MTH6412B

Implémentation d'algorithmes de recherche opérationnelle

Tournées minimales

Dominique Orban

dominique.orban@polymtl.ca

Mathématiques et Génie Industriel
École Polytechnique de Montréal

Tournées

Le problème de la tournée, ou du *cycle Hamiltonien* est un problème classique de graphes posé par Hamilton en 1859.

Étant donné un graphe non orienté, trouver un cycle qui passe par chaque sommet une et une seule fois.

Tournées

Le problème de la tournée, ou du *cycle Hamiltonien* est un problème classique de graphes posé par Hamilton en 1859.

Étant donné un graphe non orienté, trouver un cycle qui passe par chaque sommet une et une seule fois.

Condition nécessaire d'existence : le graphe doit être biconnexe. Cette condition n'est pas suffisante.

Tournées

Le problème de la tournée, ou du *cycle Hamiltonien* est un problème classique de graphes posé par Hamilton en 1859.

Étant donné un graphe non orienté, trouver un cycle qui passe par chaque sommet une et une seule fois.

Condition nécessaire d'existence : le graphe doit être biconnexe. Cette condition n'est pas suffisante.

Il n'existe pas d'algorithme connu efficace pour résoudre ce problème.

Tournées

Le problème de la tournée, ou du *cycle Hamiltonien* est un problème classique de graphes posé par Hamilton en 1859.

Étant donné un graphe non orienté, trouver un cycle qui passe par chaque sommet une et une seule fois.

Condition nécessaire d'existence : le graphe doit être biconnexe. Cette condition n'est pas suffisante.

Il n'existe pas d'algorithme connu efficace pour résoudre ce problème.

Notre problème est plus compliqué : trouver une tournée *minimale*.

Tournées

Le problème de la tournée, ou du *cycle Hamiltonien* est un problème classique de graphes posé par Hamilton en 1859.

Étant donné un graphe non orienté, trouver un cycle qui passe par chaque sommet une et une seule fois.

Condition nécessaire d'existence : le graphe doit être biconnexe. Cette condition n'est pas suffisante.

Il n'existe pas d'algorithme connu efficace pour résoudre ce problème.

Notre problème est plus compliqué : trouver une tournée *minimale*.

Restons optimistes.

Tournées minimales approchées

Trouver une tournée minimale est difficile sans utiliser d'heuristiques sophistiquées. Cependant, on peut parfois trouver de bonnes tournées minimales *approchées* en utilisant les outils développés dans les laboratoires précédents.

Tournées minimales approchées

Trouver une tournée minimale est difficile sans utiliser d'heuristiques sophistiquées. Cependant, on peut parfois trouver de bonnes tournées minimales *approchées* en utilisant les outils développés dans les laboratoires précédents.

On examine deux algorithmes :

1. l'algorithme de Rosenkrantz, Stearns et Lewis (simple) ;
2. l'algorithme de Held et Karp (plus difficile).

Tournées minimales approchées

Trouver une tournée minimale est difficile sans utiliser d'heuristiques sophistiquées. Cependant, on peut parfois trouver de bonnes tournées minimales *approchées* en utilisant les outils développés dans les laboratoires précédents.

On examine deux algorithmes :

1. l'algorithme de Rosenkrantz, Stearns et Lewis (simple) ;
2. l'algorithme de Held et Karp (plus difficile).

On suppose notre graphe *complet*.

Algorithme de Rosenkrantz, Stearns et Lewis

Condition : $c(u,w) \leq c(u,v) + c(v,w)$.

1. Choisir un nœud qui jouera le rôle de racine ;
2. calculer un arbre de recouvrement minimal en utilisant cette racine ;
3. ordonner les sommets du graphe suivant un parcours en préordre de l'arbre de recouvrement minimal (i.e., dans l'ordre de visite) ;
4. cet ordre détermine une tournée dans le graphe de départ.

Algorithme de Rosenkrantz, Stearns et Lewis

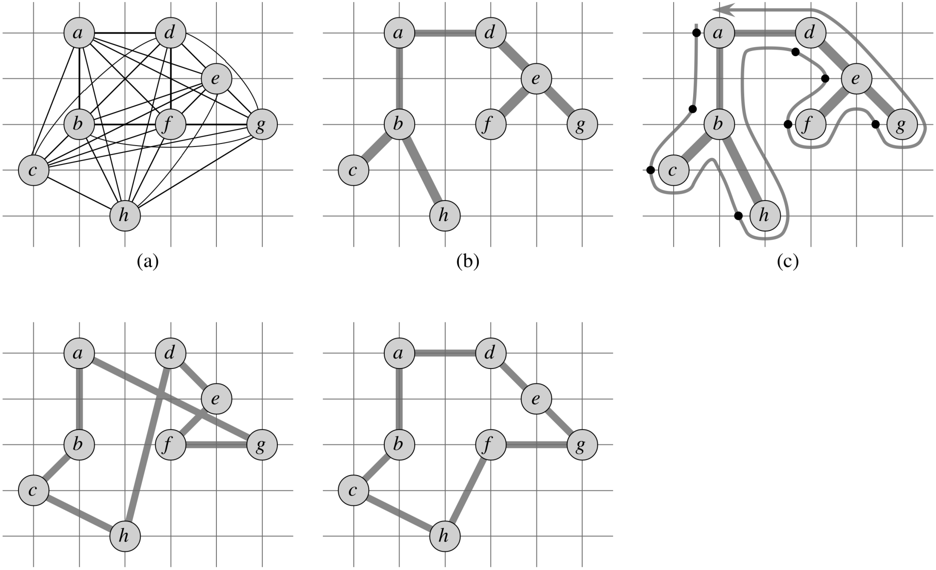
Condition : $c(u,w) \leq c(u,v) + c(v,w)$.

1. Choisir un nœud qui jouera le rôle de racine ;
2. calculer un arbre de recouvrement minimal en utilisant cette racine ;
3. ordonner les sommets du graphe suivant un parcours en préordre de l'arbre de recouvrement minimal (i.e., dans l'ordre de visite) ;
4. cet ordre détermine une tournée dans le graphe de départ.

Théorème

L'algorithme de Rosenkrantz, Stearns et Lewis fournit une tournée dont le poids est inférieur à 2 fois le poids d'une tournée optimale.

Illustration



(a) (b) (c)

(d) (e)

Algorithme de Held et Karp

(voir notes manuscrites)

Références à lire

1. *The Traveling-Salesman Problem and Minimum Spanning Trees* (Held et Karp) : introduction, sections 1 et 4 (attention, l'algorithme de la section 4 n'est pas celui qu'on demande d'implémenter mais aide à la compréhension) ;
2. *The Traveling-Salesman Problem and Minimum Spanning Trees: Part II* (Held et Karp) : introduction, sections 1, 2 et 4 ;
3. *An Effective Implementation of the Lin-Kernighan Traveling Salesman Heuristic* (Helsgaun) : section 4.1 (l'algorithme de Held et Karp se trouve à la page 25).

Quatrième partie de ce laboratoire

1. Implémenter l'algorithme de Rosenkrantz, Stearns et Lewis ;
2. implémenter l'algorithme de montée de Held et Karp (HK) ;
3. l'algorithme contient plusieurs paramètres :
   3.1 Kruskal vs. Prim ;
   3.2 le choix du sommet privilégié (la racine) ;
   3.3 le choix de la longueur de pas t (HK) ;
   3.4 le choix du critère d'arrêt (HK).
4. en jouant sur ces paramètres, identifier les meilleurs tournées possibles sur les problèmes du TSP symétrique (vous pouvez utiliser différents paramètres sur différents problèmes) ;
5. illustrer graphiquement les tournées identifiées et exprimer l'erreur relative avec une tournée optimale[1] pour chacun des deux algorithmes ;
6. je dois pouvoir reproduire vos résultats en passant une instance du TSP en argument à un programme principal.

---

1. comopt.ifi.uni-heidelberg.de/software/TSPLIB95/STSP.html