7968 TANRÉ Mathieu

Chemin optimal dans un environnement incertain

Les GPS actuels fonctionnent sur des environnements dont les paramètres sont déterminés grâce à des données en temps réel.
Sans connexion satellite, ce type de GPS n'est plus utilisable. Étudier le plus court chemin dans un graphe en se basant sur des données statistiques permet une approche différente du problème.
Dans un monde où les échanges s'intensifient, il est nécessaire de développer des méthodes d'optimisation de déplacement, que cela soit en distance, en temps ou en coût, permettant d'économiser notamment du carburant. L'augmentation du trafic des voitures autonomes dans les prochaines années requerra l'utilisation d'algorithmes du plus court chemin performants.

Ce TIPE fait l'objet d'un travail de groupe.

Liste des membres du groupe :

- *BENHANA Bilal*

Positionnement thématique (ETAPE 1)

MATHEMATIQUES (Mathématiques Appliquées), INFORMATIQUE (Informatique Théorique), INFORMATIQUE (Informatique pratique).

Mots-clés (ETAPE 1)

Mots-Clés (en français)	**Mots-Clés** (en anglais)
Graphe	*Graph*
Recherche de chemin	*Path finding*
Optimisation	*Optimization*
Transport	*Transport*
Modélisation	*Modelling*

Bibliographie commentée

Un réseau routier ou ferroviaire se représente à l'aide d'un graphe. Il est composé de noeuds et d'arêtes, modélisant par exemple des villes et les routes les reliant. À chaque arête peut être affecté un poids, représentant une distance, un temps ou même un coût. Les graphes ont été utilisés par Euler lorsqu'il entreprit de résoudre le problème des sept ponts de Königsberg, ce qui a notamment donné naissance aux graphes eulériens. [1]

Les problèmes d'optimisation de déplacement peuvent être divisés en deux grandes catégories : l'optimisation monocritère ou multicritère.
Pour l'optimisation monocritère, le but est de trouver la meilleure solution suivant un unique paramètre (coût, temps, taux d'erreur, ...). En revanche, l'optimisation multicritère consiste à trouver la meilleure solution en fonction d'au moins deux de ces paramètres. [2]

De manière générale on cherche à travailler avec des vecteurs de décision, par exemple représenté par des n-uplets dont chaque composante correspond à un paramètre du problème. La résolution des problèmes d'optimisation monocritère dépendant d'un unique paramètre peut se faire avec une méthode déterministe ou bien une méthode stochastique. [3]

La méthode déterministe se caractérise par une exploration exhaustive de l'espace des vecteurs de décision. L'algorithme du plus court chemin renvoie un résultat exact.

La méthode stochastique, généralement utilisée pour les problèmes pour lesquels il n'existe pas un algorithme de résolution optimal en un temps polynomial, ne réalise pas une exploration exhaustive de l'espace des vecteurs de décision. En conséquence, elle ne fournit pas forcément le vecteur optimal mais un vecteur approché. Néanmoins, en général, la qualité des vecteurs fournis est très bonne. L'algorithme A* utilise cette méthode.

Dans un graphe pondéré, les poids peuvent dépendre du temps : ils sont dit dynamiques. Par exemple, un voyageur qui parcourt un trajet quotidiennement peut connaître des temps de déplacement différents sur ce trajet en raison des fluctuations de l'intensité du trafic entre les heures de pointes et les heures creuses.

On peut également utiliser des poids stochastiques : on introduit une distribution de probabilité discrète et on affecte à chacune des probabilités un poids déterministe. Cela permet de mieux refléter la réalité, car dans un réseau routier, il peut arriver qu'il y ait des incidents localisés tels que des zones de travaux sur la chaussée, de mauvaises conditions climatiques, des accidents ou des pannes de véhicules. Par conséquent, un réseau stochastique est une représentation plus réaliste d'un réseau routier réel qu'un réseau déterministe. [4]

L'algorithme de Dijkstra permet de trouver le plus court chemin en utilisant une méthode déterministe. Il peut être implémenté à l'aide de tas pour parvenir à un complexité en $O(|A|*\log(|S|))$, avec A l'ensemble des arcs du graphe et S l'ensemble de ses sommets. [5]

La complexité d'un algorithme importe peu dans un graphe de quelques dizaines de sommets et d'arcs. Mais dès que l'on souhaite travailler sur des réseaux bien plus grands, il est indispensable d'utiliser des algorithmes optimaux, sans quoi le temps de calcul d'un plus court chemin pourrait être gigantesque.
En plus de celui de Dijkstra, de nombreux autres algorithmes existent. C.Prins a realisé une étude comparative du temps d'exécution de ces divers algorithmes. Il constate que les algorithmes les plus efficaces ne sont pas les mêmes en fonction de la densité du graphe. [6]

La densité d'un graphe est le rapport entre le nombre d'arêtes du graphe et le nombre d'arêtes maximal que ce graphe peut comporter. Un graphe est dit creux si ce rapport est proche de 0, et dense si il est proche de 1. La densité d'un réseau routier est souvent proche de 0. [7]

Problématique retenue

Comment optimiser un déplacement, routier ou ferroviaire par exemple, dans un environnement incertain dont les paramètres dépendent du temps ?

Objectifs du TIPE

- Se familiariser avec les **graphes orientés pondérés**
- Distinguer les poids **déterministe** et **stochastique**
- Modéliser un réseau routier ou ferroviaire en s'intéressant à la notion de **densité d'un graphe**
- Comprendre les algorithmes de Bellman Ford, Dijkstra et Floyd-Warshall, les **implémenter** en Python, étudier leur **complexité**, et déterminer lequel est le plus **pertinent** en fonction de l'environnement à l'aide d'une étude empirique et théorique.

Références bibliographiques (ETAPE 1)

[1] Claudi Alsina : Plans de métro et réseaux neuronaux : *Collection le monde est mathématique*

[2] Mohamed Mejdi Hizem : Recherche de chemins dans un graphe à pondération dynamique : application à l'optimisation d'itinéraires dans les réseaux routiers : *https://tel.archives-ouvertes.fr/tel-00344958/document*

[3] Farida Manseur : Algorithmes pour un guidage optimal des usagers dans les réseaux de transport : *https://tel.archives-ouvertes.fr/tel-01760491/document*

[4] Aurelie Bousquet : Optimisation d'itinéraires multimodaux fondée sur les temps de parcours à l'échelle d'une agglomération urbaine dense : *https://tel.archives-ouvertes.fr/tel-00563197/document*

[5] Jean-Michel Helary : Le plus court chemin : *http://www.numdam.org/article/RO_1983__17_4_357_0.pdf*

[6] C. Prins : Comparaison d'algorithmes de plus courts chemins sur des graphes routiers de grande taille : *Recherche opérationnelle, tome 30, no 4 (1996), p. 333-357*

[7] Gabriele De Luca : Graphs: Sparse vs Dense : *https://www.baeldung.com/cs/graphs-sparse-vs-dense*

DOT

[1] *Février 2020 : Familiarisation avec la notion de graphe. Lecture du livre de Claudi Alsina ([1]).*

[2] *Avril 2020 : Début de l'implémentation avec le langage de programmation Python. Utilisation uniquement des poids déterministes statiques, avec l'algorithme de Dijkstra classique.*

[3] *Mai 2020 : Découverte des poids stochastiques et dynamiques, permettant de modéliser un environnement incertain et dépendant du temps. Implémentation de ces poids.*

[4] *Juin 2020 : Utilisation de la bibliothèque Graphviz, permettant d'afficher les graphes et faciliter les tests.*

[5] *Octobre 2020 : Réécriture du code en utilisant la programmation orientée objet, permettant un code plus léger et adapté aux différents type de poids.*

[6] *Janvier-février 2021 : Application de l'algorithme au réseau de métro parisien grâce au développement d'une interface graphique à l'aide de la bibliothèque Tkinter d'édition des graphes et de calcul du plus court chemin pour tous types de poids.*

[7] *Mai 2021 : Implémentation des algorithmes de Belllman-Ford, Floyd-Warshall et Dijkstra avec tas, étude de leur complexité et comparaison du temps d'exécution de ces algorithmes en fonction du nombre de noeuds et de la densité du graphe.*