École polytechnique | Majeure informatique
Promotion 2000

Cours de conception et analyse d'algorithmes
PC 6 – algorithmes d'approximation

1. Problème du voyageur de commerce

Ce problème consiste en la donnée d'un graphe complet non-orienté à n sommets numérotés de 1 à n (et donc $n(n-1)/2$ arêtes) ; à chaque arête est associée une distance. Le voyageur de commerce réside au sommet 1 du graphe, et cherche à organiser sa tournée de façon à partir de 1, revenir en 1, passer une fois et une seule par chacun des autres sommets du graphe, et ce en parcourant la distance minimale. La version “décision" de ce problème est, en toute généralité, un problème NP-complet.

A. On suppose dans cette partie que la distance vérifie l'inégalité triangulaire : pour tous sommets u, v et w, $d(u,v) \leqslant d(u,w) + d(w,v)$. On considère l'algorithme suivant :
- calculer un arbre couvrant minimal T enraciné en 1.
- renvoyer le trajet obtenu en regardant l'ordre dans lequel les sommets apparaissent lors d'un parcours en profondeur de l'arbre T.

Soit H la tournée ainsi obtenue.

i. Quelle est la complexité de cet algorithme ?

ii. Soit H^* une tournée optimale. Comparer la longueur totale de H^* et le poids d'un arbre couvrant minimal (on pourra considérer la tournée à laquelle on enlève une arête).

iii. Le *parcours complet* de l'arbre couvrant minimal liste les sommets quand ils sont visités une première fois lors d'un parcours en profondeur, mais également lorsqu'ils sont à nouveau traversés après la visite d'un sous-arbre. Soit W un itinéraire déduit d'un parcours complet de l'arbre T. Montrer que $c(W) = 2c(T)$.

iv. Montrer que $c(W) \geqslant c(H)$.

v. Cet algorithme est-il ε-approché pour $\varepsilon < 1$?

B. Dans cette partie, on étudie à nouveau des stratégies exactes pour résoudre le problème du voyageur de commerce. Dans cette question, le graphe considéré n'est plus symétrique, à savoir que l'on dissocie l'arête (i,j) de l'arête (j,i) (mais le graphe reste complet).

a. À chaque arête (i,j), on associe une variable x_{ij} à valeurs 0 ou 1. On note d_{ij} la distance associée à cette arête. Traduire par des contraintes à satisfaire par les x_{ij} le fait que dans un parcours valide chaque nœud a exactement une arête entrante et une arête sortante. En déduire une formulation d'une condition nécessaire simple pour le problème du voyageur de commerce sous forme de programme linéaire en nombres entiers.

b. On ajoute au problème n variables réelles u_i de signe quelconque, et, pour tout couple (i,j) tel que $2 \leqslant i \neq j \leqslant n$, la contrainte $u_i - u_j + nx_{ij} \leqslant n-1$.

i. Soit $\hat{x}_{ij}$ une solution. On considère l'ensemble des arêtes telles que $\hat{x}_{ij} = 1$. Montrer qu'il ne peut y avoir dans cet ensemble d'arêtes un cycle qui évite 1. En déduire que toute solution est une tournée.

ii. Montrer qu'à toute tournée correspond une solution de l'ensemble de contraintes de **a.** et **b.** On pourra considérer les valeurs $u_1 = 0$, $u_i = t$ si le sommet i est le t-ième sommet visité par la tournée.

c. Donner un programme linéaire mixte entier/réel résolvant le problème du voyageur de commerce. Quelle technique peut-on appliquer à ce programme linéaire ?

d. On appelle 1-arbre un graphe de sommets $\{1, 2, \ldots, n\}$ constitué d'un arbre construit sur les sommets $\{2, \ldots, n\}$ et de deux arêtes incidentes sur le sommet 1. En particulier, toute tournée est un 1-arbre.

i. Donner un algorithme polynomial pour trouver un 1-arbre de poids minimal dans un graphe non-orienté quelconque à n sommets.

ii. Exploiter **i.** dans une stratégie de type *branch-and-bound*.

2. Stable de taille maximale

Dans un graphe symétrique $G = (X, E)$, un ensemble *stable* est un sous-ensemble Y de X tel qu'aucune des arêtes de G n'a ses deux extrémités dans Y. MAXSTABLE est le problème consistant à trouver un ensemble stable de cardinal maximum dans un graphe symétrique. On se propose d'examiner les algorithmes approchés de MAXSTABLE.

i. À partir d'un graphe $G = (X, E)$, on construit le graphe $G^{(2)}$ dont l'ensemble des sommets est $X \times X$ et dont les arêtes sont les paires $\{(x, x'), (y, y')\}$ telles que soit $x = y$ et $(x', y') \in E$, soit $(x, y) \in E$. Montrer que, si G admet un ensemble stable de taille k, alors $G^{(2)}$ admet un ensemble stable de taille k^2. Inversement, soit m le cardinal d'un stable de $G^{(2)}$, montrer qu'il existe un stable de G de cardinal supérieur à $\sqrt{m}$.

ii. Montrer que, de tout algorithme ε-approché de MAXSTABLE, on peut déduire un algorithme ε'-approché, où

$$\varepsilon' = 1 - \sqrt{1 - \varepsilon}.$$

iii. En déduire que s'il existe un algorithme ε-approché de MAXSTABLE pour un $\varepsilon < 1$, alors le seuil de ce problème est égal à 0.

3. Répartition de tâches

On se propose dans cet exercice d'étudier un algorithme d'approximation du problème de répartition de tâches suivant. On se donne k machines de performances identiques, et n tâches $t_1, \ldots, t_n$; on attribue à chaque tâche t_i un poids w_i. On suppose que les tâches ne peuvent être effectuées que par groupes $G_1, \ldots, G_\ell$ (non disjoints a priori), et l'on souhaite trouver l'ensemble de k groupes de tâches $\{G_{i_1}, \ldots, G_{i_k}\}$ qui maximise

$$\sum_{t_j \in G_{i_1} \cup \ldots \cup G_{i_k}} w_j.$$

i. Étudier le cas où $k = 2$, $n = 4$, $w_i = i$, $G_1 = \{t_3, t_4\}$, $G_2 = \{t_2, t_4\}$, $G_3 = \{t_1, t_3\}$.

ii. Décrire un algorithme de type glouton pour ce problème. Cet algorithme est-il optimal ?

iii. On cherche à estimer la performance de cet algorithme par rapport à la solution optimale. On note W_h le poids de la réunion des sous-ensembles sélectionnés à l'étape h. Montrer que

$$kW_h \geqslant (k-1)W_{h-1} + W_{\text{opt}}.$$

iv. Déduire de la question précédente que $W_k \geqslant \left(1 - \frac{1}{e}\right) W_{\text{opt}}$.