Un algorithme de recuit simulé pour le problème de tournées de véhicules avec fenêtres de temps et contraintes de synchronisation

Sohaib Afifi, Duc-Cuong Dang, Aziz Moukrim

▶ **To cite this version:**

Sohaib Afifi, Duc-Cuong Dang, Aziz Moukrim. Un algorithme de recuit simulé pour le problème de tournées de véhicules avec fenêtres de temps et contraintes de synchronisation. ROADEF 2013, 14ième congrès annuel de la Société française de Recherche Opérationnelle et d'Aide à la Décision, Feb 2013, Troyes, France. hal-00789236

HAL Id: hal-00789236
https://hal.science/hal-00789236

Submitted on 17 Feb 2013

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.

Un algorithme de recuit simulé pour le problème de tournées de véhicules avec fenêtres de temps et contraintes de synchronisation

Sohaib Afifi, Duc-Cuong Dang, Aziz Moukrim

Université de Technologie de Compiègne
Laboratoire Heudiasyc, UMR 7253 CNRS, 60205 Compiègne, France
{sohaib.afifi,duc-cuong.dang,aziz.moukrim}@hds.utc.fr

Mots-clés : *tournées de véhicules, synchronisation, recuit simulé, destruction/construction.*

1 Introduction

Les problèmes de tournées de véhicules ou Vehicle Routing Problems (VRP) [4] sont des problèmes d'Optimisation Combinatoire très étudiés en Recherche Opérationnelle. De manière informelle, considérant une flotte de véhicules disponibles et des clients répartis géographiquement à servir, la résolution d'un problème de tournées de véhicules consiste à déterminer pour chaque véhicule quels clients il doit visiter en précisant dans quel ordre, de sorte à ce que l'activité soit traitée le plus efficacement possible.

De nombreuses variantes du problème ont été étudiées. Nous nous intéresserons aux problèmes de tournées de véhicules avec des contraintes temporelles et des contraintes de synchronisation (VRPTWSyn). Il s'agit d'une variante du VRPTW où certains clients demandent plus d'une visite en même temps, indépendamment des véhicules. L'exemple le plus immédiat est celui des soins à domicile où certaines tâches doivent être réalisées par au moins deux assistants médicaux. Comme une extension du VRPTW, le VRPTWSyn est clairement NP-difficile.

Le VRPTWSyn a été modélisé la première fois dans [2] pour application dans des services de soins pour des personnes âgées. A notre connaissance, les méthodes de résolution de la littérature se basent toutes sur la programmation linéaire en nombres entiers (PLNE) [1, 2]. Dans ce travail, nous proposons un algorithme de Recuit Simulé (Simulated Annealing, SA) pour résoudre le VRPTWSyn. Cet algorithme intègre plusieurs techniques de recherche locale adaptées aux contraintes de synchronisation. Notre algorithme est capable de produire des solutions de qualité dans des temps de calcul plus courts en comparant avec les méthodes de la littérature [1, 2]. De plus, plusieurs nouvelles meilleures solutions ont été obtenues.

2 Le recuit simulé

L'idée principale d'un algorithme de SA [3] est d'accepter les solutions dégradées avec une certaine probabilité dans l'espoir d'échapper aux optima locaux et d'explorer l'espace des solutions le mieux possible. Dans notre SA, nous utilisons la destruction/construction avec priorités durant la phase de diversification. Une recherche locale avec les deux voisinages 2-opt* et or-opt est également appelée à chaque fois pour intensifier la recherche. L'adaptation de ces opérateurs à la synchronisation fait appel au calcul du chemin critique et celui de la fermeture transitive.

Nous avons testé notre méthode sur les instances introduites dans [2]. Ce benchmark comporte dix ensembles d'instances regroupées en trois catégories selon le nombre de clients. Chaque ensemble comprend cinq variantes selon la largeur de la fenêtre de temps. L'algorithme a été codé en C++ et compilé avec le GCC de GNU dans un environnement Linux.

Cette configuration est comparable à l'environnement de calcul utilisé dans [1, 2]. Selon le protocole proposé dans [1], toutes les méthodes ont été testées avec les variantes S (petites), M (moyennes) et L (grandes) fenêtres de temps.

Tableau 1 présente la comparaison de nos résultats avec ceux de la littérature. Les instances pour lesquelles toutes les méthodes fournissent les mêmes résultats sont exclues de ce tableau. Les colonnes n, m, s et *Best* présentent respectivement pour chaque instance : le nombre de clients, le nombre de véhicules, le nombre de clients à synchroniser et la meilleure solution trouvée. Le symbole (*) indique que l'optimalité de la meilleure solution a été prouvée par une méthode exacte. Les autres colonnes sont : MIP pour les résultats du solveur CPLEX reportés dans [2], H pour l'heuristique de *local-branching* proposée dans [2], BP1 et BP2 pour les résultats des deux algorithmes de *branch-and-price* présentés dans [1], et SA pour notre recuit simulé. Les deux colonnes Sol et CPU correspondent respectivement à la meilleure solution trouvée par chaque méthode et au temps de calcul associé.

On remarque que l'algorithme SA trouve toutes les solutions optimales connues (44 parmi 60) dans des temps de calcul très courts par rapport aux autres méthodes. De plus, SA améliore strictement les solutions pour les neuf instances : 7M, 7L, 8L, 9S, 9M, 9L, 10S, 10M et 10L.

Data	n	m	s	*Best*	MIP		H		BP1		BP2		SA	
					Sol	CPU	Sol	CPU	Sol	CPU	Sol	CPU	Sol	CPU
1L	20	4	2	3.39*	3.44	3600.00	**3.39**	120.00	**3.39**	107.41	**3.39**	11.91	**3.39**	0.29
2L	20	4	2	3.42*	3.58	3600.00	**3.42**	120.00	**3.42**	2.72	**3.42**	7.41	**3.42**	0.64
3M	20	4	2	3.33*	3.41	3600.00	**3.33**	120.00	**3.33**	17.57	**3.33**	4.31	**3.33**	0.92
4M	20	4	2	5.67*	5.91	3600.00	5.75	120.00	**5.67**	27.53	**5.67**	2.55	**5.67**	0.72
4L	20	4	2	5.13*	5.83	3600.00	5.30	120.00	**5.13**	9.74	**5.13**	7.69	**5.13**	4.66
6S	50	10	5	8.14*	-	-	-	-	**8.14**	3600.00	**8.14**	197.92	**8.14**	93.78
6M	50	10	5	7.70*	-	-	-	-	7.71	3600.00	**7.70**	3600.00	**7.70**	190.13
6L	50	10	5	7.14*	-	-	-	-	**7.14**	3279.48	**7.14**	3600.00	**7.14**	2440.95
7S	50	10	5	8.39*	-	-	-	-	**8.39**	1472.39	**8.39**	169.30	**8.39**	163.03
7M	50	10	5	7.49	-	-	-	-	7.67	3600.00	7.56	3600.00	**7.49**	199.23
7L	50	10	5	6.86	-	-	-	-	6.88	3600.00	6.88	3600.00	**6.86**	144.94
8S	50	10	5	9.54*	-	-	-	-	**9.54**	931.95	**9.54**	850.52	**9.54**	149.95
8M	50	10	5	8.54*	-	-	-	-	**8.54**	3600.00	**8.54**	3490.57	**8.54**	276.46
8L	50	10	5	8.07	-	-	-	-	8.62	3600.00	8.11	3600.00	**8.07**	335.72
9S	80	16	8	12.13	-	-	-	-	-	3600.00	12.21	3600.00	**12.13**	397.876
9M	80	16	8	10.94	-	-	-	-	11.74	3600.00	11.04	3600.00	**10.94**	641.838
9L	80	16	8	10.67	-	-	-	-	11.11	3600.00	10.89	3600.00	**10.67**	376.24
10S	80	16	8	8.82	-	-	-	-	-	3600.00	9.13	3600.00	**8.82**	3099.28
10M	80	16	8	8.01	-	-	-	-	8.54	3600.00	8.10	3600.00	**8.01**	757.87
10L	80	16	8	7.75	-	-	-	-	-	3600.00	-	3600.00	**7.75**	3247.71

TAB. 1 – Comparaison des résultats et le temps CPU

Ces résultats sont encourageants pour l'application du recuit simulé pour résoudre des problèmes de tournées de véhicules. Ils montrent également que l'opérateur destruction/réparation et différentes recherches locales peuvent être efficacement adaptées à la synchronisation.

Remerciements. Ce travail a été partiellement supporté par le Conseil Régional de Picardie et le Fonds Européen de Développement Régional (FEDER), dans le cadre du projet PRIMA.

Références

[1] D Bredström and M Rönnqvist. A branch and price algorithm for the combined vehicle routing and scheduling problem with synchronization constraints. (February), 2007.

[2] D Bredström and M Rönnqvist. Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, 191(1) :19–31, November 2008.

[3] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220 :671–680, 1983.

[4] P. Toth and D. Vigo. *The Vehicle Routing Problem*. Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics, 2002.