École polytechnique — Majeure informatique

Promotion 2001

**Cours de conception et analyse d'algorithmes**

PC 9 – revisions

**1. $\mathcal{NP}$ complétude** Montrer que le problème suivant est NP-complet.

**Entrée :** un graphe $G = (V, E)$ et un entier positif $k$.
**Sortie :** décider s'il existe deux sous-ensembles disjoints $S_1$ et $S_2$ de $V$ tels que $|S_1| = |S_2| = k$ et qui soient des stables de $G$, c'est-à-dire sans aucune arête entre deux sommets de $S_1$ ni entre deux sommets de $S_2$.

**2. Flots**

Construire un graphe orienté avec 4 sommets, tel que chaque sommet ait au plus 2 arêtes en parallèle le menant vers le même autre sommet, et que les degrés entrants et sortants des sommets soient : $(5, 4), (3, 3), (1, 2)$ et $(2, 2)$. Généraliser votre approche en un algorithme.

**3. Programmation dynamique.**

Soit $T = (V, E)$ un arbre dont les arêtes ont des poids $w : E \to \mathbf{R}^+$. Soit $\operatorname{dist}(u, v)$ la distance entre $u$ et $v$ dans l'arbre (somme des poinds des arêtes du chemin). Un $k$-clustering optimal est un ensemble $C$ de $k$ sommets de $V$ qui minimise

$$\sum_{u \in V} \min_{v \in C} \operatorname{dist}(u, v).$$

Concevoir un algorithme pour calculer un $k$-clustering optimal dans un arbre.

**4. Plus courts chemins.**

Soient $n$ nombres réels $a_1 \geq a_2 \geq \cdots \geq a_n$. On souhaite les partitionner en groupes de façon que : (1) chaque groupe contienne au moins $p$ nombres ; (2) chaque groupe soit composé de nombres consécutifs dans la liste ; et (3) la somme des carrés de la déviation des nombre à la moyenne de leur groupe soit minimale : soit $\bar{a}(S) = \sum_{i \in S} a_i / |S|$ la moyenne du groupe $S$ ; alors cette somme, pour $S$, est $\sum_{i \in S} (a_i - \bar{a}(S))^2$. Montrer comment modéliser ce problème par un problème de plus court chemin. Illustrer pour la donnée suivante : $p = 2$, $n = 6$, $(a_i) = .5, .8, 1.1, 1.5, 1.6, 2$.

**5. Programmation linéaire.**

Soit $H$ une matrice à entrées dans $\{0, 1\}$ et à $n_1$ lignes et $n_2$ colonnes. On appelle *vecteur* de $H$ une ligne ou une colonne de $H$. On dit qu'un ensemble d'entrées 1 est un *indépendant* s'il n'y en a pas deux qui apparaissent sur le même vecteur. On dit qu'un ensemble de vecteurs de la matrice est une *couverture* de $H$ si ils contiennent tous les 1 de $H$. Par le théorème max-flot-min-cut, montrer que le nombre maximum de 1 indépendants égale le nombre minimum de vecteurs dans une couverture.

**5. Programmation linéaire et approximation.**

Le problème du couplage équilibré est le suivant. L'entrée est un graphe biparti $G = (V \cup W, E)$. Il s'agit de choisir une arête incidente à chaque sommet $v \in V$ de façon à minimiser la *charge* (nombre d'arêtes incidentes) maximale des sommets de $W$.

Interprétation : $V$ est un ensemble de tâches, $W$ est un ensemble de personnes, les arêtes nous disent qui est capable de faire quoi, et il faut décider qui fait quoi pour que toutes les tâches soient faites et que la charge de travail maximal soit minimisée.

1. Modéliser le problème comme programme linéaire en entiers.
2. Ecrire sa relaxation linéaire.
3. Soit $(x^*)$ la solution optimale du programme linéaire, de valeur $\Delta^*$. Proposer une façon d'en déduire une solution faisable du problème initial par arrondi randomisé.
4. Soit $(\hat{x})$ la solution obtenue. Soit $w \in W$. Calculer la charge moyenne de $w$.
5. En appliquant une borne de type Chernov, montrer que la probabilité que la charge de $w$ excède sa moyenne d'un montant de $E$ est au plus $1/2|W|$, pour $E$ bien choisi du genre : $\max\{3\ln(2m), \sqrt{3\ln(2m)\Delta^*}\}$, mais vous n'êtes pas obligés d'être aussi précis).
6. En déduire que cet algorithme donne, avec probabilité au moins $1/2$, une solution qui excède l'optimal par au plus $E$.