Déneiger Montréal

L’objectif de cet exercice était de permettre à la ville de Montréal d’effectuer ses opérations de déneigement (épandage de sel et déblaiement de la neige) tout en respectant le budget de la ville ainsi qu’en fournissant un service de qualité.

Malgré l’objectif en apparence simple de cet énoncé, nous nous sommes heurté à plusieurs difficultés que nous allons détailler dans ce document.

▼ Les données utilisées et les contraintes

▼ Les données

Pour pouvoir lancer une simulation sur la ville de Montréal il a fallu réussir à la modéliser sous la forme d’un graph non-orienté d’abord, pour le passage du drone, puis un orienté pour maintenir le respect du code de la route par nos appareils.
Pour ce faire, nous avons utilisé une bibliothèque Python du nom de OSMNX, qui permet la création de télécharger, modélisation et analyser des réseaux de rues dans des villes depuis OpenStreetMap qui est une carte collaborative et gratuite.

Ensuite nous avons utilisé Folium qui permet de créer une carte, d’y ajouter des représenations des arrêtes d’un graph ainsi que de créer des animations en temps réel.

Ces outils sont très puissants, documentés et permettent d’obtenir énormément d’information sur les rues de la ville de Montréal, notamment leur longueur et leur sens de circulations.

Nous avons de plus effectués de nombreuses recherches pour comprendre et analyser les opérations de déneigement déjà en place, nous avons pu ainsi trouver ces données:

	Routes (km)	Temps d’une opération	Prix (1 opération)
Outremont	74	2h40	1M
Verdun	101	3h54	1.5M
Anjou	235	8h05	3.5M
Rivière-des-prairies-pointe-aux-trembles	718	25h23	10.6M
Le Plateau-Mont-Royal	169	6h30	2.5M
Total	1 128 (41,7%)		
Montréal	2700	96h	40M

▼ Les contraintes

Bien évidement, le déneigement doit se faire en temps réel et non pas à la vitesse de l’exécution d’un programme sur une machine. Il a donc fallu logiquement ajouter une dimension “real-time” au parcours de nos véhicules, en fonction de leur vitesse respectives.

De plus, nous sommes conscient que si le but est de réduire les coût liés à ces opérations qui peuvent parfois doubler d’une année à l’autre, comme à Saint-Honoré**.**
Il est donc judicieux de se fixer une limite d’employé et de véhicule afin de ne pas dépasser les nombres moyens de ces dernières années (3000 employés et 2000 véhicules). Dans notre cas, nous étudions 5 quartiers qui couvrent approximativement 40% de la surface totale de Montréal, nous nous limiterons donc, dans la limite du possible, à 800 véhicules par quartiers et 1200 employés.

Il semble évident que certaines rues doivent être priorisées dans ces opérations, comme celles qui permettent aux services de sécurité publique de circuler ou bien les axes principaux des arrondissement.

Les risques de bris mécanique de nos véhicules ne sont pas à exclure, lors, par exemple de l’aspiration de déchets enfouis. Nous sommes aussi conscient que les machines ne sont pas les seules à fatiguer et qu’il est important d’avoir un effectif en rotation, c’est pour cette raison que nous prévoyions 1,5 employés par véhicules.

Pour finir il faut prendre en compte la durée de ces opérations et comprendre à quel moment de la journée ou de la nuit il est le plus judicieux de les effectuer pour éviter la circulation ou les stationnements gênants qui sont dénombrés chaque années à plus de
50 000.

▼ Les hypothèses et choix de modélisation

Avant de pouvoir commencer à trouver une solution algorithmique viable, il est nécessaire d'établir une modélisation efficace de notre problème. En l'occurrence, le projet étant centré autour de la circulation, il semblait évident que les graphes constituent la meilleure manière de représenter une ville, ainsi que des quartiers, et plus généralement, un réseau.

L'implémentation par graphe pose une question bien spécifique : que faut-il utiliser en tant qu'arête, et que faut-il utiliser en tant que nœud ? Nous avons tout d'abord débuter par la solution qui semble à première vue la plus évidente : c'est-à-dire considérer les intersections de routes comme des nœuds, ainsi que les routes comme étant des arêtes. La ville de Montréal et ses quartiers sont donc divisés en plusieurs sous graphes.

▼ Les solutions, indicateurs et les scénarios

▼ Parcours du drone

Le parcours du drone doit répondre à des contraintes bien précises : parcourir toutes les arêtes du graphe, retourner au point de départ, et ce de la manière la plus efficace possible. Il s'agit ici en réalité d'un problème bien connu : le Chinese Postman Problem (Problème du Facteur). La solution est constituée de trois étapes :

- **Vérification de l'eulérienité du graphe :** un graphe est dit eulérien si et seulement si tous ses sommets ont un degré pair et qu'il est connexe (ou chaque composant connexe est eulérien).
- **La conversion en graphe eulérien :** si le graphe n'est pas eulérien, il est donc nécessaire de le convertir comme tel en ajoutant des arêtes, c'est ce qu'on appelle le graphe augmenté. Pour cela, on doit ajouter le moins d'arêtes possible pour que tous les sommets aient un degré pair.
- **La détection du circuit eulérien :** en utilisant l'algorithme de dijkstra, nous pouvons ainsi facilement détecter un circuit eulérien le plus court possible dans le graphe augmenté. Ce circuit correspond alors au circuit du drone.

▼ Déneigement

Dans un premier temps, les opérations de déneigement nécessitent la division des quartiers de la ville de Montréal en sous graphes, utilisés afin de répartir les déneigeuses, sous le principe du "diviser pour mieux régner". Pour diviser le graphe principal en sous-graphe, l'algorithme de Girvan Newman est utilisé. Il détecte les communautés en supprimant progressivement les arêtes du réseau original. Les composantes connectées du réseau restant sont les communautés. Au lieu d'essayer de construire une mesure qui nous indique quelles arêtes sont les plus centrales pour les communautés, l'algorithme de Girvan-Newman se concentre sur les arêtes qui sont les plus susceptibles d'être « entre » les communautés.

Comme expliqué dans la partie Implémentation, les sous-graphes produits peuvent éventuellement créer des graphes non fortement connexes, qui modéliseraient une impasse c'est-à-dire une intersection de laquelle ne peut revenir une déneigeuse sans violer le code de la route. Il fallait donc faire en sorte de rendre les sous graphes eulériens, afin que tous les nœuds puissent être accédés et qu'on puisse en revenir. Nous utilisons donc un algorithme que nous avons créée "Zaky Euler", suivant le principe suivant : pour chaque sous-graphe, commencer par le nœud ayant le degré le plus élevé, à partir de ce nœud, essayer de trouver le chemin le plus court vers tous les nœuds des mêmes sous-graphes ; s'il n'existe pas, utiliser le chemin le plus court en passant par le graphe parent.

Cet algorithme nous permet donc de nous assurer que les sous graphes soient bien eulériens, et donc de trouver les circuits des déneigeuses. La probabilité qu'il soit appelé est assez faible, étant donné que la configuration des routes d'une ville est déjà pensée à ne pas créer d'impasse.

▼ Les indicateurs

💰 Pour simplifier la visualisation des données, nous avons assumé qu'en moyenne une déneigeuse ne nécessite q'un employé. Nous avons donc pris en considération le salaire médian d'un conducteur ou conductrice de machinerie de déneigement à Montréal qui est de 32$CAD de l'heure.
À ces salaires viennent s'ajouter les prix des véhicules et du drone, qui ont un coût fixe, kilométrique et/ou horaire.

⏱️ La deuxième contrainte évidente est celle du temps. Il est parfois important de favoriser la rapidité d'exécution plutôt que le coût purement économique, cela permet de limiter les aléas auxquels l'on s'expose. Pour rappel, le temps moyen réel suite à une chute de 15cm de neige est de 4 jours, ainsi, plus les opérations durent, plus l'ont exposent à de nouvelles intempéries. Dans une région aussi fortement exposé, il est important de pouvoir régler rapidement ce problème et permettre aux habitants de circuler en toute sécurité.

📏 La distance parcourue par les véhicules et leur conducteurs est importante à prendre en compte pour mesurer l'efficacité des trajets choisis par nos différentes solutions.
Si l'on doit couvrir un réseau de rues de 100km, alors il faut que la distance du trajet choisis se rapproche le plus possible de la distance totale présente sur le réseau.

Nous pouvons ainsi estimer une fonction de coût total:

$C_I(n, t_h, d) = n * (\frac{500t_h}{24} + 1,1(d + t_{<8}) + 1,3t_{>8})$
Avec
$t_{<8}$ le nombre d'heure du tarif "premières heures", n le nombre de véhicule de type 1, d la distance parcourues par ces véhicules.
Pareil pour les type 2 avec leurs tarifs.

$C_e(e, t_h) = 32 * t_h * e$
Avec
e le nombre d'employés et t_h le nombre d'heures de travail.

$C_{drone}(t_j, d) = 0,01d + 100t_j$

Ainsi, on estime des poids pour chaque indicateurs et on obtient une fonction de "score":

$C_{total} = 0.4 * \frac{(C_I + C_{II} + C_e + C_{drone})}{C_{quartier}} + 0.3 * \frac{T_h}{t_{quartier}} + 0.3 * \frac{D_{parcourue}}{D_{quartier}}$

▼ Les limites

Tous les algorithmes de sous-divisions de graphes étant basés sur la séparation de nœuds, et non pas d'arêtes, cela signifie que des situations telles que la création d'un sous graphe non fortement connexe pose alors un problème, puisqu'en l'occurrence, une déneigeuse serait bloquée dans une impasse.

La deuxième modélisation est l'inverse de la première : routes en tant que nœuds, et arêtes en tant qu'intersections et carrefours. Dans ce cas, la création d'impasse est impossible, et la division en sous graphe est beaucoup plus efficace.

Nous avons décidé de continuer d'utiliser la première approche en raison d'une contrainte de temps

Un soucis engendré par cette approche est par exemple qu'il arrive qu'une déneigeuse passe par un endroit qui a déjà été déneigé. Toutefois, il vaut mieux cela qu'aucun passage du tout... s'il y a eu beaucoup de neige par exemple !

▼ Sources

https://www.journaldemontreal.com/2024/01/22/les-defis-du-deneigement-a-montreal

https://ville.montreal.qc.ca/pls/portal/docs/PAGE/ARROND_CDN_FR/MEDIA/DOCUMENTS/DENEIGEMENT.PDF

https://montreal.ca/articles/le-deneigement-sur-le-plateau-22510#:~:text=Déblaiement de la neige,de pistes cyclables du Plateau.

https://meteocentre.com/intermet/precipitation/neige_tempete.htm