

## Optimisation hivernale

Patel Shubhamkumar, Julien Antoine, Pham Tan Khanh et Zabat Akram

### Survol du réseau routier via le drone

Pour pouvoir analyser l'ensemble des routes, i.e visiter l'ensemble des arrêtes du graphe ; nous avons décidé d'apparenter le problème à celui du postier chinois. Ce problème consiste à trouver un plus court chemin dans un graphe connexe non orienté qui passe au moins une fois par chaque arête du graphe et revient à son point de départ (1).

L'utilisation du drone nous permet de s'affranchir des contraintes physiques posées par les routes. Il est possible de se déplacer dans l'espace librement. N'importe quel type de graphe peut alors être traité comme 1 seule composante fortement connexe, ce qui ouvre le champ à un ensemble d'optimisations que nous verrons plus tard.

La clé du problème est de trouver un couplage parfait de poids maximal (2). Cette étape rajoute des arrêtes par rapport au graphe original. Ces arrêtes seront plus tard utilisées pour permettre au drone de prendre des raccourcis.

Une fois le graphe transformé le chemin parcouru par le drone correspond à un cycle eulérien. Pour optimiser ce cycle, lorsque le drone visite à la suite plusieurs arrêtes sur lesquels il est déjà passé, ces arrêtes sont alors supprimés et remplacé par 1 seule arrête. Cela évite au drone de devoir repasser par des routes déjà scannées et diminue son temps de trajet et la distance parcourue.

Afin de constater au mieux les résultats de l'algorithme il est recommandé de l'utiliser avec l'option -a pour afficher l'animation du drone parcourant la ville.

### Trajet d'un appareil de déblaiement

La fonction de résolution implémentée ici est un moyen simple de résoudre le problème de la recherche de chemin pour la ou les déneigeuses. Pour commencer simple, nous avons implémenté un programme pour résoudre des graphes dirigés en recherchant des graphes eulériens car ils peuvent être le cas le plus simple à résoudre à première vue. En suivant, (3) et (4), nous avons commencé par implémenter une fonction pour trouver des chemins eulériens dans un graph donné. Dans un premier temps, nous commençons par une conversion de notre edgelist en entrée en un objet Graph de la bibliothèque networkx. Avoir un objet Graph sera très utile car nous aurons accès à des fonctions prédéfinies pour trouver et travailler avec des graphes eulériens.

Une fois que nous avons un graphe G, nous commençons par vérifier que le graphe donné est dirigé et qu'il est soit connecté soit faiblement connecté. Un graphe est dirigé s'il a des arêtes avec une direction donnée, ce qui signifie que les arêtes indiquent une règle de traversée avec une direction d'un nœud u vers v dans le graphe. Un graphe connecté est un graphe où, étant donné deux sommets u et v, il existe un chemin de u à v. Et un graphe faiblement connecté est un groupe de nœuds qui sont mutuellement accessibles (u depuis v et inversement) en supprimant les directions donc en rendant le graph non dirigé.

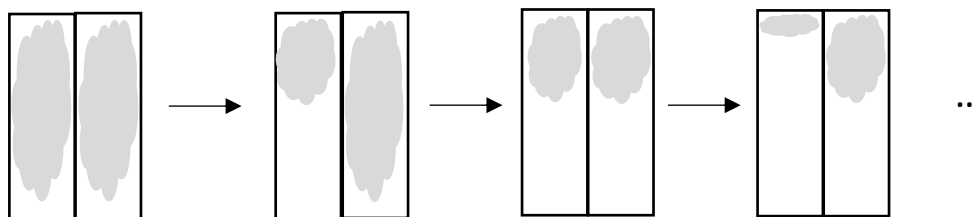
Après avoir vérifié la connectivité du graphe et son orientation, nous sommes prêts à trouver un circuit eulérien dans un graphe.

Nous commençons ensuite par vérifier si le graphe est déjà eulérien. Pour ce faire, nous utilisons la définition des graphes eulériens, qu'un graphe n'est eulérien que si chacun de ses sommets ont un degré pair, ce qui signifie que la somme de toutes les connexions entrantes et sortantes pour chaque nœud doit résulter en un nombre pair. Si le graphe est bien eulérien, nous procédons en recherchant un circuit eulérien. Si le graphe n'est pas eulérien, nous essayons de le rendre eulérien pour extraire un éventuel circuit eulérien.

La fonction renvoie un circuit eulérien que les déblayeuses peuvent utiliser pour nettoyer la neige des routes le plus rapidement possible.

Il est à noter que cette méthode n'est pas optimisée pour obtenir les meilleurs résultats. Les facteurs tels que les routes les plus utilisées, les niveaux de neige les plus élevés, les endroits les plus peuplés, la représentation géométrique des routes, le nombre de voies à nettoyer pour chaque route, et de nombreux autres facteurs n'ont pas été pris en considération lors de la mise en œuvre de cette solution. Une optimisation supplémentaire peut être utilisée pour nettoyer efficacement les routes.

Cela peut sembler tout à fait inefficace, mais en cas de problème du nombre de voies que la déneigeuse peut nettoyer, on peut imaginer que la déneigeuse procède en nettoyant chaque voie pour une voie donnée de la route en premier. Par exemple, étant donné une route à 2 voies, nous pouvons imaginer une procédure de nettoyage telle que cela ci :



Scénario

La neige peut être déneigée sur une voie que sur une autre pour éviter de visiter chacun les mêmes sommets pendant le processus de déneigement.

On peut imaginer un sous-groupe de notre graph qui aurait la priorité d'être les premiers endroits à être nettoyé en fonction de différents critères qui pourrait être mis en place selon les besoins et l'utilisation des routes à nettoyer.

Dans le cas où nous voudrions travailler avec plus d'un appareil de déblaiement, (5) propose une solution en utilisant le min-max k-Chinese postman problem qui trouve un ensemble de circuits pour chaque déblayeuse. Les auteurs utilisent une recherche avec tabou pour obtenir une solution optimale (6).

## Bibliographie

1. **Problème du postier chinois.** *Wikipedia.* [En ligne] [https://fr.wikipedia.org/wiki/Problème\\_du\\_postier\\_chinois](https://fr.wikipedia.org/wiki/Problème_du_postier_chinois).
2. Vie, Jill-Jênn et Dürr, Christoph. *Programmation efficace: Les 128 algorithmes qu'il faut avoir compris et codés en Python au cours de sa vie.* 2016. 9782340010055.
3. *A general implementation of Eulerian path.* Edwardo, S.Rivera-Hazim et Humberto, Ortiz-Zuazaga. 2015.
4. Ng, Peh H. *Designing Efficient Snow Plow Routes: A Service-Learning Project.*
5. *A tabu search algorithm for the min-max k-Chinese postman problem.* Dino Ahr, Gerhard Reinelt. Institute of Computer Science, University of Heidelberg : s.n., 2006. 0305-0548.
6. *Fondements et applications des méthodes de recherche avec tabous.* Soriano, P. et Gendreau, M. s.l. : RAIRO, 1997, Vol. 31.
7. *Snow Plow Route Optimization Webinar.* *Youtube.* [En ligne] <https://youtu.be/OugB2V8h71c>.
8. *Optimization Models for a Real-World Snow Plow Routing Problem.* Kinable, J., Hoeve, W-J. van et Smith, S. F. 2016. 978-3-319-33953-5.