École polytechnique
Promotion 2008

Majeure informatique

Cours *Conception et analyse d'algorithmes*

TD 8 – Complexité paramétrique

10 novembre 2010

1. Problème du placement d'usines

Une entreprise souhaite installer des usines (en nombre n) dans un nouveau territoire, elle a pour cela acquis n terrains et, après une étude minutieuse, on a pu déterminer pour chaque couple d'usines U_i, U_j le nombre $v_{i,j}$ de véhicules qui devront les relier chaque mois et pour chaque couple d'emplacements E_p, E_q le coût $c_{p,q}$ du voyage d'un véhicule entre ces emplacements. Il s'agit alors de déterminer l'affectation optimale des usines sur les emplacements.

i. On souhaite résoudre le problème par une technique de *branch and bound*, pour cela chaque sommet de l'arbre d'exploration représentera une situation où certaines usines auront été affectées en des emplacements et les autres ne seront pas encore affectées.

Donner une borne inférieure du coût de la meilleure affectation possible à partir d'une situation donnée. Pour cela, il faudra diviser les voyages dont on détermine le coût en trois catégories : ceux entre usines déjà affectées, ceux entre chaque usine affectée et les usines non-affectées et ceux entre les usines non-affectées. On rappelle que le produit scalaire de $a_1, \ldots, a_m$ par une permutation de $b_1, \ldots, b_m$ est minimal lorsque l'on trie ces deux vecteurs et que l'on multiplie le plus grand a_i par le plus petit b_j et ainsi de suite.

ii. Quel est la complexité du calcul de la borne inférieure en fonction du nombre p d'usines affectées et $q = n - p$ d'usines non affectées.

iii. On suppose qu'il y a 4 usines A, B, C, D et 4 emplacements a, b, c, d et que le nombre d'échanges de véhicules et leurs coûts sont donnés par les tableaux ci-dessous. Évaluer le coût de l'affectation $\pi : A \mapsto a,\ B \mapsto b,\ C \mapsto c,\ D \mapsto d$ et en déduire que le sous-arbre pour lequel A est affecté à d ne sera pas exploré.

Échanges	A	B	C	D
A	0	2	2	3
B	2	0	2	1
C	2	2	0	3
D	3	1	3	0

Coûts	a	b	c	d
a	0	2	2	1
b	2	0	1	3
c	2	1	0	2
d	1	3	2	0

iv. Pour l'exemple ci-dessus dessiner l'arbre d'exploration par la méthode *branch and bound* en faisant apparaître uniquement les sous-arbres explorés.

2. Couverture de points par des lignes

On considère le problème suivant dont on peut montrer qu'il est NP-complet :

Instance : Un ensemble de points p du plan et un entier k.

Paramètre : k

Question : Est-ce que P peut être couvert par k droites ?

Montrer que ce problème a un noyau de taille k^2 et qu'il est donc FPT en k.

3. Ensemble indépendant pour les graphes planaires

Dans le cas général, ENSEMBLE INDÉPENDANT n'est pas connu pour être FPT en k. On veut montrer dans cet exercice que lorsque l'on se restreint aux graphes planaires, on peut résoudre ENSEMBLE INDÉPENDANT en $\mathcal{O}(6^k n)$.

i. En utilisant la formule d'Euler, montrer qu'un graphe planaire à n sommets possède au plus $3n - 6$ arêtes. On rappelle que la formule d'Euler relie le nombre d'arêtes, de faces et de sommets d'un graphe planaire dessiné dans le plan de la manière suivante :

$$\# \text{ sommets} + \# \text{ arêtes} - \# \text{ faces} = 2$$

ii. Déduire de la question précédente que tout graphe planaire a n sommets possède un sommet de degré au plus 5.

iii. En déduire que ENSEMBLE INDÉPENDANT pour les graphes planaires peut être résolu en temps $\mathcal{O}(6^k n)$.

4. Réduction à un noyau de couverture par sommets

On souhaite monter que COUVERTURE PAR SOMMETS de taille k pour les graphes à n sommets peut être résolu en temps $\mathcal{O}(kn + 5^{k/4} \cdot k^2)$.

i. Montrer que COUVERTURE PAR SOMMETS de taille k possède un noyau de taille $\mathcal{O}(k^2)$, en s'appuyant sur la remarque suivante : si un graphe a un sommet de degré $> k$ alors ce sommet appartient à toute couverture de taille k sur ce graphe.

ii. On note $\Delta(G)$ le degré maximum d'un sommet de G. Montrer que si $\Delta(G) \leq 2$, alors une couverture par sommets de taille optimale peut être facilement calculée.

iii. Déduire de **ii.** que COUVERTURE PAR SOMMETS de taille k pour les graphes à n sommets et m arêtes peut être résolu en temps $\mathcal{O}(5^{k/4}(n + m))$.

iv. Conclure en appliquant l'algorithme obtenu en **iii.** au noyau obtenu en **i.**.