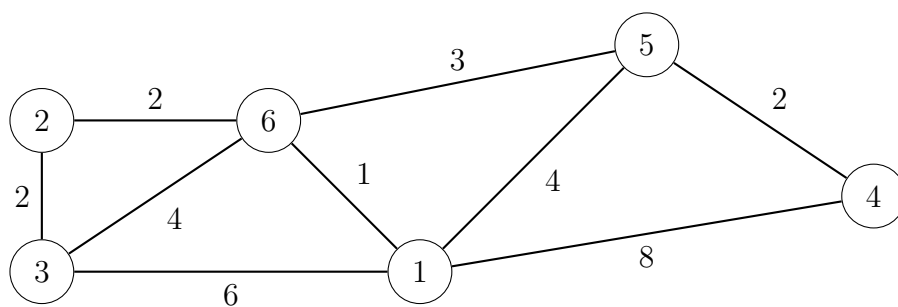


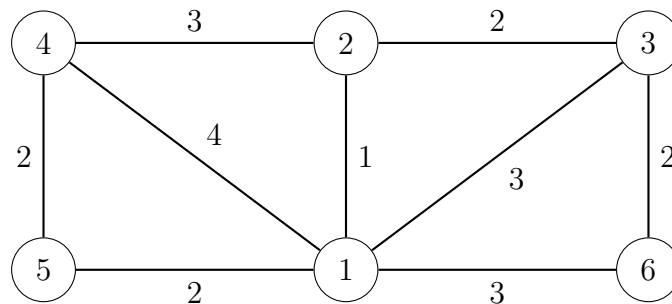
1 Présentation générale

L'objectif de ce TP est de vous faire découvrir certaines heuristiques classiques utilisées pour traiter le problème du voyageur de commerce. Dans ce sujet, nous travaillerons avec deux jeux de données fictives afin de pouvoir appliquer les heuristiques sur des exemples. Dans chacun des cas, le sommet 1 est l'usine et les autres sont des magasins.

Jeu de données 1 :



Jeu de données 2 :



2 Heuristiques gloutonnes

2.1 Tournée croissante

Le principe de cette heuristique est simple, on commence par l'usine et on prend les magasins dans l'ordre dans lequel ils nous sont donnés.

Question 1 : Donnez la tournée croissante pour les deux jeux de données.

Question 2 : Déterminez la distance parcourue dans les deux cas.

2.2 Plus proche voisin

L'heuristique du plus proche voisin est l'une des heuristiques les plus naturelle pour ce problème. Elle consiste basiquement à se rendre au magasin visité le plus proche. Plus précisément, on procède de la façon suivante :

1. On se place sur l'usine.
2. Tant qu'il y a des magasins non visités, on cherche le magasin non visité le plus proche de là où l'on se trouve et l'on s'y rend.
3. Quand tous les magasins sont visités, on rentre à l'usine.

Question 3 : Donnez la tournée obtenue pour les deux jeux de données.

Question 4 : Déterminez la distance parcourue dans les deux cas.

Question 5 : Quelles structures de données utiliseriez vous pour coder cette heuristique ?

2.3 Insertion proche et loin

Si les précédentes heuristiques pouvaient s'appliquer directement, les heuristiques d'insertion nécessitent de définir au préalable deux petites notions.

Distance d'un lieu L à un couple de lieu (A,B) : Soient A,B,L trois lieux (usine(s) ou magasin(s)), la distance de L au couple (A,B) se calcule de la façon suivante :

$$L.distance(A, B) = A.distance(L) + L.distance(B) - A.distance(B)$$

Question 6 : Calculez la distance de l'usine au deux premiers magasins dans le premier jeu d'essai.

Question 7 : Que représente cette distance ?

Distance d'un lieu L à une tournée T : On définit la distance d'un lieu L à une tournée T comme le minimum des distances de L aux différents couples de lieux successifs de la tournée.

Par exemple si $T=(A,B,C,D)$:

$$L.distance(T) = Min(L.distance(A, B), L.distance(B, C), L.distance(C, D), L.distance(D, A))$$

Question 8 : Calculez la distance de l'usine à la tournée composée des 4 premiers magasins du premier jeu d'essai

Question 9 : Que représente cette distance ?

Maintenant que nous disposons de ces deux notions, nous pouvons nous en servir pour construire des tournées.

Insertion au plus proche : La première heuristique d'insertion fonctionne de la façon suivante :

1. On commence par trouver les deux lieux les plus éloignés l'un de l'autre : A et B. On définit la tournée $T=(A,B)$.
2. Tant qu'il existe des lieux non visités, on cherche le lieu (non visité) dont la distance à la tournée T est la plus petite.
3. On insère ce lieu dans la tournée là où cela augmente le moins la longueur de la tournée.

Question 10 : Faites tourner cette algorithmme sur un exemple avec votre enseignant.

Question 11 : Donnez le pseudo-code permettant de trouver les deux lieux les plus éloignés.

Question 12 : Donnez le pseudo-code permettant de trouver le lieu non visité le plus proche d'une tournée.

Insertion au plus loin : il existe une variante de cette heuristique qui consiste à ajouter à chaque étape non pas le lieu le plus proche de la tournée mais celui le plus loin de la tournée.

Question 13 : Faites tourner cette algorithmme sur un exemple avec votre enseignant.

3 Heuristiques de recherche locale

Dans cette SAÉ, nous allons nous intéresser à une heuristique de recherche locale. Pour cette heuristique, nous aurons besoin de définir le "voisinage" d'une tournée dans notre cas.

Voisinage d'une tournée : Deux tournées T_1 et T_2 sont dites voisines si on peut passer de T_1 à T_2 en inversant juste deux lieux successifs dans la tournée T_1 .

Question 14 : Listez tous les tournées voisines de la tournée (1,2,3,4,5,6).

Le principe de notre heuristique de recherche locale est le suivant :

1. On part d'une tournée T donnée par l'une des heuristiques précédentes.
2. On regarde si une des voisines de T est meilleurs que T .
3. Si on en trouve une, on remplace T par cette tournée et on recommence.
4. Si on n'en trouve pas, on s'arrête (et on renvoie T).

Question 15 : Faites tourner cette heuristique sur le premier jeu de données en partant de la tournée croissante.

Question 16 : Proposez des variantes de cette heuristique.