

# Gestion algorithmique d'un réseau étendu dans le temps

José Luis Figueroa González, Alain Quilliot , Hélène Toussaint, Annegret Wagler  
LIMOS UMR 6158 CNRS /Université Clermont-Auvergne, Campus Universitaire des Cézaux,  
Aubière, France  
{jose\_luis.figueroa\_gonzalez, alain.quilliot, helene.toussaint,  
annegret.wagler}@uca.fr

**Mots-clés** : *optimisation combinatoire, réseaux étendus dans le temps, problème de repositionnement d'objets.*

## 1 Introduction

Les réseaux étendus dans le temps (*Time-Expanded Networks* ou TEN en anglais), construits en considérant les nœuds d'un réseau de base sur un certain espace-temps, sont de puissants outils de modélisation pour les problèmes impliquant des mécanismes de synchronisation. Ils sont bien adaptés aux représentations basées sur des modèles multiflots entre la production et la consommation de ressources. Mais il est difficile de dériver des algorithmes à partir de ce type de formulations, à cause à la fois de leur taille et de la propagation des erreurs induite par l'application de techniques d'arrondi à l'espace temporel. Afin de contourner ces difficultés, on peut essayer plusieurs types de schémas de décomposition : hiérarchique (par exemple, Benders, voir [1]), horizontal (par exemple, Lagrangien, voir [2]) ou multi-niveaux (voir [3], [4], [5]). Mais tous ces schémas posent la difficile question des interactions entre leurs composantes.

## 2 Projection/Remontage

Nous implémentons ici un autre type de schéma de décomposition, que nous appelons schéma de décomposition *Projection/Remontage* : on recherche d'abord (étape de *Projection*) une bonne solution projetée sur le réseau de base  $N$ , et on transforme (étape *Remontage*) cette solution projetée en une solution TEN complète. Définir l'étape *Projection* de manière formelle est plutôt intuitif, puisqu'il s'agit simplement de sauter la dimension temporelle du problème. Nous avons déjà commencé à étudier cette étape *Projection* dans une contribution précédente (voir [6]), en définissant un modèle *projeté* renforcé par des contraintes *d'élimination des sous-tours étendus*. Mais l'étape de *Remontage* peut être comprise de plusieurs manières. Nous avons déjà étudié une approche de *Remontage Fort*, (voir [7]) qui exigeait que la projection d'une solution TEN coïncide pleinement avec sa *empreinte* résultante de l'étape de *Projection*. Puisque cette approche s'est révélée trop restrictive, nous proposons ici une approche plus flexible qui garantit la faisabilité de l'étape de *Remontage*, et dérivons des algorithmes qui pourraient être appliqués efficacement à des contextes réactifs.

## 3 Problème de Repositionnement d'Objets

Bien que notre approche soit générique, nous nous référons ici, par souci de compréhension, à un modèle de multiflot lié à la gestion d'un : *Problème de Repositionnement d'Objets (Item Relocation Problem* ou IRP en anglais) (voir [10], [8], [9] et [11]). Nous nous appuyons sur celui-ci en raison à la fois de sa simplicité et de la continuité qu'il induit par rapport à nos

travaux précédents (voir [6], voir [7]). Ainsi, à partir de ce problème de référence IRP et de son modèle *projeté* Project\_IRP, nous définissons ce que nous appelons le problème de *Remontage Partiel* WEAK\_LIFT, qui exige que la projection d'une solution TEN coïncide partiellement avec une solution du modèle *Projeté*.

Tout d'abord nous améliorons le modèle projeté Project\_IRP de telle sorte que le pipeline :

$$\text{IRP} \rightarrow \text{Solution projetée} \rightarrow \text{Remontage partiel}$$

donne toujours une solution réalisable. Ensuite, nous définissons une formulation paramétrique du problème WEAK\_LIFT qui nous assure sa faisabilité et concevons un programme linéaire en nombres entiers exact pour cette formulation paramétrique.

Nous décrivons des algorithmes efficaces et bien adaptés aux instances à grande échelle impliquant des exigences de réactivité, et concevons un schéma heuristique *pipe-line* impliquant deux composants principaux : un problème TRANSFER, dont la résolution produit une sorte de sous-réseau *flottant* du réseau étendu dans le temps et ensuite un problème CARRIER\_ROUTE qui peut être transformé en un problème de tournées de véhicules avec des contraintes temporelles.

## 4 Résultats et Conclusions

Nous énonçons des résultats de complexité sur ces deux composants TRANSFER et CARRIER\_ROUTE, décrivons en détail les algorithmes associés et effectuons des expériences numériques. Finalement, nous concluons par quelques questions ouvertes.

## Références

- [1] F. Bendali, J. Mailfert, and A. Quilliot, *Flots entiers couplés par une contrainte de capacité*, Investigacion Operativa, 9, 2001.
- [2] M. Minoux, *Network synthesis and optimum network design problems : models, methods and applications*, Networks 19, p 313-360, 1989.
- [3] S. Alizadeh, P. Marcotte, G. Savard, *Two stage stochastic bilevel programming in transportation*, Transport. Res. B 58, p 92-105, 2013.
- [4] T. Kleinert, M. Labbe, I. Ljubic, *A survey on mixed integer programming in bi-level optimization*, EURO Jour. Comput. Optimization, 2021.
- [5] K. Stoilova, T. Stoilov, *Bi-level optimization application for urban traffic management*, Proc. 2020 FEDCSIS WCO Conf., p 327-336, 2020.
- [6] J. L. Figueroa Gonzalez, M. Baiou, A. Quilliot, H. Toussaint, A. Wagler, (2022). Branch-and-Cut for a 2-Commodity Flow Relocation Model with Time Constraints. Combinatorial Optimization. ISCO 2022. Lecture Notes in Computer Science, vol 13526. Springer, Cham.
- [7] J. L. Figueroa Gonzalez, A. Quilliot, H. Toussaint, A. Wagler, (2024) *Managing Time Expanded Networks : The Strong Lift Issue*, Graphs and Combinatorial Optimization : from Theory to Applications. AIRO Springer Series, vol 13. Springer, Cham.
- [8] M. Dell'Amico, E. Hadjicostantinou, M. Iori, S. Novellani, *The bike sharing rebalancing problem*, Omega 45, p 7-19, 2014.
- [9] M. Benchimol, P. Benchimol, B. Chappert, A. Taille, F. Laroche, F. Meunier, L. Robinet, *Balancing the stations of a self service bike hire system*, RAIRO-OR 45, p 37-61, 2011.
- [10] D. Chemla, F. Meunier, *Bike sharing systems : Solving the static rebalancing problem*, Discrete Optimization 10 (2), p 120-146, 2013.
- [11] F. T. Raviv and M. Tzur, *Static repositionning in a bike sharing system*, EURO Jour. Transport. Logist. 2, p 187-229, 2013.