

ERO

Projet de recherche opérationnelle

— Auteurs: —

Alexandre Dias
Bastien Coutadeur
Martin Chevereau
Alexandre Castello
Gaëtan Pusel

Mai 2021



1 Méthode de résolution

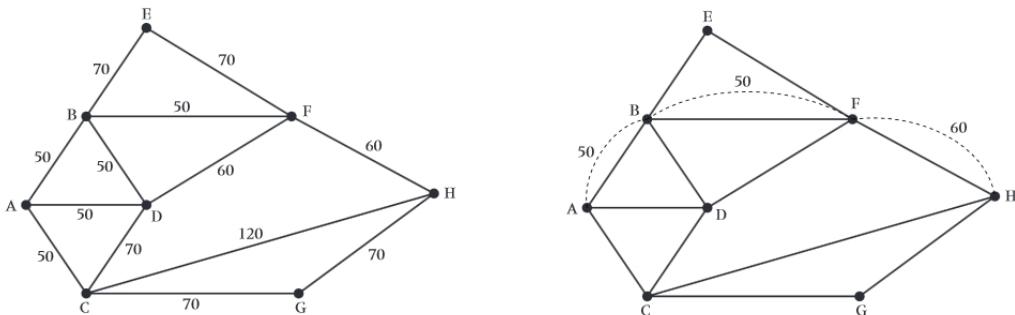
Afin de résoudre le problème de déneigement de Montréal nous avons mise en place une stratégie itérative d'attaque. En effet, nous avons identifié des sous problèmes résolubles de difficulté croissante, d'abord du coté de la théorie puis du coté de la pratique

1.1 Théorique

Tout d'abord nous avons fait une approche théorique

Nous nous sommes penchés sur la résolution du *Problème du postier chinois* étant donné sa ressemblance avec le déneigement de Montréal (visiter toutes les rues d'une ville). Les graphes étudiés étaient non dirigés et pondérés. Dans un premier temps nous avons étudier le cas trivial d'un graphe eulérien(sans sommets de degré impairs). La solution consistant évidemment à trouver un cycle eulérien.

La suite de la résolution du problème a donc consisté à rendre le graphe étudié eulérien.Nous avons donc résolu le cas où seul deux sommets étaient de degré impair. La solution a été de dupliquer les arrêtes présentes sur le plus cours chemin entre ces sommets afin de minimiser le coût du cycle final.



Enfin, dans le cas général, nous avons identifié les sommets de degré impair. Le même processus. sera donc effectué sur des paires de sommets afin de rendre le graphe eulérien. La minimisation du poids des arrêtes à rajouter s'est faite par un couplage parfait de poids minimum.

Cependant la solution théorique obtenue a vite montré ses limites appliquée à un cas réel. Tout d'abord utiliser un graphe orienté s'avère nécessaire afin de représenter une ville. De plus la taille plus conséquente des graphes à étudier, a soulevé des problèmes d'optimisation, notamment lors du couplage des sommets.

Pour les graphes non eulériens, il nous faut d'abord les rendre eulérien avant d'appliquer la même méthode. C'est ici que la difficulté se rajoute, pour rendre un graphe eulérien, où nous avons utilisé un algorithme de Dijkstra pour générer les plus courts chemins entre noeuds de degré impair.

La partie la plus complexe a été de générer la combinaison des paire de ces noeuds impair avec un poids minimum avant de pouvoir les ré-implémenter dans notre graphe. Une implémentation à base de liste d'adjacence nous a beaucoup aidé pour le parcours de graphe.

1.2 Pratique

La partie pratique doit prendre en compte les limitations d'une utilisation en temps réel. De fait, les algorithmes permettant le meilleur parcours auraient pu être remplacé par un algorithme offrant un temps de calcul plus court mais offrant un parcours moins optimisé. Néanmoins nous nous sommes axé sur un parcours optimal qui est séparé en plusieurs petit sous parcours afin d'obtenir un temps raisonnable.

Afin de récupérer des cartes routières nous avons utilisé osmnx, se basant sur la base de données de OpenStreetMap.

Nous avons commencé par séparer le déneigement de Montréal en plusieurs petits sous-problèmes

- Faire marcher osmnx et charger une petite ville dessus
- S'informer sur le framework osmnx en lisant la documentation
- Faire des recherches sur l'état de l'art du chinese postman

<https://github.com/matejker/everystreet> semblait répondre à nos attentes, malheureusement après des tests pour essayer de faire marcher ce github, il s'avère qu'il utilise un autre package qui utilise une vieille version de networkx.

Après cette tentative infructueuse d'utiliser un programme déjà construit, nous avons décidé de regarder un blog qui expliquait le chinese postman dans un cas réel avec osmnx

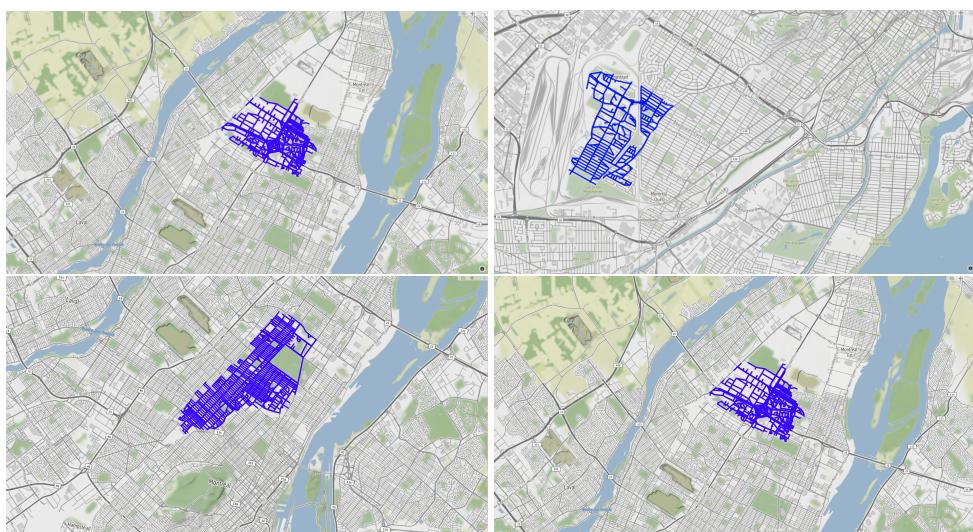
Il séparait le problème du chinese postman en quatre grandes steps :

- 1. Trouver tous les nœuds avec un degré impair.
- 2. Ajoutez des arêtes au graphe de façon à ce que tous les nœuds de degré impair deviennent pairs.
- 3. Étant donné un point de départ, trouver le tour eulérienne sur l'ensemble des données augmentées.
- 4. Puis calculer l'itinéraire du début à la fin

Ayant maintenant compris comment implémenter le Chinese Postman nous nous sommes posé des questions sur comment nous allions l'utiliser sur la totalité de Montréal

Nous nous sommes axé sur le déblaiement par arrondissement, en effet Montréal est séparé en 19 arrondissements et ils sont responsables en matière d'urbanisme. Il paraissait donc plus logique de séparer Montréal en arrondissement plutôt que en utilisant une séparation aléatoire ou arbitraire.

On a donc choisi de s'orienter plus vers le calcul d'une solution plus optimale tout en séparant la tâche grâce aux 19 arrondissements



Example de 4 Borough de montréal

Afin d'analyser les résultats que l'on obtenait en sortie nous avons produits des statistiques montrant la longueur de chemin, le pourcentage de chemin que l'on visite plusieurs fois.

Ce qui nous a permis de réaliser que nous visitons bien toutes les rues mais l'inconvénient était que l'on prend beaucoup de temps à faire tourner cette solution et donc il faut la lancer sur tous les différents borough de Montréal pour pouvoir tout déneiger.

Name of Borough	Hampstead
Length of path	76547.68 m
Length of the original map	26556.99 m
Length spent retracing edge	49990.70 m
Percent of mileage retraced	188.24%
Number of edges in circuit	571
Number of edges in original graph	219
Number of nodes in original graph	138
Number of edges traversed more than once	352

Afin d'analyser ces données de tailles de chemins à emprunter nous avons fais des recherches sur le prix de l'essence à Montréal (130 cents / Litres)

La consommation au canada pour les véhicules lourds comme les deneigeuses serait de 40L au 100 kilomètre donc 0,40L par kilomètre

On peut avoir une formule mathématique pour savoir le prix pour un déblayement : $P_e = \text{prix essence en cents} / \text{Litres}$ (130 cents / L)

L = longueur du chemin à parcourir en kilomètre

C = consommation en L / kilomètre (0,40)

P = Prix du déblayement

$$P = P_e * L * C$$

Donc si on l'applique pour l'arrondissement de Hampstead : $L = 76 \text{ km}$

$$P = 130 * 76 * 0,40 = 3952 \text{ cents} = 39\$CA$$

pour déneiger le burrow de Hampstead

Et nous pouvons même calculer le prix utilisé pour repasser sur des routes ce qui ne participe au déneigement : $L = 50 \text{ km}$

$$P = 130 * 50 * 0,40 = 2600 \text{ cents} = 26\$CA$$

avec s'il le faut un achat de déneigeuse coûtant approximativement 44 000 dollars canadiens

Les limites de la solution sont donc majoritairement le fait que en essayant d'avoir la solution optimal nous repassons beaucoup de fois sur le même chemin, et l'algorithme est assez lent.

2 Sources :

Blog expliquant Chinese Postman <https://www.datacamp.com/community/tutorials/networkx-python-graph-tutorial>

Prix essence a montreal
<https://www.essencemontreal.com/prices.php?l=f&tab=prices&l=f&prov=QC&city=Montreal>

Consomation deneigeuse
<https://tc.canada.ca/fr/services-generaux/politiques/annuel-2014-camions-porteurs-tracteurs-semi-remorques-fourgons-marchandise>