

# Recherche à voisinage large pour le problème de tournées de véhicules à voyages multiples

ROADEF 2016

Véronique François<sup>1</sup>, Yasemin Arda<sup>1</sup>, Yves Crama<sup>1</sup>, Gilbert Laporte<sup>2</sup>

<sup>1</sup> HEC- Ecole de gestion de l'Université de Liège

QuantOM, 4000 Liège, Belgique

{veronique.francois,yasemin.arda,yves.crama}@ulg.ac.be

<sup>2</sup> HEC-Montréal

CIRRELT, Université de Montréal, Montréal QC H3T 1J4 CANADA

{gilbert.laporte}@cirrelt.ca

**Mots-clés :** *tournées de véhicules, voyages multiples, recherche à voisinage large.*

## 1 Introduction

Nous étudions le problème de tournées de véhicules avec voyages multiples [6] dans lequel chaque véhicule est autorisé à effectuer un ou plusieurs voyages durant sa période de travail afin de servir un ensemble de clients géographiquement distribués. Ce type de problème est pertinent notamment lorsque les temps de trajet entre les clients sont élevés ou lorsque les commandes sont volumineuses comparées à la capacité des véhicules. Des opérateurs de recherche locale spécialement destinés au cas des voyages multiples sont proposés. Leur efficacité est évaluée par comparaison avec une approche combinée souvent utilisée dans la littérature qui associe des heuristiques développées pour le problème de tournée des véhicules à des techniques de bin packing ayant pour but d'affecter les voyages créés aux véhicules disponibles (voir notamment [4] et [6]).

## 2 Description du problème

Dans le problème de tournées de véhicules avec voyages multiples, une flotte de  $m$  véhicules est basée sur un unique dépôt et doit satisfaire les commandes des clients. Les véhicules sont de capacité homogène. La durée maximale de leur période de travail est limitée. L'objectif est de minimiser le temps de trajet total de la flotte.

Dans une solution du problème à voyages multiples, une tournée désigne l'ensemble des voyages effectués par un même véhicule. Chaque tournée commence et se termine par un dépôt. Elle contient un sous-ensemble de clients ainsi qu'éventuellement un ou plusieurs retours au dépôt. Nous appelons « dépôts internes » les retours au dépôt effectués en cours de tournée.

### 3 Méthode

Nous effectuons la comparaison entre les opérateurs spécifiques et l'approche combinée dans un cadre métaheuristique commun : la recherche adaptative à voisinage large [5].

#### 3.1 Opérateurs spécifiques

Les opérateurs de retrait et d'insertion du problème classique de tournées de véhicules sont modifiés pour pouvoir être utilisés dans le cas des voyages multiples.

Lorsqu'un client doit être inséré dans une solution partielle d'un problème avec voyages multiples, on envisage la possibilité d'insérer dans le même temps un retour au dépôt juste avant et/ou juste après ce client dans la séquence des visites effectuée par le véhicule. Lors du retrait d'un client, un dépôt interne est supprimé si la contrainte de capacité propre au véhicule et au voyage concerné le permet. De cette façon, les aspects de routage et d'affectation des voyages aux différents véhicules sont traités simultanément.

Les voisinages classiques d'amélioration (2-opt, échange, relocalisation...) sont quand-à-eux modifiés en considérant les dépôts internes comme de simples clients.

#### 3.2 Algorithmes

Deux algorithmes sont comparés dans ce travail : l'ALNSM (Adaptive Large Neighborhood Search with Multi-trip operators) et l'ALNSP (Adaptive Large Neighborhood Search with bin Packing).

L'ALNSM explore l'espace de solutions du problème avec voyages multiples dans lequel les contraintes de temps limitant la durée de la période de travail sont relâchées, entraînant potentiellement la présence d'heures supplémentaires. Cette violation est pénalisée dans la fonction objectif. Les solutions sont représentées par  $m$  tournées à voyages multiples. A chaque itération de l'ALNSM, ces tournées sont modifiées via le retrait et la réinsertion de clients mais également de dépôts internes. Différentes heuristiques de retrait et d'insertion basées sur les opérateurs spécifiques sont utilisées.

Dans l'ALNSP, l'espace de solutions du VRP est exploré de façon transitoire. Les solutions sont représentées par un nombre variable de véhicules (habituellement plus grand que  $m$ ). Chacun d'entre eux part du dépôt, sert une séquence de clients et revient au dépôt. A chaque itération, une solution de VRP est modifiée via le retrait et l'insertion de clients. Sous certaines conditions, cette solution est ensuite transformée en solution du problème avec voyages multiples grâce à une procédure de bin packing, permettant de déduire un nombre d'heures supplémentaires associé. Les routes (ou voyages) qui constituent les solutions de VRP visitées alimentent une mémoire adaptative de routes [4] utilisée régulièrement pour relancer la recherche à partir d'une nouvelle solution de VRP.

#### 3.3 Configuration

La paramétrisation des deux algorithmes est effectuée via irace [2], un outil de configuration automatique, sur un ensemble d'instances d'entraînement construites de façon à reproduire sommairement les caractéristiques des instances de référence communément utilisées dans la littérature pour mesurer la qualité des résultats. L'outil de configuration est utilisé non seulement pour attribuer une valeur aux paramètres numériques de nos ALNS mais également pour prendre une décision entre différents choix de conception des algorithmes, chacun de ces choix étant représenté par un paramètre booléen. A titre exemplatif, l'inclusion d'une heuristique d'insertion

ou de retrait particulière dans l'algorithme final, ou encore l'utilisation d'une procédure d'amélioration lorsqu'une nouvelle solution est acceptée, relèvent de paramètres booléens.

Pour chacun des deux ALNS :

- irace est utilisé sur les instances d'entraînement et renvoie plusieurs configurations statistiquement équivalentes en termes de qualité des résultats obtenus;
- la configuration qui obtient les meilleurs résultats sur les instances d'entraînement en termes de faisabilité est sélectionnée.

## 4 Résultats et expériences

La configuration choisie pour chaque algorithme est testée cinq fois sur chaque instance. L'ALNSM produit de très bons résultats bien que sans atteindre l'état de l'art [1]. Les résultats obtenus avec l'ALNSP sont excellents. L'ALNSM permet d'obtenir une solution faisable pour respectivement 97 des 99 instances de référence dont la faisabilité est prouvée. L'ALNSP obtient quant-à-lui une solution faisable pour chacune de ces 99 instances. Sur les 42 instances dont la valeur optimale est connue grâce à la méthode exacte développée par [3], l'ALNSM et l'ALNSP atteignent respectivement l'optimum pour 17 et 33 instances. Les meilleures solutions connues sont améliorées pour respectivement deux instances de références par l'ALNSM et pour dix instances par l'ALNSP.

Des expérimentations sont conduites et différents aspects des deux algorithmes sont examinés. L'outil de configuration automatique est ici utilisé à des fins d'analyse et non pas uniquement pour améliorer la qualité des résultats. Ces analyses sont conduites de différentes manières :

- les valeurs des paramètres des différentes configurations produites par irace sont examinées systématiquement de façon à obtenir de premiers indices sur l'importance de chacun d'entre eux. Par exemple, un paramètre numérique prenant des valeurs dispersées sur une très large plage peut indiquer que sa présence au sein de l'algorithme a peu d'importance ;
- la valeur de certains paramètres est imposée (ou contrainte) au moment de la configuration automatique afin d'observer l'impact sur la valeur des autres paramètres ainsi que sur les résultats numériques ;
- certaines configurations sont parfois légèrement modifiées avant d'être à nouveau testées, et ce après vérification de plusieurs hypothèses de travail, afin d'évaluer l'impact d'un paramètre en particulier sur la qualité des résultats obtenus.

Ces expériences nous permettent de tirer plusieurs conclusions, en particulier au sujet de l'utilité de certains composants.

Notamment, nous remettons en question l'efficacité de la roulette qui vise à sélectionner les heuristiques les plus performantes en cours d'algorithme dans la recherche adaptative à voisinage large [5]. Lorsqu'une présélection d'heuristiques de retrait et d'insertion est effectuée par l'outil de configuration automatique, le mécanisme adaptatif de la roulette n'impacte plus la qualité des solutions. Ce résultat nous semble d'autant plus significatif qu'à notre connaissance, il n'est pas reporté dans la littérature relative aux applications de l'ALNS au VRP. Dans le cas présent, le caractère adaptatif de la recherche à voisinage large ne démontre pas son utilité.

L'une des autres conclusions dégagée par nos expériences est que le rôle de la mémoire adaptative de routes utilisée dans l'ALNSP peut certainement être envisagé comme une perturbation occasionnelle plus que comme une pierre angulaire de la méthode. Il est par contre crucial de guider la recherche dans l'ALNSP en utilisant une fonction objectif qui pénalise les heures supplémentaires, et ce même en explorant l'espace de solutions du VRP.

## 5 Conclusions et perspectives

Les opérateurs spécifiques proposés pour le problème de tournées de véhicules à voyages multiples produisent de très bons résultats bien que légèrement en deçà de ceux obtenus via l'approche combinée. Nous proposons d'effectuer la même comparaison pour des problèmes plus contraints qui rendent le bin packing difficile, par exemple en présence de fenêtres de temps ou de backhauling. En effet, l'approche combinée permet de varier très rapidement l'affectation mais aussi la séquence des voyages de chaque véhicule. Au contraire, les opérateurs spécifiques tendent à rendre ces changements plus lents. Nous suspectons que cette caractéristique peut constituer un avantage dans le cas où la faisabilité est fortement dépendante du séquençement des clients.

Nous sommes convaincus que la réalisation d'expériences permettant de comprendre le fonctionnement des composants des algorithmes proposés est cruciale. Plus que la production de solutions de qualité, la compréhension des mécanismes de recherche locale sous-jacents constitue selon nous une contribution importante et trop souvent oubliée dans la littérature relative au VRP.

## Références

[1] D. Cattaruzza, N. Absi, D. Feillet, T. Vidal, A memetic algorithm for the multi trip vehicle routing problem. *European Journal of Operational Research*, 236: 833-848, 2014.

[2] M. López-Ibáñez, J. Dubois-Lacoste, T. Stützle, M. Birattari, The irace package, iterated race for automatic algorithm configuration, Tech. Rep. TR/IRIDIA/2011-004, IRIDIA, Université Libre de Bruxelles, Belgium, URL <http://iridia.ulb.ac.be/IridiaTrSeries/IridiaTr2011-004.pdf>, 2011.

[3] A. Mingozzi, R. Roberti, and P. Toth. An exact algorithm for the multitrip vehicle routing problem. *INFORMS Journal on Computing*, 25 :193-207, 2013.

[4] A. Olivera, O. Viera, Adaptive memory programming for the vehicle routing problem with multiple trips. *Computers & Operations Research*, 34 : 28-47, 2007.

[5] D. Pisinger, S. Ropke, A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34 : 2403-2435, 2007.

[6] E. Taillard, G. Laporte, M. Gendreau, Vehicle routing with multiple use of vehicles. *Journal of the Operational Research Society*, 47 : 1065-1070, 1996.