TD n°11

Variations sur les chemins

Exercice 1 [**Échec de Dijkstra**] Montrer que Dijkstra peut échouer à trouver des plus courts chemins quand les "longueurs" sont négatives, quand bien même le problème a une solution (c'est-à-dire quand il n'y a pas de cycle de poids négatif).

Exercice 2 [**Chemin de capacité maximum**] On a un graphe valué. La **capacité** d'un chemin est la valeur minimum d'un arc (ou d'une arête) de ce chemin. Proposez un algorithme qui calcule un chemin de capacité maximum entre deux points donnés.

Exercice 3 [**Chemin avec réduction de gros**] On s'intéresse maintenant au problème du chemin le plus économique dans le modèle suivant :
- Le coût de traverser un nœud est une constante (la même pour tous les nœuds) C_N
- Le coût de traverser un arc uv est sa valeur $w(uv)$.
- Pour un chemin plus long qu'une certaine valeur L_{50} une réduction de 50% s'applique (le coût total est divisé par deux).
- Et pour tout chemin de longueur entre L_{20} et L_{50} une réduction de 20% s'applique

La fonction w et les constantes C_N, L_{20} et L_{50} sont des données fournies avec le graphe.

Discutez des diffucultés posées par ce modèle et proposez un algorithme.

Exercice 4 [**Comparaison Dijkstra / Floyd-Warshall**] Discutez de l'efficacité comparée des deux algorithmes suivants, pour cacluler la *matrice des distances* d'un graphe (distances entre tous couples de points) :
- Floyd-Warshall
- n fois Dijkstra, un par sommet du graphe.

Exercice 5 [**fermeture transitive**] On considère un graphe orienté. La fermeture transitive est définie ainsi. Pour tous sommets u, v et w, s'il y a des arcs uv et vw mais pas uw, on rajoute l'arc uw (qui est appelé *arc de transitivité*). On itère cette opération jusqu'à ce que plus aucun arc ne puisse être ajouté.

Proposez un algorithme calculant la fermeture transitive d'un graphe et évaluez sa complexité. Donnez une borne inférieure pour le calcul de la fermeture transitive et comparez.