Majeure Informatique

Analyse et Conception d'Algorithmes

Contrôle classant

Mercredi 20 Mars 2002

Documents autorisés : polycopiés et notes de cours

1 Le couplage maximal en $O(n^{5/2})$

On considère dans ce problème un graphe biparti ayant pour ensemble de sommets $X = Y \cup Z$, tel que Y et Z sont disjoints, et pour ensemble d'arêtes E ; chaque arête e de E a une extremité dans Y et l'autre dans Z.

On rappelle qu'un *couplage* est un sous-ensemble C de E tel que deux arêtes de C n'aient pas d'extremité commune.

Dans la suite on appelle *chemin d'amélioration* du couplage C une suite d'arêtes $e_0, e_1, \ldots, e_{2k-1}, e_{2k}$ formant un chemin (de longuer $2k+1$) tel que :

- $e_i \in C$ si et seulement si i est impair.

- les arêtes e_0 et e_{2k} sont incidentes à un sommet qui n'est extremité d' aucune arête de C.

Dans la suite on note $U \oplus V$ la différence symétrique des deux ensembles U et V, il s'agit de l'ensemble formé des éléments de U et de ceux de V qui ne sont pas dans l'intersection $U \cap V$.

1. Soit C un couplage et P un chemin d'amélioration de C montrer que $C \oplus P$ est aussi un couplage. Quelle est la relation entre les nombres d'éléments $|C|$ et $|C \oplus P|$ de C et $C \oplus P$?

2. Montrer que si C et C' sont deux couplages tels que $|C'| = |C| + q$, il existe alors q chemins d'amélioration $P_1, \ldots, P_q$ tels que

$$C' = C \oplus P_1 \oplus \ldots \oplus P_q$$

3. Soit s le nombre d'arêtes d'un couplage maximal et soit C un couplage ayant $r < s$ arêtes, montrer quíl existe un chemin d'amélioration de C de longueur inférieure ou égale à

$$2\lfloor\frac{r}{s-r}\rfloor + 1$$

Dans la suite on s'interesse à l'algorithme de calcul du couplage maximal qui construit itérativement un couplage à partir de l'ensemble vide, en recherchant à chaque étape un chemin d'amélioration de longueur minimale. Soit de manière plus précise :

$C_0 = \emptyset; i = 0$

Tant qu'il existe un chemin d'amélioration pour C_i faire

– P_i = un chemin d'amélioration de longueur minimale pour C_i

– $C_{i+1} = C_i \oplus P_i$

– $i = i + 1;$

On notera qu'à chaque étape on a $|C_i| = i$.

4. Donner une évaluation de la complexité de l'algorithme ci-dessu en fonction de n, m le nombre de sommets et d'arêtes du graphe G et s le nombre d'arêtes du couplage maximum.

5. On note $r = s - \lceil\sqrt{s}\rceil$ montrer que la longeur de P_r est inférieure ou égale à $2\lceil\sqrt{s}\rceil + 1$. En déduire que les longueurs des P_i, qui sont toutes des entiers impairs, ne peuvent pas prendre plus de $2\lceil\sqrt{s}\rceil + 2$ valeurs différentes.

6. Montrer que si P est un chemin d'amélioration de longueur minimale de C et P' un chemin d'amélioration de $C \oplus P$ alors

$$|P'| \geq |P| + |P \cap P'|$$

On pourra utiliser le fait que $P \oplus P' \oplus C = C \oplus P_1 \oplus P_2$ pour deux chemins d'amélioration P_1 et P_2 de C.

7. En déduire que les longueurs $\ell(P_i)$ des chemins P_i construits par l'algorithme satisfont :

$$\ell(P_1) \leq \ell(P_2) \leq \ldots \leq \ell(P_s)$$

8. En regroupant certaines étapes de l'algorithme ci-dessus proposez un algorithme de complexité $O(n^{3/2})$ pour le calcul du couplage maximal dans un graphe biparti.

2 Ordonnancement

On considère dans cet exercice des tâches $T_1, T_2, \ldots, T_n$ à réaliser sur p processeurs identiques séquentiel. Chaque tâche T_i, a une durée d'exécution d_i. On considère le problème P-AFFECTE il s'agit d'affecter les tâches aux processeurs de façon que la durée maximale d'utilisation de ceux-ci soit la plus petite possible.

1. On dispose de 3 processeurs et on doit exécuter des tâches de durées $4, 3, 2, 7, 2, 1, 6, 3, 5, 6, 3$ donner l'afectation optimale qui utilise chacune des machines 14 unités de temps.

2. Montrer à l'aide d'une réduction à un problème étudié en cours que pour chaque valeur de $p > 1$ P-AFFECTE est NP-complet. On précisera en détail les constructions effectuées et on justifiera toutes les étapes avec soin.

On considère l'algorithme de type on-line glouton qui consiste à considérer toutes les tâches dans un ordre quelconque et à affecter chaque tâche T_i au processeur la moins chargé au moment où l'on considère celle-ci.

Ainsi sur l'exemple précédent on obtient les affectations suivantes de durées pour les différents processeurs :

$$\{4, 1, 6, 3\}, \{3, 2, 3, 5\}, \{2, 7, 6\}$$

3. On note u_i la somme des durées des tâches affectées par l'algorithme glouton au processur i, h un processeur tel que u_h est maximale et j la dernière tâche affectée au processeur h.

Donner une relation entre u_i et $u_h - d_j$.

4. En déduire que l'algorithme glouton est $2 - 1/p$ approché.

5. Donner un exemple ayant $p(p-1)+1$ tâches pour p processeurs, pour lequel la solution optimale est de durée maxiamle p et l'algorithme glouton donne une durée de $2p-1$ pour l'un des processeurs.

6. L'algorithme qui effectue la même itération que l'algorithme glouton mais qui commence par trier les tâches dans l'ordre croissant est-il meilleur ?