CORRIGÉ DE LA PC9

G. HANROT

1. Appellons 2-STABLE le problème considéré. On va montrer sa NP-complétude par réduction de STABLE. Montrons tout d'abord qu'il est dans NP : un certificat est constitué de la donnée des deux stables, tous deux de cardinal k ; on vérifie qu'il s'agit bien d'une solution en vérifiant que les deux ensembles sont disjoints (coût naïf $O(k^2)$), et que ce sont bien des stables (coût naïf $O(k^2n^2)$, en énumérant les arêtes possibles, et pour chacune d'entre elles en parcourant l'ensemble de toutes les arêtes du graphe pour l'y chercher). Le probleme 2-STABLE est donc bien dans NP.

Considérons maintenant une instance $(G = (S, A), k)$ de STABLE. On construit alors deux copies disjointes du graphe, $G_1 = (S_1, A_1)$ et $G_2 = (S_2, A_2)$, et un graphe $G' = (S_1 \cup S_2, A_1 \cup A_2 \cup \{\{x, y\}, x \in S_1, y \in S_2\})$.

Si G admet un stable U de taille k, il correspond à U des sous-ensembles U_1 de S_1 et U_2 de S_2, qui sont des stables de G_1 et de G_2. Comme G_1 et G_2 sont disjoints et qu'on n'ajoute aucune arête entre éléments de G_1 ou entre éléments de G_2 dans G', il s'ensuit que U_1 et U_2 sont deux stables disjoints de G' de cardinal k.

Inversement, supposons que G' admet deux stables U_1 et U_2 de cardinal k. Supposons par exemple que $U_1 \cap S_1 \neq \emptyset$, et soit x_1 dans cet ensemble. Si $x_2 \in U_1 \cap S_2$, par construction de G', il existe une arête $\{x_1, x_2\}$ dans G', ce qui contredit le fait que U_1 est un stable. Par suite, $U_1 \cap S_2 = \emptyset$, et $U_1 \subset S_1$. Mais alors on peut construire l'ensemble U correspondant dans S, qui, par construction de G', sera un stable de G de cardinal k.

On a donc bien montré que STABLE(G, k) $\Leftrightarrow$ 2-STABLE(G', k), et donc on a réduit 2-STABLE à STABLE. Il s'ensuit que 2-STABLE est NP-complet.

2. L'énoncé impose de traduire des contraintes différentes sur le flot entrant et sortant dans un sommet donné ; cela impose, en pratique, de dupliquer chaque sommet dans la construction du réseau de transport. À chaque sommet u on associe donc deux sommets u_1 (qui modélisera les arêtes entrantes dans u) et u_2 (qui modélisera les arêtes sortantes de u). Pour chaque couple (u, v) de sommets, on relie donc u_2 à v_1 par une arête de capacité 2. On ajoute en outre une source s et un puits t ; s est relié à u_2 par une arête de capacité le degré sortant de u, t est relié à u_1 par une arête de capacité le degré entrant dans u ; ces arêtes permettent de traduire les contraintes sur les degrés entrants et sortants.

L'idée de cette construction est que le degré entrant et sortant sont deux quantités sans lien (il n'y a pas de conservation, contrairement au flot), et qu'il faut donc complètement décorréler les deux parties, en construisant deux sommets non reliés entre eux. Les liens depuis la source et vers le puits fournissent le flot qui circulera, et assurent les limitations sur les degrés entrants et sortants.

Date: 30 mars 2004.

Le flot maximal est majoré par la coupe qui isole s, qui vaut la somme des degrés sortants, et par la coupe qui isole t, qui vaut la somme des degrés entrants. Inversement, si une solution au problème initial existe, il existe un flot dans lequel toutes les arêtes sortant de s sont saturées, et donc de valeur la somme des degrés sortants. Par suite, le flot maximal fournit, si elle existe, une solution au problème de départ. Notons que toutes les capacités sont entières, et donc que le flot l'est également.

3. On va utiliser une technique de programmation dynamique ; pour ce faire, il faut relier un k-clustering optimal pour T, que l'on enracine en un sommet v quelconque, aux l-clusterings optimaux des sous-arbres enracinés en les fils de v.

Un k-cluster C de T définit des l_i-clusters de chacun des sous-arbres T_i avec $\sum_i l_i = k-1$ si $v \in C$, $\sum_i l_i = k$ sinon. Dans chacun de ces sous-arbres on prend c_i le point du cluster le plus proche de v.

4. Considérons le graphe G dont les sommets sont l'ensemble des parties $P_{i,j} = \{a_k, k \in [i,j]\}, |j-i| \geq p-1$. Le nombre de telles parties est $\sum_{i=1}^{n-p+1} n-(i+p-1) = O(n^2)$, donc polynômial. On ajoute un sommet de départ $P_{0,0}$ et un sommet d'arrivée $P_{n+1,n+1}$. On relie tous les sommets $P_{u,v}$ et $P_{u',v'}$ dès que $u' = v+1$, par un lien de valeur $\sum_{i\in P_{u',v'}} (a_i - \overline{a}(P_{u',v'}))^2$, étant donné que les liens vers $P_{n+1,n+1}$ ont valeur 0.

La donnée d'un chemin le plus court dans ce graphe entre $P_{0,0}$ et $P_{n+1,n+1}$ correspond alors à la donnée d'une partition (c'est assuré par le choix des arêtes du graphe) en parties d'au moins p éléments consécutifs (assuré par la construction des $P_{i,j}$, de sorte que la somme des $\sum_{i\in P_{u',v'}} (a_i - \overline{a}(P_{u',v'}))^2$ soit minimale, donc une solution au problème de départ.

5. On considère le graphe biparti dont les sommets sont les vecteurs lignes et colonnes de la matrices, et une arête relie un vecteur ligne à un vecteur colonne s'il se trouve un 1 à leur intersection.

On ajoute en outre une source reliés à toutes les lignes et un puits auquel sont reliées toutes les colonnes ; on donne capacité 1 à toutes les arêtes.

Un flot entier dans ce réseau est la donnée d'un ensemble d'arêtes saturées ; comme en outre le flot entrant dans chaque sommet-ligne et le flot sortant de chaque sommet colonne est au plus 1, on voit qu'au plus une arête sortant de chaque ligne et au plus une arête entrant dans chaque colonne est saturée ; un flot correspond donc à un choix d'entrées 1 de la matrice de sorte qu'il n'y ait pas deux 1 dans le même vecteur, donc un ensemble indépendant, et la valeur du flot est égale au cardinal de l'ensemble indépendant.

Une couverture de H est la donnée d'un ensemble L_1 de lignes et C_1 de colonnes ; notons L_2 et C_2 l'ensemble des lignes et des colonnes qui ne sont pas dans la couverture. Le fait que ce soit une couverture indique qu'aucune arête ne va de L_2 à C_2. Considérons alors la coupe $\{s\} \cup L_2 \cup C_1, L_1 \cup C_2 \cup \{t\}$. La valeur de cette coupe est $|L_1| + |C_1|$, soit le cardinal de la couverture.

Inversement, on se donne une coupe de la forme $\{s\} \cup L_2 \cup C_1, L_1 \cup C_2 \cup \{t\}$. La valeur de cette coupe est $|L_1| + |C_1| + [L_2, C_2]$, où ce dernier terme désigne le nombre d'arêtes de L_2 vers C_2. S'il existe une arête (l, c) de L_2 vers C_2, faire passer l dans L_1 augmente la valeur de la coupe de 1 (l'arête (s, l)) et la diminue de $[\{l\}, C_2] \geq 1$, donc ne fait pas croître la valeur de la coupe. Il en résulte qu'on peut supposer que dans une coupe minimale

$\{s\} \cup L_2 \cup C_1, L_1 \cup C_2 \cup \{t\}$ il n'y a pas d'arête entre L_2 et C_2 ; dans ce cas $L_1 \cup C_1$ est une couverture. Par suite, coupe minimale et couverture minimale ont même valeur. Comme la première partie a montré que flot maximal et nombre maximum de 1 indépendants coïncident, il s'ensuit, par le théorème mincut-maxflow, le résultat demandé.

6. 1. On associe une variable entière x_a à chaque arête ; cette variable vaut 0 ou 1 selon que a est choisie ou non. La contrainte en chaque sommet v s'écrit alors $\sum_{a=(v,w)\in G} x_a = 1$, où la somme est prise sur les arêtes issues de s. On cherche alors à minimiser $\max_{w\in W} \sum_{a=(v,w)} x_a$. Pour ce faire, on introduit une variable supplémentaire m, que l'on va minimiser, avec les contraintes $\sum_{a=(v,w)} x_a \leq m$ pour tout w.

Noter que dans la solution optimale, une de ces contraintes est exacte, sans quoi on peut diminuer la valeur de m, et donc qu'on a bien $m = \max_{w\in W} \sum_{a=(v,w)} x_a$. On a donc

$$\begin{cases} \min m \\ \sum_{a=(v,w)} x_a \leq m, \quad \forall w \in W \\ \sum_{a=(v,w)} x_a = 1 \quad \forall v \in V \\ x_a \in \{0,1\} \quad \forall a \end{cases}$$

2. La relaxation linéaire consiste à remplacer la contrainte $x_a \in \{0,1\}$ par $0 \leq x_a \leq 1$.

3. Pour que la solution soit faisable, il faut choisir pour chaque v un des $x_{(v,w)}$ à mettre à 1. Soit v un sommet fixé ; on définit alors une variable aléatoire A_v à valeurs dans $\{(v,w)\}$ par $\Pr(A_v = (v,w)) = x_{(v,w)}$. Cela consiste à choisir une arête issue de v au hasard, la probabilité de chaque arête étant liée à la valeur x_a produite dans le programme linéaire.

4. Les ensembles $\{(v,w)\}$ étant disjoints, on trouve donc que la probabilité qu'une arête (v,w) donnée soit choisie est $x_{(v,w)}$; notons X_a la variable aléatoire qui vaut 1 si a est choisie, 0 sinon. La charge moyenne de w est $E[\sum_{a=(v,w)} X_a] = \sum_{a=(v,w)} E[x_a] = \sum_{a=(v,w)} x_a \leq \Delta^*$, avec égalité pour un certain $w \in W$, cf. la remarque concluant le 1.

5. On peut appliquer la borne de Chernov, puisque la variable aléatoire $\sum_{a=(v,w)} x_a$ est une somme de variables aléatoires indépendantes, les a étant issues de sommets v distincts. La borne de Chernov donne

$$\Pr\left(\sum_{a=(v,w)} X_a > (1+\delta) \sum_{a=(v,w)} x_a\right) \leq \exp\left(\sum_{a=(v,w)} x_a \left(\delta - (1+\delta)\log(1+\delta)\right)\right).$$

On montre facilement que $(\delta - (1+\delta)\log(1+\delta))/\delta^2$ croît sur $[0,1]$, et que son maximum est donc $1 - 2\log 2 \leq -1/3$, d'où la borne – réécrite en terme de $\delta' = \delta \sum_{a=(v,w)} x_a$) :

$$\Pr\left(\sum_{a=(v,w)} X_a > \sum_{a=(v,w)} x_a + \delta'\right) \leq \exp\left(-\delta'^2/(3 \sum_{a=(v,w)} x_a)\right),$$

si $\delta' < \sum_{a=(v,w)} x_a$.

Inversement, $(\delta - (1+\delta)\log(1+\delta))/\delta$ décroît sur $[1,+\infty[$, et est donc $< -1/3$, d'où si $\delta' > \sum_{a=(v,w)} x_a$,

$$\Pr\left(\sum_{a=(v,w)} X_a > \sum_{a=(v,w)} x_a + \delta'\right) \leq \exp(-\delta'/3),$$

Par suite, pour $\delta' = \sqrt{3\log(2|W|)\Delta^*}$, on trouve une borne supérieure de $1/2|W|$, car $\Delta^* \geq \sum_{a=(v,w)} x_a$, dès lors que

$$\sqrt{3\log(2|W|)\Delta^*} < \sum_{a=(v,w)} x_a,$$

en utilisant la première borne.

Dans le cas contraire, la seconde borne donne, pour $\delta' = 3\log(2|W|) \geq 3\log 2 > 1$,

$$\Pr\left(\sum_{a=(v,w)} X_a > \sum_{a=(v,w)} x_a + \delta'\right) \leq 1/(2|W|).$$

D'où le résultat avec $E = \delta'$.

5. Pour obtenir au plus l'optimum plus E, il faut obtenir au plus l'optimum plus E pour tous les sommets. La probabilité d'échec pour un sommet donné est au plus $1/(2|W|)$, et la probabilité d'échec est au plus la somme de ces probabilités, donc au plus $1/2$. L'arrondi randomisé du programme linéaire répond donc à la question avec probabilité au moins $1/2$. On peut, bien sûr, améliorer la probabilité en répétant l'arrondi un certain nombre de fois en en choisissant la solution minimale parmi celles trouvées.

Remarque : Les deux versions simplifiées de la borne de Chernov établies dans cet exercice sont plus facilement exploitables que la borne explicite :

$\Pr(\sum X_i > \mu + \delta) \leq \exp(-\delta^2/3\mu)$ si $\delta < \mu$, $\Pr(\sum X_i > \mu + \delta) \leq \exp(-\delta/3)$ si $\delta > \mu$, avec $\mu = E[\sum X_i]$.

Comme la borne de Chernov donne $\exp(-\delta^2/\mu)$ quand δ tend vers 0 et $\exp(-\delta\log\delta)$ quand δ tend vers $+\infty$, on voit que ces estimations sont d'assez bonne qualité.