



HAL
open science

Un algorithme efficace pour le problème de plus court chemin multi-objectif

Antoine Giret, Yannick Kergosien, Emmanuel Neron, Gaël Sauvanet

► To cite this version:

Antoine Giret, Yannick Kergosien, Emmanuel Neron, Gaël Sauvanet. Un algorithme efficace pour le problème de plus court chemin multi-objectif. ROADEF 2018, Feb 2018, Lorient, France. hal-01654185

HAL Id: hal-01654185

<https://hal.science/hal-01654185>

Submitted on 3 Dec 2017

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 efficace pour le problème de plus court chemin multi-objectif

Antoine Giret¹, Yannick Kergosien¹, Emmanuel Néron¹, Gaël Sauvanet²

¹ Université de Tours, CNRS, LI EA 6300, ROOT ERL CNRS 6305, Tours, France
{antoine.giret,yannick.kergosien,emmanuel.neron}@univ-tours.fr

² La Compagnie des Mobilités, Tours, France
gael.sauvanet@geovelo.fr

Mots-clés : *plus court chemin, labelling, label-correcting, optimisation, multi-objectif, MOSP*

1 Introduction

Le problème du plus court chemin est un problème bien connu de l'optimisation combinatoire. Depuis les premières études réalisées par [1] et [2], plusieurs variantes de ce problème sont toujours étudiées dont le problème du plus court chemin multi-objectif (*MOSP problem*) défini comme suit. Soit $G = (\mathcal{V}, \mathcal{A})$ un graphe orienté avec \mathcal{V} l'ensemble des noeuds et \mathcal{A} l'ensemble des arcs. Soit c_{ij}^k le coût du critère $k \in \mathcal{K}$ associé à l'arc $(i, j) \in \mathcal{A}$, avec \mathcal{K} l'ensemble des critères et $c_{ij}^k \geq 0 \forall i, j, k$. Le problème consiste à trouver un ensemble de chemins \mathcal{P} allant d'une source s vers une destination t en minimisant des fonctions objectif de type somme. Le résultat du *MOSP problem* est l'ensemble des chemins strictement non-dominés aussi appelé front de Pareto. Un algorithme résolvant le *MOSP problem* – appelé *Label-Correcting with Dynamic update of Pareto Front (LCDPF)* – sera présenté, ainsi que son adaptation au cas des solutions extrêmement supportées. Des résultats expérimentaux seront présentés pour montrer l'efficacité de l'algorithme.

2 L'algorithme LCDPF

L'algorithme LCDPF utilise une méthode à deux phases introduite par [8] pour résoudre les problèmes bi-objectifs. La première phase est composée de deux étapes. Le but de la première étape est de déterminer des solutions initiales du front de Pareto. La seconde étape a pour but de déduire des bornes inférieures et supérieures sur les noeuds pertinents et d'exclure un maximum de noeuds non pertinents (c.-à-d. n'appartenant à aucun chemin du front de Pareto final). Ces deux étapes sont basées sur un ensemble de résolutions de problème mono-objectif sur un graphe inversé. La seconde phase a pour but de trouver l'ensemble des solutions non-dominées, en utilisant les informations obtenues lors de la première phase. Cette phase est basée sur l'algorithme classique de *label-correcting* introduit par [5]. Des contributions ont été faites, dont la principale est la mise à jour dynamique du front de Pareto, à partir de n'importe quel noeud. Cette mise à jour dynamique permet d'écarter de manière plus efficace des étiquettes (chemin partiels) lors de la recherche.

3 Résultats expérimentaux

Dans le but de comparer nos résultats avec la littérature, nous avons réalisé nos tests sur les graphes du 9^e challenge DIMACS. Ces graphes représentent la ville de New York (264K noeuds / 734K arcs), la baie de San Francisco (321K noeuds / 800K arcs) et l'état de Floride (1,1M

noeuds / 2,7M arcs). Dans chacune des comparaisons, les mêmes paires source / destination ont été utilisées. Le tableau 1 compare les temps d'exécution des algorithmes KDLS proposés par [4] et ceux de LCDPF. La colonne $|S|$ indique le nombre de solutions sur le front de Pareto final pour chaque instance. Le tableau 2 montre les temps d'exécution moyens de l'algorithme blSET proposé par [6], de l'algorithme Pulse proposé par [3] et ceux de LCDPF. Le tableau 3 compare les temps d'exécution de l'algorithme RL BSP proposé par [7]. Seules les solutions extrêmement supportées sont recherchées. Dans le plus part des cas, l'algorithme LCDPF possède les meilleurs performances.

#	KDLS ∞	KDLS 5	KDLS 4	KDLS 3	KDLS 2	KDLS 1	KDLS 0	LSDPF	$ S $
1	973.29	922.62	920.45	799.68	539.08	572.62	499.16	0.13	1,089
2	-	26,331.69	29,985.80	24,325.03	25,228.97	21,363.73	25,781.52	0.55	1,469
3	6.86	7.33	7.07	6.61	5.73	4.23	5.15	0.38	16

TAB. 1 – Comparaison avec [4]

	blSET	pulse	LCDPF	$ S $
FLA S	330,122.3	349.8	1,389	14.7
FLA M	562,195.9	348,114.7	1,647	116.1
FLA L	2,627,432.9	888,586	3,045	552.3

TAB. 2 – Comparaison avec [6] et [3]

		SLSET	RLBSP	LSDPF	LSDPF supported
FLA	avg	119,170	12,800	26,749.3 / 2,863.9*	1,878.4
	min	33,680	7,410	300	40
	max	571,920	28,010	600,000	5925

TAB. 3 – Comparaison avec [7]

Références

- [1] R. Bellman. On a routing problem. *Quarterly of applied mathematics*, 16(1) :87–90, 1958.
- [2] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1) :269–271, 1959.
- [3] D. Duque, L. Lozano, and A. L. Medaglia. An exact method for the biobjective shortest path problem for large-scale road networks. *European Journal of Operational Research*, 242(3) :788–797, 2015.
- [4] E. Machuca and L. Mandow. Lower bound sets for biobjective shortest path problems. *Journal of Global Optimization*, 64(1) :63–77, 2016.
- [5] E. Q. V. Martins. On a multicriteria shortest path problem. *European Journal of Operational Research*, 16(2) :236–245, 1984.
- [6] A. Raith. Speed-up of labelling algorithms for biobjective shortest path problems. In *Proceedings of the 45th annual conference of the ORSNZ. Auckland, New Zealand*, pages 313–322, 2010.
- [7] A. Sedeno-Noda and A. Raith. A dijkstra-like method computing all extreme supported non-dominated solutions of the biobjective shortest path problem. *Computers & Operations Research*, 57 :83–94, 2015.
- [8] E. Ulungu and L. Teghem. The two phases method : An efficient procedure to solve biobjective combinatorial optimization problems. *Foundations of Computing and Decision Sciences*, 20(2) :149–165, 1995.