Résolution de problèmes de tournées par la recherche des groupes homologiques du graphe modifié par force layout

Guillaume Bouleux[1], Lorraine Trilling[2], Gaia Bonomi[2], Antoine Chardon[2]

[1] Univ. Lyon, UJM-Saint Etienne, INSA-Lyon, DISP, EA 4570, F-69621 Villeurbanne, France

[2] Univ. Lyon, INSA-Lyon, UCBL, Univ Lumière Lyon 2, DISP, EA4570, 69621 Villeurbanne, France

`{guillaume.bouleux, lorraine.trilling, gaia.bonomi, antoine.chardon}@insa-lyon.fr`

Mots-clés : *VRP, topologie, groupe homologique, force layout, PLNE, smith normal form*

1 Introduction

La résolution de problèmes de tournées de véhicule (VRP) nécessite bien souvent une approche par Programmation Linéaire (PL) en intégrant les différentes contraintes structurelles du problème sous-jacent. Il s'agit en fait de trouver dans le graphe associé un circuit hamiltonien. Fréquemment pour les instances de grande taille, les contraintes conduisent à des temps de résolution trop importants, il apparaît alors nécessaire de guider l'algorithme.

Nous proposons donc d'utiliser le groupe homologique [4] pour guider la recherche d'optimum, en utilisant le force layout [6] pour la prise en compte de contraintes spécifiques au problème. Afin d'illustrer l'idée proposée dans cet article, prenons le cas d'une tournée où un ensemble de praticiens doivent rendre visite à des patients. Dans cet exemple, leur position est représentée par la FIG. 1 ci-dessous.

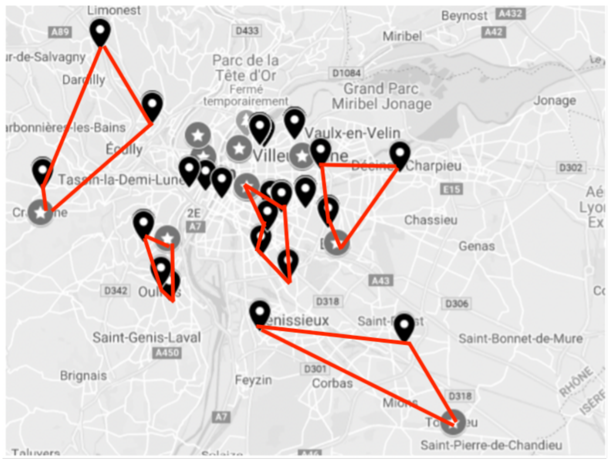

FIG. 1 – Positionnement de praticiens (étoile) et patients (marqueur noir et blanc)

2 Utilisation du groupe homologique

Nous proposons dans un premier de temps de nous intéresser à la répartition de ces patients et de chercher s'il n'existe pas des cercles naturels dans le graphe complet associé. Cela correspond en la recherche d'une information sur la topologie du graphe, ici une information dite **homologique**. En effet, l'utilisation de la topologie algébrique par la computation et la recherche des groupes homologiques du graphe, i.e. chercher les cercles intrinsèques (par extension, les sphères, hypersphères), permet de décrire la relation de cycle (circuit fermé) formé par

l'ensemble des patients-praticiens dans notre exemple, cf. cycles rouges sur la FIG. 1. L'idée est que s'il existe une telle information alors nous pourrons déjà proposer une tournée reliant un praticien à un ensemble de patients et résoudre directement le problème. Sinon, la recherche des groupes homologiques pourra être utilisée comme une heuristique pour guider la résolution. Cette idée a déjà été explorée dans des contextes de traitement de l'image [4, 2, 5] mais pas encore à notre connaissance pour la résolution de problèmes de tournées de véhicule au sens large (TSP, MTSP, Pickup & Délivery, Dial A Ride, ...).
Pour trouver les groupes homologiques, il faut dans un premier temps définir l'ordre des simplexes (noté i) qui seront utilisés, i.e un point ($i = 0$), un segment ($i = 1$), un triangle ($i = 2$), etc. On représente ensuite tous les simplexes du graphe par une matrice dite de boundary et nommée ∂_i. Cette matrice porte toute l'information homologique puisqu'elle est l'application linéaire décomposant le graphe dans la base des simplexes d'ordre i. La relation de quotient liant le noyau de ∂_i (information de cycle) et l'image de ∂_{i+1} (le cycle ne doit pas être un cycle formé d'un simplexe d'ordre supérieur) correspond aux générateurs homologiques.
Le premier objectif de cet article est de rappeler les fondements mathématiques du groupe homologique et de sa computation, qu'elle soit empirique par l'homologie persistante [1] ou explicite par l'algèbre matriciel [3] qui grâce à la décomposition de la matrice ∂_i sous une forme faisant apparaître ses pivots (Smith normal Form) permet d'atteindre les générateurs. Il s'agira ensuite de proposer un ensemble de résultats simples pour illustration.

3 Traduction des contraintes grâce au Force-Layout

Si maintenant il existe un certain nombre de contraintes dans le problème d'optimisation, par exemple une contrainte qui associe des patients aux praticiens, alors nous pouvons déformer le graphe complet afin de changer spatialement la disposition des noeuds du graphe et ensuite résoudre le problème. Nous transférons donc la contrainte matricielle du PL (ou PLNE selon le type de problème), dans le graphe directement. Ce procédé correspond au procédé et algorithme de Force-Layout [6].
Le challenge attendu ici est de traduire les contraintes habituelles, en forces d'attraction et de répulsion afin de déplacer les éléments du graphe (patients, villes, ...). Il sera ensuite possible d'utiliser l'information homologique pour aider encore une fois l'algorithme de résolution.

Références

[1] Guillaume Bouleux, Maël Dugast et Eric Marcon. « Information Topological Characterization of Periodically Correlated Processes by Dilation Operators ». In : *IEEE Transactions on Information Theory* 65.10 (2019), p. 6484-6495.

[2] Tamal K. Dey, Anil N. Hirani et Bala Krishnamoorthy. « Optimal homologous cycles, total unimodularity, and linear programming ». In : *Proceedings of the Annual ACM Symposium on Theory of Computing* January (2010), p. 221-230.

[3] Jean-Guillaume Dumas et al. « Computing Simplicial Homology Based on Efficient Smith Normal Form Algorithms ». In : *Algebra, Geometry and Software Systems*. Sous la dir. de Michael Joswig et Nobuki Takayama. Berlin, Heidelberg : Springer Berlin Heidelberg, 2003, p. 177-206.

[4] Emerson G. Escolar et Yasuaki Hiraoka. « Optimal Cycles for Persistent Homology Via Linear Programming ». In : sous la dir. de K. Fujisawa, Y. Shinano et H. Waki. Tokyo : Springer, 2016. Chap. Optimization on the Real world, p. 79-96.

[5] Ippei Obayashi. « Volume-Optimal Cycle : Tightest Representative Cycle of a Generator in Persistent Homology ». In : *SIAM Journal on Applied Algebra and Geometry* 2.4 (2018), p. 508-534.

[6] Ashley Suh et al. « Persistent Homology Guided Force-Directed Graph Layouts ». In : *IEEE Transactions on Visualization and Computer Graphics* 26.1 (2020), p. 697-707.