présidente :* **Mme Taïcir LOUKIL**, Prof. Ens. Sup à la FSEGS-Sfax

*Rapporteurs :* **M. Cyril FONLUPT**, Professeur à l'université du Littoral

**M. Talel LADHARI**, Maître de Conférences à l'ESSEC-Tunis

*Examineur :* **M. Jaouad BOUKACHOUR**, Maître de Conférences HDR à l'université du Havre



*Laboratoire de Gestion Industrielle et Aide à la Décision (GIAD),  
Faculté des Sciences Economiques et de Gestion de Sfax, Tunisie*



*Laboratoire de Mathématiques Appliquées du Havre (LMAH),  
Université du Havre, UFR Sciences et Techniques du Havre, France*

---

# Remerciements

Cette page est de loin la plus difficile à écrire, principalement parce que c'est la plus ingrate. Il m'est impossible en effet d'énumérer de manière exhaustive toutes les personnes qui ont contribué au succès du long cycle de mes études supérieures. Je me contenterai donc d'en citer quelques unes, tout en m'excusant sincèrement auprès de celles qui en sont absentes.

Je tiens tout d'abord à exprimer ma reconnaissance à mes directeurs de recherche, M. Habib Chabchoub professeur à l'université de Sfax, directeur de l'IHEC et directeur de GIAD et M. Adnan Yassine professeur à l'université du Havre et directeur de LMAH pour leurs implications à la réalisation de ce travail de recherche et pour le support qu'ils m'ont apporté. Je tiens tout particulièrement à les remercier pour leurs aides et leurs précieux conseils tout au long de ces années.

Mes remerciements vont ensuite aux membres du jury :

À Mme Taicir LOUKIL, Professeur à l'université de Sfax qui a accepté de présider cette thèse.

Mes remerciements vont également à M. Cyril FONLUPT, Professeur à l'université du Littoral, et à M. Talel Ladhari, Maître de Conférences HDR à l'université de Tunis pour avoir acceptés d'être rapporteurs de cette thèse.

Je remercie aussi M. Jaouad BOUKACHOUR, Maître de Conférences HDR à l'Université du Havre, d'avoir accepté de participer à mon jury.

Ce travail est issu d'une collaboration entre le Laboratoire de Gestion Industrielle et Aide à la Décision (GIAD) de Sfax (Tunisie) et le Laboratoire de Mathématiques Appliquées du Havre (LMAH) (France). Je tiens à remercier M. Adnan et M. Chabchoub qui ont permis cette co-tutelle, en espérant que celle-ci marque le début d'une collaboration

---

des plus fructueuses entre les deux laboratoires.

Mes remerciements vont aussi à toute de l'équipe de LMAH je tiens à lui exprimer toute ma gratitude pour m'avoir accueilli au sein du laboratoire.

Je me dois également de remercier mes parents qui, par leurs encouragements et leur soutien, m'ont permis d'aller de l'avant dans mes projets.

---

**Résumé :** Cette thèse étudie une problématique d'optimisation concernant la gestion de la distribution au niveau de la chaîne logistique. Elle aborde les problèmes de tournées de véhicules avec flotte limitée hétérogène, à travers le développement de stratégies de résolution avec des méthodes heuristiques. Tenant compte des contraintes de capacité et de limitation du nombre de la flotte hétérogène de véhicules disponibles, nous avons étudié trois variantes des problèmes de tournées de véhicules : le problème de tournées avec flotte limitée hétérogène, le problème de tournées de véhicules avec flotte limitée et transporteur externe et le problème de tournées de véhicules dynamiques. Nous avons appliqué des algorithmes de recherche tabou, des algorithmes évolutionnaires et l'algorithme de colonie de fourmis pour apporter des solutions efficaces à ces différents problèmes.

**Mots clés :** *Logistique, Transport, Problèmes des tournées de véhicules, Flotte limitée, Métaheuristiques.*

**Abstract:** This thesis investigates an optimization problem concerning the distribution management in the supply chain. It addresses the Vehicle Routing Problems (VRP) with Heterogeneous Limited Fleet, through the development of resolution strategies with heuristics methods. Taking into account capacity constraints and the limited number of heterogeneous fleet of vehicles available, we studied three variants of vehicle routing problems: the Heterogeneous Fixed Fleet Vehicle Routing Problem (HFFVRP), the Vehicle Routing Problem with Private fleet and common Carrier (VRPPC) and the Dynamic Vehicle Routing Problem (DVRP). We have applied Tabu search algorithms, evolutionary algorithms and ant colony algorithm to provide effective solutions to these problems.

**Keywords:** *Logistics, Transportation, Vehicle Routing Problem, Limited fleet, Metaheuristics.*

---

# Contents

<b>List of Figures</b>	<b>ix</b>
<b>List of Tables</b>	<b>xi</b>
<b>List of Algorithms</b>	<b>xiii</b>
<b>Introduction générale</b>	<b>1</b>
<b>1 Introduction</b>	<b>7</b>
1 Research Objectives of the thesis . . . . .	8
2 Contributions . . . . .	11
<b>2 Literature review</b>	<b>15</b>
1 Introduction . . . . .	15
2 Related works . . . . .	16
3 Travelling Salesman Problem (TSP) . . . . .	18
4 Vehicle Routing Problem (VRP) . . . . .	19
4.1 Distribution network . . . . .	19
4.2 Mathematical model . . . . .	20
4.2.1 Parameters . . . . .	21
4.2.2 Decisions variables . . . . .	21
4.2.3 Formulation . . . . .	21
5 Variants of the vehicle routing problem . . . . .	22
5.1 Capacitated vehicle routing problem . . . . .	23

5.2	Multi-Period Vehicle Routing Problem . . . . .	23
5.2.1	Period Vehicle Routing Problem . . . . .	24
5.2.2	Inventory Routing Problem . . . . .	25
5.3	Vehicle Routing Problem with limited fleet . . . . .	25
5.3.1	M-Vehicle Routing Problem with Time Windows (SATISFIABLE) . . . . .	26
5.3.2	M-Vehicle Routing Problem with Time Windows (DISSATISFYING) . . . . .	26
5.3.3	M-Heterogeneous Vehicle Routing Problem (M-HVRP) . . . . .	27
5.4	Vehicle Routing Problem with full truckload . . . . .	27
5.4.1	Vehicle Routing Problem with return truckload . . . . .	28
5.4.2	VRP with pick up and delivery with full truckload . . . . .	28
5.5	Vehicle Routing Problem with Profit . . . . .	28
5.6	Vehicle Routing Problem with Private fleet and Common carrier . . . . .	29
5.7	Dynamic Vehicle Routing Problem . . . . .	30
6	An overview of proposed approaches for the VRP . . . . .	31
6.1	Metaheuristics optimization via memory . . . . .	32
6.1.1	Tabu Search algorithm . . . . .	32
6.1.2	Variable Neighborhood Search algorithm . . . . .	33
6.1.3	Ejection chains neighborhood . . . . .	33
6.1.4	Adaptive memory . . . . .	34
6.2	Evolutionary algorithms metaheuristics . . . . .	34
6.2.1	Genetic algorithm . . . . .	34
6.2.2	Iterated Density Estimation Evolutionary Algorithms . . . . .	36
6.2.3	Artificial Ant Colony . . . . .	37
7	Our personal contributions to solve some variants of the VRP problems . . . . .	37
7.1	First line of research: A Hybrid Tabu Search to Solve the Heterogeneous Fixed Fleet Vehicle Routing Problem . . . . .	40
7.2	Second line of research: Heuristic Search Techniques to Solve the Vehicle Routing with Private Fleet and Common Carrier . . . . .	40
7.3	Third line of research: Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier. . . . .	41
7.4	Fourth line of research: Solving the Dynamic Vehicle Routing Problem by means of Artificial Ant Colony. . . . .	41

8	Conclusion . . . . .	41
<b>3</b>	<b>M-Heterogeneous Vehicle Routing Problem</b>	<b>43</b>
1	Introduction . . . . .	43
2	Literature review . . . . .	45
3	Heterogeneous Vehicle Routing Problem . . . . .	47
3.1	Description of the problem . . . . .	47
4	Vehicle Routing Problem with Private fleet and Common carrier . . . . .	48
4.1	Formulation of the VRPPC . . . . .	48
4.1.1	Index . . . . .	48
4.1.2	Parameters . . . . .	49
4.1.3	Decision variances . . . . .	49
5	Metaheuristics approaches to solve the M-VRP . . . . .	50
5.1	A Hybrid Tabu Search to solve the Heterogeneous Fixed Fleet Vehicle Routing Problem . . . . .	50
5.1.1	Initialization (Step 1 in the Adaptive Memory) . . . . .	51
5.1.2	Construction of solution (step 2 in the Adaptive Memory) . . . . .	53
5.1.3	Solution improvements (Step 3 in the Adaptive Memory) . . . . .	53
5.1.4	Updating the Adaptive Memory (Step 4 in the Adaptive Memory) . . . . .	55
5.2	Computational results . . . . .	56
5.2.1	Implementation and instances . . . . .	56
5.2.2	Parameter settings . . . . .	59
5.2.3	Evaluation method . . . . .	59
5.3	Heuristic Search Techniques to Solve the Vehicle Routing with Private Fleet and Common Carrier . . . . .	63
5.3.1	Initial solution . . . . .	63
5.3.2	Neighborhood Structure . . . . .	63
5.3.3	Tabu List . . . . .	68
5.3.4	Aspiration criterion . . . . .	68
5.3.5	Ejection chains . . . . .	68
5.3.6	Intensification . . . . .	69
5.3.7	Diversification . . . . .	69
5.3.8	Stopping condition . . . . .	70

5.4	Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the VRPPC . . . . .	72
5.4.1	Hybrid IDEA to solve the VRPPC . . . . .	73
5.4.2	Vehicle routing representation . . . . .	73
5.4.3	2-opt Local Search . . . . .	74
5.4.4	Initialization . . . . .	75
5.4.5	Selection operators . . . . .	76
5.4.6	Probabilistic Model . . . . .	76
5.4.7	Replacement . . . . .	77
5.4.8	Stopping criterion . . . . .	77
5.5	Experiments results . . . . .	78
5.5.1	Implementation and instances . . . . .	78
5.5.2	Parameter settings . . . . .	78
5.5.3	Evaluation method . . . . .	82
6	Conclusions . . . . .	92
<b>4</b>	<b>The Dynamic Vehicle Routing Problem</b>	<b>95</b>
1	Introduction . . . . .	95
2	Main definition of the Dynamic Vehicle Routing Problem . . . . .	97
3	Related Works . . . . .	97
3.1	Pick-up and delivery vehicle Routing Problem . . . . .	98
3.2	Dynamic Pick-up and delivery vehicle Routing Problem . . . . .	98
4	A framework of the Dynamic Pick-up and delivery vehicle Routing Problem	101
4.1	The Static Pick-up and delivery vehicle Routing Problem . . . . .	101
4.2	The Dynamic Pick-up and delivery vehicle Routing Problem . . . . .	102
4.3	Ant Colony Optimization . . . . .	103
5	Application . . . . .	105
5.1	Merging of New Event Procedure (MNEP) . . . . .	105
5.2	Artificial Ant Colony with 2-opt local search to solve the DPDVRP	106
5.2.1	Solutions representation . . . . .	107
5.2.2	Solution construction . . . . .	107
5.2.3	Pheromone Trail update . . . . .	109
5.2.4	Local Pheromone Update . . . . .	109
5.2.5	Global Pheromone Update . . . . .	110
5.2.6	Local Search . . . . .	110

5.3	Experimental Results and Performance Comparison . . . . .	112
5.3.1	Problem instances . . . . .	112
5.3.2	The experimental environment . . . . .	112
5.3.3	Results discussion . . . . .	113
5.4	Conclusion . . . . .	120
<b>5</b>	<b>Conclusion</b>	<b>121</b>
1	Research work summary . . . . .	121
2	Resulting papers . . . . .	125
	<b>Conclusion générale</b>	<b>127</b>
	<b>Annexe 1</b>	<b>133</b>
1	Appendix A. Best solutions found . . . . .	133
1.1	The best routes constructed with available vehicles . . . . .	133
	<b>Bibliography</b>	<b>143</b>

---

# List of Figures

2.1	classification of vehicle routing . . . . .	16
2.2	A feasible tour in a seven-city traveling salesman problem . . . . .	19
2.3	Representation of a vehicle Routing Problem . . . . .	20
2.4	Solution of Vehicle Routing Problem . . . . .	20
3.1	Example of solution of VRP with 9 customers . . . . .	64
3.2	1-exchange for a single customer (shift) . . . . .	64
3.3	1-exchange for a 2 customers (swap) . . . . .	65
3.4	1-exchange applying in the external transporter (shift process) . . . . .	65
3.5	1-exchange applying in the external transporter (swap process) . . . . .	66
3.6	2-exchange for a two customers (shift) . . . . .	66
3.7	2-exchange for a two customers (swap moves) . . . . .	67
3.8	2-exchange applying in the external transporter (shift process) . . . . .	67
3.9	2-exchange applying in the external transporter (swap process) . . . . .	67
3.10	General Scheme of the IDEA / 2-opt local search . . . . .	72
3.11	vehicle routing representation . . . . .	74
3.12	2-Opt intra route movement . . . . .	75
3.13	Comparison results for the homogeneous instances . . . . .	85
3.14	Comparison results for the heterogeneous instances . . . . .	85
3.15	Effect of the number of generation in the fitness value . . . . .	86
3.16	Efficiency of the IDEA/ 2-opt . . . . .	86
4.1	Solution of Static vehicle routing . . . . .	99

4.2	Dynamic vehicle routing . . . . .	99
4.3	Solution of a Dynamic vehicle routing . . . . .	100
4.4	Experiment for selection of the shortest branches by a colony of ants: (a) at the beginning of the experiment and (b) at the end of the experiment. . .	104
4.5	Solution representation . . . . .	107

---

# List of Tables

2.1	Overview of different problems . . . . .	31
3.1	Specifications of eight benchmark problems with at most six types of vehicles	57
3.2	Specifications for five new test problems with at most six types of vehicles	58
3.3	Computational results for TSAM algorithms on eight test problems . . . . .	59
3.4	A comparison of TSAM, HCG, BATA, and HRTR according to overall costs	60
3.5	Comparative result on five new test problems . . . . .	60
3.6	Percent deviation results for HFFVRP algorithms on eight test problems .	62
3.7	Characteristics of instances with homogeneous limited fleet . . . . .	79
3.8	Characteristics of the instances with heterogeneous limited fleet . . . . .	80
3.9	TS/EC parameters . . . . .	81
3.10	Comparative result for the homogeneous instances . . . . .	83
3.11	Comparative result for the heterogeneous instances . . . . .	84
3.12	Results for the homogeneous instances . . . . .	88
3.13	Results for the heterogeneous instances . . . . .	89
3.14	Best known solution for the homogeneous limited fleet instances . . . . .	90
3.15	Best known solution for the heterogeneous limited fleet instances . . . . .	91
4.1	Parameter calibration of our experimental environment . . . . .	113
4.2	Computational results for the AAC for the dynamic delivery instances . . .	114
4.3	Computational results for the AAC for the dynamic pickup instances . . .	116
4.4	Performance comparison of our proposed approach for the dynamic delivery routing instances . . . . .	118

4.5 Comparison processing time results for the dynamic routing instances . . . 119

---

## List of Algorithms

1	Pseudo code of Tabu local search . . . . .	54
2	Initial solution algorithm . . . . .	63
3	Pseudo code of Tabu Search Ejection Chains algorithm . . . . .	71
4	Pseudo code of initial solution algorithm . . . . .	75
5	Pseudo code of IDEA/2-Opt algorithm . . . . .	77
6	Pseudo code of an artificial ant colony based on 2-Opt . . . . .	106
7	Initialization algorithm . . . . .	108

---

# Introduction générale

La logistique est apparue dans les domaines militaire et civil depuis plusieurs siècles. Les calculs de besoins, de délais de transport, d'espace de ramassage et de stocks faisaient apparaître une sorte de nouvelle science que l'on n'appelait pas encore recherche opérationnelle mais qui en avait déjà un peu l'esprit et que l'on appela logistique pour en consacrer le caractère *logico-mathématique*.

L'activité de transport est le coeur même de la logistique, c'est l'un de ses postes de coûts les plus importants de telle sorte que l'organisation logistique est souvent déterminée par l'optimisation des coûts de transports. Quand on parle de transport, on pense directement à l'organisation des tournées de véhicules. Une meilleure organisation de ces dernières présente un potentiel d'économies majeur. C'est cette importance accrue des problèmes d'optimisation des tournées dans le secteur de transport qui a attiré de plus en plus les chercheurs et les gestionnaires d'entreprises. Toute entreprise moderne a son propre programme de transport.

Par exemple, Cooper et al. (1991) soutiennent que le transport et le stockage de marchandises occupent le centre de toute activité logistique. Nous rappelons que La logistique est la fonction responsable de l'écoulement de matériaux provenant des fournisseurs d'une organisation, par des opérations au sein de cette organisation, et ensuite aux clients.

De plus, au cours des dernières années, le monde a vécu une forte et spectaculaire augmentation des prix des carburants. Cette hausse est susceptible de perdurer au cours des décennies à venir. Etant donné que cela engendre un des problèmes les plus critiques quant à l'utilisation des véhicules, la gestion efficace du coût de ces carburants est devenue par la force des faits accomplis un parmi les enjeux majeurs de ce siècle.

---

Généralement, les projets de transport sont destinés à améliorer la prospérité économique et sociale des personnes. Ainsi, de nombreux pays à travers le monde investissent massivement, et de manière très importante, sur l'amélioration de ces transports. Les objectifs étant très ambitieux, des investissements plus importants sont toujours nécessaires.

Le problème de tournées de véhicules (PTV) tient une place centrale dans la gestion de la distribution. Son importance économique a incité les chercheurs universitaires et les industriels à trouver des moyens pour s'acquitter efficacement du transport de biens et services. Le premier article portant sur le problème de tournées de véhicules a été publiée vers la fin des années 1950 par Dantzig et Ramser (1959). Ainsi, la plupart des entreprises qui doivent livrer des produits à plusieurs clients sont confrontées à ce problème. Dans la littérature le PTV a été formulé en plusieurs formes comme celle présenté dans notre rapport de Fisher and Jaikumar (1981).

Le rôle de la distribution dans le modèle de gestion de chaîne d'approvisionnement s'est considérablement étendu pour s'écarter de sa vision classique. En effet, le point de vue conventionnel de l'activité traitait uniquement le transport et l'entreposage. Comme proposition de minimisation du coût du carburant, la planification optimisée des tournées des véhicules semble être une voie digne d'exploration. Cela permet de réduire le coût du transport.

Ce travail de recherche est motivé par le désir de trouver une modélisation réaliste et qui soit la plus fidèle possible de l'état réel du transport via la prise en compte de l'aspect d'hétérogénéité des véhicules (les véhicules ne sont pas identiques et ils n'ont pas tous la même capacité) et de limitation de la flotte de la plupart des industriels (chaque entreprise possède un nombre limité de véhicules). Ces aspects ont souvent été étudiés séparément par plusieurs chercheurs. Laporte (1992) a donné une revue sur les méthodes exactes pour la résolution du problème de tournées de véhicules. Toth et Vigo (2002) ont présenté quelques modèles et approches de résolution exacte. Quelques tentatives d'études couplées de ces effets ont déjà été avancées. Klinkiewicz et al. (1990) ont proposé une planification de tournées en présence du transporteur externe en se basant sur une méthode classique. Diaby et Ramesh (1995) ont traité le problème de tournées de véhicule avec transporteurs externes en utilisant un seule véhicule dont l'objectif était de décider quels clients qui seront servis par un transporteur externe et d'optimiser les clients restants. Chu (2005) et Bolduc et al. (2007, 2008) ont utilisé des heuristiques constructives pour la résolution du problème de tournées de véhicules avec flotte limitée et transporteur externe. Nous proposons de formuler des nouvelles formulations des problèmes, d'avancer quelques méthodes de résolutions (voir les travaux d'Euchi and Chabchoub (2010), Eu-

---

chi et al. (2011)), et de proposer de nouvelles méthodologies qui peuvent contribuer à l'amélioration des réflexions avancées par la communauté scientifique.

## **Objectifs de la thèse**

Le travail de recherche effectué dans cette thèse fait appel à une approche de recherche à voisinages et certaines approches méta-heuristiques pour résoudre le problème de tournées de véhicules avec flotte limitée hétérogène.

Etant donné tout le potentiel envisageable concernant les multiples possibilités de développer des liens commerciaux avec l'industrie en résolvant les problèmes de transports les concernant, nous nous sommes donnés comme défi la proposition de méthodologies et le développement de solutions pratiques optimisant les tournées de véhicules avec flotte limitée hétérogène. Afin d'atteindre cet objectif, nous avons identifié les points clés suivants:

1. Une hybridation avec la recherche tabou : Cette méthode permet à la recherche de voisinage de type chaînes d'éjection d'être intégrée dans l'algorithme tabou. Cela permet de savoir si l'intégration des voisinages de type chaînes d'éjection avec les métaheuristiques est capable de générer des solutions de bonne qualité.
2. Combiner la mémoire adaptative (AMP) avec la métaheuristique tabou. L'AMP et son mécanisme de mise à jour des solutions enregistrées permettent relativement la plus importante diversification et l'aboutissement à de bonnes solutions. Ces solutions seront enregistrées et utilisées au cours du processus de recherche. L'utilisation qui en découle aura lieu entre les différentes phases de voisinages au cours de la métaheuristique.
3. Chercher à développer une méthodologie pour étudier cette problématique. Notre approche est basée sur les métaheuristiques évolutives qui rendent possible la planification de tournées tout en déterminant les clients qui seront servi à l'aide d'un transporteur externe.
4. Utiliser l'optimisation par colonies de fourmis pour résoudre le problème de tournées de véhicules dynamique. Il s'agit de l'implémentation d'une méthode de colonie de fourmis basée sur une recherche locale de type 2-Opt. Dans cette application nous avons opté pour la résolution du problème de collecte de tournées de véhicules

---

dynamiques, fait qui jusque là semble non abordé dans les différentes sources de littérature de notre état de l'art.

5. Déterminer l'état de l'art actuel du problème de tournées de véhicules avec flotte limitée hétérogène puis explorer l'utilisation de différentes méthodes de résolution dans le cas de quelques variantes du dit problème. La voie à suivre étant l'amélioration des méthodes récentes qui traitent les problèmes fortement contraints.

Cette thèse vise l'étude de quelques problèmes d'optimisation de la logistique du transport et se rapporte à la proposition des métaheuristiques pour la résolution de ces différents problèmes. Cela se fait par l'étude des problématiques d'optimisation dans la chaîne logistique en s'intéressant à la partie gestion de distribution.

La thèse est organisée comme suit:

Le chapitre 1 représente l'introduction générale de ce manuscrit. Il présente l'ensemble des raisons qui ont motivé ce travail de recherche et expose une présentation non exhaustive des contributions accomplies.

Le chapitre 2 commence en premier lieu par un état de l'art de la littérature consacrée aux problèmes de tournées de véhicules en général : nous rappelons donc les notions de base et nous présentons quelques formulations mathématiques et variantes du problème de transport. En deuxième lieu, nous détaillons les différentes approches et méthodes de résolution dont on fera appel par la suite. En dernier lieu, nous proposons de nouvelles idées que nous développons et qui constituent notre apport personnel dans la résolution des variantes considérées.

Les M-problèmes de tournées de véhicules (M-VRP) sont abordés dans le chapitre 3 dédié à :

- l'extension de ces problèmes au cas où la flotte hétérogène est limitée ;
- la présence du transporteur externe.

Il s'agit d'un problème complexe, avec des contraintes de capacité et de limitation du nombre de la flotte hétérogène de véhicules disponibles. Cette extension s'inscrit dans le cadre d'un intérêt pratique, celui d'une représentation plus réaliste des phénomènes observés. Ensuite, nous avons proposé une méthode itérative, hybride et coopérative de résolution, puis, nous avons étudié le problème de tournées de véhicules avec flotte limitée et transporteur externe. La résolution de ce dernier cas permet la prise de décision stratégique favorisant :

- 
- soit le maintien d'une flotte privée ;
  - soit opter pour un transporteur externe ;
  - soit de faire appel à une combinaison des deux premières options.

Compte tenu de la nature fortement combinatoire de ce problème, les instances peuvent être de très grande taille, par conséquent, les méthodes exactes peuvent s'avérer inefficaces. Afin de contourner ce problème, nous avons opté pour les approches heuristiques et métaheuristiques.

En se basant sur les réflexions et méthodes empiriques, nous avons implémenté en un premier temps, une méthode hybride avec une mémoire adaptative pour résoudre le problème de tournées de véhicules avec une flotte limitée hétérogène. En deuxième temps, deux approches ont été proposées pour résoudre le problème de tournées de véhicules avec flotte limitée et transporteur externe. La première approche se traduit par une technique de recherche heuristique basée sur des voisinages de types chaînes d'éjection, et la deuxième constitue une méthode évolutionnaire avec une recherche locale 2-opt.

Au travers des expérimentations menées avec cette hybridation, nous montrons que l'incorporation d'une méthode efficace (ici la méthode d'échange d'arcs où voisinage2-opt) qui donne une forte amélioration de la qualité des résultats. De plus, et bien que le problème soit difficile, les méthodes proposées résolvent des problèmes de taille très importante.

Le Chapitre 4 traite le problème de tournées de véhicules dynamique (PTVD). Certaines données du problème en question ne sont pas connues à l'avance. Elles constituent de nouvelles informations révélées en ligne après l'exécution des routes par les véhicules. Les nouvelles informations correspondent souvent à l'apparition, au cours du temps, de nouveaux clients qui doivent être inclus dans les routes actuelles.

L'exécution manuelle de l'organisation et la planification d'un tel service se révèle longue et fastidieuse. L'optimisation proposée des démarches visant l'atteinte d'une solution de ce problème combinatoire passe par diverses heuristiques permettant la planification de tournées de véhicules.

Dans ce chapitre nous abordons la résolution de la variante dynamique avec la méthode de colonie de fourmis. La décision prise à partir des expérimentations est la planification des itinéraires et le ré-ordonnement de services des clients. Autrement, les demandes des clients sont traitées de façon dynamique et la résolution du problème se poursuit en parallèle et en temps réel avec la simulation de nouveaux clients.

---

On termine la rédaction par une conclusion générale montrant les apports personnels et les nouvelles contributions que nous avons apportés à la résolution à ces problèmes de tournées de véhicules.

---

---

# Chapter 1

---

## Introduction

Logistics has emerged in the military and civilian domain for several centuries. The calculations of needs, traveling time, space and inventory collection showed a new kind of science that were not yet called operational research but who had a little spirit and that the was called to devote the logistics logico-mathematical nature. It is the function responsible for the flow of materials from suppliers into an organization, through operations within the organization, and then out to customers. The role of distribution in the supply chain management model has extended considerably from the conventional view of the activity as being concerned solely with transport and warehousing. For example Cooper et al. (1991) maintain that the transport and storage of goods are at the centre of any logistics activity.

Moreover, in recent years the world has experienced a sharp and dramatic increase in fuel prices. Efficient management of fuel cost has become a critical issue in today's of using vehicles. Many countries around the world are making large investments in transportation improvements, but even greater investments are still needed. This increase is likely to continue over the coming decades. Transport projects are generally intended to improve the economic and social care of people. Thus, many countries around the world are investing heavily, and very importantly, on improving the transport. One method of managing the increase of fuel cost is to reduce the cost of transportation, and planning of vehicle routing.

Vehicle Routing Problem (VRP) holds a central place in distribution management. Its economic importance has encouraged academic researchers and industry to find ways to effectively fulfill the transportation of goods and services. The VRP dates back to the end of the fifties of the last century when Dantzig and Ramser (1959) set the mathematical programming formulation and algorithmic approach to solve the problem of delivering gasoline to service stations. Several models have been developed such as the mathematical

model presented in our thesis by Fisher and Jaikumar (1981).

The role of distribution in the model of supply chain management has grown considerably to depart from its traditional view. Indeed, the conventional view of the business dealt only transportation and storage. Proposal as to minimize the cost of fuel, the optimized planning of vehicle routing seems to be a worthy exploration. This allows thereby reducing the transportation costs.

This research is motivated by the desire to find a realistic model and that is as faithful as possible to the state transport through the consideration of the appearance of heterogeneous vehicles (vehicles are not identical and they not all have the same capacity) and limited fleet of most industrial (each company has a limited number of vehicles).

These aspects have often been studied separately. Laporte (1992) which give a review on the exact methods for solving the vehicle routing problem, and Toth and Vigo (2002) presents some models and exact approaches to solve this type of problem.

Kilniewicz et al. (1990) presented the problem in a context to divide the customers into sectors and the private fleet size and common carrier must be determined for each sector. Diaby and Ramesh (1995) have been studied the problem and they fixed an objective to decide which customers are served by external carriers and to optimize the remaining customers. Recently, Chu (2005) and Bolduc et al. (2007, 2008) have used constructive heuristics for solving the vehicle routing problem with limited fleet and external carrier.

We propose to formulate new problems studies and put forward some methods of resolutions (e.g. Euchl and Chabchoub (2010), Euchl et al. (2011)). Thus, new methodologies described in this thesis may contribute to the improvement of thinking advanced by the scientific community.

## 1 Research Objectives of the thesis

In recent years, interest in meta-heuristic approaches such as simulated annealing, tabu search, variable neighborhood search and genetic algorithms, evolutionary algorithm, ant colony (for VRP) has increased due to the ability of these approaches to generate solutions which are better than those generated from sequential heuristics alone. This research uses metaheuristics algorithms to solve the vehicle routing problem with heterogeneous limited fleet.

Given the potential for possible multiple opportunities to develop trade links with industry in solving transport problems, we set a challenge the proposed methodologies

and the development of practical solutions optimizing vehicle routing with heterogeneous limited fleet. To achieve this goal, we identified the following key points:

1. Hybridization with tabu search: This method allows the search for neighborhood-type ejection chains to be incorporated into the tabu algorithm. This shows whether the integration of neighborhoods like ejection chains with metaheuristics is able to generate good solutions.
2. We combine the adaptive memory procedure (AMP) with tabu search metaheuristic. The AMP and its mechanism for updating stored solutions allow a comparatively large pool of good and diversified solutions to be stored and used during the search process, alternating between small and large neighbourhood stages during the metaheuristic course.
3. We seek to develop a methodology to address this decision of planning of tours. Our approach is based on the evolutionary metaheuristics ones making it possible to plan routing while determining the customers to be useful via an external transporter.
4. We use an ant colony optimization to solve the dynamic vehicle routing problem. We implement an artificial ant colony based on 2-Opt local search. In this application we chose to solve the dynamic pick up vehicle routing problem, in fact which seems hitherto not addressed in the various sources of literature of our state of the art.
5. To explore the state of the art of vehicle routing problem with heterogeneous limited fleet and investigate the use of different methods in the case of some variants of the heterogeneous vehicle routing problem with limited fleet. The way forward is the improvement of recent methods that deal with highly constrained problems.

This thesis aims to study some optimization of logistics and transport problems and refers to the proposal of metaheuristics for solving these variants of problems. This is done through the study of optimization problems in the supply chain by focusing on the management part of distribution.

The thesis is organized as follows:

Chapter 1 represents the introduction of the main manuscript. It presents all the reasons that motivated this research and presents a non exhaustive presentation of completed contributions.

Chapter 2 begins firstly by a state of the art of literature on vehicle routing problems in general: thus we remind the basic notions and we present some mathematical formulations

and variants of the transportation problem. Secondly, we detail the different approaches and methods which we will identify thereafter. Finally, we propose new ideas that we develop and that's represent our personal contribution to the solving of the considered variants.

In chapter 3, we deal with the M- Vehicle Routing Problems dedicated to:

- the extension of this problem in the case of the heterogeneous limited fleet;
- the presence of an external carrier.

This is a complex problem with capacity constraints and the limited number of heterogeneous fleet of available vehicles. This extension is part of practical interest, which is a more realistic representation of the observed phenomena. Then, we proposed an iterative, hybrid and cooperative methods; thereafter we have studied the vehicle routing problem with limited fleet and external carrier (VRPPC).

Solving the VRPPC case makes the strategic decision support:

- maintain a private fleet;
- either for external carrier;
- or to use a combination of the two options

Given the highly combinatorial nature of this problem, instances can be very large; therefore, the exact methods may be ineffective. To solve this problem, we opted for the heuristics and metaheuristics algorithms.

Based on the empirical methods, we implemented in the first step, a hybrid method with an adaptive memory to solve the vehicle routing problem with heterogeneous limited fleet. In the second step, two approaches have been proposed to solve the vehicle routing problem with limited fleet and common carrier. The first approach is a heuristic search technique based on ejection chains neighborhoods, and the second is an evolutionary approach with 2-opt local search.

Through experiments conducted with this hybridization, we show that the incorporation of an effective method (here the exchange arcs method or the 2-opt), which gives a strong improvement in the quality of results. Furthermore, the proposed methods solve the large instances of the problem.

Chapter 4 deals with dynamic vehicle routing problem (DVRP). In DVRP, some data about the problem are not known before hand. That is, new information is revealed online,

after the routes are executed by the vehicles. The new information often corresponds to the appearance of a new vertex (customer) that must be included into the current routes.

The organization and the planning of such a service prove to be long and tedious if it is done manually. In order to bring a solution for complex combinatorial problem, various heuristics make it possible to build planning's of vehicle routing.

In this chapter we discuss the solving of the dynamic vehicle routing problem with an artificial ant colony. The decision taken from the experiments is the planning of the routes and the re-scheduling of the customers. Otherwise, the customers' requests are handled dynamically and the resolution of the problem is made in parallel and in the real-time with the simulation of new customers.

## 2 Contributions

This thesis has dealt with three variants of heterogeneous vehicle routing problem with fixed fleet while addressing the following special cases:

- Case where the fleet is limited and heterogeneous;
- Case taking into account the external carrier in addition to a limited and heterogeneous fleet;
- Case of the dynamic vehicle routing problem.

In the following we present the main contributions of this thesis:

### **1st case: proposed ideas and developed strategies**

The first contributions appear in Chapter 3. We remind that it has the solving of vehicle routing problem with heterogeneous limited fleet as well as on the case of the presence of an external carrier.

The presence of the limited fleet constraint makes the problem more complex and realistic. The choice of a good metaheuristic can provide better results.

We propose in this work the description of the strategies which we provide examples of applications. Through these examples we focus on the optimization of empirical research by using a hybridization method with tabu search. These strategies allow the solving of the vehicle routing problem with heterogeneous limited fleet. The comparative results

favor the proposed strategies at the expense of those derived through the application of the methods described in literature.

These strategies help to find best satisfied solutions by using the concept of adaptive memory. The development of these strategies has allowed us to propose a new algorithm which has been the subject of a publication in Logistics Research (international journal) (Euchi and Chabchoub (2010)).

### **2nd case: proposed ideas and developed strategies**

In Chapter 3, we present work related to solve the vehicle routing problem with limited fleet and common carrier. The considered problem is solved by two methods.

The first method derived from the idea that we are proposing and that involves the application of the ejection chains neighborhoods structures in the tabu search algorithm (Euchi and chabchoub (2010)). We have seen the effectiveness of this algorithm to solve this variant of vehicle routing, which demonstrates the relevance of the idea.

The second method is the application of a version that we have improved from the. The key idea is reflected by a combination of 2-Opt local search and use of the probabilistic model (Euchi et al. (2011)). These results prove the satisfaction and the performance of the Iterated Density Estimation Evolutionary Algorithm (IDEA), whose results were published in the International Journal of Applied Metaheuristic Computing (International Journal of Range A) and in the International Journal of Universal Computer Science. These same results have been a subject of a communication published in IEEE simultaneously with our participation at the international conference "International Conference on Computers Industrial Engineering (Euchi and Chabchoub (2009)).

### **3rd case: proposed ideas and developed strategies**

Finally, we discuss a dynamic vehicle routing problem. In this case we are interested in the variant which considers the dynamic delivery pickup problems.

We draw the attention to two facts:

- Taking into account the dynamic pickup seems ever addressed in literature. To the best of our knowledge, our attempt in this regard could be the first in the field.

When handling large instances, good results are recorded following the improvement of the strategy and the combination method of ant colony in one side and the local

search method to another. The comparative results highlights those deduced from our method depend on those provided by the state of the art.

- This work was presented through two communications and at two international conferences. The first submission made during the progress of work, is partial when the second traces the total work done in this direction. The enthusiasm generated after these communications has encouraged us to submit our results to the international journal "Applied Mathematics and Computation"(review of row A) in the form of paper.



---

---

# Chapter 2

---

## Literature review

*In this chapter, we are going to present a literature review of the general vehicle routing problem then we are going to focus on the problematic approaches in this thesis. For each one of these problematic, a positioning compared to this state of the art will be developed.*

*In the first part a synthesis of literature review devoted to the Complex and Logistic Transport problems in general and a transport problem with limited fleet and common carrier in particular with stochastic and dynamic vehicle routing problem.*

*In the second part, a description of the vehicle routing problem with mathematical formulations will be presented. The different variants of this problem will be approached.*

### 1 Introduction

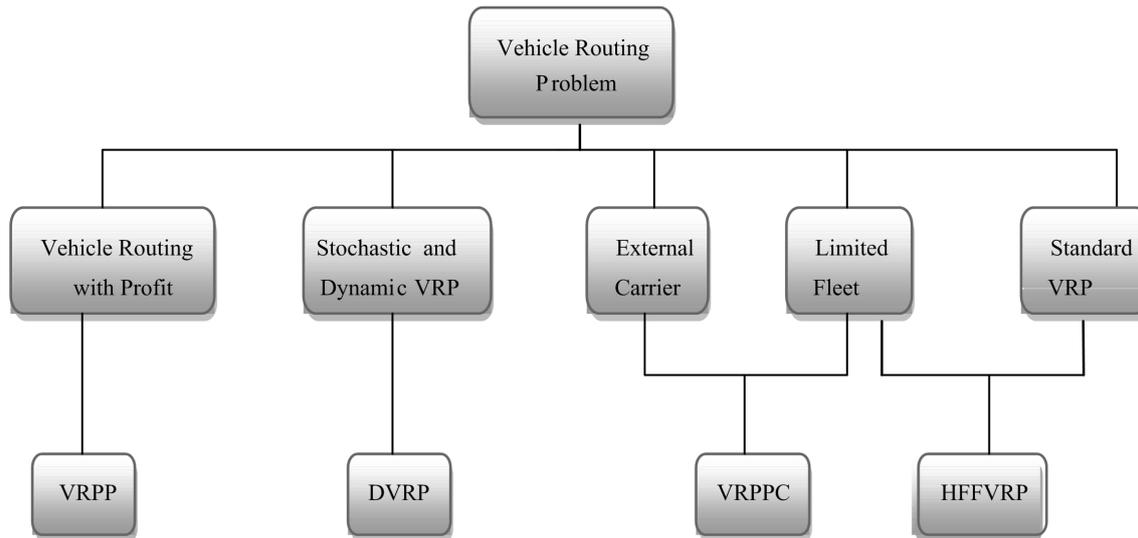
As we already mentioned in the introduction of this thesis, coordination of the various entities of the vehicle routing problem represents a major challenge today. The line of research that we have exposed the face of this problem is to define a tool to assist decision-making for planning the vehicles routing. To define precisely the problem and to make it necessary to focus on certain points from both a state of the art of academic research and industrial practices.

We'll explain more precisely what is the Vehicle Routing Problems (VRPs)? This chapter aims at positioning this thesis work in the broad field of research on Vehicles Routing issues. The purpose of this study is to bring the complete problem of vehicle routing adapted to the problems of classical literature.

## 2 Related works

The VRP and its generalizations have been widely studied in recent years.

Figure 2.1 illustrates the classification of the VRP



**Figure 2.1** – classification of vehicle routing

In general, distribution problems are in the ability to meet requests from several customers (destinations) from one or several warehouses (sources), at lower cost. Normally, these problems are solved by placing the customers demands in some way in vehicles of limited capacity.

The optimization of a distribution network involves two types of decisions that allow us to reduce costs or increase the level of service. Both types are strategic decisions and operational decisions. Strategic decisions are not taken every day, and their effects are medium and long term. The location of warehouses or transshipment nodes, the allocation of products to warehouses, the frequency of delivery and composition of the fleet vehicles are examples of strategic decisions.

The operational decisions are taken daily and their effects are immediate. For example, the allocation of customers to warehouses, the allocation of products to customers, mode of transportation and / or the choice of routes to take is indeed a Vehicle Routing Problem (VRP).

The VRP is a part of the combinatorial optimization problems most studied. This problem holds the attention of several researchers for many years, and everywhere in the

world. Several authors have made literature review that deal with problems tours.

The first paper dealing with the vehicle routing was published in the late 1950's by Danzig and Ramser (1959). This problem, more often called Vehicle Routing Problem (VRP), draws a large number of researchers because it is theoretically very interesting. Moreover, applications of VRP are numerous. Thus, most companies which must deliver a product to several customers are facing this problem. The literature of the VRP is therefore very large. These include Bodin et al. (1983), Laporte (1992a, 1992b, 1993), Laporte and Osman (1995) who presented the problems of Vehicle Routing as problems easy to explain but difficult to solve.

More specifically, two types of problems have been addressed in literature: VRP with limited fleet and the VRP with limited fleet and external carrier. Some authors studied the VRP with limited fleet including Osman and Salhi (1996), Gendreau et al. (1999), Taillard (1999) and Tarantilis et al. (2004). All these authors considered a heterogeneous limited fleet with sufficient capacity to serve all customers. Another interesting idea was made by Fisher and Jaikumar (1981) in their article on the development of a heuristic in the Vehicle Routing problem. They used an algorithm of assignment to assign each order to a vehicle. The objective was to minimize the total cost of tours. On the other hand,  $N$ , the number of vehicles available is given, and thus is not to minimize. One advantage of this algorithm is that it always gives a feasible solution if there is one.

Also Frederickson et al. (1978) address the problem of building tours with  $N$  vehicles. Their objective is to minimize the maximum length of tours. Although this problem brings together the VRP and it appears in reality. Similarly, the paper of Renaud and Boctor (2002) presents the VRP with a heterogeneous fleet. The fleet of vehicles can be made up of vehicles with different capacities. Also, fixed and variable costs of vehicles may differ from one vehicle to another. The hiring of part or all of the fleet is possible providing the advantage of flexibility since the composition of the fleet may change frequently. The objective is to minimize the total cost consisting of fixed and variable costs of vehicle use. Moreover, we must find better allocation of vehicles to different routes.

At the level of VRP with limited fleet and external carrier, Volgenant and Jonker (1987), which demonstrated that the problem involving a fleet of one single vehicle and external carriers, can be rewritten as Travelling Salesman Problem (TSP). This problem was also studied by Diaby and Ramesh (1995) whose objective was to decide which clients to visit ugly external carriers and to optimize the tour of remaining customers.

## 2.3 Travelling Salesman Problem (TSP)

---

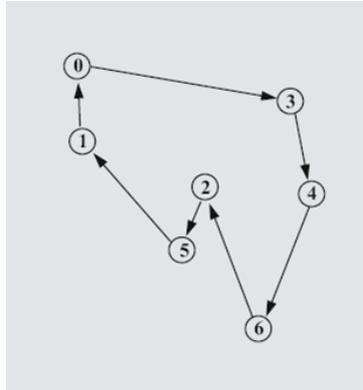
Also among the literature we find in Fisher et al. (1982) who have also included a choice of using an external carrier or its own fleet in developing their application. Hall and Racer (1995) have studied this same choice with the approximation continues. Brown et al. (1987) incorporate this decision in their model to allocate oil trucks, also Yang et al. (2000) incorporate the problem in their distribution to stores in detail.

Generally, when the company does not have enough vehicles to carry out all tours we must decide which tours will be made by an external carrier and which will be made by the fleet of the company. The Vehicle Routing Problem with Private fleet and Common carrier (VRPPC) is a more complex problem because it involves an internal fleet of several vehicles. To our knowledge, VRPPC was introduced by Chu (2005) and Bolduc et al. (2007). In his article Chu presents the mathematical modelling of the problem and solves it with heuristic economies improved interstate and intrastate routes. Thereafter, Bolduc et al. (2007) have improved the results of Chu using more sophisticated exchanges.

## 3 Travelling Salesman Problem (TSP)

The name of Traveling Salesman Problem - TSP for the optimization problem is believed to have originated in the United States. It is a prominent illustration of a class of problems in computational complexity theory which is classified as NP-hard. However, it would be reasonable to say that a systematic study of the TSP as a combinatorial optimization problem began with the work of the Dantzig et al. (1954). Applegate et al. (2003), Johnson and McGeach (1997), and Junger et al. (1995) are excellent survey and computational aspects of the TSP.

Traveling salesman problem considers a salesman who needs to visit each of  $n$  cities, which we shall enumerate as  $\{0, 1, \dots, n - 1\}$ . His goal is to start from his home city, 0, and make a tour visiting each of the remaining cities once and only once and then returning to his homecity. We assume that the "distance" between two cities,  $c_{ij}$ , is known (distance does not necessarily have to be distance-it could be travel time or, even better, the cost of travel) and that the salesman wants to make the tour that minimizes the total distance. This problem is called the traveling salesman problem. The following figure 2.2 shows an example with seven cities.



**Figure 2.2** – A feasible tour in a seven-city traveling salesman problem

## 4 Vehicle Routing Problem (VRP)

The VRP generally determine the best routes or schedules to support and / or to deposit passengers or goods across sites, while respecting certain requirements and optimizing the cost associated with the solution of the problem. This part is devoted to describe the vehicle routing problem. The VRP, the generic name given to all these problems or vehicles, must perform actions on a network. Danzig and Ramser have proposed a mathematical formulation which can be explained by the patterns that follow. The first is the problem with the warehouse and operations materialized by the crosses.

### 4.1 Distribution network

The vehicles generally move on a transportation system (e.g. Clarke and Wright (1964)). Whether an aircraft or vessel which must comply with air or sea routes, a train or a car. The geographical area is usually represented by a directed graph or symmetrical. Each node represents a remarkable place (a city, a warehouse, a customer, etc.) Each edge connects symmetrical and each arc directed a connection between these places (e.g. Figure 2.3).

The graph can also submit for each arc or edge one or more useful assessments to the problem. They may represent a distance or a cost toll, particular restrictions, or a travelling time depending on the type of vehicle.

Figure 2.4 shows the solution with three rounds of vehicles that start and finish filing satisfying all the demands of operations.

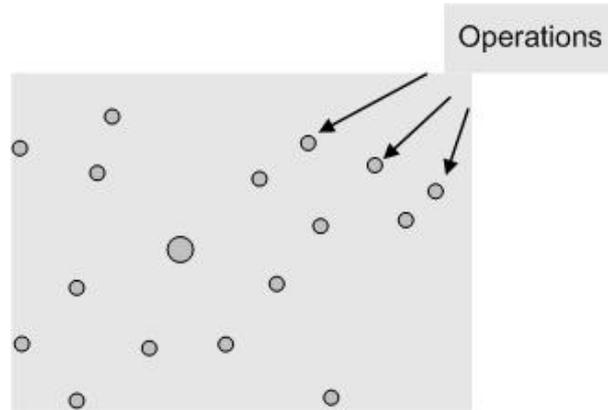


Figure 2.3 – Representation of a vehicle Routing Problem

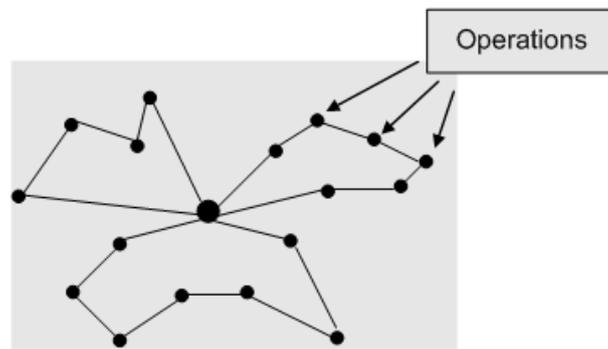


Figure 2.4 – Solution of Vehicle Routing Problem

## 4.2 Mathematical model

The Vehicle Routing Problem can be defined as a problem which many customers must serve starting from a single depot with known demands.

Mathematically, the Vehicle Routing Problem is defined on an undirected graph  $G = (V, A)$  where  $V = \{v_0, v_1, \dots, v_n\}$  is the vertex set and  $A = \{(i, j) : i, j \in V, i \neq j\}$  is the arc set. Vertex  $v_0$  represent depots at which are located at most  $m$  identical vehicles of capacity  $Q$ . With each customer  $i \in V \setminus \{0\}$  is associated with a non-negative demand  $q_i \leq Q$ . A distance  $d_{ij}$  is associated with each arc  $(i, j) \in A$ , these symmetrical distances i.e. that  $d_{ij} = d_{ji} \forall i, j \in A$ . A limit  $l$  can also be being imposed over the maximum duration of the tours.

In some versions of the problem, the number of vehicles is given a priori. In the others, the number of vehicles is a decision variable. The tours must make it possible to visit all the customers once and only once. The vehicles are assigned to the tours so as to

## 2.4 Vehicle Routing Problem (VRP)

---

minimize the objectives which for example can be the covered distance to visit all the customers. The problem consists of determining a set of  $m$  vehicle routes such that:

1. Starting and ending at the depot;
2. Each customer is visited by exactly one vehicle;
3. The total demand of any route does not exceed  $Q$ , and
4. The total routing cost is minimized.

There are many formula of the vehicle routing problem. The following formula is drawn by Fisher and Jaikumar (1981). First of all let us define the set of the necessary variables to carry out the mathematical formulation.

### 4.2.1 Parameters

$m$  : Number of available vehicles

$N$  : Number of customers to be visited. The customers are numbered from 1 to  $n$  and the depot has the number 0

$Q_k$  : Capacity of the vehicle  $k$

$q_i$  : Demand of customer  $i$

$d_{ij}$  : Outdistance between the customer  $i$  and  $j$ .

### 4.2.2 Decisions variables

$y_{ik}$ : Binary decision variable which is equal to 1 if the truck  $k$  travels from customer  $i$  to customer  $j$  and 0 otherwise.

$x_{ijk}$ : Binary decision variable which is equal to 1 if the truck  $k$  travels from customer  $i$  to customer  $j$  and 0 otherwise.

### 4.2.3 Formulation

Fisher and Jaikumar (1981) formulate the VRP as follows:

## 2.5 Variants of the vehicle routing problem

---

$$\text{Minimize} \quad \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^K d_{ij} x_{ijk} \quad (2.1)$$

subject to

$$\sum_{i=1}^n q_i y_{ik} \leq Q_k \quad k \in \{1, \dots, m\} \quad (2.2)$$

$$\sum_{k=1}^K y_{ik} = \begin{cases} k & i=0 \\ 0 & i=1, \dots, n \end{cases} \quad y_{ik} = \{0, or 1\} \quad i \in \{0, \dots, n\} \quad k \in \{1, \dots, m\} \quad (2.3)$$

and with constraints of the Traveling Salesman problem (TSP)

$$\sum_{i=0}^n x_{ijk} = y_{jk} \quad j \in \{0, \dots, n\} \quad k \in \{1, \dots, m\} \quad (2.4)$$

$$\sum_{j=0}^n x_{ijk} = y_{ik} \quad i \in \{0, \dots, n\} \quad k \in \{1, \dots, m\} \quad (2.5)$$

$$\sum_{I \in S} \sum_{J \in S} x_{ijk} \leq |S| - 1 \quad \forall S \in \{1, \dots, n\} \quad k \in \{1, \dots, m\} \quad 2 \leq |S| \leq n - 1 \quad (2.6)$$

$$x_{ijk} = \{0 \text{ or } 1\} \quad i, j \in \{0, \dots, n\} \quad k \in \{1, \dots, m\} \quad (2.7)$$

This formula (2.1) can minimize the distance travelled by all vehicles. Constraint (2.2) ensures that the loading of vehicles respects their capacity. Constraint (2.3) ensures that each route begins and ends at the depot and that each customer is assigned to a single vehicle. Constraints (2.4) to (2.7) avoid sub-tours and that each customer is visited only one time. Hence, there are the constraints used for the travelling salesman problem.

## 5 Variants of the vehicle routing problem

During these recent years of research for other problems, derivatives vehicle routing problem, made their appearances. These appearances are due mainly to the activities of researchers who work more and more on the problems of transport and distribution that companies face. In what follows we will present the main problems derived from Vehicle Routing Problem. Our primary interest in the problem to optimize the management of a limited fleet homogeneous or heterogeneous vehicle finite capacity, to domicile it in the

## 2.5 Variants of the vehicle routing problem

---

same depot, and to visit a set of geographically dispersed customers, who formulate a demand ( or an offer, but not both) known in the presence of an external carrier. In this section, we are presenting a brief period mono certain extensions of the vehicle routing problem, the Capacitated VRP in Section 5.1 and the Multi period VRP in Section 5.2. We also deal with two multi-period problems : Period Vehicle Routing Problem, Inventory Routing Problem. The VRP with a limited fleet in Section 5.3. We present the VRP with full truckload in Section 5.4. Section 5.5 deal with the vehicle routing with profit. After, we introduce the Vehicle Routing Problem with Private fleet and common Carrier in Section 5.6. Finally, Section 5.7 located the Dynamic VRP in Section.

### 5.1 Capacitated vehicle routing problem

The most elementary version of the vehicle routing problem is the capacitated vehicle routing problem (CVRP). The CVRP is described as follows:  $n$  customers must be served from a unique depot. Each customer asks for a quantity  $q_i$  of goods ( $i = 1, \dots, n$ ) and a vehicle of capacity  $Q$  is available to deliver goods. Since the vehicle capacity is limited, the vehicle has to periodically return to the depot for reloading. In the CVRP, it is not possible to split customer delivery. Therefore, a CVRP solution is a collection of tours where each customer is visited only once and the total tour demand is at most  $Q$ . The goal is to find a set of tours of minimum total travel time.

**Data:** A set of nodes customer (having a demand) and edges (provided with costs) and an unlimited fleet of vehicles with uniform capacity  $Q$  on the basis of a single depot.

**To find:** Vehicles tours, satisfying each demand once and only once and respecting the constraints of capacity.

**To Minimize:** The total cost of transport related to the arcs borrowed by the vehicles and/or the fixed costs associated with the use of the vehicles.

The CVRP can be formulated as a general vehicle routing problem where constraints (2) are added.

### 5.2 Multi-Period Vehicle Routing Problem

More recent than the classical vehicle routing problems, the period vehicle routing problems consider a planning horizon where a vehicle may make several routes. The period vehicle routing problem is to be delivered by a range of customers, the quantity of demand for one or several products on a horizon of time. In this problem, the quantity of products delivered to a customer, allows the latter to cover its needs until the next visit

## 2.5 Variants of the vehicle routing problem

---

of the vehicle. As a result, the vehicles can make several trips. The main purpose of this problem is divided into two parts: the first is to schedule deliveries for each customer over a predetermined time, and the second is to organize the vehicles tours to make deliveries required while optimising the total cost of transport. We describe in this section two types of multi-periods problems : the PVRP (Period Vehicle Routing Problem) and IRP (Inventory Routing Problem). These two problems generally correspond to industrial problems but addressed in different ways. Among these issues there are generally the delivery of liquid gas and oil, the distribution of beverages, and more generally periodic distributions of goods and consumption needing several deliveries on the planning horizon.

### 5.2.1 Period Vehicle Routing Problem

In the PVRP, consider that each customer have to be served a certain number of times in the horizon, and that this number constitutes data of the model. The goal to be used each customer as many as time as necessary. To each customer a total of possible sequences of deliveries is allotted, corresponding to the days of deliveries. The PVRP can thus break up into two problems: assignment of sequences of delivery to the customers and resolution of a classical vehicle routing problems per day of the horizon. The definition of the problem is presented as follows:

**Data:** A set of nodes customer (having a demand); and time windows (flexible and hard); a set of edges (provided with costs) and an unlimited fleet of vehicles with fixed capacity  $Q$  on the basis of a single depot.

**To find:** A set of vehicles tours, satisfying each demand one and only once in the horizon of planning and respecting the constraints of capacity of the vehicles.

**To Minimize:** The total cost of transport related to the arcs borrowed by the vehicles.

This problem was formulated the first time by Beltrami & Bodin (1974) [8] concerning the domestic garbage collection. Two approaches are proposed:

- Affect to each customer a sequence of delivery to solve the vehicle routing problem for every day of the horizon.
- Construct the tours then affect them to days while respecting sequences of delivery.

However, the frequency of visiting of the customers is a hard constraint which can penalize the resolution. More recent methods slacken this constraint and allow a customer to be visited more often than necessary. This relaxation is particularly useful in Period Vehicle Routing Problem with Service Choice (Francis et al. (2004)) where the frequency of visiting is a decision variable of the problem.

### 5.2.2 Inventory Routing Problem

For twenty years, Inventory Routing Problem has aroused much more interest on behalf of the researchers. It is about a more global solution than for the PVRP, so far as he also integrates problems of inventory control. In Inventory Routing Problem, the concept of sequence of delivery becomes implicit, and is not any more one fact of the case. Each customer has a rate of consumption which is generally known and a stock of product at the beginning of horizon. The goal is to avoid the out-of-stock conditions at the customer. One can then cut out the IRP in three problems: quantity and inventory control of each customer to deliver in order to avoid the out-of-stock conditions, assignment of the customers at the days of delivery, and design and optimization of the tours.

**Data:** A set of node customers (with an inventory) and consumption of this stock in the planning horizon; a set of edges (provided with costs) and an unlimited fleet of vehicles with fixed capacity  $Q$  on the basis of a single depot.

**To find:** A set of vehicles tours, avoiding stock outs at the customer, within constraints of capacity of vehicles and respecting the constraints of capacity of the vehicles.

**To Minimize:** The total cost of transport related to the arcs borrowed by the vehicles.

### 5.3 Vehicle Routing Problem with limited fleet

Traditionally, we differentiate the fleets on a criterion from homogeneity. It is said that a homogeneous fleet if each vehicle is identical in all points. A fleet is known as heterogeneous if the capacity is not the same for each vehicle, or so various competences exist according to vehicles. In the case of multi-depot (MDVRP) problems, departure / arrival for each vehicle can vary, which may be seen as an aspect of heterogeneous fleet. The problem can then be broken down into two sub-problems, which are the assignment of the demands to the depots, then the resolution of a VRP for each depot (Cluster first, route second). We can refer to Tansini et al. (2001) for the case of vehicle routing problem multi-depot (vehicles have specific points of departure and arrival).

For a few years, a new type of vehicle routing problem arouses a growing interest: the vehicle routing problems with limited fleet, called m-Vehicle Routing Problem (m-VRP). If there is an assignment making it possible to satisfy all the demands on the horizon of planning, the problem is known as satisfiable and we solve it according to the present objective. Otherwise, it defines a secondary objective and we start again the resolution of the problem. Typically, this objective will aim at maximizing the number of

## 2.5 Variants of the vehicle routing problem

---

satisfied demands. However, it can vary according to the constraints (time windows hard or flexible, etc). The satisfaction of each demand becomes a decision variable then, and it is thus necessary to adapt the consequently objective.

We will define three statements of the problem according to these two alternatives.

### 5.3.1 M-Vehicle Routing Problem with Time Windows (SATISFIABLE)

The case of satisfaisables problems presents no complications concerning the definition of the objective: since there is a solution satisfying all the demands, it is considered that a valid solution must satisfy all demands, whether such "meeting" or deferrable. The objective is then to minimize the total cost of the tours. There is not fixed cost associated with the creation with a tour and the cost compared to the distance. The objective of the satisfiable problems is to minimize the total travelled distance.

The definition of the problem is as follows:

**Data:** A set of nodes customer (having a demand) and time windows; a set of edges (provided with costs); Limited fleet of vehicles.

**To find:** A set of tours respecting the constraints of capacity of the vehicles, hard time windows and constraints on depots.

**To Minimize:** The total cost of transport related to the arcs borrowed by the vehicles.

### 5.3.2 M-Vehicle Routing Problem with Time Windows (DISSATISFYING)

The case of dissatisfying problems corresponds the company's willingness to test the possibilities of excessive size of the entrance to the problem. The underlying idea is that if there are more choices for applications, it can produce better solutions. The insatisfiable problems therefore correspond to satisfiable problems plus deferrable demands, not criticism. The goal is then considered minimizing the number of deferrable demands dissatisfied. Each meeting must be satisfied, and constitutes a constraint.

The definition of the problem is as follows:

**Data:** A set of nodes customer (having a demand) and time windows; a set of edges (provided with costs); Limited fleet of vehicles.

**To find:** A set of tours respecting the constraints of capacity of the vehicles, hard time windows of the jobs actually carried out and constraints on depots.

**Minimising:** The number of satisfied customers.

### 5.3.3 M-Heterogeneous Vehicle Routing Problem (M-HVRP)

This section considers an important variant of the VRP in which a fleet of vehicles characterized by different capacities and costs is available for distributing activities. The problem is known as the Mixed Fleet VRP or as the Heterogeneous Fleet VRP, and was first considered in a structured way in Golden et al. (1984). Case-studies and applications related to the solution of Heterogeneous VRPs can be found in Semet and Taillard (1993), Rochat and Semet (1994), Brandao and Mercer (1997), Prins (2002), Wu et al. (2005) and Moghaddam et al. (2006).

## 5.4 Vehicle Routing Problem with full truckload

In most vehicles Routing Problem, it is assumed that charges are partial (less than truckload) i.e. that one must visit several clients to load the truck. A full load more often called "full truckload" (FL) means that the control of a single client can fill the capacity of the truck. Arunapum et al. (2003) address a variant of VRP recital full loads, it is the "vehicle routing with full truckload" (VRPFL). This problem is to determine the tours with a minimum cost and to deliver a predetermined number of full loads between pairs of cities specified using a fleet of vehicles located in one or more depots. Every tour must satisfy the time windows in each city where a shipment is made. Thus, trucks visit pairs of cities unlike VRP visiting only cities. The algorithm also takes into account the constraints of time windows.

Less-than-truckload shipments are arranged so that the driver of the tractor trailer picks up the shipment along a short route and brings it back to the terminal, where it is later transferred to another truck. This second truck brings the shipment, along with other small shipments, to another city's terminal. The less-than-truckload shipment is transferred from truck to truck until it finally reaches its destination. Full truckload carriers normally deliver a semi trailer to a shipper who will fill the trailer with freight for one destination. After the trailer is loaded, the driver returns to the shipper to collect the required paperwork (i.e. Bill of lading, Invoice, and Customs paperwork) and depart with the trailer containing freight. In most cases the driver then proceeds directly to the consignee and delivers the freight him or herself.

Occasionally, a driver will transfer the trailer to another driver who will drive the freight the rest of the way. Full Truckload (FTL) transit times are normally constrained by the driver's availability according to Hours of Service regulations and distance. It is

## 2.5 Variants of the vehicle routing problem

---

normally accepted that Full Truckload drivers will transport freight at an average rate of 47 miles per hour (including traffic jams or queues at intersections). Finally, when a truck on the route is not in motion, a penalty period is imposed. The goal is to minimize the movement of empty vehicles as they offer no value added to the final product.

### 5.4.1 Vehicle Routing Problem with return truckload

The VRP with return truckload means that after having made its deliveries, return the depot must be made in transporting the goods. Osman and Wassan (2002) describe two constructive heuristic of routes to generate an initial solution. These routes have improved thanks to a reactive tabu search metaheuristic. The concept reagent can trigger an exchange between neighbouring structures to intensify and diversify phases of research. The algorithm was based on economies of Clarke and Wright (1964) and included several other phases of local improvement. The algorithm was tested on real problems and helped improve the solutions obtained by dispatcher's experience.

### 5.4.2 VRP with pick up and delivery with full truckload

Like the travelling salesman problem, the VRP can also help make different types of operations. These operations can be pickups and / or deliveries. When these two types of operations are combined, we must make the pickings before deliveries associated. In addition, it is possible that loading is complete or partial according to the weight or space used in the truck. Also, constraints delivery with the return pickings may be imposed.

One article discusses the case of pickups and deliveries in a context of full loads (full truckload). It is the paper of Gronalt et al. (2003). The pickups are made at some distribution centres and 48 orders are delivered to customers. They try to minimize the movement of vehicles with no charge, therefore having no value added to the product. They are based on the problem of pickup and delivery with the constraints of time windows to develop four different heuristic savings based on Clarke and Wright (1964) to solve the problem. They bring together the concepts of opportunity costs and values of regret in the calculation of savings for this algorithm. A lower bound is set, the latter represents a solution with no movement of empty vehicles.

## 5.5 Vehicle Routing Problem with Profit

A recent paper proposed by Feillet et al. (2005) elaborated on the Profitable Arc Tour Problem (PATP) which is a generalization of vehicle routing problem where it is not

## 2.5 Variants of the vehicle routing problem

---

necessary to visit all vertices of the given graph. The profit is known a priori and associated with each customer. PATP can be formulated as a discrete bicriteria optimization problem where the two goals are maximizing the profit and minimizing the traveling cost. It is also possible to define one of the goals as the objective function and the other as a satisfiability constraint.

The extension of the PATP to multiple vehicles is referred to as Vehicle Routing Problem with Profits (VRPP). In one version, this is known as Selective Traveling Salesman Problem (STSP), Orienteering Problem (OP), or Maximum Collection Problem (MCP) in the literature, the objective is the maximization of the collected profit so that the total traveling cost (distance) does not exceed an upper bound. The other version, named as the Prize Collecting Traveling Salesman Problem (PCTSP), is concerned with determining a tour with minimum total traveling cost where the collected profit is greater than lower bound. Feillet et al. (2005) provided an excellent survey of the existing literature on Traveling Salesman Problem with Profits (TSPP). Their survey presents various modeling approaches to TSPP and exact as well as heuristic solution techniques.

The multi-vehicles version of the OP is called the Team Orienteering Problem (TOP) which is studied by Chao et al. (1996a, 1996b). The authors propose a 5 step metaheuristic based on deterministic annealing for its solution. Butt and Cavalier (1994) address to the Multiple Tour Maximum Collection Problem (MTMCP) in the context of recruiting football players from high schools. They propose a greedy tour construction heuristic to solve this problem. Later on Butt and Ryan (1999) develop an exact algorithm for the MTMCP based on branch and price solution procedure. Recently, Euchi and Chabchoub (2010) are using the metaheuristics approaches to solve the PATP, they apply a tabu search and a variable neighborhood search embedded in adaptive memory procedure.

## 5.6 Vehicle Routing Problem with Private fleet and Common carrier

The only difference between the vehicle routing problem with limited fleet and the vehicle routing problem with limited fleet and common carrier is the presence of external carrier. Generally, when the company does not have enough vehicles to carry out all the tours, it relies on external carrier. The network studied consists of a depot and several customers. One or more products are distributed and ordered all units are available in the depot during the planning of transport. Each customer must be served one and only once by

## 2.5 Variants of the vehicle routing problem

---

the fleet by an internal or external carrier. The internal fleet is composed by a limited number of vehicles. The ability of vehicles is determined in terms of units produced.

The definition of the problem is as follows:

**Data:** A set of nodes customer (having a demand); a set of edges (provided with costs); Limited and private fleet of vehicles with fixed capacity; External carrier.

**To find:** A set of tours respecting the constraints of capacity of the vehicles, and the constraints of depots.

**To Minimize:** Transportation cost:

- Fixed and variable costs of the internal fleet
- Fixed cost of the external transporter

## 5.7 Dynamic Vehicle Routing Problem

During recent years, distribution systems have become increasingly complex. Another complicating distribution is the increased focus on timeliness in the distribution chains, as intelligent planning offers potential savings in capital bindings related to stock and distribution. This fact has caused an increasing interest in dynamic transportation models and systems in which data are considered to be dependent.

We are considering an alternative dynamic version of the VRP, in which the customers have uncertain demands. In the dynamic VRP, a number of customer requests are available initially, while others become available during the execution plan. Specifically, the vehicles carry multiple types of product, each customer has an initial reported demand for certain quantities of each product, when the vehicles reaches the customer, the customer may change this request based on the current contents of the vehicle.

The problem is defined as follows:

**Data:** A set of nodes customer (having a dynamic demand); a set of edges (provided with costs); unlimited fleet of vehicles with uniform capacity on the basis of a single depot.

**To find:** A set of vehicles tours, satisfying each dynamic demand and only once and respecting the constraints of capacity of the vehicles.

**To Minimize:** The total travel cost.

The table below [2.1](#) summarizes the different variants that were actually considered in the literature, together with the corresponding references:

## 2.6 An overview of proposed approaches for the VRP

**Tableau 2.1** – Overview of different problems

<b>Problem</b>	<b>Fleet size</b>	<b>Fixed costs</b>	<b>Routing costs</b>	<b>References</b>
CVRP	Unlimited	Considered	Dependant	[12] [17] [24] [30] [82] [123]
VRPTW	Unlimited	Not Considered	Dependant	[8] [18] [67] [101] [91]
FFVRP	limited	Considered	Dependant	[39] [17] [103] [111] [116]
PVRP	Unlimited	Not Considered	Dependant	[100]
SDVRP	Unlimited	Considered	Dependant	[3] [5] [70] [111] [126]
FSMFD	Unlimited	Considered	Dependant	[29] [79] [98]
FSMD	Unlimited	Not Considered	Dependant	[33] [68] [91]
FSMF	Unlimited	Considered	Independant	[33] [68] [91] [98] [109] [113]
VRPHF	Unlimited	Not Considered	Dependant	[23] [60] [88] [119] [121]
VRPPC	limited	Considered	Dependant	[7] [13] [14] [26] [71] [81]
VRPP	Unlimited	Considered	Dependant	[6] [11] [20] [22] [45] [50] [51]
DVRP	limited	Considered	Dependant	[106] [107] [114] [94]

When:

CVRP: Capacitated Vehicle Routing Problem

VRPTW: Vehicle Routing Problem with Time Windows

FFVRP: Vehicle Routing Problem with Fixed Fleet

multi-periods VRP : multi-periods Vehicle Routing Problem

SDVRP: Split Delivery Vehicle Routing Problem

FSMFD: Fleet Size and Mix VRP with fixed costs and Vehicle Dependent Routing costs

FSMD: Fleet Size and Mix VRP with Vehicle Dependent Routing Costs

FSMF: Fleet Size and Mix VRP with fixed costs

VRPHF: Vehicle Routing Problem with Heterogeneous Fleet

VRPPC: Vehicle Routing problem with Private fleet and Common carrier

VRPP: Vehicle Routing with Profit

DVRP: Dynamic Vehicle Routing Problem

## 6 An overview of proposed approaches for the VRP

The metaheuristics are from now on regularly employed in all sectors of engineering. In this section, we provide an overview of heuristic methods for operational research that are somewhat familiar with the basic application in this thesis.

The first part is devoted to the detailed presentation of the metaheuristics via memory, and then in the second part we describe other evolutionary metaheuristics.

### 6.1 Metaheuristics optimization via memory

In this section we consider the metaheuristics with memory as the Tabu Search, Variable Neighborhood Search algorithm, Ejection chains neighborhood and the adaptive memory.

#### 6.1.1 Tabu Search algorithm

Tabu Search (TS) is one of the most widely used metaheuristics. The TS is described, introduced and refined by Glover (1989). As many other metaheuristics, the success of TS is, in large part, due to its ability to steer the search process from getting stuck in a local optimum. This is achieved by allowing a move to a neighboring solution that may result in deterioration in the objective value but that simultaneously avoids cycling back through previous moves. TS procedures exploit the short term memory i.e. the Tabu list, which keeps track of recently visited solution or their attributes. A move to a neighboring solution is permitted if the neighboring solution is neither contained in the Tabu list nor possesses an identical attribute (e.g. objective value) to a solution in that list.

However, a move to a neighboring solution could be basically selected on some aspiration criteria even if it is prohibited by the Tabu list. For example, in most Tabu Search applications, a particular move may be permitted even if it (or its attribute) is contained in the tabu list as long as such a move will result in a solution that is superior to the best solution obtained thus far.

Tabu Search has become the focus of numerous comparative studies and practical applications in recent years as in Brandão (2009). Fruitful discoveries about preferred strategies for solving difficult optimization problems have surfaced as a result. We first describe the basic concepts of TS. The method begins with a complete, feasible solution (obtained, e.g., by a constructive heuristic) and, just like local improvement, it continues developing additional complete solutions from a sequence of neighbourhoods.

We suppose that we are trying to minimize a function  $f(S)$  over some domain and we apply the so-called "best improvement" version of TS, i.e., the version in which one chooses at each iteration the best available move (this is the most commonly used version of TS).

### 6.1.2 Variable Neighborhood Search algorithm

Variable neighbourhood search (VNS) (e.g. Hansen and Mladenovic (2001)) is a recent metaheuristic for solving combinatorial and global optimization problems whose basic idea is systematic change of neighbourhood within a local search. This helps the VNS to explore neighbourhoods which are distant from the current solution and to jump to new solutions. The basic VNS is made up of three stages: shaking, local search and move. More details can be found in Hansen and Mladenovic (2001). A local search is applied repeatedly to obtain the local optima from the current solution. Originally the basic VNS approach was a descent method. It does not accept a worsening solution to get out of local optima since the neighbourhood structures are varied regularly. Since the local optima in one neighbourhood structure is not necessarily a local optima in another neighbourhood structure, the change of the neighbourhood structures can be undertaken during the search. The termination criteria may be selected as a maximum number of iterations, the CPU time or a certain number of iterations without improvement.

### 6.1.3 Ejection chains neighborhood

The use of compound neighbourhoods adds another level of sophistication to the procedures of generation of movements. The ejection chains combine and generalize the ideas based on alternate ways of the graph theory (e.g. Harary (1969)), the construction of networks based on exchanges (e.g. Lawler (1976)), and the restriction of structures to solve the Integer linear Programming Problem (e.g. Glover (1977)). All these fields have a compound neighbourhood search and suggest large varieties of new approach to solve combinatorial optimization.

By interpreting the definition of Glover (1992), we preserved for an ejection chains the flowing characteristics:

- A neighbourhood of a simple movement is included on each level in a sequence of neighbourhoods to render more complex movements.
- The evaluation of a movement on each level does not depend on the movements carried out on the former level of the chains.
- The passage of a level to the following reproduces on incomplete structure but guard on high level of legitimacy.

- On each level, the obtaining of a complete structure is obtained by application of the appropriate movements to the closing of the chains.

### 6.1.4 Adaptive memory

The AMP was first proposed by Rochat and Taillard (1995) as an enhancement of Tabu Search (TS) to solve the Vehicle Routing Problem (VRP). It was motivated by the work of Glover regarding surrogate constraints Glover (1977). An important principle behind AMP is that good solutions may be constructed by combining different components of other good solutions. A memory containing components of visited solutions is kept. Periodically, a new solution is constructed using the data in the memory and improved by a local search procedure. The improved solution is then used to update the memory.

A pseudo code of the AMP is given below:

1. Initialize the memory  $M$ .
2. While a stopping criterion is not met, do:
  - Construct a new solution  $s$  combining components of  $M$ .
  - Apply a local search procedure to  $s$  (let  $s^*$  be the improved solution).
  - Update  $M$  using components of  $s^*$ .

## 6.2 Evolutionary algorithms metaheuristics

Evolutionary algorithms, as the name implies, are a class of metaheuristics that emulate natural evolutionary processes. Sometimes the adjective "genetic" is used instead of "evolutionary". A major portion of the Michalewicz and Fogel (2000) book is devoted to the subject. Another general reference is Reeves (1993) in which several applications (with associated references) are discussed, including the travelling salesman problem, vehicle routing.

### 6.2.1 Genetic algorithm

The widespread term Evolutionary algorithms appeared in 1993 as the title of a new journal entitled Evolutionary Computation published by the MIT Press, and then it was widely used to designate all the techniques based on the metaphor of the biological evolution theory. However, some specialists use the term "Genetic Algorithms" to designate

## 2.6 An overview of proposed approaches for the VRP

---

any evolutionary technique even though they have few common points with the original propositions of Holland and Goldberg.

The genetic algorithm (GA) was invented by John Holland and his colleagues in the early 1970s, inspired by Darwin's theory. The idea behind GA is to model the natural evolution by using genetic inheritance together with Darwin's theory. In GA, the population consists of a set of solution or individuals instead of chromosomes.

Genetic algorithms are the most well known and robust methods. The most important stage in designing a GA is to represent solution with chromosome. A chromosome should be able to reflect the features of a problem and represent them properly, and produce a more suitable solution for the objective function, through an evolutionary process, by a genetic operator. To solve an optimization problem, genetic algorithm start with the chromosomal representation of a parameter set, which is to be encoded as a finite size string over an alphabet of finite length. Usually, the chromosomes are strings of 0 and 1. Each chromosome actually refers to a coded possible solution. A set of such chromosomes in a generation is called a population, the size of which may be constant or may vary from one generation to another.

The fitness/objective function is chosen depending on the problem to be solved, in such a way that the strings (possible solutions) representing good points in the search space have high fitness value. The frequently used genetic operators are selection, crossover and mutation operators. These are applied to a population of chromosomes to yield potentially new offspring.

It seems that the population diversity and the selective pressure are the two most important factors in the genetic algorithm. They are strongly related, since an increase in the selective pressure decreases the population diversity and vice versa. If the population becomes too homogeneous the mutation will almost be the only factor causing variation in the population. Therefore, it is very important to make the right choice when determining a selection method for genetic algorithm. A selection mechanism is necessary when selecting individuals for both reproducing and surviving. A few methods are available and they all try to simulate the natural selection, where stronger individuals are more likely to reproduce than the weaker ones.

The main genetic operator is crossover, which simulates a reproduction between two organisms, the parents. It works on a pair of solutions and recombines them in a certain way generating one or more offsprings. The offsprings share some of the characteristics

of the parents and in that way the characteristics are passed on to the future generation. It is not able to produce new characteristics.

Mutation is the process by which a random alteration in the genetic structure of a chromosome takes place. Its main objective is to introduce genetic diversity into the population. It may so happen that the optimal solution resides in a portion of the search space that is not represented in the population's genetic structure. The process will therefore be unable to attain the global optima. In such situations, only mutation can possibly direct the population to the optimal section of the search space by randomly altering the information in a chromosome. Mutation aids a genetic algorithm to break free from fixation at any given point in the search space.

### 6.2.2 Iterated Density Estimation Evolutionary Algorithms

In recent years, an increasing interest has been stressed on a class of competent probabilistic and graphical model based genetic algorithm (GA), which is commonly called as the Estimation of Distribution Algorithm (EDA) (see e.g. Larranaga (2002), Mühlenbein et al. (1999) and Bosman and thierens (2002)). In EDA, there is neither crossover nor mutation operators, the new solutions are generated through sampling an estimated probability distribution of the promising solutions. EDA has successfully extracted the global information about the search space so far, and has solved many problems of bounded difficulty at a single or multiple hierarchical levels requiring only polynomial number of fitness evaluations.

Although EDA has gained a significant success both in theory and application, much work still needs to be done to improve its performance. Among them is to design a flexible model to estimate the probability distribution of promising solutions. This topic commonly named as structure selection and parameters learning is the kernel of EDA which directly determines the quality of its final solution. During the process to select a proper structure, both the reliability of the selected structure and the time requirement to get such a model should be considered. It may not be an advisable alternative to build a perfect structure, but its time consumption is not practical realistic. In addition, a simple model sometimes may be better than a complex one, because a complex model usually tends to overfit the data when the population size is small.

Another deficiency of EDA is its single convergence. All the promising solutions tend to converge to one global optimal point at the final stage of evolution. This is because

## 2.7 Our personal contributions to solve some variants of the VRP problems

---

the distribution of promising solutions to be estimated is always assumed uni-modal and the structure to reflect relationships among variables is considered unity.

IDEA iterates the three steps listed below, until some termination criterion is satisfied:

1. Select good candidates (i.e., solutions) from a (initially randomly generated) population of solutions.
2. Estimate the probability distribution from the selected individuals.
3. Generate new candidates (i.e., offspring) from the estimated distribution.

### 6.2.3 Artificial Ant Colony

Artificial Ant Colony(Dorigo et al.(1996)) is a metaheuristic in which a colony of artificial ants cooperates in finding good solutions to discrete optimization problems. Each ant of the colony exploits the problem graph to search for optimal solutions. An 'artificial ant', unlike natural counterparts, has a memory in which it can store information about the path it follows. Every ant has a start state and one or more terminating conditions. The next move is selected by a probabilistic decision rule that is a function of locally available pheromone trails, heuristic values as well as the ant's memory. Ant can update the pheromone trail associated with the link it follows. Once it has built a solution, it can retrace the same path backward and update the pheromone trails. ACO algorithm is interplay of three procedures as described in Dorigo and Stutzle (2004).

## 7 Our personal contributions to solve some variants of the VRP problems

This section aims to set this work of thesis in the broad field of research on issues of vehicle routing problems. The work performed has resulted in several contributions of different types. We will study and analyze the tool used to solve the HFFVRP as well as the presence of an external carrier, and then we will treat the solution of a Dynamic Vehicle Routing Problem.

In the following we describe a generic model which combines respectively, three problems tackled in this thesis: the HFFVRP ( Eq2.8), the VRPPC (Eq 2.9) and the DVRP (Eq 2.8).

## 2.7 Our personal contributions to solve some variants of the VRP problems

---

- Parameters:

$n$ : number of customers

$m$ : number of vehicles

$n_k$ : number of vehicles of type  $k$

$t$ : elapsed service time, such that  $t \in [0, D]$

$\hat{V}$ : set of non-served customers

$v_s^k$ : last customer served by vehicle  $k$

$F_k$ : Fixed cost of vehicle  $k$

$c_{ijk}$ : Variable cost for vehicle  $k$  traveling from customer  $i$  to customer  $j$

$L_i$ : Fixed cost to supply customer  $i$  with the external transporter

$q_i$ : Demand of customer  $i$

$Q_k$ : Capacity of vehicle  $k$

$Q_k^t$ : remaining available capacity of vehicle  $k$  at time  $t$

$D_k^t$ : remaining available time of vehicle  $k$  at time  $t$  before reaching upper bound  $D$

- Decisions variables:

$$x_{ijk} = \begin{cases} 1 & \text{if the vehicle } k \text{ travels from customer } i \text{ to customer } j \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ik} = \begin{cases} 1 & \text{if the demand of customer } i \text{ is supplied by the vehicle } k \\ 0 & \text{otherwise} \end{cases}$$

$$Z_i = \begin{cases} 1 & \text{if the demand of customer } i \text{ is supplied by the external transporter} \\ 0 & \text{otherwise} \end{cases}$$

- Objectives functions:

$$\text{Minimize} \quad \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m c_{ijk} x_{ijk} \quad (2.8)$$

$$\text{Minimize} \quad \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m c_{ijk} x_{ijk} + \sum_{k=1}^m F_k y_{0k} + \sum_{i=1}^n L_i Z_i \quad (2.9)$$

- Constraints:

$$\sum_{k=1}^m y_{0k} \leq m \quad \forall k \in \{1, \dots, m\} \quad (2.10)$$

## 2.7 Our personal contributions to solve some variants of the VRP problems

---

$$\sum_{i=1}^n q_i y_{ik} \leq Q_k \quad \forall k \in \{1, \dots, m\} \quad (2.11)$$

$$\sum_{i=0}^n \sum_{k=1}^m y_{ik} x_{ijk} = 1 \quad \forall j \in \{0, \dots, n\} \quad (2.12)$$

$$\sum_{i=0}^n \sum_{j=0}^n x_{ijk} \leq n_k \quad \forall j \in \{0, \dots, n\} \quad (2.13)$$

$$\sum_{k=1}^m y_{ik} + Z_i = 1 \quad \forall i \in \{1, \dots, n\} \quad (2.14)$$

$$\sum_{k=1}^m y_{ik} = 1 \quad \forall i \in \{1, \dots, n\} \quad (2.15)$$

$$\sum_{i=1}^n q_i y_{ik} \leq Q_k^t \quad \forall k \in \{1, \dots, m\} \quad (2.16)$$

$$\sum_{i=0|i \neq h}^n x_{ihk} + \sum_{j=0|j \neq h}^n x_{hjk} = y_{ik} \quad \forall h \in \{0, \dots, n\}, k \in \{1, \dots, m\} \quad (2.17)$$

$$\sum_{i=0}^n \sum_{j=0}^n c_{ijk} x_{ijk} \leq D_k^t \quad \forall k \in \{1, \dots, m\} \quad (2.18)$$

$$\sum_{ij \in S} x_{ijk} \leq |S| - 1 \quad \forall S \subseteq \{2, \dots, n\}, k \in \{1, \dots, m\} \quad (2.19)$$

## **2.7 Our personal contributions to solve some variants of the VRP problems**

In our work, we are interested in three alternatives variant of heterogeneous fixed fleet vehicle routing problems that are grouped according to 4 area of research:

### **7.1 First line of research: A Hybrid Tabu Search to Solve the Heterogeneous Fixed Fleet Vehicle Routing Problem**

One of the most significant problems of supply chain management is the distribution of products between locations. The delivery of goods from a warehouse to local customers is a critical aspect of material logistics. The Heterogeneous Fixed Fleet Vehicle Routing Problem (HFFVRP) is a variant of the Vehicle Routing Problem (VRP) that aims to provide service to a specific customer group with minimum cost using a limited number of vehicles. We assume that the number of vehicles is fixed.

We must decide how to make the best use of the fixed fleet of vehicles. In the generic model we are interested to solve the model where the objective is to minimize the function 2.8 subject to constraints 2.10, 2.11, 2.12 and 2.13. In this work we describe a Tabu Search algorithm embedded in the Adaptive Memory (TSAM) procedure to solve the HFFVRP.

### **7.2 Second line of research: Heuristic Search Techniques to Solve the Vehicle Routing with Private Fleet and Common Carrier**

The Vehicle Routing Problem with Private fleet and common Carrier consists of serving all customers in such a way that:

Each customer is served exactly once either by a private fleet vehicle or by a common carrier vehicle, all routes associated with the private fleet start and end at the depot. Each private fleet vehicle performs only one route and the total demand of any route does not exceed the capacity of the vehicle assigned to it.

The total cost is minimized over the objective function 2.9 respecting the constraints 2.10, 2.11, 2.14. In practice, several common carriers may be used to serve any of the customers unvisited by the private fleet.

### **7.3 Third line of research: Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier.**

In this work we apply an evolutionary algorithm based on 2-opt local search to solve the vehicle routing problem with private fleet and common carrier.

### **7.4 Fourth line of research: Solving the Dynamic Vehicle Routing Problem by means of Artificial Ant Colony.**

In this area of research we use a hybrid artificial ant colony optimization algorithm to solve the Dynamic Vehicle Routing Problem such a way that:

- Not all information relevant to the planning of the routes is known by the planner when the routing process begins.
- Information can change after the initial routes have been constructed.

We are interested to solve the mathematical model which minimize the objective function 2.8 while considering the constraints 2.10, 2.15, 2.16, 2.17 and 2.18.

## **8 Conclusion**

This chapter presented a state of the art of VRP. We clarified the general framework vehicle routing and various parameters and issues that such problems can make. Turning an overview and a description of the VRP, we have listed the different variations of the vehicles routing. The VRPs in spite of many years of research, are a subject of topicality which always draws researchers attentions. It is possible to conclude that the VRP, too complex to be solved, are still very present in literature. In addition, new problems appear by adding additional constraints. These constraints will come closer and closer to reality.

After the presentation of the state of the art for the vehicle routing problems we will be concentrated to some variants that deal with heterogeneous and limited fleet. Therefore the aim of the next chapter is to provide a complete description of the M-Vehicle routing problems and to propose some applications with metaheuristics algorithm for solving these variants.



---

---

## Chapter 3

---

# M-Heterogeneous Vehicle Routing Problem

*A bad management of routing and deliveries between the sites of the same company or towards the sites involve consequent cost of transport. When shipping alternatives exist, the selection of the appropriate shipping alternative (mode) for each shipment may result in significant cost savings. In this chapter we consider two class of vehicle routing in which a fixed internal fleet is available at the warehouse and the fixed fleet in the presence of an external transporter.*

*The Heterogeneous Vehicle Routing Problem (HVRP) is a variant of Vehicle Routing Problem (VRP) that aims to provide service to a specific customer group with minimum cost using a limited number of vehicles. We assume that the number of vehicles of the fleet is fixed. We must decide how to make the best use of the fixed fleet of vehicles.*

### 1 Introduction

The modern global economy is moving increasingly towards open markets and encourages exchanges of resources across the globe. This new trend leads to greater competitiveness in the pursuit of efficiency at all levels, inter alia in the distribution of goods and services. Thus, management problems of vehicle fleets very complex and considerable size have emerged. Accordingly, any improvement in the techniques of solving this problem has a major impact on the economic front.

The management of supply chains requires the integration of different models, which hitherto (until now) were often treated separately and not jointly. In particular, production management (related to internal logistics) and transport problems (related to external logistics) have mutual interference that must be taken into account. As in production management, all types of decisions should be considered the structure of the logistics chain and its rules of operation between partners who may belong to different

legal entities.

Numerous organizations are involved in the production and distribution of goods. Very often trucks with different physical, operational and cost characteristics are available to distribute such goods, and the shipper has to decide which shipments to assign to each truck for delivery. Many manufacturers and distributors use private fleets, or common carrier, for the purpose of collecting and delivering shipments for their facilities. In addition to offering greater control over goods movement, private fleets may reduce costs over common carrier prices. Whereas common carriers typically require that shipments be processed at consolidation terminals, private fleets can transport shipments directly from origin to destination via multiple stop routes.

Planning the selection and utilization of transport vehicles is one of a trilogy of problems along with facility location and inventory policy determination that is essential to good strategic planning.

There has been growing interest in truck service selection and the deregulation of truck common carrier. This has had particular impact on organization that use privately-owned vehicles, since they are responsible for the utilization of the fleet and must make choices that determine the balance between common carrier and private carrier usage.

The common management problem of delivering finished goods or picking up raw materials involves choices: whether to use privately-owned vehicles or common carrier, and to what extent each should be used. If private vehicles are used, routing the vehicles for best utilization, sizing the vehicles, and determining the number needed are common choices that must be made. If common carrier used, rate negotiation, shipment consolidation, and routing are important consideration. Multiple shipping modes must be considered when one faces economies of scale in shipping.

There are also a multitude of firm specific factors that will make either private or common carrier more efficient and productive. Owing to its scale economies, a common carrier may be able to offer a lower price, for small shipments in particular. To take an extreme example, few companies would use a private fleet to retrieve small packages or letters from their vendors. These can be more economically handled by the postal service, packages services, or less-than-truckload (LTL) carriers.

The operations research community shows that the VRP is one of its great success stories. The interplay between theory and practice is recognized as a major driving force for this success. Many variants and extensions of VRP have been subject of research during the last four decades. Some well studied characteristics include the existence of demands, fixed fleet and heterogeneous vehicles.

## 2 Literature review

The VRP is one of the optimization problems most studied. This problem holds the attention of several researchers for many years, and everywhere in the world. The VRP was firstly introduced by Dantzig and Ramser (1959) and since then it has been widely studied. It is a complex combinatorial optimization problem. The problem involves a fleet of vehicles set-off from a depot to serve a number of customers at different geographic locations with various demands. Several authors have made a literature review that deal with vehicle routing these include those of Bodin et al. (1983), Laporte (1992a, 1992b, 1993), and Toth and Vigo (2002). Like other authors, Golden and Assad (1988) presented the problems of VRP as problem easy to explain but difficult to solve.

More specifically, two types of problems have been addressed in literature: the VRP with limited fleet and the VRP with private fleet and common carrier. Some authors have studied the problem of vehicle routing with limited fleet include Osman and Salhi (1996), Gendreau et al. (1999), Taillard (1999), Tarantilis et al. (2004), Choi and Tcha (2007) and Li et al. (2007). All these items using a mixed fleet limited but sufficient capacity to serve all customers.

VRPs are divided into various areas. In most of the studies, the vehicle fleets that are studied consist of homogeneous or heterogeneous vehicles (e.g., Tarantilis et al. (2004) and Taillard (1999)). Problems related to providing service through fixed fleets are complicated in comparison to unlimited fleet VRP. Heterogeneous VRP is studied in two different ways. On the one hand some researchers make an assumption that there are an unlimited number of vehicles of each type and they try to find the optimal set of vehicles to be scheduled in the problem. This is called the Fleet Size and Mix VRP (FSMVRP). On the other hand, other researchers study the case where there is a fixed vehicle fleet and try to schedule this fleet of vehicles to the customers in an optimal way. This problem is called Heterogeneous Fixed Fleet VRP (HFFVRP).

In the literature, three variants of VRP with heterogeneous fleet (HFVRP) have been studied. The first one is introduced by Golden et al. (1984), in which variable costs are uniformly given over all vehicle types with the number of available vehicles assumed to be unlimited for each type. The second version considers the variable costs depending on vehicle type, which is neglected in the first version. The third one, called Heterogeneous Fixed Fleet VRP (HFFVRP), generalizes the second version by limiting the number of available vehicles of each type.

Due to the complexity of the HFFVRP no exact algorithms have ever been presented

for it. It is widely studied by heuristic design as those proposed in Salhi et al. (1992) and Osman and Salhi (1996). Recently, the solution methods for the HFFVRP have substantially progressed in Rochat and Taillard (1995), Rego and Roucairol (1996), Taillard (1999), Tarantilis et al. (2004), Li et al. (2007) and Choi and Tcha (2007). Classical heuristics for the HFFVRP including the saving based algorithms are presented in Desrochers et al. (1991). Golden et al. (1984) develop a savings heuristic to solve the Fleet Size and Mix VRP as well as techniques for generating a lower bound and an underestimate of optimal solutions. Gendreau et al. (1999) have proposed the Tabu Search algorithm for the Fleet Size and Mix VRP (FSMVRP).

Taillard (1999) has presented a Heuristic Column Generation method (HCG) for solving the Heterogeneous Fixed Fleet VRP (HFFVRP). A new metaheuristic called Back-tracking Adaptative Threshold Accepting (BATA) was developed by Tarantilis et al. (2004) in order to solve the HFFVRP. Also, Li et al. (2007) developed a record-to-record travel algorithm for the Heterogeneous Fleet VRP (HFVRP). Recently, Choi and Tcha (2007) propose a column generation method to solve the HFFVRP. They have built an integer programming model and solved the linear relaxation by column generation technique.

The second extension of VRP is the VRPPC. Despite its wealth and abundance of work that are devoted to him, the VRP with Private fleet and common Carrier (VRPPC) represent only a subset of a larger family known as VRP.

At the level of routing problem with external carrier are Volgenant and Jonker (1987), which demonstrated that the problem involving a fleet of one single vehicle and external carriers can be rewritten as Traveling Salesman Problem - TSP. This problem also have been studied by Diaby and Ramesh (1995) whose objective was to decide that customers visited with external carrier and optimize the tour of remaining customers.

Several approaches have been used to solve the classical VRP, exact methods, heuristics and metaheuristics solution principally are proposed. The VRPPC is more complex problem because it involves an internal fleet of several vehicles.

To our knowledge, the VRPPC was introduced by Ball et al. (1983) were among the first authors to tackle this problem of determining optimal homogeneous fleet size in the presence of an external carrier.

Kilniewicz et al. (1990) presented the problem in a context to divide the customers into sectors and the private fleet size and common carrier must be determined for each sector. Chu (2005) put forward an interesting formulation for the VRPPC and solve it with a heuristic economies improved by inter and intra routes. There after, Bolduc et

al. (2007) have improved the results of Chu using more sophisticated customer exchanges using two different initial solutions.

Recently, Bolduc et al. (2008) used an efficient application of a local descent technique based on different neighborhood structures which is enhanced by two diversification strategies, a randomized construction procedure and a perturbation mechanism.

## 3 Heterogeneous Vehicle Routing Problem

In this section, we are interested to the HFFVRP that involves a limited number of vehicles which can be heterogeneous or homogeneous. In this problem, the aim is to provide service to the customer group with minimum cost.

### 3.1 Description of the problem

The HFFVRP can be described as follows: Let  $N = (1, \dots, n)$  be the set of customers and  $G = (V, A)$  be a directed graph where  $V = v_0, \dots, v_n$  is the vertex set and  $A = \{(v_i, v_j), v_i, v_j \in V, i \neq j\}$  is the arc set. The vertex  $v_0$  represents a depot at which is grouped a fleet of vehicles while the remaining vertices correspond to cities or customers. Each customer  $v_i$  has a non-negative demand  $q_i$ . Denote by  $z_k$  the fixed cost of a vehicle  $k$ ,  $g_k$  its variable cost per distance unit, and  $Q_k$  its capacity.  $c_{ijk}$  represent the cost of the travel from customer  $i$  to customer  $j$  with a vehicle of type  $k$ . There are a several types of vehicles, with  $T$  denoting the set of such types.  $n_k$  is the number of vehicles of type  $k$ . In this version of the HFFVRP, the values of  $n_k$  are fixed. Then, the number of vehicles of type  $k$  is limited and the fleet is known in advance. Let  $m = |T|$  represent the sum of routes realized of  $n_k$  for all types of vehicles. With each arc  $(v_i, v_j)$  is associated a distance  $d_{ij}$ .

The HFFVRP consists of designing a set of vehicle routes, each starting and ending at the depot such that each customer is visited exactly once by exactly one vehicle of the available fleet, the total demand of a route does not exceed the capacity of the vehicle assigned to it, the route length constraint is maintained, and the total cost is minimized.

Using binary decision variable:

$$x_{lk} = \begin{cases} 1 & \text{if the } l^{\text{th}} \text{ route is selected and performed by a vehicle of type } k \\ 0 & \text{otherwise} \end{cases}$$

A feasible HFFVRP solution exists only if all customers are served by exactly one vehicle of the available fleet composition. Given the above condition, the objective of HFFVRP is to minimize both the total travelling cost by vehicles and the fixed costs such

that all the problems' constraints are satisfied.

## 4 Vehicle Routing Problem with Private fleet and Common carrier

One of the most general versions of the VRP is the VRPPC, which can be formally described in the following way. Let  $G = (V, A)$  be a graph where  $V = \{0, 1, \dots, n\}$  is the vertex set and  $A = \{(i, j) : i, j \in V, i \neq j\}$  is the arc set. Vertex 0 is a depot, while the remaining vertices represent customers. A private fleet of  $m$  vehicles is available at the depot. The fixed cost of vehicle  $k$  is denoted by  $f_k$ , its capacity by  $Q_k$ , and the demand of customer  $i$  is denoted by  $q_i$ . A travel cost matrix  $(c_{ij})$  is defined on  $A$ . If travel costs are vehicle dependent, then  $c_{ij}$  can be replaced with  $c_{ijk}$ , where  $k \in 1, \dots, m$ . Each customer  $i$  can be served by a vehicle of the private fleet, in which case it is called an internal customer or by a common carrier at a cost equal to  $e_i$ , in which case it is called an external customer. The VRPPC consists of serving all customers in such a way that:

- 1) Each customer is served exactly once either by a private fleet vehicle or by a common carrier vehicle.
- 2) All routes associated with the private fleet start and end at the depot.
- 3) Each private fleet vehicle performs only one route.
- 4) The total demand of any route does not exceed the capacity of the vehicle assigned to it.
- 5) The total cost is minimized. In practice, several common carriers may be used to serve any of the customers unvisited by the private fleet.

Typically, the one selected is the lowest cost carrier. It is not necessary to specify the routes followed by the common carrier because it charges a fixed amount  $e_i$  for visiting customer  $i$ , irrespective of visit sequence.

### 4.1 Formulation of the VRPPC

#### 4.1.1 Index

$n$  : Number of customers

$m$  : Number of vehicles in the internal fleet

$i \in 0, \dots, n$  : Index of customer (depot = 0)

$j \in 0, \dots, n$  : Index of customer (depot = 0)

### 3.4 Vehicle Routing Problem with Private fleet and Common carrier

---

#### 4.1.2 Parameters

$F_k$  : Fixed cost of vehicle

$c_{ijk}$  : Variable cost for vehicle traveling from customer  $i$  to customer  $j$

$L_i$  : Fixed cost to supply customer  $i$  with the external transporter

$q_i$  : Demand of customer  $i$

$Q_k$  : Capacity of vehicle  $k$

#### 4.1.3 Decision variables

The formulation uses the following variables:

$$x_{ijk} = \begin{cases} 1 & \text{if the vehicle } k \text{ travels from customer } i \text{ to customer } j \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ik} = \begin{cases} 1 & \text{if the demand of customer } i \text{ is supplied by the vehicle } k \\ 0 & \text{otherwise} \end{cases}$$

$$Z_i = \begin{cases} 1 & \text{if the demand of customer } i \text{ is supplied by the external transporter} \\ 0 & \text{otherwise} \end{cases}$$

Bolduc et al. (2008) formulate the VRPPC as follows:

$$\text{Minimize} \quad \sum_{k=1}^m F_k y_{0k} + \sum_{i=0}^n \sum_{j=0}^n \sum_{k=1}^m c_{ijk} x_{ijk} + \sum_{i=1}^n L_i Z_i \quad (3.1)$$

subject to

$$\sum_{k=1}^m y_{0k} \leq m \quad (k = 1, \dots, m) \quad (3.2)$$

$$\sum_{j=0|j \neq h}^n x_{hjk} = \sum_{i=0|i \neq h}^n x_{i0k} = y_{hk} \quad (h \in \{0, \dots, n\}; k \in \{1, \dots, m\}) \quad (3.3)$$

$$\sum_{k=1}^m y_{ik} + Z_i = 1 \quad i \in \{1, \dots, n\} \quad (3.4)$$

$$\sum_{i=1}^n q_i y_{ik} \leq Q_k \quad k \in \{1, \dots, m\} \quad (3.5)$$

$$\sum_{ij \in S} x_{ijk} \leq |S| - 1 \quad \forall S \subseteq \{2, \dots, n\} \quad k \in \{1, \dots, m\} \quad (3.6)$$

The objective functions (3.1) minimize the transportation cost: fixed and variable cost of the internal fleet and fixed cost of the external transporter. Constraint (3.2) specify that at most  $m$  private fleets vehicles can be used in the solution, while constraint (3.3) indicate that the same vehicle  $k$  must enter and leave customer  $h$ , (3.4) apply that each customer is served either by the private truck or the external transporter. (3.5) Ensure that the vehicle capacity is never exceeded. (3.6) Are the sub tour-breaking constraints.

## 5 Metaheuristics approaches to solve the M-VRP

In this section we present three applications to solve different problem with metaheuristics. The first application is the adaptation of the tabu search algorithm to solve the heterogeneous fixed fleet vehicle routing. In this part of this thesis, a mathematical model of the HFFVRP is first given in the section 7 in chapter 2 (Literature review) conducted by the objective function 2.8 with the constraints 2.10, 2.11, 2.12 and 2.13.

The second applications consider the solving of the VRPPC. This research presents a mathematical model specified with the objective function 2.9 respecting the constraints 2.10, 2.11, 2.14 and a heuristic search technique based on the Tabu Search with ejection chains neighborhood (HST) and an Iterated Density Estimation Evolutionary Algorithm (IDEA) metaheuristics to solve the vehicle routing problem with private fleet and common carrier (VRPPC).

### 5.1 A Hybrid Tabu Search to solve the Heterogeneous Fixed Fleet Vehicle Routing Problem

Our proposed solution algorithm is based on the Adaptive Memory Procedure (e.g., Golden et al. (1997), Arntzen et al. (2006)) to solve the HFFVRP. The AMP was first proposed by Rochat and Taillard (1995) as an enhancement of TS to solve the VRP. It was motivated by the work of Glover regarding surrogate constraints, Glover (1977).

An important principle behind AMP is that good solutions may be constructed by combining different components of other good solutions. A memory containing components of visited solutions is kept. Periodically, a new solution is constructed using the data in the memory and improved by a local search procedure. The improved solution is

then used to update the memory.

Our algorithm uses a TS embedded in adaptive memory, the choice of our solution procedure is based on the successful of TS to solve a wide range of challenging problems. A key feature of TS is its use of adaptive memory to enhance a search strategy.

A pseudo code of the Adaptive Memory Procedure (AMP) is given below:

1. Initialize the memory  $M$ .
2. While a stopping criterion is not met, do:
  - Construct a new solution  $s$  combining components of  $M$ .
  - Apply a tabu Search algorithm to  $s$  (let  $s^*$  be the improved solution).
  - Update  $M$  using components of  $s^*$ .

In the sequel, we investigate and develop a TS heuristic embedded in Adaptive Memory Procedure (TSAM) to solve the HFFVRP (e.g. Euch and chabchoub (2010)).

The TS heuristic embedded in Adaptive Memory (TSAM) proposed herein for the HFFVRP can be roughly characterized into four steps: Initialization, construction of solution, solution improvements and updating the Adaptive Memory Procedure (AMP). The details of each step of the TS Adaptive Memory (TSAM) are presented below:

#### 5.1.1 Initialization (Step 1 in the Adaptive Memory)

To begin with, a certain amount of storage space called adaptive memory is allocated within the AMP. A set of routes is generated by the nearest neighbour method and is stored in the adaptive memory. The feasibility in the initialization is made with respect to the route length constraint and by avoiding the violation of capacity constraints.

For each arc  $(v_i, v_j)$  in  $A$ , we receive a profit value based on the distance travelling from  $v_i$  to  $v_j$  and the number of times this arc appears in the solution denoted by  $g(v_i, v_j)$  if the vehicle of type  $k$  traveling from customers  $v_i$  to  $v_j$  ( $v_i, v_j \in l_k$ ).

We are going to skew the selection in favour of the most profitable vertices. To generate an initial solution, define an advantage to insert the arc  $(v_i, v_j)$  in the solution by:

$$av_{ijk} = \begin{cases} \frac{(g(v_i, v_j) - c_{lk})}{c_{ijk}} & \text{if } (v_i, v_j) \in \text{routel}_k \\ \frac{1}{c_{ijk}} & \text{otherwise} \end{cases}$$

### 3.5 Metaheuristics approaches to solve the M-VRP

To choose an arc of a set  $\bar{A} \subseteq A$ , let

$$av_{\bar{A}k} = \sum_{(v_i, v_j) \in \bar{A}} av_{ijk}$$

To choose an arc  $(v_i, v_j)$  in  $\bar{A}$  we proceed by firstly selecting randomly a number  $\alpha \in [0]av_{\bar{A}k}$ . Then select the arc  $(v_{i\tau}, v_{j\tau})$  such that  $\tau$  is the smallest integer value such that  $\sum_{l=1}^{\tau} av_{i_l j_l} > \alpha$ . If  $\sum_{l=1}^{\tau_k} av_{i_l j_l} \geq \alpha$ . The procedure to generate  $m$  routes of the adaptive memory sums up as follows:

1. **Step1:** Create  $m$  vehicle routes containing only an arc  $(v_0, v_0)$ . Also create the set  $M = \{(v_i, v_j) : c_{v_0 v_i k} + c_{v_i v_j k} + c_{v_j v_0 k} \leq c_{max}\}$ . Let  $k = 1$  and move to Step 2.
2. **Step 2:** If  $\bar{A}$  contains only an arc  $(v_0, v_0)$ , select by roulette wheel an arc  $(v_i, v_j)$  from  $M$  and insert it into route  $l_k$ . Let  $\bar{A} = A - (v_i, v_j)$  and  $c_k = c_{v_0 v_i} + c_{v_i v_j} + c_{v_j v_0}$ . Otherwise continue the construction of route  $l_k$  as follows:
  - Choose with roulette wheel an arc  $(v_i, v_j) \in \bar{A}$
  - Insert an arc  $(v_i, v_j)$  and select two vertices  $p$  and  $q$  in  $l_k$ , so that the evaluation functions  $\{(av_{pvik} + av_{vivjk} + av_{vjvqk} - av_{pvqk} : c_{pvik} + c_{vivjk} + c_{vjvqk} - c_{pvqk} \leq c_{max}\}$  will be maximal.  
If such  $(v_i, v_j)$  is not found (because  $c_k + c_{pvik} + c_{vivjk} + c_{vjvqk} - c_{pvqk} \geq c_{max}$ ), go to Step 3.
  - Insert  $(v_i, v_j)$  between  $p$  and  $q$  and adjust the length  $c_{ijk}$  of route:  $l_k := c_{ijk} + c_{pvik} + c_{vivjk} + c_{vjvqk} - c_{pvqk}$ .
  - Let  $\bar{A} = A - (v_i, v_j)$
  - if  $\bar{A} \neq \emptyset$ , repeat Step 2
3. **Step3:** Let  $k = k + 1$ . If  $k \leq m$ , go back to Step 2.

The Hybrid TSAM in the initialization phase may be started from heuristic created solution. In the first Step we consider that all vehicles are at the depot. Second for every Step we select a customer based in the constructive methods described above and we insert it in the best position that minimizes the total cost. In the next phase we try to construct a solution and to repair and improve the solution constructed from the routes generated in the AMP.

#### 5.1.2 Construction of solution (step 2 in the Adaptive Memory)

The TSAM procedure starts from an initial solution  $s$  constructed in the AMP. Then to improve the solution we use the regret heuristic used by Potvin and Rousseau (1993), Liu and Shen (1999). Generate an order  $\{a_{k_1}, a_{k_2}, \dots, a_{k_N}\}$  in which the routes of the AMP are considered.

The regret heuristic works in the following way:

1. Initialization

- For every artificial vertex (chosen vertex)  $v_i \in V$  : find the closest transport vertex  $v_j$  and the second closest transport vertex  $v_z$  .
- Calculate a regret-value  $REG_i = c_{iz} - c_{ij}$ .
- Sort the regret-value in descending order
- Allocate the closest transport vertex to the artificial vertex according to this order; if a transport vertex is the closest to two or more artificial vertex, it is assigned to the one with the highest regret value.

2. Continue with the same procedure until all of the transport vertices are assigned to a tour. Always find the closest and second-closest transport vertex to the last included vertex.

#### 5.1.3 Solution improvements (Step 3 in the Adaptive Memory)

In order to improve the solution, we propose to use a TS algorithm as a local search. We give a pseudo-code of the proposed algorithm in algorithm 1.

---

**Algorithm 1** Pseudo code of Tabu local search

---

```

1: begin Tabu Search
2:  $S_0$  initial solution
3:  $S_{current} = S_0; S_{best} = S_0; \theta = 0; TL = 0$ 
4:   for ( $j = 1; j \leq nbclient; j++$ ) do
5:     //apply Nearest Neighbour
6:      $S = NearestNeighbour(S_{current});$ 
7:     for ( $i = 1; i \leq \theta_{max}; i++$ ) do
8:       //find the best non tabu solution
9:       Update  $TL$ 
10:      //use the permutation local search
11:       $\hat{S} = permutation(S_{current});$ 
12:       $S_{current} = 2 - move(\hat{S});$ 
13:       $value = evaluate(S_{current});$ 
14:      if ( $value < S_{best}$ ) then
15:         $S_{best} = value;$ 
16:         $\theta++;$ 
17:      endif
18:      Update  $TL;$ 
19:    endfor
20:  endfor
21: end Tabu Search

```

---

#### *Initial solution*

The TS starts from an initial solution  $s$  constructed with the nearest neighbourhood method where customers are placed in an array sorted in the increasing order of demand. In this method, the customer with the biggest demand is appended to a route. When the next to-be inserted customer's distance exceeds the length of cycles on the current route, a new route is initiated.

#### *Neighbourhood structures*

The TS algorithm that we have implemented uses two structures of neighbourhoods:

- Permutation-neighborhood:

Let  $v_i$  and  $v_j$  be two vertices on two different routes  $l_i(s)$  and  $l_j(s)$ . A permutation-move consists of replacing  $l_{v_i}(s)$  and  $l_{v_j}(s)$  by  $(l_{v_i}(s) - v_i) + v_j$  and  $l_{v_j}(s) - v_j + v_i$ , respectively.

- 2-move-neighborhood:

In a 2-move, vertex  $v_i$  is moved from its route to a route  $l \neq l_{v_i}(s)$ . Route  $l$  can be an empty route. Hence,  $l_{v_i}(s)$  and  $l$  are replaced by  $l_{v_i}(s) - v_i$  and  $l + v_i$ , respectively.

A conventional Tabu List (TL) contains pairs  $(v_i, l)$  with the condition that it is forbidden

to move customer  $c$  to circuit  $l$ . A move  $(v_i, l)$  is considered as Tabu if  $(v_i, l) \in TL$ . The Tabu Search is stopped where  $\theta_{max}$  iterations have been performed without improving the best solution found  $s^*$ . To improve the solution generated for each algorithm, we use two improvement procedures:

***Exchanging of arcs between two routes***

For randomly selected route  $l_1$  and  $l_2$  from the current solution  $s$ , these improvement procedures make one-arc exchange between route  $l_1$  and  $l_2$ , Brandão (2009). Starting with the first arc on  $l_1$ . We scan from the first to the last arc on  $l_2$  to examine if an exchange of the first arc on  $l_1$  with the current arc on  $l_2$  makes the total cost of the two routes shorter. The exchange is made immediately when such arc on  $l_2$  is found. Then, the procedure is repeated with the second arc on  $l_1$  and scanning arc on  $l_2$ . This procedure stops when all one-arc exchanges between  $l_1$  and  $l_2$  leading to improve tour duration have been performed.

***A random arc-insertion procedure***

For a select vehicle cycle  $\tau$  of duration  $D(\tau)$ , this semi-greedy random insertion approach randomly removes a subset of arcs from the cycle and re-inserts them in the resulting partial cycle in a greedy way.

#### 5.1.4 Updating the Adaptive Memory (Step 4 in the Adaptive Memory)

The strategy adopted in this paper is based in the framework of tabu search algorithm, but also borrows some heuristic ideas from the greedy constructive heuristics mentioned before. The main features of the algorithm are in the constructive greedy methods used in the different phases to improve the solution. Below, the implementation of each part of the TSAM to solve the HFFVRP is described. The Steps of the hybrid metaheuristic are summarised as follows:

1. **Step1:**Generate  $m$  routes derived from solution nearest neighbour methods.  
Start with  $AMP = \phi$ .
2. **Step2:** While a stopping criterion is not met, do:  
Initialize the  $m$  routes to  $AMP', s = \phi$ .  
Apply the constructive method mentioned in initialization Step.  
Repeat, while  $AMP' = \phi$   
Choose randomly a route  $l \in AMP'$ .  
Let  $s' = s' \cup l$ .

For each route  $l \in AMP'$ , where  $l \cap l' \neq \emptyset$ ,  
 Let  $AMP' = AMP' \setminus \{l'\}$   
 Apply regret heuristic

3. **Step3:** Improve the new constructed solution.

For each route  $l$  in  $s$ , let  $AMP = AMP \cup l$ . Find a solution  $s^*$ (best solution) by considering the continuous routes in  $AMP$ .

Apply Tabu Search algorithm

Exchange vertices between two routes and perform the random vertex-insertion procedure

4. **Step4:** Update the  $AMP$  by inserting the new constructed routes and removing routes (if necessary) which belong to the worst solutions.

## 5.2 Computational results

This section reports on the performance of the proposed TS heuristic embedded in Adaptive Memory (TSAM) over the benchmark test problem of Golden et al. (1998) and Li et al. (2007).

### 5.2.1 Implementation and instances

We consider two sets of instances to evaluate the performance of TSAM algorithm. The first set is composed by the eight tests problems developed by Golden et al. (1998) for the vehicle fleet size and mix routing problem which can be viewed as a special case of HFFVRP where the travel costs are the same for all vehicles types and the number of vehicles of each type is limited. The specifications for the HFFVRP problem set are given in Table 3.1. We use the numbering scheme (problem 13... problem 20) given by Golden et al. (1984).

The second set is composed by the five new tests problems developed by Li et al. (2007), selected from the large-scale vehicle routing with 200 - 360 customers from Golden et al. (1998) and adapted to the HFVRP (Table 3.2). These problems contain between 50 and 100 vertices as well as the depot all randomly located over a square. They have fixed fleet, capacity restrictions, no route length constraints, and no service times at the vertices. Moreover euclidean distances are used in the entire problem.

The algorithm described here has been implemented in C++ using Visual Studio C++ 6.0. Experiments are performed on a PC Pentium 4, 3 GHz with 512MB of RAM.

**Tableau 3.1** – Specifications of eight benchmark problems with at most six types of vehicles

Problem	n	Vehicle A			Vehicle B			Vehicle C			Vehicle D			Vehicle E			Vehicle F			%						
		$Q_A$	$f_A$	$\alpha_A$	$n_A$	$Q_B$	$f_B$	$\alpha_B$	$n_B$	$Q_C$	$f_C$	$\alpha_C$	$n_C$	$Q_D$	$f_D$	$\alpha_D$	$n_D$	$Q_E$	$f_E$		$\alpha_E$	$n_E$	$Q_F$	$f_F$	$\alpha_F$	$n_F$
13	50	20	20	1	4	30	35	1.1	2	40	50	1.2	4	70	120	1.7	4	120	225	2.5	2	200	400	3.2	1	95.39
14	50	120	100	1	4	160	1500	1.1	2	300	3500	1.4	1													88.45
15	50	50	100	1	4	100	250	1.6	3	160	450	2	2													94.76
16	50	40	100	1	2	80	200	1.6	4	140	400	2.1	3													94.76
17	75	50	25	1	4	120	80	1.2	4	200	150	1.5	2	350	320	1.8	1									95.38
18	75	20	10	1	4	50	35	1.3	4	100	100	1.9	2	150	180	2.4	2	250	400	2.9	1	400	800	3.2	1	95.38
19	100	100	500	1	4	200	1200	1.4	3	300	2100	1.7	3													76.74
20	100	60	100	1	6	140	300	1.7	4	200	500	2	3													95.92

$n$ : number of customers;

$Q_t$ : capacity of vehicles type  $t$  ( $t = A, B, C, D, E, F$ );

$f_t$ : fixed cost of vehicles type  $t$ ;

$\alpha_t$ : variable cost per unit distance of vehicles type  $t$ ;

$n_t$ : number of vehicles type  $t$  available;

%;  $100 \times (\text{totaldemand} / \text{totalcapacity})$ .

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.2** – Specifications for five new test problems with at most six types of vehicles

Problem	Vehicle A		Vehicle B		Vehicle C		Vehicle D		Vehicle E		Vehicle F		%													
	$n$	$Q_A$	$f_A$	$\alpha_{A N_A}$	$Q_B$	$f_B$	$\alpha_{B N_B}$	$Q_C$	$f_C$	$\alpha_{C N_C}$	$Q_D$	$f_D$		$\alpha_{D N_D}$	$Q_E$	$f_E$	$\alpha_{E N_E}$	$Q_F$	$f_F$	$\alpha_{F N_F}$						
H1	200	50	20	1	8	100	35	1.1	6	200	50	1.2	4	500	120	1.7	3	1000	225	2.5	1	93.02				
H2	240	50	100	1	10	100	1500	1.1	5	200	3500	1.2	5	500	120	1.7	4					96.00				
H3	280	50	100	1	10	100	250	1.1	5	200	50	1.2	5	500	120	1.7	4	1000	225	2.5	2	94.76				
H4	320	50	100	1	10	100	200	1.1	8	200	400	1.2	5	500	120	1.7	2	1000	225	2.5	2	1500	250	3	1	94.12
H5	360	50	25	1	10	100	80	1.2	8	200	150	1.5	5	500	320	1.8	1	1000	225	2.5	2	2000	250	3	1	92.31

### 5.2.2 Parameter settings

The Tabu Search procedure employs a set of parameters whose values need to be set before the algorithm is run. These parameters include the number of Tabu iteration  $N_{max}$ , tour improvement frequency  $\chi$ , tour selection parameter  $T$ , number of neighbourhood solutions generated in the Tabu Search  $\beta$  and maximum non-improvement iterations  $\theta$  and the size of tabu list  $TL$ . Our preliminary experiments show that the suitable length of tabu list increases as the ratio of the number of customers ( $n$ ) to the number of vehicles ( $m$ ) becomes larger for the considered problem. These parameters were determined as the basis of a number of preliminary runs. The values of these parameters are defined as follows:  $N_{max}=200; \chi=6; T=30; \beta=200, \theta=100$  and  $TL = 10 + n/m$ .

### 5.2.3 Evaluation method

The results produced by our algorithms have been compared with those produced by the algorithms of Taillard (1999), Tarantilis et al. (2004) and Li et al. (2007).

We begin the presentation of the results by examining, in Table 3.3, the efficiency of the procedure of TSAM. This table shows the strong performance of the TSAM algorithm in the form of the quality of solution and in the best CPU time.

**Tableau 3.3** – Computational results for TSAM algorithms on eight test problems

TSAM (Tabu Search embedded in adaptive memory)					
Problem number	n	Total vertices used	Fixed cost	variable cost	Time
13	50	15	1650	1477.34	3
14	50	6	6800	590.00	8
15	50	9	2050	1019.69	3
16	50	9	2200	1112.92	2.26
17	75	10	1035	1022.31	31.19
18	75	13	1930	1768.51	25.35
19	100	8	9500	1104.87	82.94
20	100	13	3200	1510.72	71.09

*n*: number of customers

*Time*: processing time in seconds

In Table 3.4 we describe the efficiency of the TSAM algorithm over other metaheuristics presented in the literature. It gives the comparison results between the TSAM and the other methods proposed by Taillard (1999), Tarantilis et al. (2004), Li et al. (2007). We observe that in seven out of eight test problems, the TSAM finds a better solution.

In Table 3.5 we report the comparative result on five new test problems proposed by Li et al. (2007). It is interesting to observe that Over the five large instances, four new

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.4** – A comparison of TSAM, HCG, BATA, and HRTR according to overall costs

Problem	Taillard		Tarantilis et al.		Li et al.		our algorithm	
	HCG	Time	BATA	Time	HRTR	Time	TSAM	Time
13	1518.05	476	1519,96	843	1517.84	358	<b>1477.34</b>	3
14	615.64	575	611,39	387	607.53	141	<b>590.00</b>	8
15	1016.86	335	1015,29	368	1015.29	166	1019.69	3
16	1154.05	350	1145,52	341	1144.94	188	<b>1112.92</b>	2.26
17	1071.79	2245	1071,01	363	1061.96	216	<b>1022.31</b>	31.19
18	1870.16	2876	1846,35	971	1823.58	366	<b>1768.51</b>	25.35
19	1117.51	5833	1123,83	428	1120.34	404	<b>1104.87</b>	82.94
20	1559.77	3402	1556,35	1156	1534.17	447	<b>1510.72</b>	71.09

**HCG** Heuristic column generation solution from Taillard (1999), Sun Sparc workstation, 50 MHz;

**BATA** Backtracking adaptive threshold accepting solution from Tarantilis et al.(2004), Pentium II, 400 MHz, 128MB RAM;

**HRTR** Record-to-record travel solution from Li et al.(2007), Athlon, 1 GHz, 256MB RAM;

**TSAM** Tabu search adaptative memory, Pentium IV, 3 GHz, 256MB RAM

**Tableau 3.5** – Comparative result on five new test problems

Problem	n	Li et al.		our algorithm	
		HRTR	Average CPU (s)	TSAM	Average CPU(s)
H1	200	12067.65	687.82	11742.62	413.05
H2	240	10234.40	995.27	10103.87	724.00
H3	280	16231.80	1437.56	16231.80	1060.18
H4	320	17576.10	2256.35	17529.21	1755.37
H5	360	21850.41	3276.91	20996.15	2355.47

**HRTR** Record-to-record travel solution from Li et al. (2007), Athlon, 1 GHz, 256MB RAM;

**TSAM** Tabu Search Adaptive Memory, Pentium IV, 3 GHz, 512MB RAM.

best solutions were produced with our algorithm. In the large test problems, the TSAM yields consistently better results than the HRTR metaheuristic of Li et al. (2007).

Finally, in Table 3.6, we give the relative percentage deviation of each algorithm's solution from the best known solution. A simple criterion to measure the efficiency and the quality of an algorithm is to compute the relation percentage deviation of its solution from the best solution reported in the literature on specific benchmark instances. From this table we conclude that the solution quality of the algorithms is comparable with an

### 3.5 Metaheuristics approaches to solve the M-VRP

---

average deviation that is between 1 and 5% for the eight test problems. Our algorithm still seems to be superior in terms of solution quality with an average deviation of 0.0222%.

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.6** – Percent deviation results for HFFVRP algorithms on eight test problems

Problem	Best Known solution	Taillard	Tarantilis et al.	Li et al.	Our algorithm
		<b>HCG</b>	<b>BATA</b>	<b>HRTR</b>	<b>TSAM</b>
13	1477.34	2.7556	2.8849	2.7644	0.00
14	590.00	4.3457	3.6254	2.9711	0.00
15	1015.29	1.5144	1.3577	1.3577	0.0043
16	1112.92	3.6956	2.9292	2.8771	0.00
17	1022.31	4.8400	4.7636	3.8784	0.00
18	1768.51	5.7477	4.4014	3.1139	0.00
19	1104.87	1.144	1.716	1.401	0.00
20	1510.72	3.2467	3.0204	1.5522	0.00
Average	1200,245	3.4112	3.087	2.489	0.0005

### 5.3 Heuristic Search Techniques to Solve the Vehicle Routing with Private Fleet and Common Carrier

In this section we describe a Tabu Search algorithm based on the ejection chains movements (TS/EC) to solve the VRPPC.

The TS/EC algorithm proposed herein for the VRPPC can be roughly characterized into seven steps: Initialization, Neighborhood structure, Tabu list, aspiration criterion, Intensification, Diversification, stopping criterion step. We refer the reader to the paper of Euchi and Chabchoub (2010).

#### 5.3.1 Initial solution

The initialization step consists of generating an initial solution  $s$  where all customers  $i(i = 1, \dots, n)$  are visited by construction of  $m_{max}$  tours.

To choose an initial solution, the main heuristic used are the same presented in the paper of Euchi and Chabchoub (2009).

In order to obtain an initial feasible solution, the insertion method embedded in the following algorithm is used. Algorithm 2 describes the main initial solution heuristic used.

---

**Algorithm 2** Initial solution algorithm

---

```

1: begin
2:   sort all available vehicles in increasing order of capacity
3:   for each available vehicle  $k := 1$  to  $m_{max}$  loop
4:     for each  $v_i \in V(i = 1, \dots, n)$  loop
5:       if ( $v_i \in r_{ki}$  and  $q_i \leq Q_k$ ) then
6:         Insert  $v_i$  into route  $r_{ki}$  using the insertion method
7:       end if
8:     end for
9:     customers not inserted will be assigned to the external transporter
10:    Execute 2-Opt local search
11:  end for
12: end

```

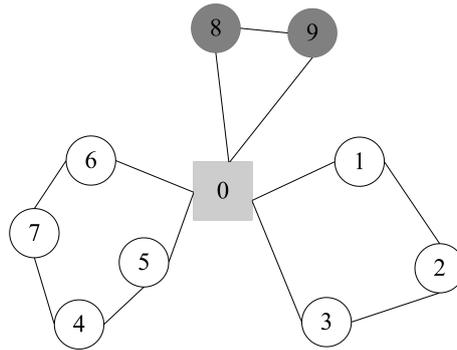
---

#### 5.3.2 Neighborhood Structure

The use of the ejection chains method requires the definition of our neighborhood structures to generate moves. The TS implementation proposed here use two structures of neighborhood to move from the current customer (current solution  $S$ ) to another one

### 3.5 Metaheuristics approaches to solve the M-VRP

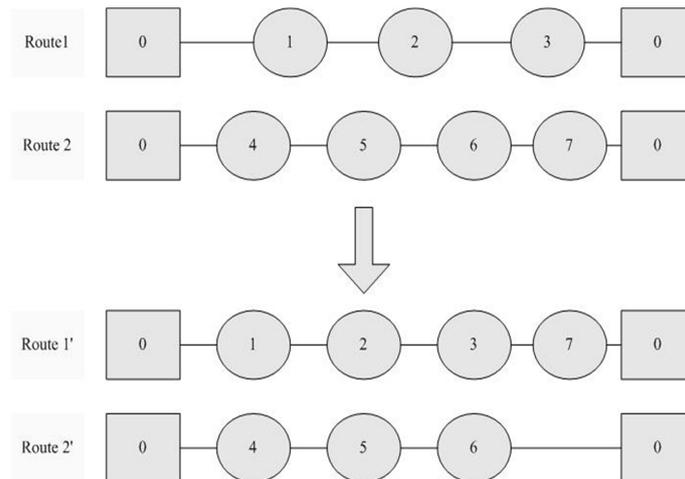
(new solution  $\hat{S}$ ) in the neighborhood of  $S$ . We adapt the 1-exchange and 2-exchange neighborhoods based on shifting and swapping vertices (customers) between a given set of routes. Figure 3.1 play an example of solution of VRP with 9 customers.



**Figure 3.1** – Example of solution of VRP with 9 customers

#### *1-exchange neighborhood*

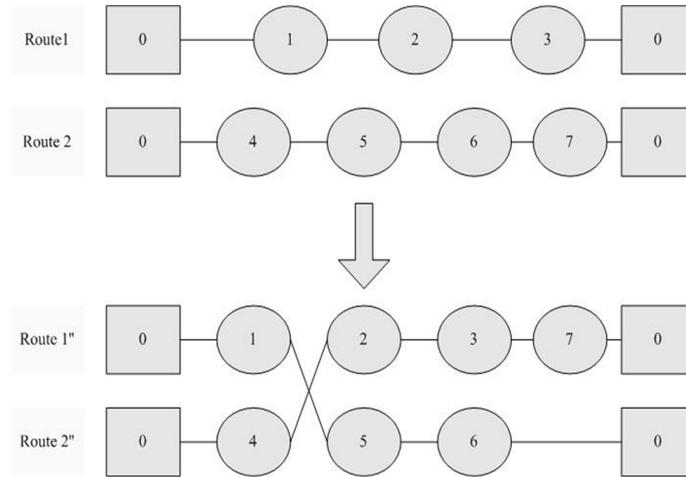
The 1-exchange neighborhood proposed in this algorithm consist in shifting a customer (vertex) from one route to another, then swapping two customers between two given routes realized by the internal fleet. When a new best solution is found, we apply the 1- exchange procedure to the route performed by the external transporter. (see the following figures 3.2, 3.3, 3.4, 3.5).



**Figure 3.2** – 1-exchange for a single customer (shift)

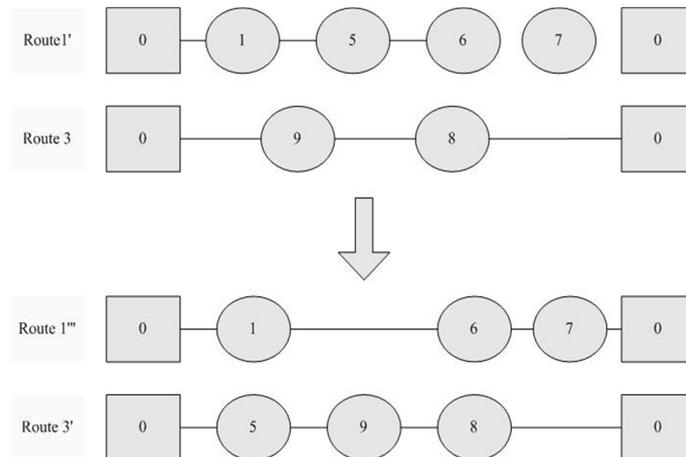
In the final step of generating a 1-exchange, when a new best solution is found, we apply the 1-exchange to the best route found with the route performed by the external

### 3.5 Metaheuristics approaches to solve the M-VRP



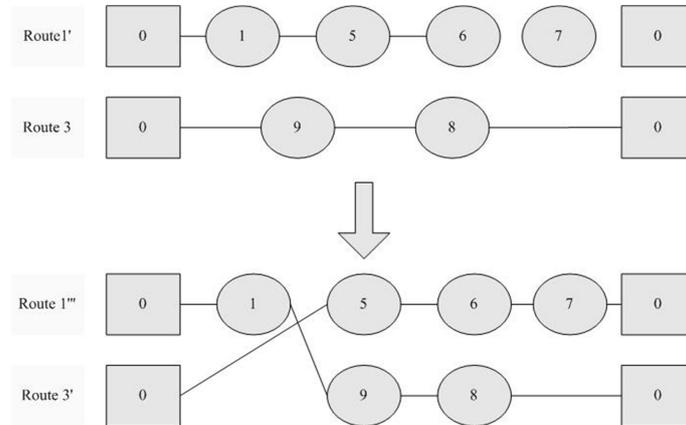
**Figure 3.3** – 1-exchange for a 2 customers (swap)

transporter. We suppose that the best routes found when applying the 1- exchange is :  
Route 1''= 0-1-5-6-7-0 then,



**Figure 3.4** – 1-exchange applying in the external transporter (shift process)

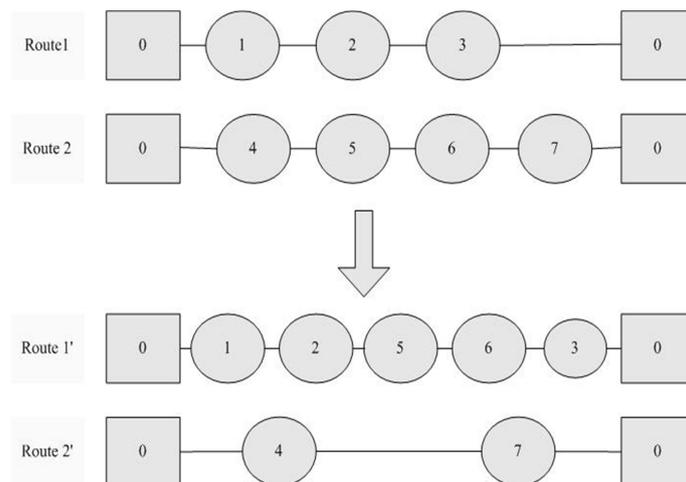
### 3.5 Metaheuristics approaches to solve the M-VRP



**Figure 3.5** – 1-exchange applying in the external transporter (swap process)

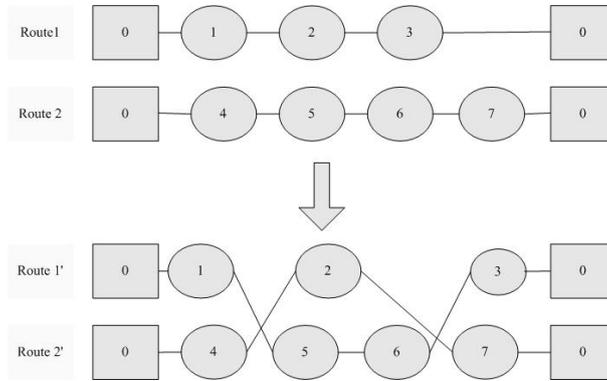
#### *2-exchange neighborhood*

The 2-exchange is the application of the 1-exchange twice. This second structure of neighborhood corresponds to moving from the current solution to a nonadjacent solution by performing the 1-exchange twice. We used the shifting and swapping moves for 2 consecutive customers between two given routes from the internal fleet, and then at each step when the solution is improved, we apply the 2-exchange to the new solution with the route performed by the external transporter. Figures 3.6, 3.7, 3.8, 3.9 shows the application of the 2-exchange to the VRP with 9 customers and 2 owned vehicles and one external transporter.

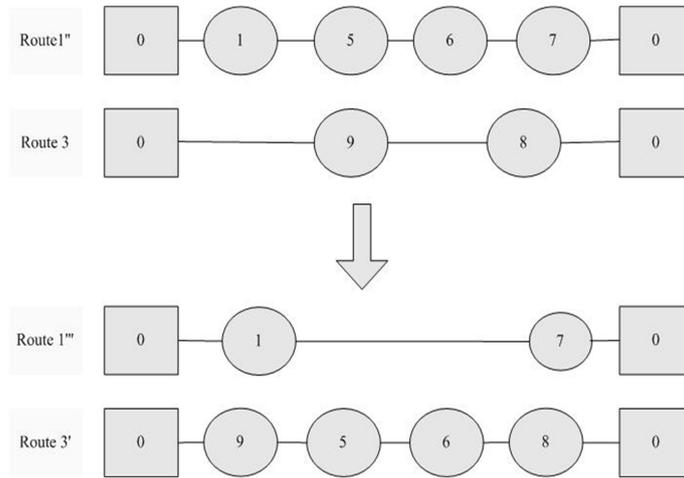


**Figure 3.6** – 2-exchange for a two customers (shift)

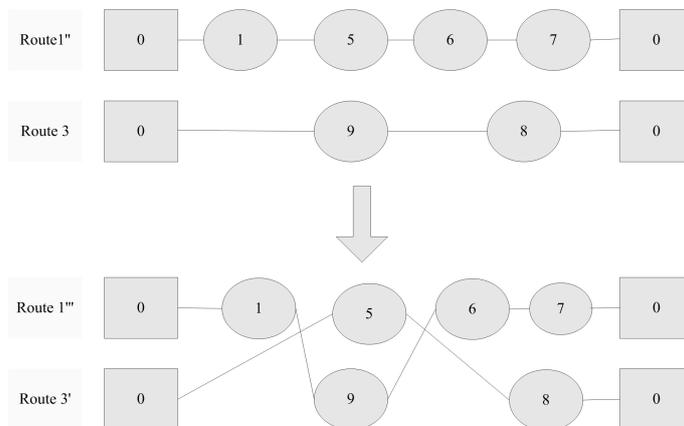
### 3.5 Metaheuristics approaches to solve the M-VRP



**Figure 3.7** – 2-exchange for a two customers (swap moves)



**Figure 3.8** – 2-exchange applying in the external transporter (shift process)



**Figure 3.9** – 2-exchange applying in the external transporter (swap process)

### 5.3.3 Tabu List

To avoid cycling, solutions possessing some attributes of recently explored ones are temporarily declared as *tabu*. The latter is achieved using a so-called short term memory. Tabu moves are represented by attributes which are stored in an ordered queue called *tabu list*. A tabu list is associated with each attribute. If the customer  $i$  is shifted from the route  $r$ . We add  $(i, r)$  as tabu and it is not possible to reinsert customer  $i$  in route  $r$  for the next iteration.

At each iteration the best solution that does not exist in the tabu list, is chosen as the new current solution. After, it is added to the tabu list and the first inserted is removed.

### 5.3.4 Aspiration criterion

The tabu status of a solution is overridden when the aspiration criterion is satisfied. It temporarily overrides the tabu status if the move is sufficiently good. In our approach, if a set of solution is selected at iteration  $t$  and this move results in a best cost for all previous iterations, then that solution is selected even if that feature is in the tabu list.

The aspiration is conducted if the solution obtained by the tabu movement is the best from any solution considered so far.

### 5.3.5 Ejection chains

At each iteration of the neighborhood search structure, ejection chains are built and the corresponding compound movements at higher evaluation is selected, if it is not tabu or satisfy the aspiration criterion.

To form a sophisticated neighborhood and to allow the search space, the ejection chains work in the following way: When the route receiving a new customer, we eject one, in a way that the solution still feasible. The customer who was ejected must naturally be integrated in another round, with the requirement by ejecting a third customer. The customer ejected is inserted to the route performed by the external transporter. Thus, at each iteration, we stay in one of the two following situations:

- The vehicle does not have a sufficient capacity to integrate the new customer. In this case, another customer must be ejected from the route and the process continues (unless the chain of the ejected-integrated customers is longer than a parameter  $h$ ).
- Or the vehicle has a sufficient capacity to integrate the customer. In this case, we have an admissible solution which we can keep it in memory before ejecting another

customer if the chain of the ejected-integrated customers is shorter than  $h$ .

Naturally, the ejection chain can be initialized from whatever  $n$  customers. Furthermore, we can try several customers during the later ejection. We can thus repeat the process for a certain number of initial customers (possibly all) and after every insertion of a customer in a route; we can also try the ejection of a number of customers outside the same route (in the external transporter). We will generally allow storing numerous acceptable solutions which will form a new neighborhood structure. If every insertion gives place to several ejections, it is clear that the size of the neighborhood grows with the value of the parameter  $h$ .

#### 5.3.6 Intensification

The principle idea of the intensification is to return periodically visit areas of the research space that seem particularly promising. In the intensification phase, the algorithm carries out for the neighborhood search from the best solution so far. If a better solution is found, the intensification phase is reactivated.

The technique adopted is rebuilding a starting solution which attempts to combine the attributes that have often been present in the best configurations. In the Tabu search algorithm the intensification step start with the best solution found created and we try to intensify the search by visiting the customers created with the routes performed with the external transporter basis on the ejection chains neighborhood.

This technique are executed periodically after the moves, if the solution is infeasible, no attempt is made to make it feasible, then the incumbent is updated, and some more time is spent in an attempt to further improve using the ejection chains. When the current intensification does not find a better solution, then a diversification phase is initialized.

#### 5.3.7 Diversification

The most common approaches here are to start from different points and to penalize frequently performed moves according to the penalty factor  $\lambda$ . The diversification phase is performed over a fixed number of iterations.

We make a diversification immediately after intensification by removing all the tabu status so as to authorize the movements which were forbidden before diversification. We thus redirect the local search towards regions of the search space not yet explored.

The key idea in the diversification is to analyze the differences between the created solutions with respect to some common attribute, and penalize search directions that

### 3.5 Metaheuristics approaches to solve the M-VRP

---

increase the number of previously selected common attributes.

#### 5.3.8 Stopping condition

The Tabu search status of a solution is overridden when these two criteria are satisfied:

- The search is stopped after a fixed number of iterations (Maximum allowable number of iteration);
- Or when a maximum allowable number of non-improving moves is reached.

Algorithm 3 describes the operational details of TS/EC procedure.

### 3.5 Metaheuristics approaches to solve the M-VRP

---

---

**Algorithm 3** Pseudo code of Tabu Search Ejection Chains algorithm

---

```
1: Notation
2:  $S_0$ :initial solution
3:  $S$ :current solution
4:  $S^*$ :Best-Known solution
5:  $N(S)$ :Neighborhood of solution  $S$ 
6:  $CL$ :Candidate List
7:  $N_{max}$ : Number of maximum iteration
8:  $N_{noimp}$ :Number of non-improvement iteration
9:  $TL$ :Tabu List
10: Begin
11:   Construct an initial solution  $S_0$  according to construction method.
12:   Generate the set  $RCL(i)$  with Eq.2
13:   Initialize  $TL, N_{max}, N_{noimp}$ 
14:   While  $(N < N_{max})$  or  $(i < N_{noimp})$  do
15:     Generate neighbor solution  $\subset N(S)$ 
16:     Apply 1-exchange neighborhood
17:     Apply 2-exchange neighborhood
18:     if  $S_0 \leftarrow S^*$  is not in  $TL$  then
19:        $S_0 := S^*$ 
20:       Update  $TL, N_{max}$ 
21:        $i++$ 
22:     else
23:       Generate solution over the ejection chain method
24:       if  $f(S) < f^*$  then
25:         Set  $f^* := f(S), S^* := S$ 
26:         Update  $TL, N_{max}$ 
27:          $i++$ 
28:       end if
29:       If no improvement of iterations is found then
30:         Apply diversification step
31:       end if
32:     end while
33: end
```

---

### 5.4 Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the VRPPC

The need for optimization using methods of operations research in VRP has become more and more important in recent years. One way of tackling the VRPPC is by allowing the search space of the IDEA with 2-opt local search. Local search remains the main practical tool for finding good solutions for large instances of the VRPPC, more details are presented in Euchí and Chabchoub (2009).

Figure 3.10 present the scheme of the IDEA / 2-opt approach.

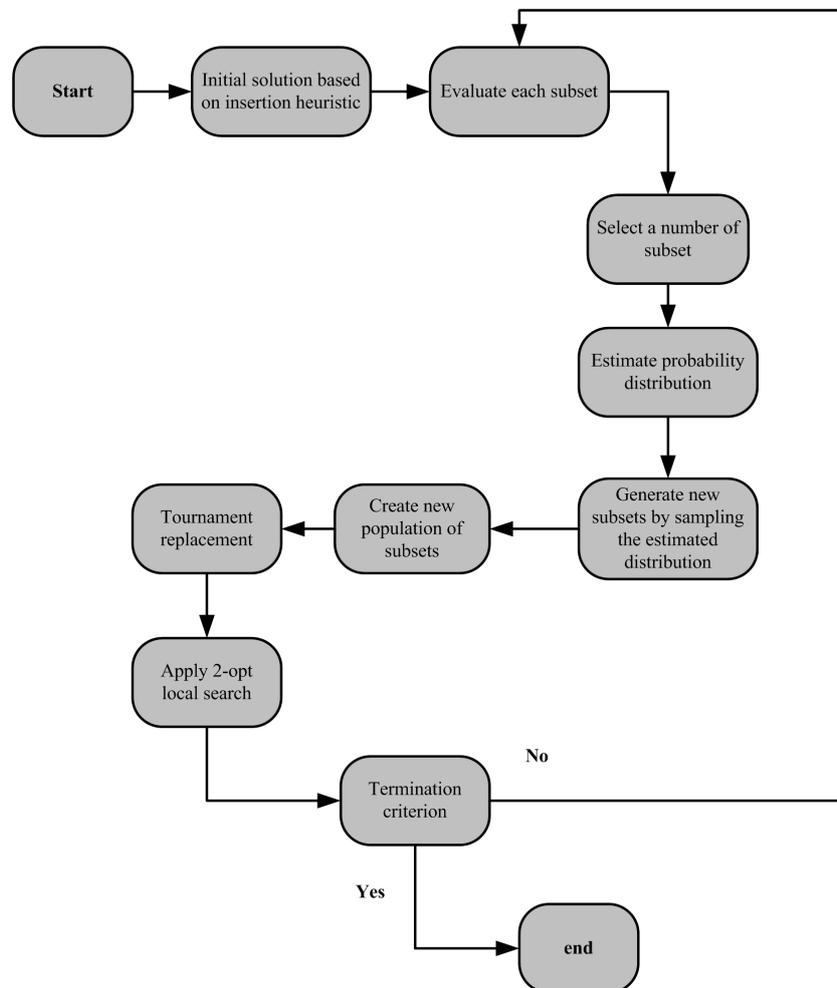


Figure 3.10 – General Scheme of the IDEA / 2-opt local search

#### 5.4.1 Hybrid IDEA to solve the VRPPC

To solve the VRPPC, it is recommended to hybridize it with a local search (e.g. Lozano et al. (2006)). In this way, we propose to use a 2-opt local search (e.g. Lin (1965)) to improve the solution generated after the creation of the initial population and after the generation of new solution. With each generation  $t$ , IDEA algorithm maintains a population  $pop(t) = \{\pi^1, \pi^2, \dots, \pi^N\}$  of  $N$  solutions and the probability matrix is

$$p(t) = \begin{pmatrix} p_{11}^t \cdots p_{1n}^t \\ \vdots \quad \vdots \\ p_{n1}^t \cdots p_{nn}^t \end{pmatrix}$$

where  $p(t)$  models the distribution of promising solutions in the search space. More precisely,  $p_{kj}(t)$  is the probability that vehicle  $k$  is assigned to customer  $j$  in the assignment.

Below the implementation of each part of the IDEA to solve the VRPPC is described, vehicle routing representation, initialization, selection operators, probabilistic model, replacement and stopping criterion.

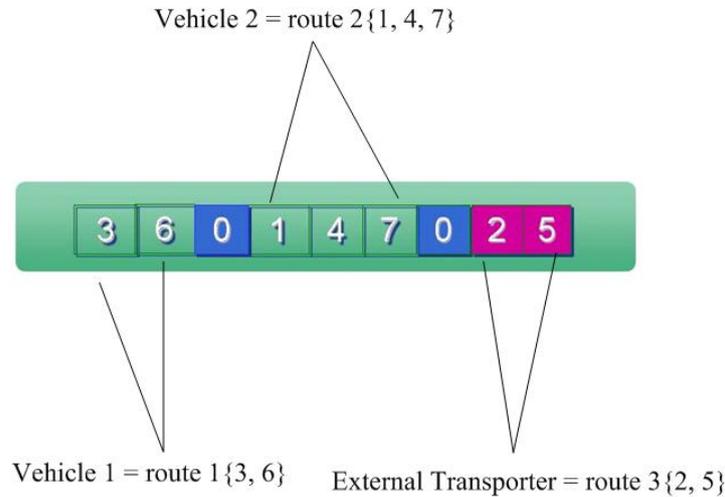
#### 5.4.2 Vehicle routing representation

A suitable presentation of solution to VRP is i.e. a chromosome consisting of several routes, each containing a subset of customers that should be visited in the same order as they appear. Every customer has to be a member of exactly one route  $r$ . The solution is coded as follows:

The solution is presented by one vector of dimension  $n + k$  where  $n$  is the number of customers and  $k$  is the number of vehicles. It is assumed that every solution is started from the depot. Each vector has a combined value of 1 to  $n$  and  $k$  value = 0. Each value  $\in [1, n]$  indicates the customer and each value 0 indicates the return to the depot (depot for the new truck) in the  $k - 1$  value 0 in the sequence. For the last value 0 indicates the end of the cycle allocated to vehicles. The last remaining customers will be assigned to the external carrier.

Figure 3.11 present the solution with one vector. Each value between 1 and  $n$  represent the index of the customers when each customer is listed in the order in which they are visited. In our example, we have 7 customers and 2 private vehicles.

The solution of the VRPPC is defined by a set of vehicles and route where each route has a sequence to serve customers.



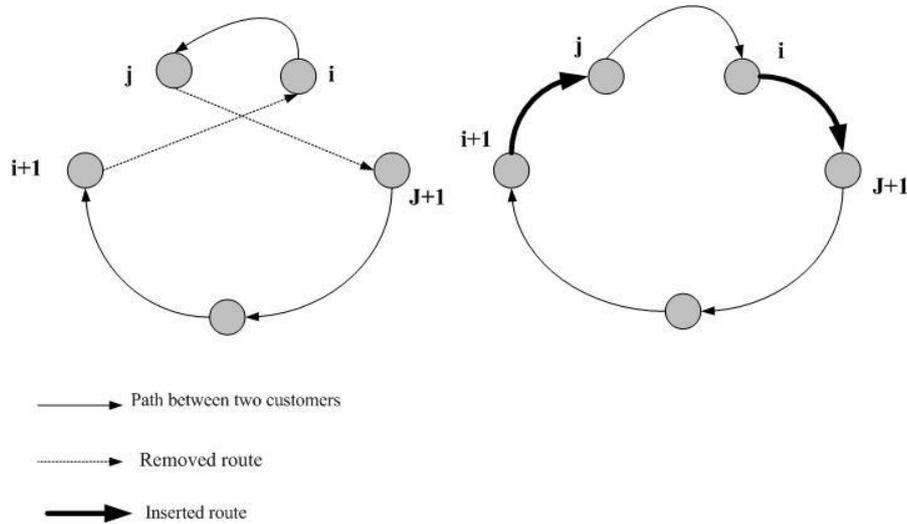
**Figure 3.11** – vehicle routing representation

The solution is represented as follows:

- Internal fleet:
  - Vehicle1 = route 1 = 3, 6;
  - Vehicle2 = route 2 = 1, 4, 7;
- External fleet:
  - External transporter = route 3 = 2, 5;

#### 5.4.3 2-opt Local Search

To improve the solution we have used the 2-opt local search. This is an intra route movement. This procedure is used in the following way. Let  $\pi$  be a solution for the VRPPC. Then its 2-opt neighbourhood  $N(\pi)$  is defined as a set of all possible solutions resulting from  $\pi$  by swapping two distinct nodes. This operator involves the substitution of two arcs,  $(i, j)$  and  $(i + 1, j + 1)$  with two others arcs,  $(i, i + 1)$  and  $(j, j + 1)$ , and the reversal of path  $p(i + 1, j)$ . Customer  $i$  on route 1 and customer  $j$  on route 2 are exchanged in the best possible way to result in a cost savings. If such a better solution is found, it replaces the current solution and the search continues. The process is repeated until no further reduction of route length is possible. Figure 3.12 illustrate the main 2-opt procedure used in this application.



**Figure 3.12** – 2-Opt intra route movement

#### 5.4.4 Initialization

In this section, insertion heuristic is developed and used for obtaining an initial solution. IDEA / 2-opt in the initialization start with the insertion heuristic and then apply the 2-opt local search to improve it as in Euchi and Chabchoub (2009).

When inserting a customer  $v_i$  into a route  $r_k$ , the least cost of insertion,  $C_{ik}$  is computed in the following manner. A customer can be inserted into a non used vehicle, with at least one customer in it. When assigning a customer to a vehicle on the route, the type of loads (demand) required by the customer and the vehicle type servicing the customer must be compatible. In order to obtain an initial feasible solution, the following algorithm 4 is used.

---

**Algorithm 4** Pseudo code of initial solution algorithm

---

- 1: **sort** all available vehicles in increasing order of capacity
  - 2:     **for** each available vehicle  $k := 1$  to  $K_{max}$  **loop**
  - 3:         **for** each  $v_i \in V(i = 1, \dots, n)$  **loop**
  - 4:             **if** ( $v_i \in r_{ki}$  and  $q_i < Q_k$ ) **then**
  - 5:                 insert a customer  $v_i$  into route  $r_{ki}$  using the insertion method
  - 6:             **end if**
  - 7:         **end for**
  - 8:     customers not inserted will be assigned to the external transporter
  - 9:     Execute 2-opt local search
  - 10:    **end for**
  - 11: **end**
- 

The last remaining customers not inserted with the insertion method will be assigned

to the external carrier.

The  $N$  resultant solutions  $\{\pi^1, \pi^2, \dots, \pi^N\}$  constitute the initial population  $Pop(0)$ . The initial probability matrix  $p(0)$  is set as  $p_{ij} = \frac{1}{N}$ .

Then the probability matrix  $p(t)$  can be updated as follows:

$$p_{ij}(t) = (1 - \beta) \frac{1}{N} \sum_{k=1}^N I_{ij}(\pi^k) + \beta p_{ij}(t-1), (1 \leq i, j \leq n) \text{ where}$$

$$I_{ij}(\pi) = \begin{cases} 1 & \text{if } \pi(i) = j \\ 0 & \text{otherwise} \end{cases}$$

$0 \leq \beta \leq 1$  is a learning rate. The bigger  $\beta$  is, the greater the contribution of the solutions in  $pop(t)$  is to the probability matrix  $p(t)$ .

#### 5.4.5 Selection operators

Selection in evolutionary algorithms meant to select the better solutions of the population to perform the variation. In our IDEA/ 2-opt local search algorithm we use the selection Pressure Towards Diversity of Bosman et al. (2002). The selection operator proposed are presented as follows: the selection operator selects  $\lceil \tau pop \rceil$  solutions, where  $pop$  is the population size and  $\tau \in \left[ \frac{1}{pop}, 1 \right]$  is the selection percentile.

#### 5.4.6 Probabilistic Model

One of the successful ways of IDEA is the use of a probabilistic model that captures the important correlations of the search distribution, assigning high probability values to the selected solution. The IDEA builds a probabilistic model with the best individuals and then sample the model to generate new ones.

This section presents an efficient technique for learning probabilistic model on the base of a probability matrix  $p = (p_{ij})_{n \times n}$ , we generate a new solution.

In order to create a new solution based on the probabilistic model we proceed as follows:

1. Divides the vehicles into two groups based on their capacity. The first group has assigned to  $\lceil \alpha n \rceil$  customers and the second one to  $n - \lceil \alpha n \rceil$  customers.
2. Vehicle  $k$  is assigned to location  $\pi(k)$ , which is the location for this customer in solution  $\pi$ .

### 3.5 Metaheuristics approaches to solve the M-VRP

---

3. Arranges the vehicles in the second group sequentially, based on the probability matrix  $p(t)$ .
4. Customers not served are assigned to the external transporter.

#### 5.4.7 Replacement

The role of the replacement is keeping the population size constant. To do so, some individuals from the population have to be substituted by some of individuals created during the probabilistic model. This can be done using the tournament replacement.

A subset of  $\alpha$  individuals is selected at random, and the worst one is selected for replacement (for  $\alpha > 1$ ).

#### 5.4.8 Stopping criterion

The stopping criterion is inherent to the complexity of used probabilistic model.

We use a maximum number of iteration  $Max_{iter}$  and maximum of execution time  $t_{max}$  as a stopping condition. We can give a pseudo-code of the proposed approach in the following algorithm 5.

---

#### Algorithm 5 Pseudo code of IDEA/2-Opt algorithm

---

```

1:  $P$ : Population size
2:  $\pi^0$ : initial solution
3:  $\pi^*, \pi^{**}$ : intermediate solution
4:  $F$ : Fitness value
5: begin
6:    $t:=0$ ;
7:    $P$ :population ( $t$ ); // initialization using the insertion heuristics
8:    $\pi^0 = \pi^0 \cup$ (Insertion heuristic)
9:    $\pi^* = 2$ -Opt local search ( $\pi^0$ ) // selection operator and probabilistic model
10:  While (stopping condition is not met) do
11:     $S =$  selection ( $pop(t)$ );
12:     $M =$  learn model ( $S$ );
13:     $Pop = S$ ;
14:    for ( $i \leq \lfloor \tau pop \rfloor$  ton  $\leq 1$ ) do
15:       $Pop = Pop \cup (Pop_s(z))$  // improve solution with local search
16:       $\pi^{**} = 2$ -Opt local search( $\pi^*$ )
17:      if ( $F(\pi^{**}) > F(\pi^*)$ ) then
18:         $\pi^* = \pi^{**}$ ;
19:      end if
20:    end for
21:     $t = t + 1$ ;
22:  endwhile
23: end

```

---

## 5.5 Experiments results

In this section we discuss the performance of our IDEA / 2-opt local search and the tabu search with ejection chains algorithm when applied on a wide set of instances taken from the literature.

### 5.5.1 Implementation and instances

In order to evaluate and to compare the performance of the proposed algorithms we will compare it with the results presented in the literature. We used the same set of test cases reported in Bolduc et al. (2008). Two sets instances were used to assess the performance of our algorithms. For the 34 instances on the first set (Table 3.7), the fleet is composed of a limited number of homogeneous vehicles, while the fleet for the 44 instances of the second set (Table 3.8) is limited and heterogeneous.

The first set is divided into two subsets: the 14 instances proposed by Christofides & Eilon (1969) and the 20 instances proposed by Golden et al. (1998). The second set is the same as presented in Bolduc et al. (2008) with the same notations and parameters. The set of instances with heterogeneous fleet (Table 3.8) are composed with the five small instances beginning with Chu-H used by Chu (2005), and the five instances B-H used by Bolduc et al. (2008) and the set generated by Christofides and Eilon (1969) and from those of Golden et al. (1998) (CE-H and G-H).

The algorithm described here has been implemented in C++ using Visual Studio C++ 6.0. Experiments are performed on a PC Pentium 4, 3.2 GHz with 512MB of RAM.

The description of the instances for the VRPPC are presented as follows:  $n$  is the number of customers,  $\bar{n}$  is the average number of customers per route,  $m$  is the number of vehicles,  $q_{min}$  and  $q_{max}$  are the lowest and highest demands, respectively,  $Q$  is the capacity of vehicle,  $f$  is the fixed cost and the vehicle variable cost is set equal to 1 per unit of distance.

### 5.5.2 Parameter settings

The TSEC and IDEA procedures employ a set of parameters which values need to be set before the algorithms are runs.

- Calibration of parameter for the TSEC algorithm :

The TSEC procedure employs a set of parameters include the number of iteration  $N_{max}$ , the number of non improvement tabu iteration  $N_{noimp}$ , tabu list  $TL$ , tour improve-

### 3.5 Metaheuristics approaches to solve the M-VRP

---

**Tableau 3.7** – Characteristics of instances with homogeneous limited fleet

Instances	$n$	$\bar{n}$	$m$	$Q$	$q_{min}$	$q_{max}$	$f$	$c$
CE-01	50	12,5	4	160	3	41	120	1
CE-02	75	8,3	9	140	1	37	100	1
CE-03	100	16,7	6	200	1	41	140	1
CE-04	150	16,7	9	200	1	41	120	1
CE-05	199	15,3	13	200	1	41	100	1
CE-06	50	12,5	4	160	3	41	140	1
CE-07	75	8,3	9	140	1	37	120	1
CE-08	100	16,7	6	200	1	41	160	1
CE-09	150	15	10	200	1	41	120	1
CE-10	199	15,3	13	200	1	41	120	1
CE-11	120	20	6	200	2	35	180	1
CE-12	100	12,5	8	200	10	50	120	1
CE-13	120	20	6	200	2	35	260	1
CE-14	100	14,3	7	200	10	50	140	1
G-01	240	34,3	7	550	10	30	820	1
G-02	320	40	8	700	10	30	1060	1
G-03	400	50	8	900	10	30	1380	1
G-04	480	60	8	1000	10	30	1720	1
G-05	200	50	4	900	10	30	1620	1
G-06	280	56	5	900	10	30	1700	1
G-07	360	51,4	7	900	10	30	1460	1
G-08	440	55	8	900	10	30	1480	1
G-09	255	23,2	11	1000	29	300	60	1
G-10	323	24,9	13	1000	25	300	60	1
G-11	399	28,5	14	1000	23	300	80	1
G-12	483	32,3	15	1000	20	300	80	1
G-13	252	12	21	1000	60	300	60	1
G-14	320	13,9	23	1000	52	300	60	1
G-15	396	15,2	26	1000	47	300	60	1
G-16	480	16,6	29	1000	42	300	60	1
G-17	240	13,3	18	200	10	40	40	1
G-18	300	13,6	22	200	10	40	60	1
G-19	360	13,9	26	200	10	40	60	1
G-20	420	13,6	31	200	10	40	60	1

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.8** – Characteristics of the instances with heterogeneous limited fleet

Instances	$n$	Vehicle A				Vehicle B				Vehicle C			
		$m$	$Q$	$f$	$c$	$m$	$Q$	$f$	$c$	$m$	$Q$	$f$	$c$
Chu-H-01	5	1	40	60	1.50	1	30	50	1.50				
Chu-H-02	10	1	75	120	1.50	1	65	100	1.50				
Chu-H-03	15	1	110	150	1.50	1	100	140	1.50	1	90	130	1.50
Chu-H-04	22	1	4500	250	1.50	1	4000	200	1.50				
Chu-H-05	29	1	4500	250	1.50	1	4000	200	1.50	1	3500	180	1.50
B-H-01	5	1	40	60	1.50	1	30	50	1.50				
B-H-02	10	1	75	120	1.50	1	65	100	1.50				
B-H-03	15	1	110	150	1.50	1	100	140	1.50	1	90	130	1.50
B-H-04	22	1	4500	250	1.50	1	4000	200	1.50				
B-H-05	29	1	4500	250	1.50	1	4000	200	1.50	1	3500	180	1.50
CE-H-01	50	2	160	140	1.00	2	192	168	1.00				
CE-H-02	75	4	112	80	1.00	5	168	120	1.00				
CE-H-03	100	2	160	112	1.00	2	200	140	1.00	2	240	168	1.00
CE-H-04	150	2	160	96	1.00	4	200	120	1.00	3	240	144	1.00
CE-H-05	199	7	160	80	1.00	5	200	100	1.00	2	240	120	1.00
CE-H-06	50	1	128	112	1.00	2	160	140	1.00	1	192	168	1.00
CE-H-07	75	4	112	96	1.00	3	140	120	1.00	2	168	144	1.00
CE-H-08	100	1	160	128	1.00	1	200	160	1.00	4	240	192	1.00
CE-H-09	150	4	160	96	1.00	3	200	120	1.00	3	240	144	1.00
CE-H-10	199	2	160	96	1.00	5	200	120	1.00	6	240	144	1.00
CE-H-11	120	2	160	144	1.00	2	200	180	1.00	2	240	216	1.00
CE-H-12	100	2	160	96	1.00	3	200	120	1.00	3	240	144	1.00
CE-H-13	120	1	160	208	1.00	4	200	260	1.00	1	240	312	1.00
CE-H-14	100	1	160	96	1.00	1	200	120	1.00	5	240	144	1.00
G-H-01	240	3	440	656	1.00	1	50	820	1.00	3	660	984	1.00
G-H-02	320	2	560	848	1.00	2	700	1060	1.00	4	840	1272	1.00
G-H-03	400	3	720	1104	1.00	3	900	1380	1.00	2	1080	1656	1.00
G-H-04	480	2	800	1376	1.00	4	1000	1720	1.00	2	1200	2064	1.00
G-H-05	200	2	720	1296	1.00	2	900	1620	1.00				
G-H-06	280	3	720	1360	1.00	2	900	1700	1.00	1	1080	2040	1.00
G-H-07	360	3	720	1168	1.00	1	900	1460	1.00	3	1080	1752	1.00
G-H-08	440	1	720	1184	1.00	2	900	1480	1.00	5	1080	1776	1.00
G-H-09	255	6	800	48	1.00	3	1000	60	1.00	3	1200	72	1.00
G-H-10	323	3	800	48	1.00	3	1000	60	1.00	6	1200	72	1.00
G-H-11	399	6	800	64	1.00	8	1000	80	1.00	1	1200	96	1.00
G-H-12	483	6	800	64	1.00	6	1000	80	1.00	4	1200	96	1.00
G-H-13	252	6	800	48	1.00	4	1000	60	1.00	10	1200	72	1.00
G-H-14	320	11	800	48	1.00	2	1000	60	1.00	11	1200	72	1.00
G-H-15	396	7	800	48	1.00	9	1000	60	1.00	10	1200	72	1.00
G-H-16	480	12	800	48	1.00	6	1000	60	1.00	11	1200	72	1.00
G-H-17	240	4	160	32	1.00	7	200	40	1.00	6	240	48	1.00
G-H-18	300	7	160	48	1.00	9	200	60	1.00	6	240	72	1.00
G-H-19	360	9	160	48	1.00	7	200	60	1.00	10	240	72	1.00
G-H-20	420	16	160	48	1.00	6	200	60	1.00	10	240	72	1.00

### 3.5 Metaheuristics approaches to solve the M-VRP

---

ment frequency  $\chi$ , tour selection parameter  $T$  and neighborhood size  $\beta$ , Tabu tenure and penalty factor  $\mu$ . In our algorithm after a number of preliminary runs we choose the good values of these parameters.

In this section the TS/EC for the VRPPC can be characterized by its capability to intensify and diversify the search by using the tour improvement frequency  $\chi$ . The number of candidate tours that will be considered for selection in initial solution construction  $T=30$  and the number of neighborhood solutions generated in the TS procedure  $\beta=100$  are chosen randomly. The tabu tenure is selected, based on the sum of the improvements from consecutive improving moves rather than from an individual move.

The objective function value is increased by the frequency of the move weighted by a penalty factor  $\mu$ . Thus, this mechanism plays an important role on diversifying the search by decreasing the possibility to select a frequently examined non improving move as the solution for the next iteration. The value of  $\mu$  is fixed to 0.5 and it affects the quality solution.

Tabu search contains several elements and strategies that should be explored to produce good solution for complex problems. The performance of our algorithm with regard to both solution quality and computation time comes from the exploitation of the tabu search framework and the choice of good parameters. The values of these parameters were determined in the basis of a number of preliminary runs. Table 3.9 present the values of these parameters.

**Tableau 3.9** – TS/EC parameters

Parameter Type	Value
Number of maximum iteration	500
Maximum non improvement iteration	200
Tour improvement frequency	6
Number of candidate tours	30
Number of neighborhood solutions	100
Tabu tenure	3
Penalty factor	0.5

- Calibration of parameter for the IDEA algorithm :

In order to have a uniform comparison, the algorithm were evaluated with a maximal number of iteration = 500, a selection percentage of  $\tau = \frac{100}{5} = 20\%$ , a population size = 100. We experimented with various factor  $\alpha \in \{1, 10\}$ . We run the algorithm to find the best results within a maximum number of  $10^5$  evaluations (After 10 runs).

### 5.5.3 Evaluation method

In order to evaluate and to compare the performance of the proposed TS with the ejection chains algorithm many experiments results were conducted.

In particular, computation results reported in this paper are compared with the best results proposed by Bolduc et al. (2008). We indicate the best and average solution value as well as the average computation time in seconds over 10 runs.

Results obtained with TS/EC algorithm over 10 runs are given in Table 3.10 and 3.11. We report in Table 3.10 results of RIP metaheuristic of Bolduc et al. (2008) and the basic TS/EC corresponding to the first set of 34 instances with homogeneous limited fleet. In Table 3.11 we present the results corresponding to the second set with heterogeneous fleet. We show that for the homogeneous and heterogeneous instances, a new best solution was found.

The solution quality of the TS/EC algorithm is measured in terms of cumulative total travel distance required to serve all demands of a given instance.

The computational results in Table 3.10 shows that our algorithm produces better results than other heuristic provides in the literature, is on average 0.12% over the best known solutions produced by the TS/EC and with the 0.24% for the RIP metaheuristic.

Table 3.11 shows for the heterogeneous limited fleet instances, that the TS/EC provides a good solution than the RIP metaheuristic with an average 0.37% and 0.96% for the RIP over the best known solution provided by the TS/EC. The results indicate that our algorithm always generates better results than the others that are tested on the whole of 44 instances. The second observation is that the values of CPU time are very weak and varied according to the size of the problem.

It can be seen from these Tables (3.10, 3.11) the TS/EC algorithm finds reasonably good and best execution time in all instances, TS/EC algorithm is faster than the RIP metaheuristic with respect of the conversion factor.

The generally favorable factor has the fast obtaining of the optimal solution from the greedy solution proposed at the beginning of the algorithm and the use of the ejection chain in order to find the neighborhood of a current given solution.

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.10** – Comparative result for the homogeneous instances

Instances	TS/EC			RIP metaheuristic	
	Best	Average	CPU (s)	Best	CPU(s)
CE-01	1118,47	1119,47	17	1132.91	25
CE-02	1810,52	1819,02	39	1835.76	73
CE-03	1940,2	1956,89	76	1959.65	107
CE-04	2538,991	2552,17	175	2545.72	250
CE-05	3124,7	3175,18	287	3172.22	474
CE-06	1207,4	1208,51	13	1208.33	25
CE-07	2006,52	2010,17	60	2006.52	71
CE-08	2072,05	2081,33	81	2082.75	110
CE-09	2438,5	2443,27	232	2443.94	260
CE-10	3429,71	3439,81	401	3464.90	478
CE-11	2332,8	2332,91	159	2333.03	195
CE-12	1953,55	1953,55	82	1953.55	128
CE-13	2859,4	2863,18	110	2864.21	188
CE-14	2213,08	2221,32	97	2224.63	110
G-01	14330,8	14336,71	407	14388.58	651
G-02	19269,804	19335,54	1102	19505.00	1178
G-03	24659,5	24925,28	1859	24978.17	2061
G-04	34675,58	34695,29	2668	34957.98	3027
G-05	14340,28	14375,42	397	14683.03	589
G-06	21838,47	21838,81	825	22260.19	1021
G-07	23713,26	23925,11	1339	23963.36	1628
G-08	30360,7	30423,48	1894	30496.18	2419
G-09	1336,25	1336,25	756	1341.17	832
G-10	1605,44	1617,18	1157	1612.09	1294
G-11	2166,45	2180,12	1366	2198.45	2004
G-12	2518,43	2520,12	2592	2521.79	2900
G-13	2268,22	2284,33	740	2286.91	802
G-14	2736,49	2744,11	1196	2750.75	1251
G-15	3194,52	3212,59	1741	3216.99	1862
G-16	3684,28	3691,45	2652	3693.62	2778
G-17	1695,82	1700,05	527	1701.58	806
G-18	2752,99	2764,71	1198	2765.92	1303
G-19	3544,63	3553,16	1221	3576.92	1903
G-20	4367,59	4369,03	2423	4378.13	2800
%deviation		0.12%	879.08	0.24%	1074.14

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.11** – Comparative result for the heterogeneous instances

Instances	TS/EC			RIP metaheuristic	
	Best	Average	CPU (s)	Best	CPU(s)
Chu-H-01	387,50	387,50	0.00	387,50	0.35
Chu-H-02	586.00	586.00	0.08	586.00	1.9
Chu-H-03	826,50	826,50	1.02	826,50	3.50
Chu-H-04	1389.00	1389.00	3.00	1389.00	5.85
Chu-H-05	1441,50	1441,50	5.70	1444.50	10.40
B-H-01	423,50	423,50	1.30	423,50	1.85
B-H-02	476,50	476,50	2.00	476,50	3.50
B-H-03	777	777	2.80	778.5	4.75
B-H-04	1564,50	1564,50	10.23	1564,50	15.85
B-H-05	1609,50	1609,50	10.29	1609,50	12.90
CE-H-01	1168,23	1183,91	15.9	1192.72	26
CE-H-02	1753,35	1781,59	53.26	1798.26	72
CE-H-03	1861,93	1922.35	90.7	1934.85	105
CE-H-04	2492,32	2493,14	195.91	2493.93	251
CE-H-05	3126,99	3177,99	407.33	3195.66	470
CE-H-06	1169,84	1202,99	17.2	1210.23	25
CE-H-07	2034,5	2041,58	62.1	2042.79	74
CE-H-08	2005,19	2009,17	106.41	2015.72	112
CE-H-09	2433,28	2444,94	257.16	2445.88	267
CE-H-10	2995,64	3265,84	259.39	3304.69	482
CE-H-11	2303,06	2306,42	112.22	2308.76	188
CE-H-12	1902,05	1902,05	106.94	1908.74	130
CE-H-13	2739,35	2824,99	156.4	2842.18	195
CE-H-14	1707,24	1752,15	100.2	1920.36	114
G-H-01	14216,9	14295,22	566.22	14408.31	647
G-H-02	18508,31	18557,82	836.9	18663.15	1254
G-H-03	25368,27	25411,28	1954.82	25561.55	2053
G-H-04	35272,66	35486,27	1996.89	35495.66	2049
G-H-05	16119	16121,01	441.39	16138.50	512
G-H-06	20065,81	20183,03	991.5	20329.04	1005
G-H-07	24691,2	24741,77	1207.6	24840.83	1608
G-H-08	27442,6	27691,33	2098.37	27710.66	2584
G-H-09	1313,63	1322,75	521.46	1346.03	814
G-H-10	1373,19	1374,1	1035.07	1575.82	1332
G-H-11	2203,9	2208,61	2005.4	2218.91	2140
G-H-12	2363,21	2368,13	2235.10	2510.07	2970
G-H-13	2213,39	2214,35	2006.25	2253.45	733
G-H-14	2652,28	2663,5	2635.77	2711.81	1246
G-H-15	3063	3116,49	1335.53	3156.93	1895
G-H-16	3433,8	3535,29	2635.77	3649.09	2785
G-H-17	1663,19	1668,01	635	1705.48	762
G-H-18	2723,39	2733,38	1005	2759.99	1299
G-H-19	3363,99	3378,77	1235.41	3517.48	1892
G-H-20	4363,88	4382,63	2135.89	4413.82	2733
%deviation		0.37%	715.74	0.96%	793.29

### 3.5 Metaheuristics approaches to solve the M-VRP

Figure 3.13 and 3.14 present the comparison of results between the IDEA / 2-opt, SRI and RIP metaheuristic in the generation space of homogeneous and heterogeneous instances. According to the figures we notice that the average percentage deviation tends towards zero for the IDEA / 2-opt algorithm both in the homogeneous and heterogeneous instances. In fact, the IDEA / 2-opt finds slightly fitter solutions than the previous approach using SRI and RIP metaheuristic.

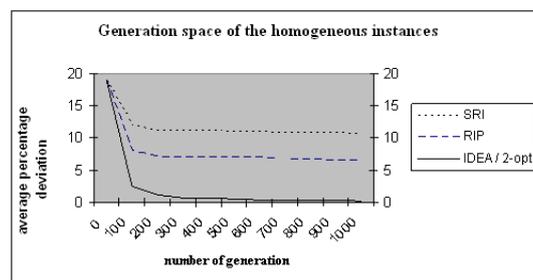


Figure 3.13 – Comparison results for the homogeneous instances

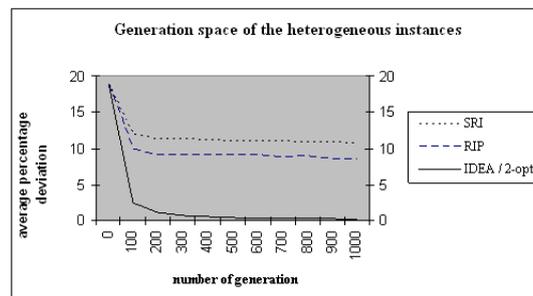


Figure 3.14 – Comparison results for the heterogeneous instances

Figure 3.15 indicate how the proposed probabilistic model in the generation of new solution in the VRPPC. We measure the efficiency of the IDEA / 2-opt changes during the optimization. Finally, in figure 3.16 we observed that with our proposed approach, we obtain quickly, a best fitness value in a number of reasonable generations.

### 3.5 Metaheuristics approaches to solve the M-VRP

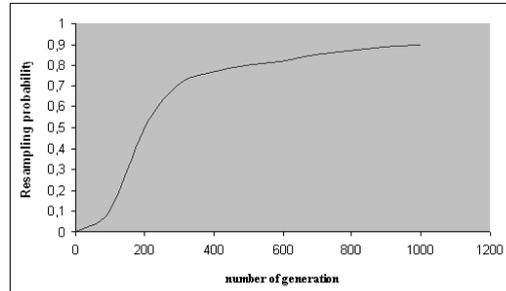


Figure 3.15 – Effect of the number of generation in the fitness value

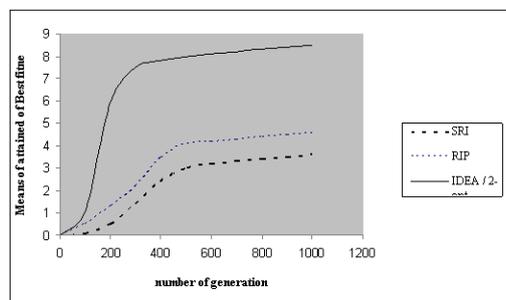


Figure 3.16 – Efficiency of the IDEA/ 2-opt

Table 3.12 gives the solution values for the instances of homogeneous fleet, while Table 3.13 contains results for the heterogeneous fleet instances. These Tables compare the results obtained with IDEA / 2-opt to those obtained with the SRI and RIP metaheuristic of Bolduc et al. (2008) for each instances.

From these tables, we can see that the IDEA/2-opt with 15000 and 30000 iterations is on average 0.23% and 0.007% respectively over the best known solutions with the 0.63% for the RIP metaheuristic. Table 14 show that the IDEA/2-opt with 15000 and 30000 iterations is on average 0.99% and 0.24% with 1.2% for the RIP metaheuristic over the best known solutions. The results in Tables 3.12 and 3.13 clearly show the strong performance of hybrid IDEA algorithm.

Over the 34 homogeneous instances, 31 best solutions were produced with our algorithm. So, for all heterogeneous instances, a new best solution was found. Although IDEA / 2-opt local search is better than the RIP metaheuristic with a good parameter setting.

We want to turn to the evaluation of the solution quality obtained by hybridization between the IDEA and the local search 2-opt over time. To that end Tables 3.12, 3.13 show for each instance the time needed to obtain solution. It can be seen from this

### 3.5 Metaheuristics approaches to solve the M-VRP

---

table that IDEA / 2-opt algorithm finds reasonably good and best execution time in all instances.

For the homogeneous instances we save the CPU time on average 346.33 and 438.28 seconds for the IDEA/2-opt with 15000 and 30000 iterations respectively, then for the heterogeneous instances we report 326.805 and 418.92 seconds. We conclude that IDEA/2-opt algorithm is faster than the RIP metaheuristic with respect of the conversion ratio.

Tables 3.14 and 3.15 gives the best solution founded with the TSEC and the IDEA/2-opt with the others methods reported in Bolduc et al. (2008).

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.12** – Results for the homogeneous instances

Instances	IDEA (15000 iteration)			IDEA (30000 iteration)			RIP	
	BS	Average	CPU	BS	Average	CPU	Solution	CPU
CE-01	1119.47	1119.47	4.21	1119.47	1119.47	7.49	1132.91	25
CE-02	1814.52	1818.12	2.55	1814.52	1814.52	3.73	1835.76	73
CE-03	1920.36	1946.97	15.71	1920.36	1920.36	22.98	1959.65	107
CE-04	2520.58	2550.74	35.64	2511.63	2521.50	37.72	2545.72	250
CE-05	3102.54	3173.81	110.04	3087.95	3099.81	172.31	3172.22	474
CE-06	1211.05	1217.51	110.98	1204.56	1206.99	131.21	1208.33	25
CE-07	2006.52	2010.17	114.53	2004.02	2008.37	159.14	2006.52	71
CE-08	2050.32	2071.33	28.56	2045.63	2051.55	48.09	2082.75	110
CE-09	2431.12	2473.72	50.67	2427.99	2431.12	53.26	2443.94	260
CE-10	3403.94	3413.81	110.29	3391.23	3404.98	112.86	3464.90	478
CE-11	2332.21	2332.21	23.15	23.29.01	2333.01	26.41	2333.03	195
CE-12	1953.55	1953.55	13.82	1950.64	1952.88	18.21	1953.55	128
CE-13	2861.39	2864.21	27.51	2857.03	2859.17	31.78	2864.21	188
CE-14	2214.14	2222.32	15.79	2214.14	2217.04	19.57	2224.63	110
G-01	14206.51	14206.51	81.12	14206.51	14206.51	81.12	14388.58	651
G-02	19171.62	19235.64	201.63	19169.84	19234.69	220.98	19 505.00	1178
G-03	24925.28	24925.28	1071.46	24763.11	24956.83	1145.91	24 978.17	2061
G-04	34645.19	34645.19	936.52	34601.79	34618.23	1985.27	34 957.98	3027
G-05	14249.82	14325.42	602.53	14249.82	14292.57	654.09	14 683.03	589
G-06	21550.39	21838.81	559.73	21550.39	21550.39	612.99	22 260.19	1021
G-07	23525.15	23925.11	764.10	23525.15	23525.15	958.36	23 963.36	1628
G-08	30025.13	30123.48	875.13	30025.13	30179.06	1072.75	30 496.18	2419
G-09	1321.73	1326.38	93.41	1316.53	1316.53	103.91	1341.17	832
G-10	1599.01	1607.18	194.23	1583.10	1596.71	204.30	1612.09	1294
G-11	2229.56	2230.12	850.99	2123.98	3180.44	1020.36	2198.45	2004
G-12	2485.00	2496.12	765.91	2485	2485.01	825.66	2521.79	2900
G-13	2268.32	2288.33	72.18	2266.21	2266.21	78.56	2286.91	802
G-14	2746.06	2704.01	548.7	2688.31	2708.99	698.71	2750.75	1251
G-15	3141.72	3242.59	1079.08	3104.68	3142.98	1249.13	3216.99	1862
G-16	3601.23	3695.45	980.31	3595.22	3598.69	1180.49	3693.62	2778
G-17	1685.22	1710.05	57.36	1631.29	1631.29	65.92	1701.58	806
G-18	2701.05	2766.15	105.16	2686.54	2688.32	125.97	2765.92	1303
G-19	3513.16	3513.16	745.69	3413.56	3421.98	988.21	3576.92	1903
G-20	4361.74	4460.38	526.77	4268.12	4360.08	784.12	4378.13	2800
%deviation	0.23%	0.49%	346.33	0.007%	0.13%	438.28	0.63%	1074.14

**BS:**Best solution

**CPU:**average CPU time en seconds

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.13** – Results for the heterogeneous instances

Instances	IDEA (15000 iteration)			IDEA (30000 iteration)			RIP	
	BS	Average	CPU	BS	Average	CPU	Solution	CPU
Chu-H-01	387.5	387.5	0.13	387.5	387.5	0.02	387.5	0.35
Chu-H-02	586.0	586.0	0.18	586.0	586.0	0.08	586.0	1.9
Chu-H-03	826.5	826.5	1.05	826.5	826.5	1.02	826.5	3.5
Chu-H-04	1389.0	1389.0	3.2	1389.0	1389.0	3	1389.0	5.85
Chu-H-05	1444.5	1444.5	6.15	1441.5	1441.5	5.7	1444.5	10.4
B-H-01	423.5	423.5	2.2	423.5	423.5	1.3	423.5	1.85
B-H-02	476.5	476.5	2	476.5	476.5	2	476.5	3.5
B-H-03	778.5	778.5	3.4	777	777	2.8	778.5	4.75
B-H-04	1564.5	1564.5	8.5	1564.5	1564.5	10.23	1564.5	15.85
B-H-05	1609.5	1609.5	9.26	1609.5	1609.5	10.29	1609.5	12.9
CE-H-01	1183.91	1186.19	3.58	1183.91	1183.91	5.18	1192.72	26
CE-H-02	1781.59	1781.59	12.15	1781.59	1781.59	16.15	1798.26	72
CE-H-03	1921.77	1938.16	40.58	1918.29	1920.35	45.98	1934.85	105
CE-H-04	2446.95	2446.95	20.62	2428.22	2431.14	24.76	2493.93	251
CE-H-05	3142.81	3159.66	245.61	2059.86	2117.07	321.83	3195.66	490
CE-H-06	1204.36	1204.89	0.55	1202.99	1202.99	1.51	1210.23	25
CE-H-07	2036.81	2040.86	10.54	2014.65	2021.78	14.6	2042.79	74
CE-H-08	1984.49	1989.36	59.37	1981.17	1981.17	77.18	2015.72	112
CE-H-09	2433.52	2436.87	87.04	2409.46	2424.94	95.73	2445.88	267
CE-H-10	3225.41	3304.21	182.05	3207.20	3265.84	191.78	3304.69	482
CE-H-11	2315.56	2348.30	72.53	2246.87	2269.42	93.62	2308.76	188
CE-H-12	1902.05	1908.54	14.25	1902.05	1902.05	23.74	1908.74	130
CE-H-13	2836.11	2840.19	88.42	2814.10	2824.99	90.47	2842.18	195
CE-H-14	1913.87	1923.44	8.54	1903.01	1912.15	12.44	1920.36	114
G-H-01	14382.07	14405.41	410.78	14345.09	14395.22	489.39	14408.31	647
G-H-02	18537.40	18557.24	769.16	18506.39	18557.82	852.33	18663.15	1254
G-H-03	25482.0	25559.09	743.24	25309.61	25511.8	972.66	25561.55	2053
G-H-04	34797.50	34960.39	678.43	34473.26	34886.27	952.57	35495.66	2049
G-H-05	15685.69	15711.43	358.39	15609.30	15621.01	376.36	16138.50	512
G-H-06	19987.69	20280.11	666.01	19960.21	20183.03	787.10	20329.04	1005
G-H-07	23639.51	23877.03	402.35	23410.63	23415.77	651.42	24840.83	1608
G-H-08	27529.83	27704.67	1001.25	27410.59	27691.33	1279.05	27710.66	2584
G-H-09	1309.41	1314.18	388.67	1304.09	1306.57	476.03	1346.03	814
G-H-10	1553.28	1573.59	741.45	1540.66	1544.10	975.70	1575.82	1332
G-H-11	2207.44	2216.93	979.98	2179.20	2198.61	1194.02	2218.91	2140
G-H-12	2455.97	2489.76	970.28	2420.02	2488.38	1682.73	2510.07	2970
G-H-13	2216.13	2221.17	368.16	2209.29	2214.35	438.71	2253.45	733
G-H-14	2698.15	2722.66	559.88	2602.33	2652.28	732.21	2711.81	1246
G-H-15	3144.65	3149.45	950.21	3101.90	3126.67	1132.09	3156.93	1895
G-H-16	3629.03	3644.28	735.69	3623.12	3635.92	856.93	3649.09	2785
G-H-17	1673.07	1702.71	400.80	1661.96	1668.01	548.05	1705.48	762
G-H-18	2732.36	2740.59	593.50	2718.41	2733.38	809.24	2759.99	1299
G-H-19	3478.48	3488.30	807.61	3408.60	3480.26	991.69	3517.48	1892
G-H-20	4382.12	4391.83	971.68	4322.97	4382.63	1182.87	4413.82	2733
%deviation	0.79%	0.99%	326.805	0%	0.24%	418.92	1.20%	791.90

**BS:**Best solution

**CPU:**average CPU time en seconds

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.14** – Best known solution for the homogeneous limited fleet instances

Instances	Best Solution	SRI		RIP		IDEA/2-opt		TS/EC	
	Z	Z	CPU	Z	CPU	Z	CPU	Z	CPU
CE-01	1119.47	1199.99	0	1119.47	25	1119.47	7.49	1118,47	17.00
CE-02	1814.52	1890.33	0	1814.52	73	1814.52	3.73	1810,52	39.00
CE-03	1920.36	2050.33	1	1937.23	107	1920.36	22.98	1940,2	76.00
CE-04	2511.63	2694.72	1	2528.36	250	2511.63	37.72	2538,991	175.00
CE-05	3087.95	3228.67	3	3107.04	474	3087.95	172.31	3124,7	287.00
CE-06	1204.56	1282.94	0	1207.47	25	1204.56	131.21	1207,4	13.00
CE-07	2004.02	2092.32	0	2006.52	71	2004.02	159.14	2006,52	60.00
CE-08	2045.63	2163.32	1	2052.05	110	2045.63	48.09	2072,05	81.00
CE-09	2427.99	2526.82	1	2436.02	260	2427.99	53.26	2438,5	232.00
CE-10	3391.23	3511.02	3	3407.13	478	3391.23	112.86	3429,71	401.00
CE-11	2329.01	2375.71	1	2332.21	195	2329.01	26.41	2332,8	159.00
CE-12	1950.64	2037.54	0	1953.55	128	1950.64	18.21	1953,55	82.00
CE-13	2857.03	2916.21	1	2858.94	188	2857.03	31.78	2859,4	110.00
CE-14	2214.14	2220.77	1	2216.68	110	2214.14	19.57	2213,08	97.00
G-01	14 160.77	14675.33	4	14160.77	651	14206.51	81.12	14330,8	407.00
G-02	19169.84	20108.84	9	19234.03	1178	19169.84	220.98	19269,804	1102.00
G-03	24 646.79	26046.80	16	24646.79	2061	24763.11	1145.91	24659,5	1859.00
G-04	34601.79	36234.51	27	34607.12	3027	34601.79	1985.27	34675,58	2668.00
G-05	14249.82	15751.31	5	14249.82	589	14249.82	654.09	14340,28	397.00
G-06	21550.39	23255.65	8	21703.54	1021	21550.39	612.99	21838,47	825.00
G-07	23525.15	25298.48	13	23549.53	1628	23525.15	958.36	23713,26	1339.00
G-08	30025.13	30899.74	18	30173.53	2419	30025.13	1072.75	30360,7	1894.00
G-09	1316.53	1378.67	4	1336.91	832	1316.53	103.91	1336,25	756.00
G-10	1583.10	1646.91	8	1598.76	1294	1583.10	204.30	1605,44	1157.00
G-11	2123.98	2238.57	14	2179.71	2004	2123.98	1020.36	2166,45	1366.00
G-12	2485	2597.14	17	2503.71	2900	2485	825.66	2518,43	2592.00
G-13	2266.21	2339.93	5	2268.32	802	2266.21	78.56	2268,22	740.00
G-14	2688.31	2825.76	8	2704.01	1251	2688.31	698.71	2736,49	1196.00
G-15	3104.68	3269.96	12	3171.20	1862	3104.68	1249.13	3194,52	1741.00
G-16	3595.22	3784.63	19	3654.20	2778	3595.22	1180.49	3684,28	2652.00
G-17	1631.29	1732.70	5	1677.22	806	1631.29	65.92	1695,82	527.00
G-18	2686.54	2821.82	8	2742.72	1303	2686.54	125.97	2752,99	1198.00
G-19	3413.56	3614.59	11	3528.36	1903	3413.56	988.21	3544,63	1221.00
G-20	4268.12	4439.45	15	4352.95	2800	4268.12	784.12	4367,59	2423.00

**SRI:** Selection - Routing - Improvement algorithm of Bolduc et al. (2007).

**RIP:** Randomized construction- Improvement-Perturbation algorithm of Bolduc et al. (2008). Xeon 3.6 GHz processor and 1.00 GB of RAM under Windows XP.

**IDEA/2-opt:** Iterated Density Estimation Evolutionary algorithm with 2-opt local search

**TS/EC:** Tabu search heuristic with ejection chains neighborhoods

### 3.5 Metaheuristics approaches to solve the M-VRP

**Tableau 3.15** – Best known solution for the heterogeneous limited fleet instances

Instances	Best Solution	SRI		RIP		IDEA/2-opt		TS/EC	
	Z	Z	CPU	Z	CPU	Z	CPU	Z	CPU
Chu-H-01	387.5	387.5	0.00	387.5	0.35	387.5	0.00	387,50	0.00
Chu-H-02	586.0	586.0	0.02	586.0	1.9	586.0	0.08	586.00	0.08
Chu-H-03	826.5	826.5	0.03	826.5	3.5	826.5	1.02	826,50	1.02
Chu-H-04	1389.0	1389.0	0.08	1389.0	5.85	1389.0	3	1389.00	3.00
Chu-H-05	1441.5	1444.5	0.09	1444.5	10.4	1441.5	5.7	1441,50	5.70
B-H-01	423.5	423.5	0.02	423.5	1.85	423.5	0.3	423,50	1.30
B-H-02	476.5	476.5	0.02	476.5	3.5	476.5	1.02	476,50	2.00
B-H-03	777	804.0	0.03	778.5	4.75	777	2.8	777	2.80
B-H-04	1564.5	1564.5	0.09	1564.5	15.85	1564.5	7.23	1564,50	10.23
B-H-05	1609.5	1609.5	0.13	1609.5	12.9	1609.5	8.29	1609,50	10.29
CE-H-01	1183.91	1 220.72	0	1191.70	26	1183.91	3.58	1168,23	15.9
CE-H-02	1781.59	1858.24	0	1790.67	72	1781.59	12.15	1753,35	53.26
CE-H-03	1918.29	1999.91	1	1919.05	105	1918.29	45.98	1861,93	90.7
CE-H-04	2428.22	2615.95	1	2475.16	251	2428.22	24.76	2492,32	195.91
CE-H-05	2059.86	3248.26	3	3146.45	490	2059.86	321.83	3126,99	407.33
CE-H-06	1202.99	1264.72	0	1204.48	25	1202.99	1.51	1169,84	17.2
CE-H-07	2014.65	2093.48	1	2025.98	74	2014.65	14.6	2034,5	62.1
CE-H-08	1981.17	2058.81	0	1984.36	112	1981.17	77.18	2005,19	106.41
CE-H-09	2409.46	2570.57	2	2438.73	267	2409.46	95.73	2433,28	257.16
CE-H-10	3207.20	3391.25	3	3267.85	482	3207.20	191.78	2995,64	259.39
CE-H-11	2246.87	2334.41	1	2303.13	188	2246.87	93.62	2303,06	112.22
CE-H-12	1902.05	1924.92	0	1908.74	130	1902.05	23.74	1902,05	106.94
CE-H-13	2814.10	2925.27	1	2842.18	195	2814.10	90.47	2739,35	156.4
CE-H-14	1903.01	1957.63	1	1907.74	114	1903.01	12.44	1707,24	100.2
G-H-01	14345.09	14599.16	4	14251.75	647	14345.09	489.39	14216,9	566.22
G-H-02	18506.39	18945.77	13	18560.07	1254	18506.39	852.33	18508,31	836.9
G-H-03	25309.61	26151.24	13	25356.93	2053	25309.61	972.66	25368,27	1954.82
G-H-04	34473.26	36519.42	22	34589.11	2049	34473.26	952.57	35272,66	1996.89
G-H-05	15609.30	17173.22	3	15667.13	512	15609.30	376.36	16119	441.39
G-H-06	19960.21	21083.42	8	19975.32	1005	19960.21	787.10	20065,81	991.5
G-H-07	23410.63	24854.96	14	23510.98	1608	23410.63	651.42	24691,2	1207.6
G-H-08	27410.59	28412.97	21	27420.68	2584	27410.59	1279.05	27442,6	2098.37
G-H-09	1304.09	1371.98	5	1331.83	814	1304.09	476.03	1313,63	521.46
G-H-10	1540.66	1599.77	8	1561.52	1332	1540.66	975.70	1373,19	1035.07
G-H-11	2179.20	2249.11	14	2195.31	2140	2179.20	1194.02	2203,9	2005.4
G-H-12	2420.02	2573.81	19	2487.38	2970	2420.02	1682.73	2363,21	2235.10
G-H-13	2209.29	2325.09	5	2239.18	733	2209.29	438.71	2213,39	2006.25
G-H-14	2602.33	2783.74	10	2682.85	1246	2602.33	732.21	2652,28	2635.77
G-H-15	3101.90	3224.50	13	3131.89	1895	3101.90	1132.09	3063	1335.53
G-H-16	3623.12	3740.85	22	3629.41	2785	3623.12	856.93	3433,8	2635.77
G-H-17	1661.96	1741.66	4	1695.75	762	1661.96	548.05	1663,19	635.00
G-H-18	2718.41	2787.10	7	2740.05	1299	2718.41	809.24	2723,39	1005.00
G-H-19	3408.60	3518.50	11	3464.70	1892	3408.60	991.69	3363,99	1235.41
G-H-20	4322.97	4362.31	15	4352.35	2733	4322.97	1182.87	4363,88	2135.89

## 6 Conclusions

In this chapter we have introduced two types of VRP. The HFFVRP and the VRPPC. We have presented a survey for this variants and we have described the specification and the formulation of the basic VRPPC variant.

The general HFFVRP and VRPPC problems are at least as complex as the VRP, which known to be NP-hard. Which prove the necessity of using heuristic/metaheuristic approaches for solving the considered problems which is the purpose of the second part of the chapter.

This work presents a TS heuristic embedded in Adaptive Memory (Euchi and Chabchoub (2009)). The computational results obtained with the TSAM metaheuristic on a set of benchmark instances compare favourably to existing literature, both with respect to solution quality and to computation time. The results of this research show that the performance of the proposed metaheuristic (TSAM) is competitive when compared with other approaches presented in the literature.

We have discussed the VRPPC and we have proposed a TS/EC algorithm to solve this variant of VRP (Euchi and Chabchoub (2010)). The proposed TS algorithm provides the best results and it outperforms those obtained by the SRI and RIP metaheuristic proposed in Bolduc et al. (2008) on the same problem sets. The initial solution and the ejection chains neighbourhoods and its mechanism to create new solution allow a comparatively large pool of good and diversified solutions to be stored and used during the search process, alternating between small and large neighbourhood stages during the metaheuristic course.

The results showed that the solution produced by our proposed approach was highly dependent on the choice of the initial solution and the use of ejection chains neighbourhood. The algorithm requires minimal computation time and it is very performing according to similar experiment presented in the literature. The results of this research show that the proposed metaheuristic is a very effective tool for finding good solutions for the VRPPC.

We have proposed an Iterated Density Estimation Evolutionary Algorithm with 2-opt local search to solve the VRPPC (Euchi and Chabchoub (2009a, 2009b), and Euchi et al. (2011)). The main features of this metaheuristic are a simple and flexible local search as well as an acceptance criterion for the search space and the use of the probabilistic model. In the process of comparing the algorithm, the contributions of the local search operator and the use of probabilistic model to the performance of the IDEA are also displayed.

The combination of the IDEA / 2-opt local search to solve the VRPPC has been shown to result in efficient algorithms for combinatorial optimization.

The results demonstrated the competitiveness and accuracy of proposed IDEA algorithm. From the experiments carried out here we can conclude that the IDEA algorithm obtained the best solution, although it was the best performer on the data sets. Moreover, the IDEA runs quickly, even for problems with many variables. The results of this research show that the proposed metaheuristics is a very effective tool for finding good solutions for the VRPPC.

We remind that this thesis has dealt with three variants of heterogeneous vehicle routing problem with fixed fleet while addressing the following special cases where the fleet is limited and heterogeneous, presence of the external carrier and the case of dynamic routing. In this chapter we have discussed the two cases presented below. Therefore the purpose of the next chapter is the study of the dynamic pickup and delivery routing problems. In chapter 4 we seek to develop a solution methodology based on artificial ant colony with 2-Opt local search to solve the considered problem.



---

---

# Chapter 4

---

## The Dynamic Vehicle Routing Problem

*The increased awareness in just-in-time supply systems with the apparition of the new advances in communication and information technologies, have recently led researchers to focus on dynamic vehicle routing problems. So, in this chapter we will scrutinize the dynamic pickup and delivery vehicle routing problem.*

*First, we discuss the differences between the traditional static vehicle routing problems and its dynamic counterparts and we conducted a literature review to get an overview of the work that has been done on the various issues. Second, we propose a solution technique to solve this variant of dynamic PDVRP.*

### 1 Introduction

The globalization of the economy leads to a rapidly growing exchange of goods on our planet. Transportation and logistics play an important role in many companies. Optimizing the processes that are involved directly influences a company's efficiency and hence can lead to better revenues. With the advances in technology in communication and navigation, companies can exert an increasing amount of direct control on their transportation and logistics processes. In almost all developed economies, and increasingly in developing ones as well, transport has become a major problem for policy-makers.

Despite short-term fluctuations in the economic climate, international trade continues to grow at a remarkable rate. Leontiades<sup>1</sup> notes that: ( Leontiades J.E. (1985) Multinational Business Strategy, D.C. Heath & Co., Lexington, MA.)

One of the most important phenomena of the 20th century has been the international expansion of industry. Today, virtually all major firms have a significant and growing presence in business outside their country of origin.

This trade is based on the recognition that an organization can buy things from a

supplier in one country, use logistics to move them, and then sell them at a profit to a customer in another country. Improved communications, transport, financial arrangements, trading agreements, and so on, mean that organizations search the world to find the best location for their operations. Then international logistics move the related materials through long and complex supply chains.

Any organization involved in transport has to consider many types of problem. We have discussed some of the most important, but there are always operational details to consider. The number of vehicles needed, type and size, special features required, routes used, assignment of loads and customers to vehicles, schedules, maintenance schedules, measures of service and quality, and so on. A basic routing problem looks for the best path for a delivery vehicle around a set of customers. There are many variations on this problem, all of which are notoriously difficult to solve.

An important subtask in this context is the operational planning of trucks or other specialized transportation vehicles. These optimization tasks are called Vehicle Routing Problems (VRP).

The high complexity and intractable nature of the dynamic VRP forms an attractive row of research motivated by the significance of the transport industry. The transportation and logistics problems are optimized using a static model, but with the increasing of the traffic, and demand and with the flexibility of customers, the growing quantity of computing, and communication, the problem is studied in dynamic conditions.

There exist many models of the supply chain logistics systems among them we consider the dynamic routing models which try to determine the optimal minimum total travel cost. The main characteristics of this type of problem are the uncertainty in the data and it can be due to different sources, and it can have different natures. The automaticity and flexibility are the main features and activities of the dynamic routing. Here we will focus on a dynamic pick-up-and-delivery routing problem, i.e., where goods are collected from various customers and then delivered to a central depot.

In this chapter we focus on the dynamic pick up and delivery vehicle routing problem. Each service request either has a combined pick-up and delivery location or only a single pick-up (or delivery) location. The dynamic vehicle routing with pickup and delivery represents an interesting research issue since it presents some characteristic facial appearance with regards to the static pickup and Delivery.

The dynamic vehicle routing may not be performed proficiently and due to this complexity little report of minimizing costs has given for this variant of problem. Therefore the responsible for administration the transport function must perfectly be educated in

the basic principles of transport cost management; should use the computer aided systems for the planning of the dynamic vehicle routing problem.

The main goal of this study is to develop an Ant Colony System based on 2-Opt local search algorithm to solve the Dynamic Pickup and Delivery VRP. In the ACS meta-heuristic, a set of agents (ants) build solutions to the given problem cooperating through pheromone-update. The success principles of ACS consist in an intelligent exploitation of the problem structure and in an effective interplay between the search space and the solution space elaborating with the local search.

The chapter is structured as follows. In Section 2 we present the main definition and framework of the DPDVRP. Section 3 is devoted to the literature review of the existing papers that deals with the DPDVRP. Section 4 give the framework of the Dynamic Pick-up and delivery vehicle Routing Problem. Section 5 contains the description of our algorithm. In section 6 we include the experiment results and the corresponding analysis to test our proposal algorithm. In section 7 we conclude the chapter.

## 2 Main definition of the Dynamic Vehicle Routing Problem

To give a main definition of the Dynamic Vehicle Routing Problem we look at the work of Psaraftis who was among the very first to consider the dynamic extension of the traditional static VRP. Therefore, Psaraftis (1988, 1995) uses the following classification of the static routing problem;

- *"if the output of a certain formulation is a set of preplanned routes that are not re-optimized and are computed from inputs that do not evolve in real-time".*

While he refers to a problem as being dynamic;

- *"if the output is not a set of routes, but rather a policy that prescribes how the routes should evolve as a function of those inputs that evolve in real-time".*

## 3 Related Works

The role of managing the planning of the vehicle routing problem holds an essential place in distribution management. The academic researchers and private companies were motivated to conduct powerfully the transportation of goods and services.

### 3.1 Pick-up and delivery vehicle Routing Problem

For all the explosive growth in the VRP literature over the past several years, Pick-up and Delivery problems are a class of VRP in which objects or people have to be transported between an origin and a destination. Several routing problems with pick-up and delivery service are reported in the literature. The number of pickup, delivery, and service locations is the same in all transportation demands in the classical models. In the VRP all transportation requests are associated with a single pickup or delivery location. In the PDP all transportation requests concern the pickup of a shipment at its origin and the delivery at its destination.

Little research in the vehicle routing problem considers the multiple pickup, delivery, and/or service locations but we reflect on Savelsbergh and Sol (1995) which represent the General Pickup and Delivery Problem (GPDP) with multiple pickup and delivery locations. In the GPDP a transportation request is composed of several shipments with different pickup and delivery locations. Each pickup location has to be visited before any delivery location. However, the sequence in which the pickup (or delivery) locations must be visited is not specified.

Ai and Kachitvichyanukul (2009) propose a particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery. Among publication that have explicitly addressed a Pick-up and Delivery VRP situation we may mention several papers, such as Psaraftis (1980, 1983), Dumas et al. (1991) proposed a branch-and-price algorithm for the multiple-vehicle Pick-up and Delivery Problem with Time Windows.

Nanry and Barnes (2000) propose a reactive tabu search for the PDPTW in which the search progress is continuously analyzed and search parameters such as the length of the tabu list are adjusted accordingly during search.

Schonberger et al. (2003) propose a Genetic Algorithm for solving the so-called Pickup and Delivery Selection Problem (PDSP) which extends the PDPTW by the decision of acceptance or rejection of transportation requests. Figure 4.1 describe the solution of a static vehicle routing problem

### 3.2 Dynamic Pick-up and delivery vehicle Routing Problem

In figure 4.2 we present a general scheme of a Dynamic VRP. Therefore the figure 4.3 present a solution for the dynamic VRP.

Until now, little research has been focused on the dynamic Pick up and Delivery Vehicle Routing Problem, where problem size and parameters changes after the vehicles

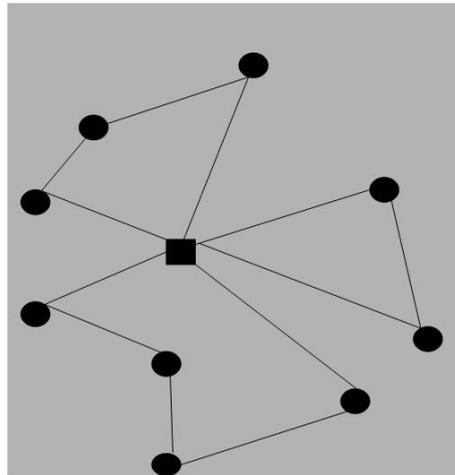


Figure 4.1 – Solution of Static vehicle routing

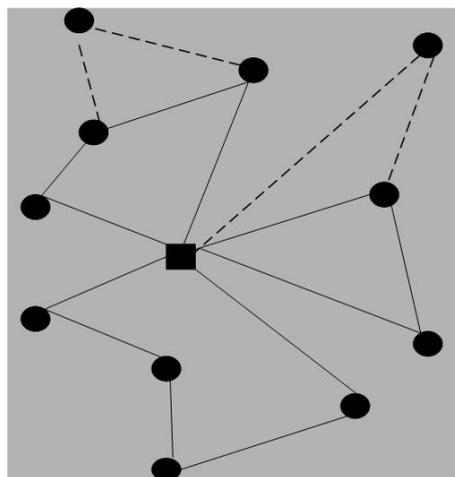
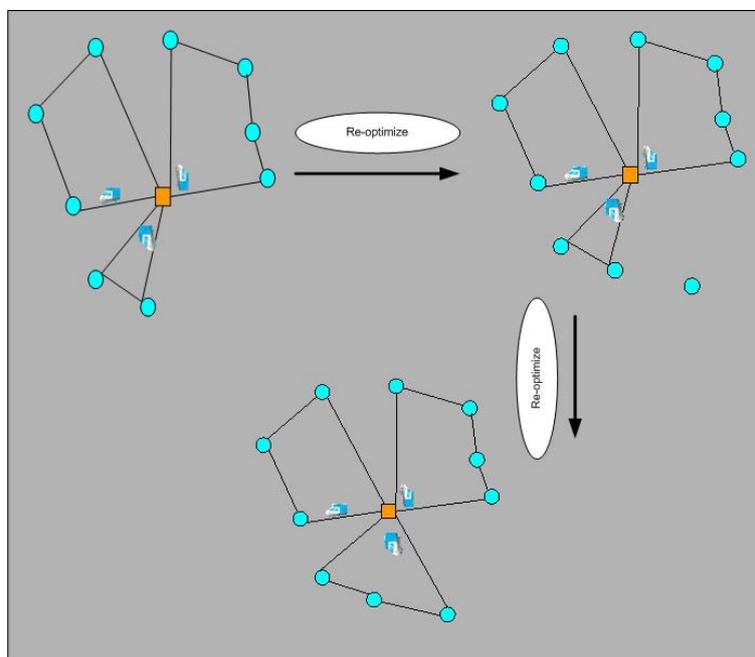


Figure 4.2 – Dynamic vehicle routing



**Figure 4.3** – Solution of a Dynamic vehicle routing

are already commissioned. We consider the paper of Bertsimas and Ryzin (1991) among the earliest work of the most basic dynamic vehicle routing problem which present an objective function with the waiting time and it describe a generic mathematical model. To our knowledge, very few published papers deal with the Dynamic Pick-up and Delivery Routing Problem. Some further work can be found on Powell et al. (1995) and Berbeglia et al. (2009).

We present a general discussion of dynamic network modeling problems that arise in logistics and distribution systems, including a priori optimization and on-line decision policies for stochastic routing problems in Powell et al. (1995). Recently, in Berbeglia et al. (2009) we surveys the subclass of dynamic pickup and delivery problems, where the objects or people have to be collected and delivered in real time. The paper discusses various issues as well as solution strategies.

Since the first formulation of the VRP by Dantzig and Ramser (1959), thousands of algorithms have been proposed for the optimal and approximate solution of the VRP, PDVRP and their variants. The immense popular of these algorithms distress the static vehicle routing problem, i.e. it is unspecified that all data is known and invariant during resolution. Only recently dynamic problems have been increasingly premeditated in the vehicle routing literature. We present the most relevant literature on DPDVRP and in

the solutions techniques devoted to it. A comprehensive discussion of dynamic vehicle routing can be found in Psaraftis (1988) and Psaraftis (1995). In the literature, there are many methods and strategies have been proposed to tackle DVRPs.

Montemanni et al. (2005) considered a DVRP as the extension to the standard VRP by decomposing a DVRP as a sequence of static VRPs and then solving them with artificial ant colony algorithm.

Gendreau et al.(1999) propose a tabu search heuristic for the DPDVRP with Time Windows. Hvattum et al. (2004) presented an approach for problems where statistical information about orders appearance is available. A neighborhood search heuristics for the dynamic dispatching problem with pick-ups and deliveries has been applied in Gendreau et al. (2006).

Savelsbergh and Sol (1998) presented a planning module designed for a transportation company, which embeds a dynamic VRP module. A survey on results achieved on the different types of DVRPs can be found in Gendreau and Potvin (1998).

## 4 A framework of the Dynamic Pick-up and delivery vehicle Routing Problem

Dynamic (or adaptive) routing goes beyond static routing by admitting the possibility of building/changing the vehicles routing solution online according to the current traffic events. It is useful to distinguish between the ability of adapting to the changing traffic conditions and to topological modifications (e.g., link/node failures, link/node addition/removal). This section defines two different types of PDVRP: the static PDVRP and the dynamic PDVRP.

### 4.1 The Static Pick-up and delivery vehicle Routing Problem

The static PDVRP is described in the following way: let  $G = (V, A)$  be a graph where  $V = \{0, 1, \dots, n\}$  is a set of vertices and  $A = \{(i, j) : i, j \in V, i \neq j\}$  is a set of arcs. Vertex 0 is a depot at which is based a fleet of  $m$  vehicles, while the remaining vertices represent customers. Each arc  $(i, j) \in A$  is associated with a non negative demand  $q_i$ . The total demand of a vehicle route may not exceed the vehicle capacity  $Q$ . The objective is to find a vehicle route starting from the depot, finishing all the requests of tasks, and ending at one of the delivery locations. The total distance travelled is minimized.

## 4.2 The Dynamic Pick-up and delivery vehicle Routing Problem

In this section we address a Dynamic DVRP, in which the distribution plan needs to be adjusted in real-time to accommodate changes in uncontrollable parameters of the delivery environment. The problem is a dynamic in a sense that customer requests are handled dynamically, and where the problem is solved in real-time.

Formally, the dynamic VRP is defined over a graph  $G = (V, A)$  as a static PDVRP such as the pick up and delivery vertices of a request cannot be visited by different routes and the pick up vertex naturally precedes the delivery vertex. At each time  $t$ , each vehicle is either serving a vertex, waiting at a vertex, or moving toward a vertex. In either of these cases, the request associated with the vertex is known at time  $t$  (see e.g. Berbeglia, G. et al. (2009)).

In dynamic environment, new orders may appear over the time, that is, the initial problem specification can change after the vehicles have started their tours. Recently Garrido and Riff (2010) they proposed a new model formulation to the dynamic vehicle routing problem (DVRP).

Garrido and Riff (2010) formulate the problem as follows:

- Parameters:

$t$ : elapsed service time, such that  $t \in [0, D]$

$m$ : number of vehicles

$\hat{V}$ : set of non-served customers

$v_s^k$ : last customer served by vehicle  $k$

$S_k^t$ : sequence of customers  $v_0^k, \dots, v_l^k$  served by the  $k$ -th vehicle before  $t$

$c_{il}$ : travel distance from  $v_i$  to  $v_l$

$q_l$ : demands delivered by/to customer  $v_l$

$Q_k^t$ : remaining available capacity of vehicle  $k$  at time  $t$

$D_k^t$ : remaining available time of vehicle  $k$  at time  $t$  before reaching upper bound  $D$

- Variables:

$$x_{lk} = \begin{cases} 1 & \text{if vehicle } k \text{ serves customer } v_l, v_l \in \hat{V} \cup \{v_s^k\} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ilk} = \begin{cases} 1 & \text{if vehicle } k \text{ goes directly from client } v_i \text{ to } v_l \\ 0 & \text{otherwise} \end{cases}$$

## 4.4 A framework of the Dynamic Pick-up and delivery vehicle Routing Problem

---

- Objective function:

$$\text{Minimize} \quad \sum_{ilk, i \neq l} c_{il} * y_{ilk} \quad (4.1)$$

- Constraints:

$$\sum_{k=1}^m x_{lk} = 1, \forall v_l \in \hat{V}, v_l \neq v_0 \quad (4.2)$$

$$\sum_l x_{lk} * q_l \leq Q_k^t, \forall k = 1, \dots, m, \forall v_l \in \hat{V} \cup \{v_s^k\} \quad (4.3)$$

$$\sum_i y_{ilk} + \sum_j y_{ljk} = x_{lk}, \forall k = 1, \dots, m, \forall v_l \in \hat{V} \cup \{v_s^1, \dots, v_s^m\} \quad (4.4)$$

$$\sum_{i,l} c_{il} * y_{ilk} \leq D_k^t, \forall k = 1, \dots, m \quad (4.5)$$

$$\sum_{i,l} y_{ilk} \leq |S|, S \subseteq V' \cup \{v_s^1, \dots, v_s^m\}, 2 \leq |S| \leq n - 1, \forall k = 1, \dots, m \quad (4.6)$$

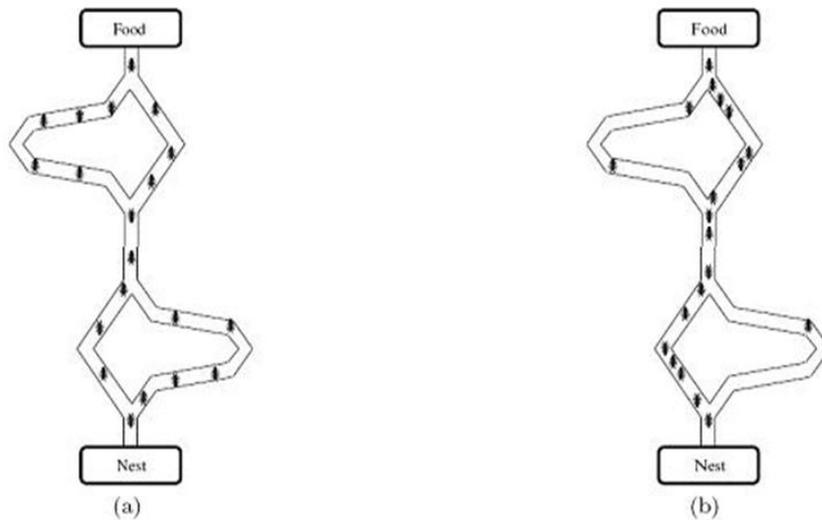
The objective function 4.1 minimize the total travel time. Constraint 4.2 limits the service of each remaining customer  $v_l$  to a single vehicle  $k$  and only once. Constraint 4.3 describe the remaining available vehicle capacities, which are updated over specific time periods. A vehicle  $k$  serves a customer  $v_l$ , if and only if,  $v_l$  belongs to its scheduling, it is handled in constraint 4.4. In constraint 4.5, travel times of each vehicle  $k$  are limited to  $D_k^t$ . Finally, constraint 4.6 establishes the maximum number of clients that can be served by each vehicle  $k$ .

### 4.3 Ant Colony Optimization

Ant Colony Optimization (ACO) is a family of optimization algorithms based on real ants' behaviour in finding a route to food nest. It has been observed in Dorigo and Stutzle (2004) that of available routes, ants find shortest route to food nest. To achieve this, ants communicate through deposition of a chemical substance called pheromone along the route. Shortest path has highest concentration leading to more and more ants using this route. The algorithm is based on an experiment conducted in 1989 by Goss et al. (1989) as shown in figure 4.4.

#### 4.4 A framework of the Dynamic Pick-up and delivery vehicle Routing Problem

---



**Figure 4.4** – Experiment for selection of the shortest branches by a colony of ants: (a) at the beginning of the experiment and (b) at the end of the experiment.

In Mullen et al. (2009) we present a review of ant algorithms. The idea behind ant algorithms is to simulate artificial stigmergy to coordinate societies of artificial ants. Initially, the ants choose one of the available paths randomly because of absence of pheromone trail on any of the paths. However, the ants that choose shorter paths, take less time to traverse it and hence pheromone deposition on the shorter paths occur earlier than the longer ones. Ants which arrive after pheromone deposition on shorter path has occurred and before ants on longer path have completed their journey prefer to choose shorter path because of higher pheromone concentration on it. More number of ants on shorter path further increases the rate of pheromone deposition on that path. Cumulatively, over time this results in highest pheromone concentration on best path and finally all the ants travel through that path only.

ACO (we refer the reader to the papers of Dorigo et al. (1999, 1996)) is a metaheuristic in which a colony of artificial ants cooperates in finding good solutions to discrete optimization problems. Each ant of the colony exploits the problem graph to search for optimal solutions. An 'artificial ant', unlike natural counterparts, has a memory in which it can store information about the path it follows. Every ant has a start state and one or more terminating conditions. The next move is selected by a probabilistic decision rule that is a function of locally available pheromone trails, heuristic values as well as the ant's memory. Ant can update the pheromone trail associated with the link it follows. Once it

has built a solution, it can retrace the same path backward and update the pheromone trails. ACO algorithm is interplay of three procedures as described in Dorigo and Stutzle (2004).

## 5 Application

In this section we have implemented an hybrid AAC-2Opt algorithm for solving the DPDVRP and their performance to give competitive and high quality solutions.

Based on the paper of Montemanni et al. (2005) and on the idea of splitting working into  $n_{ts}$  time slices proposed initially in kilby et al.(1998) and adopted in the ant based metaheuristic by Montemanni et al.(2005). Each time slices with an equal length of time  $\frac{T}{n_{ts}}$  where  $T$  is the length of the working day and we reschedule the arrival of new customers to the end of that time slice. The concept of time slice is to make a partial static VRP. Kilby et al. (1998) introduced the concept of time slice to bound the time committed to each static problem. The objective is to make an order to serve the new demands and the appearance of new customers. Also the concept of cut-off time is considered in our approach with the purpose that the orders up until a certain time are permitted, and all orders thereafter are postponed. We assume that an order has to be committed to a driver at time therefore the advanced commitment time is equal to zero.

### 5.1 Merging of New Event Procedure (MNEP)

Derived from the initialization procedure and based on splitting the working day the MNEP procedure constructs a static problems by means of a  $T_{co}$  and  $T_{ac}$  parameters. The MNEP procedure makes the assignment of the demands to the effective vehicles. It represents an interface between the appearance of new customers and the optimization methods. The first time slice corresponds to the first static problems created from the previous working day. Therefore the cutoff time parameter  $T_{co}$  controls the time dedicated to the arrival of new orders and the guideline of unserved customers.

In our implementation, basis on the time available section we try to divide the customers into  $n_{ts}$  time slices, each time slice represent a static VRP. For each static problem we apply the AAC algorithm to solve the dynamic problem.

All the orders received after the  $T_{co}$  are interpreted as being customers that were not serviced the day before and the re-optimization starts with the rest of non served customers. Each vehicle starts from the location of the last customer committed to it, with

a starting time resulting to the end of the servicing time for this customer with appropriate travel time and capacity changes after it has served all the customers previously committed to it. After each time slice and to the same static problem we keep the best solution selected and orders with a processing time starting within the next  $\frac{T}{n_{ts}} + T_{ac}$  seconds be required to be committed to their respected vehicles, i.e when a vehicle worn all its capacity, it should be return to the depot.

## 5.2 Artificial Ant Colony with 2-opt local search to solve the DPDVRP

This subsection introduces the basic concepts of Artificial Ant Colony hybridized with a 2-Opt local search to solve the DPDVRP. We describe the processing steps of the algorithm including details of the solution representation step, solution construction step, pheromone trail update step and the local search step. Algorithm 6 display a pseudo code of the ACS-2-Opt approach to solve the DPDVRP.

---

**Algorithm 6** Pseudo code of an artificial ant colony based on 2-Opt

---

```

1: Begin
2: A set of ants
3:  $iter \leftarrow 1; BESTSOL \leftarrow \phi;$ 
4: Initialize pheromone values  $\forall (i, j) \in (1, \dots, n)$ 
5: Apply algorithm 7
6: Improve the initialization by applying a Dynamic Construction method
7:  $PQ_{ij} = 0$  (on every customer for each colony)
8:   While( $iter < itermax$ )
9:     for  $j = 1$  to  $n$ 
10:      Apply the Local pheromone update according to the total cost of solution  $S$ 
      (Eq.4.8)
11:    endfor
12:    for  $a = 1$  to  $a_{max}$  do (for each ant)
13:      choose the next customer  $i$  to be served according to the BESTPOS rule (Eq.2)
14:       $BESTSOL \leftarrow BESTSOL \cup \{i\}$ 
15:    endfor
16:    Local search: apply 2-Opt method
17:     $BESTSOL \leftarrow 2 - Opt(BESTSOL)$ 
18:    if the current solution is better than the previous
19:      replace the current solution with the worst
20:    end if
21:    Global pheromone updates (Eq.4.9)
22:  end
23: end

```

---

### 5.2.1 Solutions representation

Solution representation of vehicle routing is one of the key elements for effective implementation of Ant Colony Optimization.

The majority of ACO algorithms for the VRP represent the solution as a sequence of customers to be served, which determine the relative assignment of vehicles. In our algorithm the solution representation is based on the vehicles assignment. The solution is represented as a string of length equal to the number of customers.

In the string, the corresponding value to each position  $i$ , represent the vehicle to which customer  $i$  was assigned.

To illustrate the encoding solution, we propose an example (figure 4.5) treating 10 customers and 3 vehicles. Then the customers and the vehicles allocation are represented as follows:

1	2	3	4	5	6	7	8	9	10
V1	V2	V3	V2	V1	V1	V3	V2	V1	V3

**Figure 4.5** – Solution representation

Vehicle 1: Customers 1, 5, 6, 9

Vehicle 2: Customers 2, 4, 8

Vehicle 3: Customers 3, 7, 10

### 5.2.2 Solution construction

The construction of heuristic function is very important to the search speed and the solution quality. The constructive solution proposed here it characterized by two parts: The first part of the construction is related to the static scheme of the problem, we refer the reader to the paper of Euchi and Chabchoub (2009), and the second part is related to the dynamic scheme of the VRP.

The two parts are ensured in a parallel way, after the execution of the first constructive solution, we try to improve the solution and to optimize the vehicles tours dynamically using the dynamic construction method.

In the first constructive solution, insertion heuristic is developed and used for obtaining an initial solution. AAC-2opt in the initialization start with the insertion heuristic and then apply the second dynamic constructive solution to improve it.

We first compute an order of the customer's basis on their demands; sort all customers in increasing order of demands. Then at each step, the next customer with high demand in the list is selected for insertion in a route.

When inserting a customer  $i$  into a route  $r_k$ , the least cost of insertion,  $C_{ik}$  is computed in the following manner. A customer can be inserted into a non used vehicle, with at least one customer in it. When assigning a customer to a vehicle on the route, the type of demands required by the customer and the vehicle type servicing the customer must be compatible.

In order to obtain an initial feasible solution, algorithm 7 is used:

---

**Algorithm 7** Initialization algorithm

---

```

1: begin
2:   sort all available vehicles in increasing order of capacity
3:   for each available vehicle  $k := 1$  to  $K$  loop
4:     for each customer  $i \in V(i = 1, \dots, n)$  loop
5:       if ( $i \in r_k$  and  $q_i \leq Q_k$ ) then
6:         Insert a customer  $i$  into route  $r_k$  using the insertion method
7:       end if
8:     end for
9:   end for
10: end

```

---

The ACO, such as we know it, is a sequential construction algorithm which never calls into question the decisions taken. A new way of building the solutions is proposed in this section. It is about the concept of Dynamic Construction. For the routes carried out using the first constructive method, we continue the solution construction dynamically.

According to this method, the ants dynamically build a route  $r_k$ , in which they add a new customer  $i$  while inserting it into any place of the already partial constructive route. In this manner, it is possible to add conflicts to the temporary route, then to cancel them by the insertion of other customers later in the solution construction. This way of making takes as a starting point it happens that is impossible to place a customer  $i$  without conflict after a certain time finds an advantageous place among the positions already placed.

The Dynamic construction already refers to a construction where the route  $r_k$  is built using a dynamic chain of customers denoted by (*DCC*) that we can insert into all the positions, by shifting the elements already places while preserving overall constraints of the problem. For the Dynamic Construction, the logic of solution construction is reversed.

With each step, randomly an ant must initially choose a route  $r_k$ . Then, the best position *BESTPOS* into which a customer  $i$  of this route will be inserted is selected according to a modified transition pseudo random rule according to the formula 4.7.

Formally, if during the  $t^{th}$  iteration the  $k^{th}$  ant is located in customer  $i$ , the next customer  $j$  is chosen according to the probability distribution over the set of unvisited customers ( $n - DCC$ ) defined by:

$$P_{ij}^k(t) = \begin{cases} \frac{[PQ_{ij}]^\alpha [VIS_{ij}]^\beta}{\sum_{ant \in DCC} [PQ_{ij}]^\alpha [VIS_{ij}]^\beta} & \text{if } j \in (n - DCC) \\ 0 & \text{otherwise} \end{cases} \quad (4.7)$$

Where (n-DCC) represent the number of all customers minus the set of chosen customer. To conduct the reader it necessary to define some parameters; we denote whether going from customer  $i$  to customer  $j$  led to good solutions in the past by the pheromone quantity  $PQ_{ij}$  associated with the coefficient  $\alpha$  and the visibility  $VIS_{ij}$  in significant the benefit of moving from customer  $i$  to customer  $j$  associated with the coefficient of visibility denoting by  $\beta$ . In this case of the DVRP we set the  $VIS_{ij} = \frac{1}{d_{ij}}$ .

The decision an ant makes is probabilistic in nature and influenced by the pheromone quantity and the visibility factors. We indicate how good it is to choose customer  $i$  after *BESTPOS*. The set of valid choices for an ant is denoted by  $S$ . With probability  $p_0$  where  $0 \leq p_0 < 1$  is a parameter of the algorithm, the ant chose the customer  $i \in S$  which maximizes  $[PQ_{ij}]^\alpha [VIS_{ij}]^\beta$ .

### 5.2.3 Pheromone Trail update

During the phase of solution construction, the ant can in principle also carry out some update of the values of the pheromone array, typically along the path that it is just following. When each ant has constructed a solution, it is time to deposit pheromone.

An alternative pheromone update method is to create a pheromone update thread. Additionally we can build two structures, Local Pheromone Update and Global Pheromone Update to record proper ants' solutions.

### 5.2.4 Local Pheromone Update

The first update is a local pheromone update which occurs after an ant completes its solution. This update evaporates the pheromone values along the customers visited by the ant in order to allow the succeeding ants to explore other customers. Once an arc

$(i, j)$  is chosen by an ant, its level of pheromone will be changed by applying the following local trail updating according to his rule:

$$PQ_{ij} = (1 - \rho)PQ_{ij} + \rho PQ_{ij}(0) \quad (4.8)$$

where  $\rho$  is the evaporation rate (with  $0 \leq \rho \leq 1$ ) and  $PQ_{ij}(0)$  represent the initial value of the trails. The AAC differs mainly by its pheromone function which keeps track of recently visited solution or their attributes. After locally updating the pheromone, each ant undergoes local search.

### 5.2.5 Global Pheromone Update

The second update is the global pheromone update. The pheromone values of an objective are updated by the iteration's best route (after local search) with respect to that objective. The best solution found is used for the global pheromone update. A constructive solution is stored in the pheromone trail matrix to generate a new solution by the next ants. This global pheromone update rule is defined as follows:

$$PQ_{ij} = (1 - \rho)PQ_{ij} + \frac{\rho}{cost(\pi)} \forall (i, j) \in S \quad (4.9)$$

Where  $cost(\pi)$  is the total travelling distance of solution  $S$ .

### 5.2.6 Local Search

Local search has been testified to be of great improvement in ACO metaheuristic Dorigo and Stutzle (2004). Once ants complete their solution construction phase, local search algorithms can be used to refine their solutions before using them for the pheromone update. Various experimental researches have shown that the combination of solution construction by ants and local search procedures is a promising approach. There exist a large number of possible choices when using local search in ACO algorithms. We refer the reader to the papers of Stützle and Hoos (2000) and Dorigo et al. (1996) for a recent review of these techniques.

Our implementation of local search uses the intra-route improvements. For the intra-improvement, we use a 2-opt method for each vehicle route of the DPDVRP.

In this study, to allow the search space of the ACO metaheuristic, the 2-opt local search approach (e.g. Euchi and Chabchoub (2009a)) is incorporated in ACO before the update of a global pheromone of each iteration.

The 2-opt local search Lin (1965) is used in the following way. Let  $S$  be a solution for the DPDVRP. Then its 2-opt neighbourhood  $\mathcal{N}(S)$  is defined as a set of all possible solutions resulting from  $S$  by swapping two distinct customers. This operator involves the substitution of two arcs,  $(i, j)$  and  $(i + 1, j + 1)$  with two others arcs,  $(i, i + 1)$  and  $(j, j + 1)$ , and the reversal of path  $p(i + 1, j)$ . If such a better solution is found, it replaces the current solution and the search continues. When implementing 2-Opt to each vehicle route, the termination condition used the best-accept strategy, i.e. all the neighbors are examined until no improvements can be obtained.

This method links good solutions and the routes that result from the substitution comprise the new solutions. Linking the solutions may be performed either in decision or objective space. The basic processing steps in AAC-2Opt to solve the DDVRP are summarized as follows:

---

**Step 0: Initialization**

Set  $iter = 0$ ;

Set initial value  $PQ_{ij} \forall (i,j) \in A$

Place  $m$  ants at the depot

Set  $\frac{\rho}{cost(\pi)} = 0 \forall (i,j) \in S$

**Step 1: Route construction**

**For**  $i = 0$  to  $n$  **do** ( $n$  is the number of customers)

**For** every  $k = 1$  to  $m$  **do**

**Step1.1** Build solution based on algorithm 7

**Step1.2** Apply the Dynamic Solution Construction

**Step 2: Memorize the BESTSOL found**

**Step 3: Local Pheromone Update**

Pheromone update with the formula in 4.8

**Step 4: Local Search**

Apply 2-Opt method for the improvement of *BESTSOL*

$BESTSOL \leftarrow 2 - Opt(BESTSOL)$

**Step 5: Global Pheromone Update**

$\forall (i,j) \in BESTSOL$  **do**

Pheromone update with the formula in 4.9

**Step 6: Stop criterion**

**If** the stop criterion is satisfied then stop

Otherwise go to Step1

---

### 5.3 Experimental Results and Performance Comparison

We present in this part an experimentation that aims to compare the ACS-2Opt with the proposed algorithms presented in the literature. We have performed a number of experiments and comparisons on several public data sets derived from Montemanni et al. (2005) and Kilby et al. (1998).

#### 5.3.1 Problem instances

Computational experiments are reported into two small groups delivery and pickup instances each using three sets of instances from the literature based on the paper of Kilby et al. (1998) and Montemanni et al. (2005).

The first group (delivery instances) is composed by the first set contains 13 instances with sizes varying from 75-150 customers derived from Taillard (1994). The second set consists of 7 instances with sizes varying from 50-199 customers derived from Christophides and Beasley (1984). The third set consists of 2 instances with sizes varying from 71-134 customers derived from Fisher et al. (1995).

The second group (pickup instances) is composed by the first set contains 14 instances with sizes varying from 75-385 customers derived from Taillard (1994). The second set consists of 7 instances with sizes varying from 50-199 customers derived from Christophides and Beasley (1984). The third set consists of 2 instances with sizes varying from 71-134 customers derived from Fisher et al. (1995).

#### 5.3.2 The experimental environment

We note that we have tested our proposed algorithm with the dynamic delivery routing instances and with the dynamic pickup routing instances which not considered in the other paper tackled the dynamic routing. It is the first paper dealing the instances with negative demands.

In Table 4.1, we give the parameters of the AAC algorithm to solve the DVRP. In our experimental setup and in order to standardize the benchmarks instances, Montemanni et al. (2005) explored the cutoff time  $t_{co}$  and the advanced commitment time  $t_{ac}$  which have been chosen according to the suggestions provided in Kilby et al. (1998):  $t_{co} = 750$  and  $t_{ac} = 0$  and the total length of working day  $T$  equal to 1500. In Montemanni et al. (2005) we have proved that the best number of time slices  $n_{ts}$  is 25. It is the best substitution between the objective value and computational cost.

**Tableau 4.1** – Parameter calibration of our experimental environment

Parameters type	Description	Value
$n_{ts}$	Number of time slices	25.00
$t_{co}$	Cutoff time	750.00
$t_{ac}$	Advanced commitment time	00.00
$T$	Total length of working day	Max time available section
$m$	Number of ants	8.00
$\alpha$	Coefficient of pheromone quantity	0.8
$\beta$	Coefficient of visibility	0.2
$n_{gen}$	Number of generation	10000;15000
$phe0$	Initialization of pheromone	10.00
$r$	evaporation rate	0.625
$nr$	Number of runs	10.00

The values of these parameters are defined as follows: we fixed the number of ants  $m = 8$ ; the coefficient of pheromone quantity  $\alpha = 0,8$  and the coefficient of visibility  $\beta$  equal to 0,2. For the initialization of the pheromone quantity we choose  $phe0 = 10$ ; the evaporation rate  $r$  equal to 0,625; for the preliminary experiments, we choose to run the algorithm into two values of generation  $n_{gen}$  equal to 10000 and 15000. We finish by admitting the solution after 10 runs of algorithm.

### 5.3.3 Results discussion

To evaluate the effectiveness of our proposed approach, the AAC meta-heuristic is compared with two algorithms presented in the paper of Montemanni et al. (2005), the GRASP-DVRP and the ACS-DVRP.

The Table below 4.2 gives the computational results for our proposed approach for the dynamic delivery routing instances. In this table we present the results of AAC algorithm with and without 2-Opt local search within 10000 and 15000 generations. For each instance we provide the best, average and the worst solution. From this table, we conclude that the results obtained from the AAC algorithm with 2-Opt local search are the best compared with results obtained without 2-Opt local search. Also the number of generation considered is an important factor to achieve the best solution.

Tableau 4.2 – Computational results for the AAC for the dynamic delivery instances

Instances	AAC 10000 iterations			AAC 15000 iterations			AAC-2-Opt 10000 iterations			AAC-2-Opt 15000 iterations		
	Best	Average	worst	Best	Average	worst	Best	Average	worst	Best	Average	worst
C100	1147,25	1147,25	1147,25	1014,97	1016,07	1050,29	961,10	1096,10	1106,55	1311,72	1380,25	1417,87
C100b	901,94	911,56	1009,97	977,95	978,72	989,25	900,94	1010,15	1203,62	800,93	841,44	890,34
C120	1398,15	1479,39	1484,79	923,47	1379,97	1404,73	1353,59	1370,11	1525,15	1049,47	1153,29	1390,58
C150	1368,08	1371,50	1396,96	2620,46	1391,73	1394,67	1348,88	1388,69	1475,41	2188,33	2386,93	2391,49
C199	1718,51	1722,56	1968,06	3145,84	1787,88	1748,85	1654,51	1660,41	1798,44	1650,85	1758,51	1774,11
C50	598,38	651,88	755,6	601,34	625,78	645,57	599,58	567,98	607,12	551,95	570,89	593,42
C75	1012,95	1062,95	1275,39	1086,94	1126,41	1138,60	981,57	1011,47	1077,05	1156,83	1213,45	1274,87
f134	20890,95	20171,86	21090,91	16530,43	16675,56	17583,55	15028,81	15133,88	16083,56	13015,56	15528,81	15986,84
f71	369,34	399,22	416,88	396,30	414,46	413,46	301,79	321,99	399,87	301,79	309,94	346,77
tail00a	2477,12	2617,75	2711,60	2273,97	2342,88	2414,47	2282,71	2306,82	2419,13	2194,93	2232,71	2295,61
tail00b	2464,22	2664,32	2689,69	2487,84	2458,49	2609,51	2177,19	2229,77	2367,90	2126,09	2182,61	2215,39
tail00c	1925,87	2065,28	2103,76	1815,37	1820,17	1841,08	1541,28	1582,41	1665,35	1544,50	1562,66	1620,78
tail00d	1909,13	1914,35	2047,94	2017,82	2019,19	2162,08	1839,60	1866,39	1940,52	1909,55	1912,43	1921,33
tail50a	3511,87	3689,32	3759,32	2721,55	3470,80	3640,33	3328,85	3518,31	3840,18	2999,27	3185,727	3501,83
tail50b	3099,71	3144,11	3557,31	2454,28	3266,17	3310,24	2933,40	3019,34	3327,47	2846,28	2880,57	2933,40
tail50c	3105,44	3138,17	3236,83	2644,62	2710,53	2897,23	2912,68	3002,91	2912,68	2718,36	2743,55	2779,67
tail50d	2996,11	3016,62	3657,19	3188,05	3242,06	3362,83	2950,61	3029,50	3113,56	3230,67	3345,16	3456,37
tail75a	1961,84	2005,91	2115,50	1964,36	2010,35	2058,32	1782,91	1868,29	1879,22	1755,33	1782,91	1856,66
tail75b	1619,07	1700,67	1919,41	1597,16	1769,79	1899,46	1464,56	1506,46	1698,04	1306,47	1452,26	1527,77
tail75c	1845,42	1967,47	2053,97	1506,75	1620,70	1672,97	1440,54	1476,83	1513,08	1424,76	1441,91	1528,85
tail75d	1430,66	1560,05	1656,61	1408,56	1639,48	1761,65	1399,83	1411,99	1499,01	1334,67	1422,27	1356,55
Average	2909,185	2874,53	2954,99	2541,81	2560,34	2666,62	2429,26	2395,89	2545,37	2260,42	2442,29	2526,69
Total	52365,33	57490,63	62054,94	53378,03	53767,19	55999,14	48585,35	50313,8	53452,91	40687,59	51288,27	53060,5

The Table 4.3, present the computational results for the AAC for the dynamic pickup routing instances. We notice that this paper represent the first dealing with the dynamic pickup routing instances. This table summarizes the solution quality of the AAC algorithm. We give the results of AAC within 10000 and 15000 generations with and without 2-Opt local search. For each instance we describe the best, average and the worst solution. We demonstrate the role of the 2-Opt to obtain the best results.

Tableau 4.3 – Computational results for the AAC for the dynamic pickup instances

Instances	AAC 10000 iter			AAC 15000 iter			AAC-2Opt 10000 iter			AAC-2Opt 15000 iter		
	Best	Average	worst	Best	Average	worst	Best	Average	worst	Best	Average	worst
C100D	1251,35	1453,18	1456,92	1362,29	1593,12	2048,93	1259,58	1378,13	1593,12	1096,77	1136,99	1231,63
C100bD	861,33	876,14	889,28	663,24	719,26	793,69	487,18	528,81	919,26	483,48	513,03	789,66
C120D	961,62	982,88	982,88	717,94	889,58	956,06	638,35	658,53	717,94	638,87	660,89	861,11
C150D	2751,65	2753,99	2763,45	2535,44	2654,15	2790,99	1654,15	1851,51	1756,11	1796,24	1855,56	1994,45
C199D	2737,08	2889,15	3089,15	2145,57	2280,36	2715,58	2280,36	2283,13	2605,73	2270,91	2527,95	2873,31
C50D	1171,11	1199,69	1209,32	982,06	1038,78	1227,73	687,85	743,25	1038,78	531,28	721,06	931,17
C75D	1818,47	1853,70	1913,83	1417,08	1523,46	1760,74	1141,28	1282,26	1417,08	1131,94	1243,99	1447,86
f134D	18941,94	19143,02	20343,84	15878,94	16346,96	18220,94	12381,92	12691,10	15878,94	12148,24	12312,26	18813,65
f171D	389,37	396,22	396,22	327,62	340,84	432,46	234,18	265,47	340,84	239,50	257,54	388,15
tail00aD	2131,07	2015,62	2275,20	1759,49	1834,59	1840,41	1242,28	1374,71	1454,53	1333,68	1456,35	1759,49
tail00bD	2232,19	2344,94	2286,25	1810,81	2010,23	2020,88	1329,29	1579,07	1657,30	1484,32	1531,93	1810,81
tail00cD	1800,98	1934,20	1669,82	1465,12	1494,96	1587,19	1065,23	1224,12	1262,85	1187,58	1465,12	1645,24
tail00dD	2276,34	2393,39	2257,97	1886,90	1951,25	2088,09	1359,78	1586,66	1537,53	1539,30	1555,23	1886,90
tail150aD	2154,59	2223,97	2411,75	1806,94	2099,85	2113,81	1423,50	1466,27	1802,31	1422,50	1525,62	1806,94
tail150bD	2374,92	2456,36	2607,62	2150,14	2217,52	2318,51	1559,46	1794,60	1685,93	1577,04	1741,03	2217,52
tail150cD	2128,54	2262,02	2343,28	1669,90	1740,31	2126,55	1390,73	1534,45	1371,35	1488,64	1669,90	1840,20
tail150dD	2892,12	2949,79	2974,21	2427,73	2503,48	2717,14	1714,95	1889,51	2244,10	1833,11	2228,81	2503,48
tail75aD	1737,76	1743,85	1829,36	1442,78	1443,14	1737,85	1077,94	1342,23	1476,66	1192,06	1302,17	1333,68
tail75bD	1355,85	1450,53	1495,02	1199,80	1235,88	1241,35	894,86	992,99	996,71	849,79	963,35	1617,21
tail75cD	1746,61	1800,93	1818,44	1416,40	1516,35	1572,91	1109,57	1166,30	1341,06	1131,48	1145,99	1361,29
tail75dD	1191,74	1226,41	1248,39	969,52	1048,04	1175,77	714,01	806,80	823,73	782,72	803,91	973,19
Average	2605,07	2664,28	2755,34	2201,70	2308,67	2547,02	1654,37	1845,83	2066,40	1721,87	1838,98	2385,09
Total	54706,63	55949,98	57862,2	46235,71	48482,11	53487,58	34741,84	38762,5	43394,53	36159,45	38618,68	50086,94

In order to verify the effectiveness and efficiency of the proposed algorithm, we compare the results generated with AAC-2-Opt, with those generated in the literature derived from the paper of Montemanni et al. (2005). The Table 4.4 give the performance comparison of our proposed approach for the dynamic delivery routing instances

We compared the AAC-2-Opt with the GRASP and ACS algorithms. We can see from this table that our algorithm provides the best results compared with the GRASP and ACS algorithms. For more than 71.42% instances, a new best solution was found with solution produced by the results of Montemanni et al. (2005). Over 21 instances 15 new best solution is found by the AAC algorithm.

Table 4.5 shows the comparison processing time results for the dynamic delivery routing instances and for the dynamic pickups routing instances with and without 2-Opt local search and within 10000 and 15000 generations. This table describes the pressure of the number of generations to the processing time results. When we increase the number of generation, the CPU time increased automatically, also the integration of the 2-Opt local search has an effect on the final value of the processing time.

With respect of the fixed parameters, the computational results compared very favorably with existing literature and justify the effectiveness of the AAC algorithm to solve the dynamic routing instances (e.g. see Appendix A)(2).

Tableau 4.4 – Performance comparison of our proposed approach for the dynamic delivery routing instances

Instances	GRASP-DVRP			ACS-DVRP			AAC-DVRP (with 2-Opt)		
	Best	Average	worst	Best	Average	worst	Best	Average	worst
C100	1080.33	1169.67	1119.06	973.26	1100.61	1066.16	1311, 72	1380,25	1417,87
C100b	978.39	1173.01	1022.12	944.23	1123.52	1023.60	<b>800.93</b>	841.44	890.34
C120	1546.50	1754.00	1643.15	1416.45	1622.12	1525.15	<b>1049,47</b>	1153,29	1390.58
C150	1468.36	1541.54	1501.35	1345.73	1522.45	1455.50	2188, 33	2386.93	2391,49
C199	1774.33	1956.76	1898.20	1771.04	1998.87	1844.82	<b>1650,85</b>	1758.51	1774,11
C50	696.92	757.97	719.56	631.30	756.17	681.86	<b>551,95</b>	570.89	593.42
C75	1066.59	1142.32	1079.16	1009.38	1086.65	1042.39	1156,83	1213.45	1274,87
f134	15433.84	17325.73	16458.47	15135.51	17305.69	16083.56	<b>13015,56</b>	15528.81	15986,84
f171	359.16	429.64	376.66	311.18	420.14	348.69	<b>301.79</b>	309.94	346.77
tai100a	2427.07	2583.02	2510.29	2375.92	2575.70	2428.38	<b>2194,93</b>	2232.71	2295.61
tai100b	2302.95	2636.05	2512.27	2283.97	2455.55	2347.90	<b>2126,09</b>	2182,61	2215.39
tai100c	1599.19	1800.85	1704.40	1562.30	1804.20	1655.91	1544,50	1562.66	1620,78
tai100d	1973.03	2165.39	2087.55	2008.13	2141.67	2060.72	1909,55	1912.43	1921,33
tai150a	3787.53	4165.42	3899.16	3644.78	4214.00	3840.18	<b>2999,27</b>	3185,727	3501.83
tai150b	3313.03	3655.63	3485.79	3166.88	3451.69	3327.47	<b>2846,28</b>	2880,57	2933.40
tai150c	3110.10	3635.17	3219.27	2811.48	3226.73	3016.14	<b>2718,36</b>	2743,55	2779,67
tai150d	3159.21	3541.27	3298.76	3058.87	3382.73	3203.75	3230, 67	3345.16	3456,37
tai75a	1911.48	2116.95	2005.44	1843.08	2043.82	1945.20	<b>1755,33</b>	1782.91	1856.66
tai75b	1582.24	1934.35	1758.88	1535.43	1923.64	1704.06	<b>1306,47</b>	1452,26	1527.77
tai75c	1596.17	1859.71	1674.37	1574.98	1842.42	1653.58	<b>1424,76</b>	1441.91	1528,85
tai75d	1545.21	1641.91	1588.73	1472.35	1647.15	1529.00	<b>1334,67</b>	1422.27	1356,55
Average	2510,07	2808,87	2645,84	2422,67	2745,02	2561,14	2260,42	2442,29	2526,69
Total	52731.63	58986.36	55562.64	50876,25	57645,52	53784,02	40687,59	51288,27	53060,5

Tableau 4.5 – Comparison processing time results for the dynamic routing instances

Instances	Dynamic delivery routing instances		Dynamic pickup routing instances	
	AAC-2-opt (10000 iter)	AAC-2-opt (15000 iter)	AAC-2-opt (10000 iter)	AAC-2-opt (15000 iter)
	CPU (seconds)	CPU (seconds)	CPU (seconds)	CPU (seconds)
C100(D)	697,48	1356,50	1226,45	1875,75
C100b(D)	658,16	2086,25	1540,39	1732,44
C120(D)	894,06	2471,25	2530,38	2274,34
C150(D)	1426,29	3861,15	5777,614	3683,12
C199(D)	2447,59	3001,35	4731,35	6335,05
C50(D)	360,98	525,68	264,57	491,66
C75(D)	402,06	1142,41	722,07	1369,69
f134(D)	2040,57	3417,70	1879,07	2816,52
f71(D)	718,53	1143,72	743,28	1084,15
tai100a(D)	1290,18	1917,05	1170,38	1484,74
tai100b(D)	1193,01	1887,13	1170,28	1777,49
tai100c(D)	2451,26	1929,09	1869,98	1885,74
tai100d(D)	1897,80	1991,84	1024,38	1794,60
tai150a(D)	3618,78	3323,27	1867,87	3858,07
tai150b(D)	3935,06	3687,81	2307,96	4310,42
tai150c(D)	3409,67	3867,34	2230,03	3713,04
tai150d(D)	16129,24	3496,64	2230,18	3561,16
tai75a(D)	780,44	942,42	786,75	1087,47
tai75b(D)	749,38	906,60	724,69	1089,87
tai75c(D)	750,82	906,20	705,53	1083,16
tai75d(D)	803,95	1091,89	762,25	1160,77
Total	46655,31	42198,07	31238,65	46984,51

## 5.4 Conclusion

In this chapter, we reconsidered the dynamic vehicle routing problem (DVRP). It contributed to understanding the dynamic delivery and pickup problem. Because their complexity, a few papers deal with the DVRP. Depending on the complexity of the problem, the use of approximate methods is very necessary. To solve the DVRP, we have proposed an Artificial Ant Colony based on 2-Opt local search (AAC-2-Opt). To the best of our knowledge, this is the first application dealing with dynamic pickup routing instances.

The decision taken from the experimental results is the planning of the routes and the reschedule of customers in a sense that customer requests are handled dynamically, and where the problem is solved in real-time with the simulation of new customers. As we seen in the results discussion section, our proposed approach is compared to the methods produced by Montemanni et al. (2005) on each benchmarks instance. The developed meta-heuristic integrates a 2-Opt local search, and an artificial ant colony for obtaining a good results to solve the DVRP.

The computational experiments provided prove the utility and the importance of the mixing of the 2-Opt local search with the best usage of AAC algorithm. The solution quality from the algorithm specify that the proposed approach is an essential tool, either in the case of dynamic delivery routing instances also in the case of dynamic pickup routing instances. Therefore the results show that the AAC-2-Opt algorithm produced a best results compared with the GRASP-DVRP and the ACS-DVRP of Montemanni et al. (2005). The tentative analysis reveals high-quality results compared to the best-known results published in literature.

---

---

# Chapter 5

---

## Conclusion

In this chapter we summarize the major developments represented in this thesis. We elaborate the main conclusion reported and future work to be realized. In Section 1 we give a summary of the research work developed which we discuss our contributions in this thesis where we suggest the possible future work directions. Section 2 outlines the resulting papers that have been published for journals and conference proceedings.

### 1 Research work summary

According to the Council of Supply Chain Management Professionals (CSCMP), logistics management can be defined as, "that part of supply chain management that plans, implements, and controls the efficient, effective forward and reverse flow and storage of goods, services and related information between the point of origin and the point of consumption in order to meet customers' requirements."

Transportation involves the physical movement or flow of goods. The transportation system is the physical link that connects customers, raw material suppliers, plants, warehouses and channel members. These are the fixed points in a logistics supply chain. This problem is described in literature as Vehicle Routing Problems.

In this thesis, an attempt was made to deal comprehensively with the complicated phenomenon of vehicle routing and the satisfaction of customers and optimization of tours. This work deals with meta-heuristics approaches to solve some variants of the heterogeneous fixed fleet vehicle routing problems.

In particular, we aim to provide many algorithms to solve different types of problems. Our work focuses on many subjects within the heterogeneous vehicle routing problems with limited fleet. In the last decades the ground of meta-heuristics has developed significantly. To solve hard combinatorial problems of practical sizes, several researchers have

unproven the capability of these methods within rational processing time.

In this collection we highlight the recent developments made in the area of:

- Tabu Search
- Variable Neighborhood Search
- Evolutionary-inspired algorithms like Genetic Algorithms
- Iterated Density Estimation
- Ant Colony Optimization.

To improve the solutions quality of these variants of the heterogeneous vehicle routing problems with limited fleet, various combinations of heuristics and meta-heuristics have been suggested which allows a better planning of tours.

In chapter 1 (**General Introduction**) we have introduced the methodology and the need of the new methods called meta-heuristics to improve the solution quality of the heterogeneous vehicle routing problems with limited fleet.

A comprehensive survey was described in the Chapter 2 (**Literature review**). This chapter begins with a summary on literature of vehicle routing problems in general and a heterogeneous fixed fleet vehicle routing problem with common carrier in particular. Then, a description of the considered problems with mathematical formulations will be presented. After there, a detailed of different variants of heterogeneous fixed fleet vehicle routing problems are discussed.

Chapter 3 (**M-Heterogeneous Vehicle Routing Problem**) treated the cases of M-Vehicle Routing Problems. We studied two variants taking into account the heterogeneity and the limited fleet. The two considered variants of problems deal with the following scenarios: the vehicle routing with heterogeneous limited fleet and the vehicle routing with heterogeneous limited fleet and common carrier.

The vehicle routing problem with heterogeneous limited fleet is a variant of vehicle routing that involves a limited number of vehicles which can be heterogeneous. In this problem, the aim is to provide service to the customer group with minimum cost. The question that arises is: How to exploit the limited and heterogeneous vehicles for better planning of tours while optimizing the transportation costs?

To solve this problem we developed a new hybridization considering the implementation of tabu search algorithm embedded in adaptive memory procedure. This hybridization is specific and appropriate to our situation. The metaheuristic is developed along

four stages including the development of several constructive heuristics in the purpose to improve the solution.

To evaluate the effectiveness of our approach, we provided experimentation over the benchmark test problem of Golden et al. (1998) and Li et al. (2007) for the same problems. The results produced by our algorithms have been compared with those produced by the algorithms of Taillard (1999), Tarantilis et al. (2004) and Li et al. (2007).

The comparative study with the methods of Taillard (1999), Tarantilis et al. (2004) and Li et al. (2007), observe that in seven out of eight test problems, our approach finds a better solution. It is interesting to observe that over the five large instances of Li et al. (2007), four new best solutions were produced with our algorithm. In the large test problems, the TSAM yields consistently better results than the HRTR metaheuristic of Li et al. (2007).

In second scenario we tackle the vehicle routing problem with private fleet and common carrier, the problem consist of the choice of made of transport. There has been growing interest in truck service selection and the deregulation of truck common carrier. This has had particular impact on organization that use privately owned vehicles, since they are responsible for the utilization of the fleet and must make choices that determine the balance between common carrier and private carrier usage. The problematic asked is: How to supply customers to minimize routing costs using the private fleet (fixed and variable costs) or an external carrier (visited cost)?

To address this problem, two approaches have been proposed. The first application is the adaptation of the tabu search algorithm with ejection chains neighbourhood (e.g. Euchi and Chabchoub (2009, 2010)). The second method is the application of Iterated Density Estimation Evolutionary Algorithm with 2-Opt local Search (e.g. Euchi and Chabchoub (2010,2009)).

To highlight the relevance of the results provided by our methods many experiments results were conducted. We have advanced a comparative study with the methods provided in literature and presented in the papers of Bolduc et al. (2007, 2008). Our approach, through the two proposed methods, improves the results of the literature for all considered instances of Christofides and Eilon (1969), Golden et al. (1998). Our metaheuristics have been improving all results provided by the 34 heterogeneous instances proving their effectiveness.

The algorithm requires minimal computation time and it is very performing according to similar experiment presented in literature. The results of this research show that the proposed metaheuristic is a very effective tool for finding good solutions for the VRPPC.

In Chapter 4 (**The Dynamic Vehicle Routing Problem**) we address the dynamic vehicle routing problem. Dynamic routing goes beyond static routing by admitting the possibility of building/changing the vehicles routing solution online according to the current traffic events. In dynamic optimization data such as customer demands are unknown where new customers are appeared after the execution of the tour.

The Integrating of the dynamic concepts on the combinatorial problems requires the development of effective tools for their resolution. Our approach is based on artificial ant colony with a 2-Opt local search (AAC) to solve this variant. The use of an initial algorithm to construct the initial solution has an impact in the improvement of the final solution. Also the involvement of the local search after the two structures of pheromones update has generated a good improvement in the compute of the vehicle routing costs.

To evaluate the effectiveness of our proposed approach, the AAC meta-heuristic is compared with two algorithms presented in the paper of Montemanni et al. (2005), the GRASP-DVRP and the ACS-DVRP. Computational experiments are reported into two small group's delivery and pickup instances each using three sets of instances from the literature based on the paper of Kilby et al. (1998) and Montemanni et al. (2005). The first set contains 13 instances derived from Taillard (1994). The second set contains 7 instances derived from Christophides and Beasley (1984). The third set consists of 2 instances derived from Fisher (1995). Over 21 instances a 15 new best solution are found by the AAC algorithm. Our algorithm provides the best results compared with the GRASP and ACS algorithms. For more than 71.42% instances, a new best solution was found with solution produced by the results of Montemanni et al. (2005). Our approach has solved the dynamic pick up vehicle routing problem that has not been studied in literature. To the best of our knowledge, this is the first application dealing with dynamic pickup routing instances.

We conclude that the attempts and approaches developed for different types of vehicle routing problems with heterogeneous limited fleet and the results are promising with respect to different results in literature. The vehicle routing problem with heterogeneous limited fleet is a very important area of research in combinatorial optimization. Due to its complexity, it was generally resolved by metaheuristics. In this thesis, the proposed and developed methodology proves to be effective in solving some variants of heterogeneous limited fleet vehicle routing problem. Our future research will focus on the study of all problems mentioned in literature review chapter. We try to add dynamic constraints to various problems.

Another perspective is attractive to us is the important choice of the mode of transport

and the use of the external carrier. However, in our future work we mainly concentrated in the vehicle routing problem with dynamic external carrier taking into account the limited size of the fleet.

## 2 Resulting papers

1. J. Euch, H. Chabchoub. *Hybrid metaheuristic for the profitable arc tour problem*. Journal of the Operational Research Society, doi:10.1057/jors.2010.179, 2011.
2. J. Euch, H. Chabchoub. *A Hybrid Tabu Search to Solve the Heterogeneous Fixed Fleet Vehicle Routing Problem*. Logistics Research, vol. 2 (1), pp. 3-11, 2010.
3. J. Euch, H. Chabchoub, A. Yassine. *New evolutionary algorithm to solve the vehicle routing problem with private fleet and common carrier*. International Journal of Applied Metaheuristic Computing, vol. 2 (1), pp. 58-82, 2011.
4. J. Euch, H. Chabchoub, A. Yassine. *Ant Colony System Based on 2-opt Local Search to Solve the Dynamic Vehicle Routing Problem*. The 9th international conference on Multiple Objective Programming and Goal Programming, MOPGP 2010, Abstract, pp. 60. Sousse-Tunisia, May 24-26, 2010.
5. J. Euch J, A. Yassine, H. Chabchoub. *Solving the dynamic vehicle routing by means of Artificial Ant Colony*. The 3rd International Conference on Metaheuristics and Nature Inspired Computing, META10, Djerba-Tunisia, October 27-31, 2010.
6. J. Euch J, A. Yassine, H. Chabchoub. *On the performance of artificial ant colony to solve the dynamic vehicle routing problem*. Soumis à Journal of Applied Mathematics and Computation (Elsevier, Décembre 2010).
7. J. Euch, H. Chabchoub. *Heuristic Search Techniques to Solve the Vehicle Routing with Private Fleet and Common carrier*. International Journal of Universal Computer Sciences, vol.1 (1), pp. 10-18, 2010.
8. J. Euch, H. Chabchoub. *Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier*. IEEE proceedings of 39ème International Conference on Computers and Industrial Engineering, ISBN 978-1-4244-4136-5, pp.1058-1063, Troyes-France, 6-9 July 2009.

9. J. Euchi, H. Chabchoub. *Hybrid Tabu Search heuristic with ejection chains algorithms for the Vehicle Routing Problem with Private fleet and common Carrier*. The VIII Metaheuristics International Conference, MIC 2009, Extended Abstract, pp. 151-156, Hamburg-Germany, July 2009.
10. J. Euchi, H. Chabchoub. *Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier*. Proceedings of the European Chapter on Metaheuristics, Portugal, pp. 133-140, 2009.
11. J. Euchi, H. Chabchoub. *A hybrid genetic algorithm for the vehicle routing problem with private fleet and common carriers*. 10ème Journées Nationales de l'association française de Recherche Opérationnelle et d'Aide à la Décision, ROADEF'2009, Résumé pp. 243-245. Nancy - France, février 2009.

---

## Conclusion générale

Dans cette partie du rapport nous résumons nos contributions apportées pour la résolution des problèmes étudiés en présentant d'abord l'ensemble des résultats, théoriques et numériques, auxquels nous avons abouti, et en proposant ensuite des objectifs pour la poursuite du travail entamé, et ce, sous forme d'orientations et de perspectives de recherche. Cette conclusion se structure comme suit :

D'abord nous résumons l'ensemble de nos travaux, puis nous mettons l'accent sur les résultats obtenus, ensuite nous avançons quelques idées aptes à constituer une extrapolation de celles développées et enfin une liste de nos publications et communications clôture ce rapport.

Selon la vision classique la logistique représentait, l'activité allant de la mise à disposition des produits ou le négociant jusqu'à la livraison au client. Elle s'est désormais fondue dans le concept de " supply chain management " dont l'objectif est d'optimiser la gestion des flux physiques et des flux d'information le long de la chaîne logistique depuis le fournisseur du fournisseur jusqu'au client du client.

Notre idée clé consistait en la proposition des méthodes métaheuristiques efficaces pour la résolution des différents problèmes de la Logistique et du Transport. Pour cela, nous avons développé des algorithmes qui résolvent différents types de problèmes de tournées de véhicules tenant compte de la limitation de la flotte et de son hétérogénéité. Si nous avons choisi de faire appel à des métaheuristiques c'est parce qu'ils ont montré leurs efficacité face à des problèmes de taille modérée. Nous mettons l'accent en premier lieu sur les développements récents réalisés concernant :

- 
- la recherche tabou
  - la recherche à voisinage variable
  - les algorithmes évolutionnaires (comme les algorithmes génétiques)
  - les algorithmes à estimation de distribution
  - l'optimisation par colonies de fourmis

Afin d'améliorer la qualité des solutions de ces variantes du problème de tournées de véhicules, diverses combinaisons des heuristiques et méta-heuristiques ont été avancées ce qui permet une meilleure planification des tournées.

Dans le chapitre 1 (**Introduction**) nous avons introduit la méthodologie permettant l'étude des différentes variantes du problème de tournées de véhicules.

Dans le chapitre 2 (**Literature review**) nous avons fait un état de l'art de l'essentiel des travaux réalisés en rapport avec notre problématique d'étude. Ce chapitre débute par une synthèse de revue de littérature consacrée aux problèmes de tournées de véhicules en général et les problèmes de tournées de véhicules avec flotte limitée hétérogène et transporteur externe en particulier. Par la suite, nous enchainons avec la description et la formulation mathématique du problème de tournées de véhicules. Différentes variantes de ce problème sont abordées.

Dans le chapitre 3 (**M-Heterogeneous Vehicle Routing Problems**) nous traitons le cas des m-problèmes de tournées de véhicules. Nous avons étudié deux variantes en tenant compte de l'hétérogénéité et de la limitation de la flotte. Les deux variantes en question traitent les cas de figure suivants : la tournée de véhicules avec flotte limitée hétérogène et la tournée de véhicules avec flotte limitée hétérogène et transporteur externe.

Le problème de tournées avec flotte limitée hétérogène est une variante du problème de tournées de véhicules (PTV) qui vise à fournir des services à un ensemble de clients avec un coût minimum en utilisant un nombre limité de véhicules. La question qui se pose est : Comment exploiter l'ensemble des véhicules limités et hétérogènes pour avoir une meilleure planification des tournées tout en optimisant les coûts de transport ?

Pour résoudre ce type de problème nous avons développé une nouvelle hybridation de l'algorithme tabou avec une mémoire adaptative. Cette hybridation est spécifique et adéquate à notre cas de figure. La métaheuristique est développée le long de quatre étapes dont des heuristiques constructives ont été développées pour améliorer la solution.

---

Pour évaluer l'efficacité de notre approche, recherche tabou avec mémoire adaptative, nous avons pris les instances fournies par Golden et al. (1998) et Li et al. (2007) pour le problème de tournées de véhicules avec flotte limitée hétérogène. L'étude comparative avec les méthodes de Taillard (1999), Tarantilis et al. (2004) et Li et al. (2007) a donné que parmi 8 instances, notre algorithme a fourni 7 meilleures solutions. Il est intéressant d'observer que pour les cinq grandes instances de Li et al. (2007), quatre nouvelles meilleures solutions ont été produites avec notre algorithme.

Pour le problème de tournées de véhicules avec flotte limitée hétérogène et transporteur externe, le problème consiste en un réseau contenant : un dépôt, plusieurs clients, une flotte privée limitée hétérogènes et un transporteur externe. La question que nous nous sommes posée est : Comment desservir les clients afin de minimiser les coûts de transport en utilisant la flotte privée (coûts fixes et variables) ou un transporteur externe (coût de visite)?

Pour répondre à cette problématique, deux approches ont été proposées. La première consiste en une technique de recherche heuristique basée sur des voisinages de types chaînes d'éjection. La seconde est une méthode évolutionnaire avec une recherche locale 2-opt.

Pour mettre en évidence la pertinence des résultats fournis par nos méthodes, nous avons avancé une étude comparative avec les méthodes fournies par la littérature et qui sont présentées dans les papiers de Bolduc et al. (2007, 2008). Notre approche, par le biais des deux méthodes proposées, améliore les résultats de la littérature pour toutes les instances considérées de Christofides and Eilon (1969), Golden et al. (1998). Nos métaheuristiques ont pu améliorer toutes les résultats fournis par les 34 instances de types hétérogène prouvant ainsi son efficacité.

Dans le chapitre 4 (**The Dynamic Vehicle Routing Problem**) nous traitons le problème de tournées de véhicules dynamique. Dans l'optimisation dynamique, certaines données, comme les demandes des clients sont inconnues. Un parmi la spécificité du problème de tournées de véhicules dynamique est l'apparition de nouveaux clients après le commencement des tournées. Les tournées dynamiques se diffèrent par rapport aux tournées statiques c'est qu'on admet la possibilité de la construction et de changement de la solution de tournées de véhicules après que les véhicules débutent leurs tournées.

L'intégration des concepts aléatoires ou dynamiques aux problèmes combinatoires nécessite le développement de méthodes efficaces pour leur résolution. Notre approche est basée sur une méthode de colonie de fourmis avec une recherche locale de type 2-opt. Au niveau de l'application de notre méthode, l'utilisation d'un algorithme initial

---

pour la construction de la solution a un impact au niveau de l'amélioration de la solution finale. L'implication de la recherche locale après les deux structures de mise à jour de la phéromone a engendré une bonne amélioration au niveau de traitement du calcul des coûts de tournées de véhicules.

Pour mettre en évidence notre méthode et plus spécifiquement avec la problématique abordée nous avons utilisées les instances de Taillard (1994), Christophides and Beasley (1984) et Fisher (1995). Notre méthode a été comparé avec les méthodes de résolution de Montemanni et al. (2005) sur la même base de benchmarks. Suite a cette étude comparative la méthode de colonie de fourmis a donné de bons résultats. Parmi 21 instances, 15 nouvelles meilleures solutions sont obtenus par l'algorithme de colonie de fourmis. Bien que les instances de livraison utilisées, notre approche a pu résoudre le problème de tournées de véhicules avec collecte dynamique (Dynamic pick up) qui n'a pas été étudié par la littérature. Il nous semble que notre essai pour la résolution du problème tenant compte de la collecte dynamique est la première.

Les expérimentations menées au niveau de cette variante de tournées de véhicules se révèlent l'utilité et l'importance de l'hybridation de la recherche locale 2-opt avec l'algorithme de fourmis.

On conclut que les tentatives et les approches développées aux différents types de problèmes de tournées de véhicules avec flotte limitée hétérogène ainsi que les résultats sont prometteurs par rapport aux différents résultats fournis dans la littérature.

Le problème de tournées de véhicules avec flotte limitée hétérogènes correspond à un domaine de recherche très important dans l'optimisation combinatoire. Vu sa complexité, il a été résolu généralement par des métaheuristiques.

Dans cette thèse, notre méthodologie proposées et développées prouve son efficacité pour résoudre certaines variantes de problèmes de tournées de véhicules avec flotte limitée hétérogène. Nos recherches futures se concentrent sur l'étude de tous les problèmes mentionnés dans le chapitre revue de littérature. Nous essayons d'ajouter des contraintes dynamiques aux différents problèmes.

Une autre perspective nous semble intéressante c'est le choix du mode de transport et de l'utilisation du transporteur externe. Cependant, dans nos futurs travaux nous avons essentiellement concentré dans le problème de tournées de véhicules dynamique avec transporteur externe avec la prise en compte de la taille limitée de la flotte.

---

## Liste des publications et communications

1. J. Euchi, H. Chabchoub. *Hybrid metaheuristics for the profitable arc tour problem*. Journal of the Operational Research Society, doi:10.1057/jors.2010.179, 2011.
2. J. Euchi, H. Chabchoub. *A Hybrid Tabu Search to Solve the Heterogeneous Fixed Fleet Vehicle Routing Problem*. Logistics Research, vol. 2 (1), pp. 3-11, 2010.
3. J. Euchi, H. Chabchoub, A. Yassine. *New evolutionary algorithm to solve the vehicle routing problem with private fleet and common carrier*. International Journal of Applied Metaheuristic Computing, vol. 2 (1), pp. 58-82, 2011.
4. J. Euchi, H. Chabchoub, A. Yassine. *Ant Colony System Based on 2-opt Local Search to Solve the Dynamic Vehicle Routing Problem*. The 9th international conference on Multiple Objective Programming and Goal Programming, MOPGP 2010, Abstract, pp. 60. Sousse-Tunisia, May 24-26, 2010.
5. J. Euchi J, A. Yassine, H. Chabchoub. *Solving the dynamic vehicle routing by means of Artificial Ant Colony*. The 3rd International Conference on Metaheuristics and Nature Inspired Computing, META10, Djerba-Tunisia, October 27-31, 2010.
6. J. Euchi J, A. Yassine, H. Chabchoub. *On the performance of artificial ant colony to solve the dynamic vehicle routing problem*. Soumis à Journal of Applied Mathematics and Computation (Elsevier, Décembre 2010).
7. J. Euchi, H. Chabchoub. *Heuristic Search Techniques to Solve the Vehicle Routing with Private Fleet and Common carrier*. International Journal of Universal Computer Sciences, vol.1 (1), pp. 10-18, 2010.
8. J. Euchi, H. Chabchoub. *Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier*. IEEE proceedings of 39ème International Conference on Computers and Industrial Engineering, ISBN 978-1-4244-4136-5, pp.1058-1063, Troyes-France, 6-9 July 2009.
9. J. Euchi, H. Chabchoub. *Hybrid Tabu Search heuristic with ejection chains algorithms for the Vehicle Routing Problem with Private fleet and common Carrier*. The VIII Metaheuristics International Conference, MIC 2009, Extended Abstract, pp. 151-156, Hamburg-Germany, July 2009.

- 
10. J. Euchi, H. Chabchoub. *Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier*. Proceedings of the European Chapter on Metaheuristics, Portugal, pp. 133-140, 2009.
  11. J. Euchi, H. Chabchoub. *A hybrid genetic algorithm for the vehicle routing problem with private fleet and common carriers*. 10ème Journées Nationales de l'association française de Recherche Opérationnelle et d'Aide à la Décision, ROADEF'2009, Résumé pp. 243-245. Nancy - France, février 2009.

---

# Annexe 1

## 1 Appendix A. Best solutions found

Here in the appendix we present the following routes constructed with the available vehicles to achieve the new best results realized with the AAC with 2-Opt local search. The format of these results are described in the following details in appendix A 2 :  $\langle instance \rangle : \langle bestcost \rangle : \langle routenumber \rangle : \langle depot \rangle : \langle circuit \rangle : \langle depot \rangle$ .

### 1.1 The best routes constructed with available vehicles

---

***Benchmark C100: best cost =1311, 72***

Route 1: 0-1-9-3-4-2-6-5-8-7-11-10-12-13-15-0

Route2 : 0-16-14-19-18-17-20-21-22-24-23-27-28-26-25-0

Route3 : 0-31-30-29-35-34-33-32-38-37-40-39-36-41-42-43-0

Route4 : 0-48-47-46-45-44-49-50-52-51-54-55-56-57-0

Route5 : 0-53-59-58-61-60-62-63-64-65-67-68-0

Route6 : 0-66-75-74-72-73-69-70-71-76-77-78-79-82-83-84-0

Route7 : 0-80-81-86-85-87-89-88-90-91-93-92-96-0

Route8 : 0-94-95-97-99-98-100-0

---

***Benchmark C100b: best cost =800.9371***

Route 1: 0-3-2-1-4-6-7-5-8-9-12-14-13-0

Route 2: 0-10-11-15-16-19-18-17-22-23-26-0

Route 3: 0-21-20-24-25-27-28-30-29-31-35-33-0

## 1 Appendix A. Best solutions found

---

Route 4: 0-32-34-36-39-38-37-40-41-42-44-45-0

Route 5: 0-43-46-48-50-51-52-49-47-54-53-56-0

Route 6: 0-58-59-57-55-60-61-62-63-65-0

Route 7: 0-66-64-67-69-68-70-71-73-72-74-75-77-0

Route 8: 0-79-78-76-81-80-82-83-84-85-88-0

Route 9: 0-87-86-89-91-92-93-94-95-96-97-0

Route 10: 0-98-99-100-0.

---

### ***Benchmark C120: best cost =1049, 47***

Route 1: 0-2-4-3-5-1-10-7-6-9-8-12-13-14-11-16-0

Route 2: 0-15-18-17-19-20-21-23-22-24-25-27-28-26-29-32-31-30-33-34-36-0

Route 3: 0-37-35-43-38-39-42-41-40-46-44-47-45-48-51-50-49-0

Route 4: 0-52-54-53-55-58-56-57-59-61-62-64-63-60-66-65-0

Route 5: 0-67-69-70-71-72-68-81-73-76-77-78-74-75-80-79-0

Route 6: 0-87-86-85-84-83-82-88-89-90-91-94-93-95-92-98-100-97-0

Route 7: 0-96-99-101-102-105-106-107-104-103-110-109-108-111-112-113-114-115-116-118-117-0

Route 8: 0-119-120-0

---

### ***Benchmark C150: best cost =2188, 33***

Route1:0-5-2-4-3-6-8-7-9-11-10-12-13-15-14-0

Route2:0-18-17-16-19-20-26-21-22-23-25-24-29-28-27-0

Route3:0-30-34-33-31-32-35-39-37-38-36-41-40-0

Route4:0-42-43-44-45-46-47-48-49-52-53-50-51-56-55-0

Route5:0-58-57-59-60-54-61-62-63-64-65-66-70-69-0

Route6:0-68-67-71-73-72-74-75-81-78-79-0

Route7:0-76-77-80-82-87-85-91-86-84-83-89-88-90-0

Route8:0-94-95-92-93-102-101-96-99-98-100-97-103-0

Route9:0-105-104-106-107-108-110-109-111-112-116-115-117-0

Route10:0-118-114-113-119-121-120-122-123-124-125-0

Route11:0-132-127-126-131-128-129-130-133-136-135-134-138-137-139-0

Route12:0-144-142-141-140-143-145-149-150-146-147-148-0

***Benchmark C199: best cost =1650, 85***

Route1 : 0-4-3-9-10-11-7-8-5-6-19-18-17-16-14-15-0  
Route2 : 0-13-12-20-24-21-22-23-27-26-25-29-34-35-33-0  
Route3 : 0-28-31-30-32-36-37-39-41-40-42-43-44-38-0  
Route4 : 0-45-46-49-47-48-52-50-51-62-60-59-0  
Route5 : 0-53-58-57-56-55-54-61-64-63-66-65-68-0  
Route6 :0-67-75-74-72-73-69-70-71-76-77-78-79-81-80-0  
Route7 : 0-83-84-82-86-85-87-89-88-90-96-0  
Route8 :0-94-95-97-92-98-93-91-102-101-99-100-110-109-105-0  
Route9 : 0-104-106-107-108-103-114-113-117-112-111-116-0  
Route10 : 0-115-121-120-122-126-123-124-125-118-119-137-133-0  
Route11 : 0-132-127-131-128-135-136-129-134-130-143-148-146-147-0  
Route12 : 0-152-149-138-150-139-145-144-142-151-141-140-153-0  
Route13 : 0-156-154-158-157-162-159-160-161-0  
Route14 : 0-155-168-166-167-169-164-163-165-170-171-0  
Route15 : 0-176-177-179-178-172-173-174-175-186-187-180-0  
Route16 : 0-183-182-189-181-188-185-184-195-196-0  
Route17 : 0-190-194-193-191-192-197-198-199-0

---

***Benchmark C50: best cost =551, 95***

Route1 : 0-3-2-1-5-4-7-6-8-10-9-0  
Route2 : 0-11-13-12-14-17-15-16-19-0  
Route3 : 0-18-20-21-22-23-25-24-26-0  
Route4 : 0-27-30-29-28-31-33-34-32-36-35-37-0  
Route5 : 0-38-43-41-40-42-44-39-46-47-45-48-0  
Route6: 0-49-50-0

---

***Benchmark C75: best cost =1156, 83***

Route 1: 0-3-1-2-4-5-8-7-0  
Route 2: 0-6-9-10-11-13-0  
Route 3: 0-12-14-15-17-16-19-18-0  
Route 4: 0-21-22-23-24-20-25-28-0  
Route 5: 0-26-27-29-31-32-33-0

## 1 Appendix A. Best solutions found

---

Route 6: 0-30-36-34-35-37-38-39-41-0

Route 7: 0-40-42-45-44-43-47-48-0

Route 8: 0-46-49-50-51-52-54-53-55-0

Route 9: 0-58-59-57-56-60-62-61-0

Route 10: 0-63-64-66-65-67-70-71-69-0

Route 11: 0-68-73-72-74-75-0

---

### ***Benchmark f134: best cost =13015, 56***

Route 1: 0-1-3-2-4-5-6-7-13-12-11-10-9-8-16-15-14-19-18-17-22-21-20-23-24-25-26-27-28-30-31-29-0

Route 2: 0-32-34-35-36-37-33-42-41-40-39-38-46-47-48-45-43-44-49-50-51-52-54-53-58-57-56-55-59-60-61-62-65-0

Route 3: 0-66-71-64-63-67-68-69-70-78-77-76-74-73-72-75-79-83-0

Route 4: 0-82-84-80-81-85-86-87-88-89-95-96-97-93-94-92-90-91-99-98-100-0

Route 5: 0-103-104-101-102-105-106-107-108-109-114-113-0

Route 6: 0-112-111-110-115-116-117-119-118-120-121-122-123-130-0

Route 7: 0-133-132-125-126-124-127-128-129-131-134-0

---

### ***Benchmark f71: best cost =301, 79***

Route 1: 0-4-3-5-6-1-2-8-7-9-10-0

Route 2: 0-11-13-12-16-17-15-14-18-20-19-0

Route 3: 0-22-21-24-23-25-26-27-28-30-29-31-35-36-34-32-0

Route 4: 0-33-37-38-39-41-42-40-44-43-46-45-48-47-50-49-51-52-54-56-55-53-0

Route 5: 0-57-58-59-61-60-62-63-65-64-66-67-68-69-70-71-0

---

### ***Benchmark tai100a: best cost =2194, 93***

Route 1: 0-9-8-5-4-6-3-2-7-1-11-10-15-0

Route 2: 0-14-16-12-13-17-18-0

Route 3: 0-19-20-23-21-22-24-25-26-0

Route 4: 0-28-27-29-30-31-33-32-0

Route 5: 0-34-36-37-35-38-43-41-39-44-40-0

Route 6: 0-42-47-46-48-45-49-0

## 1 Appendix A. Best solutions found

---

Route 7: 0-51-50-53-55-54-52-59-58-57-56-0  
Route 8: 0-60-61-62-64-63-65-67-0  
Route 9: 0-66-69-68-70-0  
Route 10: 0-73-74-72-71-75-77-78-79-76-82-80-81-0  
Route 11: 0-83-84-85-90-86-88-91-89-87-0  
Route 12: 0-94-95-96-92-93-97-0  
Route 13: 0-98-99-100-0

---

### ***Benchmark tai100b: best cost =2126, 09***

Route 1: 0-3-1-2-4-5-8-7-9-10-6-11-0  
Route 2: 0-13-17-16-12-14-15-18-0  
Route 3: 0-22-20-21-19-25-23-24-27-28-0  
Route 4: 0-26-29-31-33-30-34-32-0  
Route 5: 0-37-35-38-36-43-0  
Route 6: 0-41-40-42-39-49-48-47-44-46-50-45-51-0  
Route 7: 0-52-53-56-54-55-57-60-61-59-58-63-64-0  
Route 8: 0-62-65-66-67-68-69-73-72-0  
Route 9: 0-70-71-75-76-74-0  
Route 10: 0-77-80-79-78-81-82-83-85-87-86-84-88-90-89-0  
Route 11: 0-91-93-92-94-0  
Route 12: 0-96-95-99-97-0  
Route 13: 0-100-98-0

---

### ***Benchmark tai100c: best cost =1544, 50***

Route 1: 0-3-4-2-5-0  
Route 2: 0-1-9-8-7-6-10-14-13-11-12-17-16-0  
Route 3: 0-15-18-20-21-22-0  
Route 4: 0-19-25-23-24-26-29-27-0  
Route 5: 0-28-30-33-32-31-34-35-38-37-36-42-39-40-41-43-0  
Route 6: 0-46-45-47-49-44-48-56-55-53-0  
Route 7: 0-52-51-50-54-58-57-61-62-60-0  
Route 8: 0-64-59-63-66-67-65-70-68-0  
Route 9: 0-69-72-71-73-75-76-74-79-78-77-84-0  
Route 10: 0-85-82-83-80-81-88-86-0

## 1 Appendix A. Best solutions found

---

Route 11: 0-89-87-97-90-94-93-92-95-0

Route 12: 0-91-96-99-100-98-0

---

### ***Benchmark tai100d: best cost =1909, 55***

Route 1: 0-3-1-2-4-5-9-8-6-7-10-13-14-12-11-15-0

Route 2: 0-18-17-16-22-20-21-19-23-25-0

Route 3: 0-24-26-28-0

Route 4: 0-27-29-30-0

Route 5: 0-31-32-33-37-35-34-38-36-41-39-42-40-0

Route 6: 0-47-45-44-46-49-43-48-0

Route 7: 0-51-55-54-53-56-52-50-57-58-62-64-61-63-60-0

Route 8: 0-59-65-70-69-66-67-68-71-0

Route 9: 0-72-74-73-76-75-77-78-0

Route 10: 0-79-0

Route 11: 0-81-80-83-82-84-85-88-87-86-90-92-89-93-91-0

Route 12: 0-97-94-0

Route 13: 0-95-96-98-99-100-0

---

### ***Benchmark tai150a: best cost =2999, 27***

Route 1: 0-4-6-3-1-2-7-5-9-12-11-10-8-16-13-14-0

Route 2: 0-18-17-15-19-22-23-20-21-24-25-35-0

Route3 : 0-31-33-32-27-29-34-28-0

Route4 : 0-26-30-38-44-45-39-41-43-42-40-0

Route5 : 0-36-37-48-49-51-46-47-50-53-57-0

Route6 : 0-56-55-54-58-52-59-60-0

Route7 : 0-62-61-63-64-65-66-0

Route8 : 0-67-70-68-71-69-72-75-73-74-77-76-83-82-79-78-0

Route9 : 0-85-81-80-84-86-88-91-89-87-90-93-92-0

Route10 : 0-95-94-96-97-103-0

Route11 : 0-98-101-99-100-102-105-104-106-107-108-0

Route12 : 0-109-112-111-110-117-118-115-114-116-0

Route13 : 0-113-119-127-123-122-126-121-124-125-120-0

Route14 : 0-129-128-131-130-137-133-136-0

Route15: 0-132-134-135-138-140-139-141-145-143-148-149-142-144-0

Route16: 0-147-146-150-0

---

***Benchmark tai150b: best cost =2846, 28***

Route 1: 0-2-6-1-7-4-5-3-11-9-10-8-0

Route 2: 0-13-12-18-19-15-16-17-14-22-20-23-0

Route 3: 0-26-25-21-24-27-28-0

Route 4: 0-29-31-33-30-34-32-37-36-38-35-43-0

Route 5: 0-42-40-41-39-44-45-47-46-49-48-50-54-0

Route 6: 0-53-52-51-56-55-60-57-59-58-63-64-68-61-67-0

Route 7: 0-65-66-62-69-70-72-73-71-74-76-0

Route 8: 0-75-81-77-80-0

Route 9: 0-78-79-86-84-82-83-85-88-87-90-91-89-95-0

Route 10: 0-96-94-92-97-0

Route 11: 0-93-101-0

Route 12: 0-102-98-100-0

Route 13: 0-103-104-99-107-108-105-106-0

Route 14: 0-111-109-113-110-112-115-119-117-114-116-118-120-  
126-125-121-122-124-123-0

Route 15: 0-127-131-128-129-130-134-132-138-135-133-137-136-  
139-142-140-141-146-143-0

Route 16: 0-147-148-145-144-149-150-0

---

***Benchmark tai150c: best cost =2718, 36***

Route 1: 0-6-7-1-0

Route 2: 0-4-2-5-3-11-10-8-9-15-12-14-13-0

Route3: 0-18-16-19-17-0

Route4: 0-20-28-24-26-23-27-22-21-25-0

Route5: 0-29-30-32-31-35-34-33-38-39-36-37-43-42-0

Route6 : 0-40-41-46-51-48-50-0

Route7 : 0-49-44-47-45-52-56-57-55-53-54-61-58-60-62-0

Route8 : 0-59-63-64-65-66-69-68-0

Route9 : 0-70-67-75-72-71-74-73-0

Route10 : 0-76-81-79-80-83-82-0

Route11 : 0-78-77-91-87-89-86-90-85-88-84-0

## 1 Appendix A. Best solutions found

---

Route12 : 0-92-93-94-96-95-97-98-0

Route13 : 0-100-101-99-102-105-103-0

Route14 : 0-104-106-109-107-110-108-111-115-113-112-114-116-119-121-0

Route15 : 0-117-118-120-123-124-125-0

Route16: 0-127-126-122-135-129-133-128-134-132-130-131-136-139-140-138-137-143-142-141-144-145-0

Route17: 0-147-148-146-149-150-0

---

### ***Benchmark tai150d: best cost =3230, 67***

Route 1: 0-5-1-2-6-4-3-13-12-9-8-10-7-11-17-14-15-16-19-21-23-25-20-18-22-0

Route 2: 0-24-0

Route 3: 0-27-28-26-29-30-0

Route 4: 0-34-33-32-31-36-38-41-39-35-37-40-42-43-46-44-0

Route 5: 0-45-47-50-48-49-52-58-53-54-55-51-57-59-0

Route 6: 0-60-56-67-66-62-64-65-61-63-71-69-70-68-73-72-74-0

Route 7: 0-78-76-75-77-79-81-80-83-85-84-82-86-0

Route 8: 0-94-91-93-89-87-92-90-88-97-95-96-100-101-102-99-0

Route 9: 0-98-104-105-106-103-107-109-110-108-0

Route 10: 0-118-113-117-112-115-116-111-0

Route 11: 0-114-121-120-119-123-125-124-0

Route 12: 0-122-126-0

Route 13: 0-128-127-134-131-129-130-133-132-0

Route 14: 0-139-138-135-137-0

Route 15: 0-136-140-141-142-0

Route 16: 0-146-145-143-147-144-0

Route 17: 0-149-150-148-0

---

### ***Benchmark tai75a: best cost =1755, 33***

Route 1: 0-1-3-2-6-4-5-7-10-0

Route 2: 0-8-9-11-12-13-14-15-0

Route 3: 0-16-17-18-19-21-20-22-23-0

Route 4: 0-24-27-25-0

Route 5: 0-26-28-29-31-33-0

Route 6: 0-35-36-30-34-32-37-0

## 1 Appendix A. Best solutions found

---

Route 7: 0-39-38-40-44-0

Route 8: 0-45-43-42-41-47-46-50-49-48-51-52-53-54-56-55-57-60-59-58-63-0

Route 9: 0-61-62-64-65-66-0

Route 10: 0-67-69-68-73-71-72-0

Route 11: 0-70-74-75-0

---

### ***Benchmark tai75b: best cost =1306, 47***

Route 1: 0-1-6-7-0

Route 2: 0-2-4-0

Route 3: 0-3-5-10-8-11-9-12-14-13-18-17-16-0

Route 4: 0-15-19-20-25-21-0

Route 5: 0-23-24-22-27-29-28-26-33-30-32-35-31-34-36-37-0

Route 6: 0-38-39-40-41-42-43-0

Route 7: 0-44-46-45-47-49-50-0

Route 8: 0-48-51-52-58-57-56-55-53-54-59-60-62-0

Route 9: 0-61-63-64-67-65-0

Route 10: 0-68-66-70-69-73-74-71-72-0

Route 11: 0-75-0

---

### ***Benchmark tai75c: best cost =1424, 76***

Route 1: 0-3-1-2-4-7-5-6-9-8-10-11-12-14-0

Route 2: 0-13-17-16-15-18-20-19-22-23-21-26-25-0

Route 3: 0-24-27-0-28-32-31-0

Route 4: 0-29-0

Route 5: 0-30-0

Route 6: 0-36-33-37-35-34-40-39-38-42-41-43-0

Route 7: 0-44-46-45-49-47-52-51-0

Route 8: 0-48-50-53-55-54-58-56-0

Route 9: 0-57-59-60-62-61-63-66-64-65-67-71-69-70-0

Route 10: 0-68-74-73-72-75-0

---

### ***Benchmark tai75d: best cost =1334, 67***

Route 1: 0-2-3-0

## 1 Appendix A. Best solutions found

---

Route 2: 0-1-4-5-6-7-11-10-9-8-0

Route 3: 0-13-14-12-17-15-16-20-19-18-0

Route 4: 0-21-23-22-24-26-0

Route 5: 0-25-27-32-0

Route 6: 0-30-28-31-29-33-36-35-34-37-38-39-40-0

Route 7: 0-42-41-43-44-46-45-49-0

Route 8: 0-50-51-48-47-52-53-55-56-0

Route 9: 0-54-57-58-59-60-64-0

Route 10: 0-62-63-61-66-65-67-68-69-71-70-72-73-74-0

Route 11: 0-75-0

---

---

## Bibliography

- [1] Ai T.J., Kachitvichyanukul V. *A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery*. Computers & Operations Research, 36: 1693 - 1702, 2009.
- [2] Appelegate D., Cook W.J and Rohe A. *Chained Lin-Kernighan for large travelling salesman problems*. INFORMS Journal on Computing, 15(1): 82-92, 2003.
- [3] Archetti C., Savelsbergh M.W.P, Speranza M.G. *To split or not to split: That is the question*. Transportation Research Part E, 44: 114-123, 2008. [31](#)
- [4] Arntzen H., Hvattum L.M, L0kketangen A. *Adaptive memory search for multi demand multidimensional knapsack problems*. Computers & Operations Research, 33: 2508-2525, 2006.
- [5] Arunapuram Sundararajan, Mathur Kamlesh, Solow Daniel. *Vehicle Routing and scheduling with full truckloads*. Transportation Science, 37 (2): 170-182, 2003. [31](#)
- [6] Balas E. (1989). *The prize collecting travelling salesman problem*. Networks, 19: 621-636, 1989. [31](#)
- [7] Ball M.O., Golden A., Assad A., Bodin L.D. *Planning for truck fleet size in the presence of a common-carrier option*. Decision Sciences, 14; 103-120, 1983. [31](#)
- [8] Beltrami E, Bodin L. *Networks and vehicle routing for municipal waste collection*. Networks, vol.4, pages 568-581, 1974. [24](#), [31](#)
- [9] Berbeglia G., Cordeau J.-F., Laporte G. *Dynamic pickup and delivery problems*. European Journal of Operational Research, Volume 202, Issue 1, Pages 8-15, 2010.

- 
- [10] Bertsimas, D.J., van Ryzin, G.J. *A stochastic and dynamic vehicle routing problem in the Euclidean plane*. Operations Research, 39: 601-615, 1991.
- [11] Bèrubé J.F, Gendreau M., Potvin J.Y. *An exact E-constraint method for bi-objective combinatorial optimization problems: Application to the Travelling Salesman Problem with Profits*. European Journal of Operational Research, 194: 39-50, 2009. [31](#)
- [12] Bodin L.D, Golden B.L, Assad A.A, Ball M.O. *Routing and scheduling of Vehicles and crews*. The state of the Art. Computers and Operations Research,10: 69-211, 1983. [31](#)
- [13] Bolduc M.C., Renaud J. and Boctor F.F. *A heuristic for the routing and carrier selection problem*. Short communication. European Journal of Operational Research, 183, 926-932, 2007. [31](#)
- [14] Bolduc M.C., Renaud J. , Boctor F.F. and Laporte G. *A perturbation metaheuristic for the vehicle routing problem with private fleet and common carriers*. Journal of the Operational Research Society, 59, 776-787, 2008. [31](#)
- [15] Bosman P.A.N. and Thierens D. *Multi-objective optimization with diversity preserving mixture-based iterated density estimation evolutionary algorithms*. International Journal of Approximate Reasoning, 31, 259-289, 2002.
- [16] Brandão J. *A deterministic tabu search algorithm for the fleet size and mix vehicle routing problem*. European Journal of Operational Research, 195: 716-728, 2009.
- [17] Brandão J.C.S and Mercer A. *Tabu search algorithm for the multi-trip vehicle routing and scheduling problem*. European Journal of Operational Research, 100, 181-191, 1997. [31](#)
- [18] Brown G., Ellis C., Graves G.W. and Ronen D. *Real-time wide area dispatching of Mobil tank trucks*. Interfaces, Vol 17 No. 1, pp. 107-20, 1987. [31](#)
- [19] Butt S.E, Ryan D.M. *An optimal solution procedure for the multiple tour maximum collection problem using column generation*. Computers & Operations Research, 26 (4): 427-441, 1999.
- [20] Butt S.E, Cavalier T.M. *A heuristic for the multiple tour maximum collection problem*. Computers and Operations Research, 21: 101-111, 1994. [31](#)
- [21] Chao I.M, Golden B., Wasil E.A. *A fast and effective heuristic for the orienteering problem*. European Journal of Operational Research, 88: 475-489, 1996.
- [22] Chao I.M, Golden B., Wasil E.A. *The team orienteering problem*. European Journal of Operational Research, 88: 464-474, 1996. [31](#)

- 
- [23] Choi E. and Tcha D.W. *A column generation approach to the heterogeneous fleet vehicle routing problem*. Computers and operations research, 34, 2080-2095, 2007. [31](#)
- [24] Christofides N. and Eilon S. *An algorithm for the vehicle dispatching problem*. Operations Research, 20, 309-318, 1969. [31](#)
- [25] Christofides N, Beasley J. *The period routing problem*. Networks, 14:237-256, 1984.
- [26] Chu C.W. *A heuristic algorithm for the truckload and less-than-truckload problem*. European Journal of Operational Research, 165, 657-667, 2005. [31](#)
- [27] Clarke G., Wright J.W. *Scheduling of vehicles from a central depot to a number of delivery points*. Operations Research, 12, 568-581, 1964.
- [28] Cooper, J, Browne, M and Peters M. *European Logistics: Markets, management and strategy*. Blackwell, London, 1991.
- [29] Corberán A., Mota E., Sanchis J.M. *A comparison of two different formulations for arc routing problems on mixed graphs*. Computers & Operations Research, 33: 3384-3402, 2006. [31](#)
- [30] Cordeau J.F, Laporte G. *A tabu search heuristic for the static multi-vehicle dial-a-ride problem*. Transportation Research Part B: Methodological, 37 (6): 579-594, 2003. [31](#)
- [31] Dantzig G.B, Ramser J.H. *The truck dispatching problem*. Management science, 6(1), 80-91, 1959.
- [32] Dantzig G.B, Fulkerson R., Johnson S. *Solution of a Large-Scale Traveling-Salesman Problem*. Journal of the Operations Research Society of America, Vol. 2, No. 4 pp. 393-410, 1954.
- [33] Desrochers M. and Verhoog T.W. *A new heuristic for the fleet size and mix vehicle routing problem*. Computers & Operations Research, 18(3), 263-274, 1991. [31](#)
- [34] Diaby M. and Ramesh R. *The Distribution Problem with Carrier Service: A Dual Based Penalty Approach*. ORSA Journal on Computing, 7, 24-35, 1995.
- [35] Dorigo M., Stutzle T. *Ant Colony Optimization*. MIT Press, Cambridge, MA, 2004.
- [36] Dorigo M., V. Maniezzo & A. Coloni. *The Ant System: Optimization by a colony of cooperating agents*. IEEE Transactions on Systems, Man, and Cybernetics-Part B, 26(1), 29-41, 1996.
- [37] Dorigo M., Di Caro G., and Gambardella L.M . *Ant algorithms for discrete optimization*. Artificial Life, 5:137-172, 1999.

- 
- [38] Dumas Y., Desrosiers J., Soumis F. *The pickup and delivery problem with time windows*. European Journal of Operational Research, 54: 7-22, 1991.
- [39] Euchi. J, Chabchoub H. *A Hybrid Tabu Search to Solve the Heterogeneous Fixed Fleet Vehicle Routing Problem*. Logistics Research, vol 2 (1) pp. 3-11, 2010. [31](#)
- [40] Euchi. J, Chabchoub H. *Heuristic Search Techniques to Solve the Vehicle Routing with Private Fleet and Common carrier*. International Journal of Universal Computer Sciences, 1 (1), pp. 10-18, 2010.
- [41] Euchi. J, Chabchoub H. *Hybrid Metaheuristics for the profitable arc tour problem*. Journal of Operational Research Society, doi:10.1057/jors.2010.179, 2011.
- [42] Euchi J., Chabchoub H., Yassine A. *New evolutionary algorithm to solve the vehicle routing problem with private fleet and common carrier*. A paraître dans International Journal of Applied Metaheuristic Computing, vol. 2 (1), 2011.
- [43] Euchi. J, Chabchoub H. *Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier*. IEEE proceedings of 39ème International Conference on Computers and Industrial Engineering, ISBN 978-1-4244-4136-5 pp: 1058-1063. Troyes - France, 6-9 July 2009.
- [44] Euchi. J, Chabchoub H. *Iterated Density Estimation Evolutionary Algorithm with 2-opt local search for the vehicle routing problem with private fleet and common carrier*. Proceedings of the European Chapter on Metaheuristics, Portugal pages 133-140, 2009.
- [45] Euchi. J, Chabchoub H. *Tabu search metaheuristic embedded in adaptative memory procedure for the Profitable Arc Tour Problem*. IEEE proceedings of Nature and Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on , vol., no., pp.204-209, 9-11 Dec. 2009. A pdf copy of the paper is accessible at <http://ieeexplorepreview.ieee.org/stamp/stamp.jsp?tp=&arnumber=5393689> [31](#)
- [46] Euchi J, Chabchoub H. *Hybrid Tabu Search heuristic with ejection chains algorithms for the Vehicle Routing Problem with Private fleet and common Carrier*. The VIII Metaheuristics International Conference, MIC 2009, Extended Abstract, p. 151-156. Hamburg, Germany, July 2009.
- [47] Euchi J, Chabchoub H. *A hybrid genetic algorithm for the vehicle routing problem with private fleet and common carriers*. 10ème Journées Nationales de l'association française de Recherche Opérationnelle et d'Aide à la Décision, ROADEF2009, Resume pp. 243-245. Nancy - France, février 2009.
- [48] Euchi J, Chabchoub H, Yassine A. *Ant Colony System Based on 2-opt Local Search to Solve the Dynamic Vehicle Routing Problem*. The 9th international conference on

- Multiple Objective Programming and Goal Programming, MOPGP 2010, Abstract, p 60. Sousse, Tunisia, May 24-26, 2010.
- [49] Euchi J, Yassine A, Chabchoub H. *Solving the dynamic vehicle routing by means of Artificial Ant Colony*. The 3rd International Conference on Metaheuristics and Nature Inspired Computing, META10, Djerba, Tunisia, October 27-31, 2010.
- [50] Feillet D., Dejax P., Gendreau M. *The Profitable Arc Tour Problem: Solution with a Branch-and-Price Algorithm*. Transportation Science, 39(4): 539-552, 2005. 31
- [51] Feillet D, Dejax P, Gendreau M. *Traveling Salesman problems with Profits*. Transportation Science, 39(2): 188-205, 2005. 31
- [52] Fisher M.L, Jaikumar R. *A generalized assignment heuristic for vehicle routing problem*. Network, 11, 109-124, 1981.
- [53] Fisher, M.L, Greenfield, A.J, Jaikumar, R. and Lester, J.T. *A computerized vehicle routing applications*. Interfaces, vol 12, N 4, pp 42-52, 1982.
- [54] Fisher ML. *Vehicle routing*. Handbooks Operation Research Management Science, 8, 1995.
- [55] Frederickson G. N., Hecht M. S. , and Kim C. E. *Approximation algorithms for some routing problems*. SIAM Journal on Computing, 7(2):178-193, 1978.
- [56] Garrido P., Riff, M.C. *DVRP: a hard dynamic combinatorial optimization problem tackled by an evolutionary hyper-heuristic*. Journal of Heuristics, 16: 795-834, 2010.
- [57] Gendreau M., Guertin F., Potvin J., and Seguin R. *Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries*. Transportation Research Part C, 14:157-174, 2006.
- [58] Gendreau M., Potvin J.-Y. *Dynamic vehicle routing and dispatching*. In Fleet management and logistic. Kluwer Academic Publishers, T.G. Crainic, G. Laporte eds., pages 115-226, 1998.
- [59] Gendreau M., Laporte G., Semet F. *A tabu search heuristic for the undirected selective travelling salesman problem*. European Journal of Operational Research, 106 (2-3): 539-545, 1998.
- [60] Gendreau M., Laporte G., Musaraganyi C. and Taillard E.D. *A tabu search heuristic for the heterogeneous fleet vehicle routing problem*. Computers & Operations Research, 26, 1153-1173, 1999. 31
- [61] Gendreau M, Guertin F., Potvin J.-Y., and Taillard É.D. *Parallel tabu search for real-time vehicle routing and dispatching*. Transportation Science, 33:381-390, 1999.

- 
- [62] Goss S., Aron, Doneubourg J., and Pasteels J.M. *Self-organized shortcuts in the argentine ants*. *Naturwissenschaften*, 76:579-581, 1989.
- [63] Glover F. *Tabu Search - Part I*. *ORSA Journal on Computing*, 1 (3), 190-206, 1989.
- [64] Glover F. *Ejection chains, references structures and alternating path methods for the travelling salesman problem*. *Discrete Applied Mathematics*, 65, 223-253, 1992.
- [65] Glover F. *Heuristics for Integer Programming Using Surrogate Constraints*. *Decision Sciences*, 8( 1): 156-166, 1977.
- [66] Golden B.L, Laporte G., Taillard E.D. *An adaptive memory heuristic for a class of vehicle routing problems with minmax objective*. *Computers & Operations Research*, 24(5): 445-452, 1997.
- [67] Golden B. L. and Assad A.A. *Vehicle Routings: Methods and Studies*. North-Holland, Amsterdam, 1988. 31
- [68] Golden B.L., Assad A.A., Levy L. and Gheysens F. *The fleet size and mix vehicle routing problem*. *Computers & Operations Research*, 11: 49-66, 1984. 31
- [69] Golden B.L., Wasil E.A., Kelly J.P. and Chao I.M. *The impact of metaheuristics on solving the vehicle routing problem: Algorithms, problem sets, and computational results*. In: Crainic TG and Laporte G (eds). *Fleet Management and Logistics*. Kluwer: Boston, 33-56, 1998.
- [70] Gronalt, M., Hartl, R.F. and Reimann, M. *New Savings Based Algorithms for Time Constrained Pickup and Delivery of Full Truckloads*. *European Journal of Operational Research*, 151(3), 520-535, 2003. 31
- [71] Hall R.W., Racer M. *Transportation with common carrier and private fleets: system assignment and shipment frequency optimization*. *IIE Transactions*, 27, 217-225, 1995. 31
- [72] Hansen P., Mladenovic N. *Variable neighbourhood search: Principles and applications*. *European Journal of Operational Research*, 130: 449-467, 2001.
- [73] Harary F. *Graph Theory*. Reading, MA: Addison-Wesley, 1969.
- [74] Hvattum L.M., Løkketangen A., Laporte G. *Solving a dynamic and stochastic vehicle routing problem with a sample scenario hedging heuristic*. *Transportation Science*, 40(4): 421-438, 2006.
- [75] Jozefowicz N., Semet F., Talbi E.G. *An evolutionary algorithm for the vehicle routing problem with route balancing*. *European Journal of Operational Research*, 195: 761-769, 2009.

- 
- [76] Johnson D. S. and McGeoch L. A. *The Traveling Salesman Problem: A Case Study in Local Optimization. Local Search in Combinatorial Optimization*. E. H. L. Aarts and J. K. Lenstra (editors). John Wiley and Sons, Ltd., pp. 215-310, 1997.
- [77] Jünger M., Reinelt G. and Rinaldi G. *The Traveling Salesman Problem*. In Ball, Magnanti, Monma and Nemhauser (eds.). Handbook on Operations Research and the Management Sciences vol. 7. Elsevier, Amsterdam, 1995.
- [78] Keller, C. P., Goodchild M. *The multiobjective vending problem: A generalization of the traveling salesman problem*. Environ Planning B: Planning Design, 15 447-460, 1988.
- [79] Keller, C. P. *Multiobjective routing through space and time: The MVP and TDVP problems*. Unpublished doctoral dissertation, Department of Geography, The University of Western Ontario, London, Ontario, Canada, 1985. [31](#)
- [80] Kilby P, Prosser P, Shaw P. *Dynamic VRPs: a study of scenarios*. Technical Report APES-0-1998, University of Strathclyde, 1998.
- [81] Klincewicz J.G., Luss H. and Pilcher M.G. *Fleet size planning when outside carrier services are available*. Transportation Science, 24, 169-182, 1990. [31](#)
- [82] Laporte G. *The Vehicle Routing Problem: An overview of exact and approximate algorithms*. European Journal of Operational Research, 59(3): 345-358, 1992. [31](#)
- [83] Laporte G. *The traveling salesman problem: An overview of exact and approximate algorithms*. European Journal of Operational Research, 59(2): 291-247, 1992.
- [84] Laporte G. *Computer aided routing* CWI tract.75:M.W.P. Savelsbergh Mathematisch Centrum, Amsterdam, 1992, 134 pages, DFL.40.00, ISBN 90 6196 412 1. European Journal of Operational Research, 71(1), 143, 1993.
- [85] Laporte G., Osman I.H. *Routing Problems: A Bibliography*. Annals of Operations Research, 61, 227-262, 1995.
- [86] Larrañaga P. *A review on estimation of distribution algorithms*. In P. Larrañaga and J. A. Lozano, editors, Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation. Kluwer Academic Publishers, pp. 80-90, 2002.
- [87] Lawler E. L. *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart and Winston. Network, 1976.
- [88] Li F., Golden B.L. and Wasil E.A. *A record-to-record travel algorithm for solving the heterogeneous fleet vehicle routing problem*. Computers & Operations Research, 34: 2734-2742, 2007. [31](#)

- 
- [89] Libertad Tansini, Maria Urquhart, & Omar Viera. *Comparing assignment algorithms for the Multi-Depot VRP*. Tech. rept. University of Montevideo, TR0108.pdf, 2001.
- [90] Lin S. *Computer solutions of the traveling salesman problem*. The Bell System Technical Journal, 2245-2269, 1965.
- [91] Liu F.H. and Shen S.Y. *The fleet size and mix vehicle routing problem with time windows*. Journal of Operational Research Society, 50:721-732, 1999. [31](#)
- [92] Lozano J.A., Larránaga P., Inza I., and Bengoetxea E. *Towards a New Evolutionary Computation*. Springer, Berlin Heidelberg, New York, 2006.
- [93] Michalewicz Z., and Fogel D.B. *How to solve it: modern heuristics*. Springer-Verlag New York, Inc. 2000, NY, USA, 2000.
- [94] Montemanni, R., Gambardella, L. M., Rizzoli, A. E, and Donati, A. V. *Ant colony system for a dynamic vehicle routing problem*. Journal of Combinatorial Optimization, 10: 327-343, 2005. [31](#)
- [95] Mühlenbein H. and Mahnig T. *FDA - a scalable evolutionary algorithm for the optimization of additively decomposed functions*. Evolutionary Computation, 7(4):353-376, 1999.
- [96] Mullen R.J, Monekosso D., Barman S., Remagino P. *A review of ant algorithms*. Expert Systems with Applications, 36: 9608-9617, 2009.
- [97] Nanry, W.P., Barnes, J.W. *Solving the pickup and delivery problem with time windows using reactive tabu search*. Transportation Research B, 34: 107-121, 2000.
- [98] Osman I.H., Salhi S. *Local search strategies for the VFMP*. In: Rayward-Smith VJ, Osman IH, Reeves CR, Smith GD (Eds.), Modern Heuristic Search Methods. Wiley, New York, pp. 131-153, 1996. [31](#)
- [99] Osman I.H. and Wassan N. *Reactive tabu search meta-heuristic for the vehicle routing problem with backhauls*. Journal of Scheduling, Vol. 5, 263-285, 2002.
- [100] Peter Francis, Karen Smilowitz & Michal Tzur. *The period vehicle routing problem with service choice*. working paper WP 04 005.pdf, 2004. [31](#)
- [101] Potvin J.Y. and Rousseau J.M. *A parallel route building algorithm for the vehicle routing and scheduling problem with time windows*. European Journal of Operational Research, 66: 331-340, 1993. [31](#)
- [102] Powell, W.B., Jaillet, P., Odoni, A. *Stochastic and dynamic networks and routing*. In: Ball, M.O., Magnanti, T.L., Monma, C.L., Nemhauser, G.L. (Eds.), Network routing, Handbooks in Operations Research and Management Science, 8. Elsevier, Amsterdam, The Netherlands, pp. 141-295, 1995.

- 
- [103] Prins C. *A simple and effective evolutionary algorithm for the vehicle routing problem*. Computers and Operations Research, 31: 1985-2002, 2004. 31
- [104] Psaraftis, H.N. *A dynamic programming approach to the single-vehicle, many-to-many immediate request dial-a-ride problem*. Transportation Science, 14: 130-154, 1980.
- [105] Psaraftis, H.N. *An exact algorithm for the single-vehicle many-to-many dial-a-ride problem with time windows*. Transportation Science, 17: 351-357, 1983.
- [106] Psaraftis, H. N. *Dynamic vehicle routing problems*. In B. L. Golden, A. A. Assad, eds. Vehicle Routing: Methods and Studies. Elsevier North-Holland, Amsterdam pp. 223-248, 1988. 31
- [107] Psaraftis, H. *Dynamic vehicle routing: Status and prospects*. Annals of Operations Research, 61:143-164, 1995. 31
- [108] Reeves C.R. *Diversity and diversification in genetic algorithms: some connections with tabu search*. In R.F.Albrecht, C.R.Reeves and N.C.Steele (1993). Proc. of 1st International Conference on Artificial Neural Nets and Genetic Algorithms, Springer-Verlag, Vienna, 1993.
- [109] Renaud J., Boctor F. *A sweep based algorithm for the fleet size and mix vehicle routing problem*. European Journal of Operational Research, 140:618-628, 2002. 31
- [110] Rego C., Roucairol C. *A parallel tabu search algorithm using ejection chains for the vehicle routing problem*. In: Osma, I.H., Kelly J. (Eds.): Meta-Heuristics: Theory and Applications, 1996.
- [111] Rochat Y., Semet F. *A tabu search approach fo delivering pet food and flour in Switzerland*. Journal of the Operational Research Society, 45:1233-1246, 1994. 31
- [112] Rochat Y., Taillard E.D. *Probabilistic diversification and intensification in local search for vehicle routing*. Journal of heuristics, 1: 147-167, 1995.
- [113] Salhi, S., Rand, G.K. *Incorporating vehicle routing into the vehicle fleet composition problem*. European Journal of Operational Research, 66:313360, 1993. 31
- [114] Savelsbergh M.W.P. and Sol M. *DRIVE: Dynamic routing of independent VEHicles*. Operations Research, 46: 74-490, 1998. 31
- [115] Savelsbergh M.W.P. and Sol M. *The general pickup and delivery problem*. Transportation Science, 29:17-29, 1995.
- [116] Semet F., Taillard É. D. *Solving real-life vehicle routing problems efficiently using taboo search*. Annals of Operations research, 41:469 488, 1993. 31

- [117] Schönberger J., Kopfer H., and Mattfeld H. *A combined approach to solve the pickup and delivery selection problem*. In Operations Research Proceedings, pages 150-155, 2002.
- [118] Stützle T., Hoos H.H. *MAX-MIN Ant System*. Future Generation Computer Systems,16(8):889-914, 2000.
- [119] Taillard E.D. *A heuristic column generation method for the heterogeneous fleet VRP*. RAIRO, 33(1):1-14, 1999. [31](#)
- [120] Taillard ED. *Parallel iterative search methods for vehicle routing problems*. Networks, 23(8):661-673, 1994.
- [121] Tarantilis C., Kiranoudis C. and Vassiliadis V. *A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem*. European Journal of Operational Research, 152, 148-158, 2004. [31](#)
- [122] Tavakkoli-Moghaddam R., Safaei N., Gholipour Y. *A hybrid simulated annealing for capacitated vehicle routing problems with the independent route length*. Applied Mathematics and Computation, Volume 176, Issue 2, Pages 445-454, 2006.
- [123] Toth P. and Vigo D. *Models relaxations and exact approaches for the capacitated vehicle routing problem*. Discrete Applied Mathematics, 123:487-512, 2002. [31](#)
- [124] Volgenant T. and Jonker R. *On some generalizations of the traveling salesman problem*. Journal of the Operational Research Society, 38:1073-1079, 1987.
- [125] Wu P.L., Hartman J.C., and Wilson G.R. *An integrated model and solution approach for fleet sizing with heterogeneous assets*. Transportation Science, 39(1):87-103, 2005.
- [126] Yang J., Jaillet P. and Mahmassani H.S. *Study of a real-time multi-vehicle truck load pickup-and-delivery problem*. INFORMS, November, 2000. [31](#)