Théorie des graphes

M1 - IWOCS

Devoir maison 2023

Vous être vendeur de farine et vous voulez fournir les boulangeries de votre ville. Chaque boulangerie b a besoin d'une quantité $k(b)$ de farine. Vos camions ont un capacité c de farine, et doivent partir et revenir au dépôt. Vous avez une matrice de temps entre toutes les boulangeries et le dépôt. Ce problème est connu sous le nom de Capacitated Vehicle Rounting Problem (CVRP). Soit le graphe complet $G = (V, E)$ avec V les boulangeries et le dépôt, et E l'ensemble des arêtes. Le poids de l'arête $e(i, j) \in E$ est le temps de route $t(i, j)$ entre les sommets i et j. Le but est de former des tournées de coût minimum, en allant sur tous les sommets tel que la quantité d'une route ne dépasse pas la capacité. La quantité d'une route est la somme de toutes les quantités des sommets qui constituent la route. Un algorithme glouton est l'heuristique de Clarke and Wright. Elle fonctionne sur les *savings*, soit le coût qu'on peut sauver en reliant les sommets i et j, définit par

$$s(i, j) = t(D, i) + t(D, j) - t(i, j)$$

avec i et j les sommets sauf le dépôt, et D le dépôt.

L'algorithme est le suivant:

Construction des routes initiales, avec pour chaque sommet i sauf le dépôt, deux arêtes $e_{D,i}$
Calculer tous les savings $s_{i,j}$, les mettre dans la liste S
Trier S par ordre décroissant
for $s_{i,j} \in S$ **do**
  if il existe une arête $e_{D,i}$ et $e_{D,j}$ et la quantité de la route de i plus la quantité de la route de j est inférieur à la capacité du véhicule et $s_{i,j} > 0$ **then**
    Supprimer les arêtes $e_{D,i}$ et $e_{D,j}$
    Ajouter l'arête $e_{i,j}$

Algorithm 1: Algorithme de Clarke and Wright

(1) Appliquez l'algorithme sur le graphe d'exemple G, avec comme capacité du véhicule $c = 10$. Expliquez méticuleusement les étapes.

(2) Trouvez un exemple où l'heuristique de Clarke and Wright ne donne pas la solution optimale. Trouvez un exemple différent des autres étudiants.

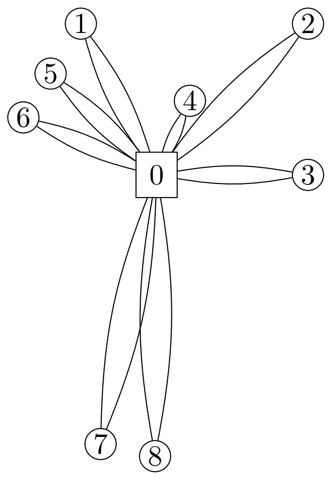
(3) Quelle est la complexité de l'algorithme ? Donnez-la et expliquez votre démarche pour la trouver.

	0	1	2	3	4	5	6	7	8
0	0	22	28	20	10	19	19	36	37
1	22	0	30	36	17	7	14	60	60
2	28	30	0	20	18	34	39	60	60
3	20	36	20	0	18	36	38	50	50
4	10	17	18	18	0	18	22	50	50
5	19	7	34	36	18	0	6	50	50
6	19	14	39	38	22	6	0	50	50
7	36	60	60	50	50	50	50	0	7
8	37	60	60	50	50	50	50	7	0

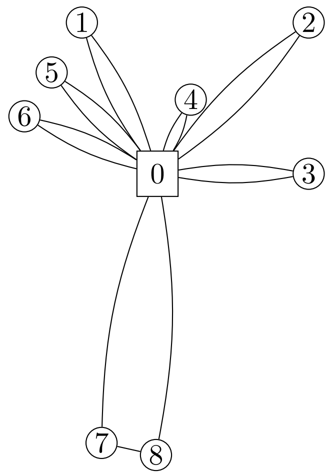
Table 1: Poids des arêtes du graphe G

Sommet s	1	2	3	4	5	6	7	8
$k(s)$	7	2	1	4	5	1	3	4

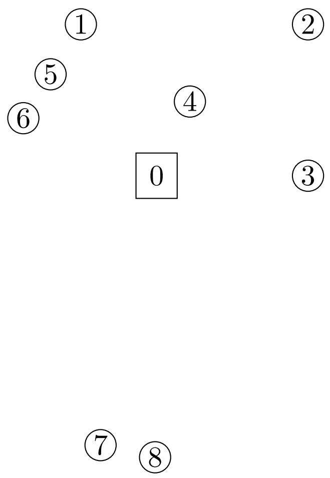
Table 2: Quantité requise par les sommets du graphe G



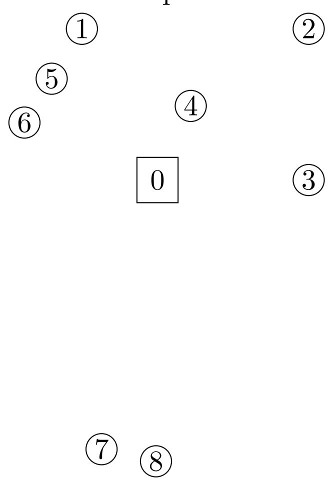
Routes initiales du graphe G



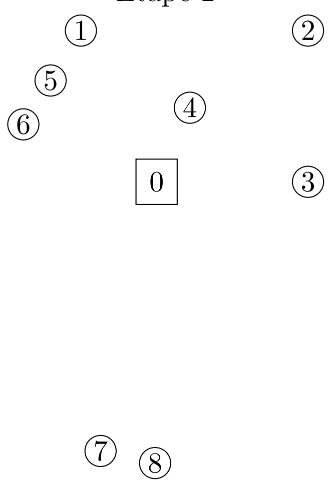
Étape 1



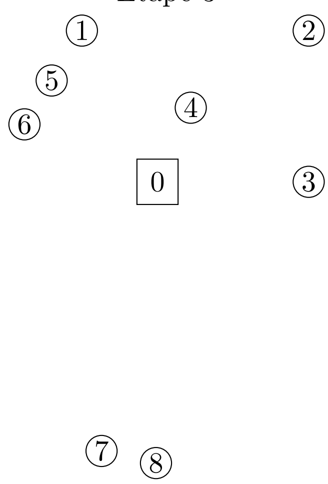
Étape 2



Étape 3



Étape 4



Étape 5