École polytechnique
Promotion 2000

Majeure 2 informatique

Cours de conception et analyse d'algorithmes

PC 5 – problèmes NP-complets

1. $\mathcal{NP}$ complétude de MAX2SAT

Etant donné un ensemble Φ de m clauses sur les variables booléennes $x_1, \ldots, x_n$, tel que chaque clause contient au plus deux littéraux, et étant donné un entier k, le problème MAX2SAT consiste à décider s'il existe une assignation des variables qui satisfasse au moins k clauses. On se propose de montrer que ce dernier est NP-complet.

i. Montrer que MAX2SAT est dans $\mathcal{NP}$.

ii. Soit une clause à trois littéraux $C = x \vee y \vee z$; vérifier que dans l'ensemble de 10 clauses donné ci-dessous (faisant intervenir une variable supplémentaire t), au plus 7 sont satisfaites, et que ce maximum n'est atteint que si x, y ou z sont vrais :

$$x,\ y,\ z,\ t,\ \overline{x} \vee \overline{y},\ \overline{y} \vee \overline{z},\ \overline{z} \vee \overline{x},\ x \vee \overline{t},\ y \vee \overline{t},\ z \vee \overline{t}.$$

ii. En déduire une réduction de 3SAT à MAX2SAT. Conclure.

2. $\mathcal{NP}$ complétude de HITTING SET

Etant donné un ensemble S de cardinal n est une collection C de sous-ensembles de S, et étant donné un entier k, le problème est de décider s'il existe un sous-ensemble S' de S de taille au plus k et qui contienne au moins un élément de chaque ensemble de C. Montrer, par réduction à partir de COUVRANT-MIN, que ce problème est $\mathcal{NP}$ complet.

3. $\mathcal{NP}$ complétude de l'isomorphisme de sous-graphes

Etant donné deux graphes G et H, le problème est de décider s'il existe une injection ϕ de V_H dans V_G telle que $\{u, v\} \in E_H$ ssi $\{\phi(u), \phi(v)\} \in E_G$. Montrer que ce problème est $\mathcal{NP}$ complet par réduction à partir de STABLE.

4. $\mathcal{NP}$ complétude de ORD-TACHES

On s'intéresse au problème suivant qui modélise l'ordonnancement de tâches sur un multiprocesseur : soit $\Gamma = (S, A)$, $A \subset S \times S$ un graphe orienté sans circuit, où les sommets représentent les tâches et l'arc (x, y) signifie que la tâche x doit être exécutée avant la tâche y. Une *décomposition compatible* de Γ est une partition de l'ensemble des sommets $S = S_1 \cup S_2 \cup \ldots \cup S_t$ (où S_i représente l'ensemble des tâches exécutée en parallèle à l'instant i), telle que si $x \in S_i$, $y \in S_j$, et $(x, y) \in A$, alors nécessairement $i < j$. La *profondeur* de la décomposition (nombre total d'étapes) est t, et sa *largeur* (nombre de processeurs nécessaire) est $l = \max \# S_t$.

Etant donné $\Gamma = (S, A)$, t et l, le problème ORD-TACHES consiste à décider s'il existe une décomposition de Γ de profondeur au plus t et de largeur au plus l.

On utilisera une réduction de CLIQUE à ORD-TACHES. Soit $G = (X, E)$ un graphe non orienté à n sommets et m arêtes ; on construit un graphe orienté Γ comme suit. Les sommets sont l'ensemble

$$S = X \cup E \cup Y \cup Z \cup T,$$

où Y, Z, T sont des ensembles finis quelconques de cardinaux respectifs p, q, r qui seront déterminés par la suite. Les arcs de Γ sont donnés par

$$A = (Y \times Z) \cup (Z \times T) \cup \{(x, e) \in X \times E, x \in e\}.$$

i. Exprimer le nombre d'éléments de A en fonction de n, m, p, q, r. Montrer qu'il existe une décomposition compatible de Γ de profondeur 3.

ii. On suppose que G n'a pas de sommet isolé. Soit $S = S_1 \cup S_2 \cup S_3$ une décomposition compatible de Γ. Montrer que $S_1 \cap E = \emptyset$ et que $S_3 \cap X = \emptyset$. Soit k le cardinal de $S_1 \cap X$. Montrer que

$$\#(S_2 \cap E) \leqslant k(k-1)/2.$$

Que dire de $S_1 \cap X$ dans les cas d'égalité ?

iii. Pour un entier k tel que $k \leqslant n$ et $k(k-1)/2 \leqslant m$, montrer que l'on peut choisir p, q, r, l de façon que G admette une clique de taille k si et seulement si Γ admet une décomposition compatible en trois sous-ensembles de même taille l. En déduire que ORD-TACHES est NP-complet.

5. CIRCUIT-SAT et SAT

Un circuit booléen est un graphe orienté sans circuit dont chaque sommet s est étiqueté soit par un littéral (si s n'a pas d'arc entrant), soit par une constante 0 ou 1 (si s n'a pas d'arc entrant), soit par l'un des symboles $\vee$ ou $\wedge$ (si s a exactement deux arcs entrants), soit par $\neg$ (si s a un unique arc entrant). Il est clair qu'à tout circuit correspond une formule booléenne et que, réciproquement, on peut associer un circuit à toute fonction booléenne. Le problème CIRCUIT-SAT consiste à déterminer s'il existe une affectation des variables qui donne la valeur vraie à la fonction booléenne associée au circuit.

La plupart des livres prennent CIRCUIT-SAT (au lieu de SAT) comme premier problème NP-complet. On se propose de montrer que le problème CIRCUIT-SAT se réduit à SAT. Pour cela, on associe à tout circuit un ensemble de clauses faisant intervenir les variables $x_1, \ldots, x_n$ qui étiquettent des sommets, ainsi qu'une variable y_S pour chaque sommet s du graphe.

i. Écrire des ensembles u de clauses faisant intervenir des variables x, y et z vérifiant respectivement :

- les clauses de u sont satisfaites si et seulement si $x = y$
- les clauses de u sont satisfaites si et seulement si $x = \bar{y}$
- les clauses de u sont satisfaites si et seulement si $x = y \vee z$
- les clauses de u sont satisfaites si et seulement si $x = y \wedge z$

ii. Montrer que CIRCUIT-SAT se réduit à SAT.

6. $\mathcal{NP}$ complétude du BIN PACKING

On se donne un ensemble de n objets de tailles respectives s_i, avec $0 < s_i < 1$, et un entier k. Le problème du BIN PACKING consiste à décider s'il est possible de ranger tous les objets dans au plus k boîtes de taille 1.

Montrer, par une réduction à partir de SUBSET-SUM, que ce problème est NP-complet.