

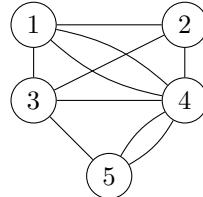
## I Coupe minimum

Soit  $G = (S, A)$  un multigraphe ( $G$  peut avoir plusieurs arêtes entre deux sommets),  $n = |S|$  et  $p = |A|$ .

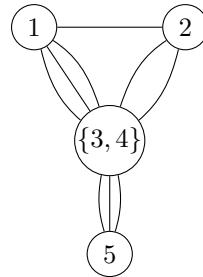
Une coupe de  $G$  est une partition  $C = (S_1, S_2)$  de  $S$  telle que  $S_1 \neq \emptyset$  et  $S_2 \neq \emptyset$ . Sa taille  $|C|$  est le nombre d'arêtes entre  $S_1$  et  $S_2$  :  $|C| = \{ \{u, v\} \in A \mid u \in S_1 \text{ et } v \in S_2 \}$ .  $C$  est une coupe minimum si elle minimise  $|C|$ .

Soient  $u, v \in S$ . La contraction  $G/\{u, v\}$  de  $\{u, v\}$  dans  $G$  est le multigraphe obtenu à partir de  $G$  en fusionnant  $u$  et  $v$  en un nouveau sommet  $uv$ , en supprimant les arêtes entre  $u$  et  $v$ , et en remplaçant chaque arête  $\{u, x\}$  ou  $\{v, x\}$  par une arête  $\{uv, x\}$ .

1. Dessiner  $G/\{3, 4\}$  si  $G$  est le multigraphe suivant :



Solution :



2. On tire aléatoirement et uniformément une coupe  $C$ . Montrer que  $\mathbb{P}(C \text{ est une coupe minimum}) \geq \frac{1}{2^n}$ .

On propose l'algorithme suivant, où on note  $S(G)$  l'ensemble des sommets et  $A(G)$  l'ensemble des arêtes d'un graphe  $G$  :

```

Entrée : Graphe connexe  $G$ 
Sortie : Coupe de  $G$ 
 $H \leftarrow G$ 
Tant que  $|S(H)| > 2$  :
  Choisir aléatoirement une arête  $\{u, v\}$  de  $H$ 
   $H \leftarrow H/\{u, v\}$ 
Renvoyer  $S(H)$ 

```

3. Montrer l'invariant : «  $H$  est connexe ».

4. On note  $c(G)$  la taille minimum d'une coupe de  $G$ . Montrer l'invariant : «  $c(H) \geq c(G)$  ». Peut-on avoir  $c(H) > c(G)$  ?

Solution : Soit  $C$  une coupe minimum de  $H$ , à une itération donnée. Alors  $C$  est aussi une coupe de  $G$ . Donc  $c(H) \geq c(G)$ .

5. Expliquer comment implémenter cet algorithme en  $O(n^2)$ .

Solution : On utilise une matrice d'adjacence  $M$  pour représenter  $H$ , où  $m_{i,j}$  est le nombre d'arêtes entre  $i$  et  $j$ . À chaque itération, on modifie en  $O(n)$  les lignes et colonnes de  $u$  et  $v$  en les additionnant sans changer la structure de la matrice (on a donc plusieurs lignes correspondant au même sommet dans  $H$ ). On met aussi 0 dans  $m_{u,v}$  et  $m_{v,u}$ . On conserve aussi, pour chaque noeud de  $H$ , un de ses représentants dans  $G$  ainsi que son degré dans  $H$ . On peut aussi générer en  $O(n)$  une arête aléatoire de  $H$  en choisissant un sommet  $u \in S(H)$  avec probabilité  $\deg(u)/|A(H)|$  puis en choisissant un voisin  $v$  de  $u$  avec probabilité  $m_{u,v}/\deg(u)$ .

L'algorithme a une complexité en  $O(n) + O(n)$  par itération, soit une complexité en  $O(n^2)$ .

Soit  $C$  une coupe minimum et  $k = |C|$ .

6. Montrer que  $p \geq \frac{nk}{2}$ .

Solution : Si  $v \in S$ ,  $\deg(v) \geq k$  (sinon,  $(\{v\}, S \setminus \{v\})$  serait une coupe de taille  $< k$ ). Donc  $2|A| = \sum_{v \in S} \deg(v) \geq nk$  d'où  $p \geq \frac{nk}{2}$ .

7. Montrer que la probabilité de ne pas choisir une arête de  $C$  lors de la première contraction est au moins  $1 - \frac{2}{n}$ .

Solution : La probabilité de choisir une arête de  $C$  lors de la première contraction est  $\frac{k}{p} \leq \frac{2}{n}$ . La probabilité de ne pas choisir une arête de  $C$  est donc  $\geq 1 - \frac{2}{n}$ .

8. Montrer  $\mathbb{P}(S(H) = C) \geq \frac{2}{n(n-1)}$ .

9. En déduire un algorithme probabiliste pour trouver une coupe minimum de  $G$  avec une probabilité au moins  $1 - \frac{1}{n}$ .

On pourra utiliser l'inégalité :  $(1 - \frac{1}{x})^x \leq \frac{1}{e}$ .

Solution : On peut répéter l'algorithme plusieurs fois et garder la coupe minimum parmi les résultats obtenus. La probabilité de ne jamais obtenir de coupe minimum est au plus :

$$\left(1 - \frac{2}{n(n-1)}\right)^{\frac{n(n-1)}{2} \ln n} \leq \left(\frac{1}{e}\right)^{\ln n} = \frac{1}{n}$$

10. Montrer qu'un multi-graphe connexe à  $n$  sommets peut avoir au plus  $\frac{n(n-1)}{2}$  coupes minimums. Donner un cas d'égalité.

Solution : Comme l'algorithme de Karger a une probabilité au moins  $\frac{2}{n(n-1)}$  de renvoyer une coupe minimum en particulier, il ne peut pas y avoir plus de  $\frac{n(n-1)}{2}$  coupes minimums. De plus, cette borne est atteinte pour les cycles de taille  $n$ , donc la borne est optimale.

## II Ensemble indépendant

Soit  $G = (S, A)$  un graphe non-orienté. Un ensemble indépendant (ou : stable) de  $G$  est un sous-ensemble  $I \subseteq S$  tel que  $\forall u, v \in I, \{u, v\} \notin A$ . On note  $\alpha(G)$  la taille du plus grand ensemble indépendant de  $G$ .

On considère les problèmes de décision suivants :

**INDEPENDANT**

Entrée : un graphe  $G$  et un entier  $k$ .

Sortie : est-ce que  $\alpha(G) \geq k$  ?

**CLIQUE**

Entrée : un graphe  $G$  et un entier  $k$ .

Sortie : est-ce que  $G$  possède une clique de taille  $k$ , c'est-à-dire un sous-ensemble  $C \subseteq S$  tel que  $\forall u, v \in C, \{u, v\} \in A$  ?

1. On admet que CLIQUE est NP-complet. Montrer que INDEPENDANT est NP-complet.

Solution : INDEPENDANT ∈ NP car on peut vérifier en temps polynomial qu'un ensemble de sommets est indépendant. Soit  $G = (S, A)$  une instance de CLIQUE. On construit  $\bar{G} = (S, \bar{A})$  où  $\bar{A} = \{(u, v) \in S^2 \mid u \neq v \text{ et } (u, v) \notin A\}$ . Alors  $C$  est une clique de taille  $k$  dans  $G$  si et seulement si  $C$  est un ensemble indépendant de taille  $k$  dans  $\bar{G}$ . Comme CLIQUE est NP-complet et que CLIQUE se réduit en temps polynomial à INDEPENDANT, INDEPENDANT est NP-complet.

2. Décrire un algorithme efficace pour calculer  $\alpha(G)$  si  $G$  est un arbre.

Solution : On peut enraciner  $G$  en un arbre  $a$  et calculer récursivement la taille maximum  $f(a)$  d'un ensemble indépendant contenant la racine de  $a$  et la taille maximum  $g(a)$  d'un ensemble indépendant ne contenant pas la racine de  $a$  :

$$f(a) = 1 + \sum_{c \text{ fils de } a} g(c)$$

$$g(a) = \sum_{c \text{ fils de } a} \max(f(c), g(c))$$

On peut aussi calculer taille maximum  $h(a)$  d'un ensemble indépendant de  $a$  avec :

$$h(a) = \max(1 + \sum_{c \in \text{petits fils de } a} h(c), \sum_{c \in \text{fils de } a} h(c))$$

On considère l'algorithme suivant pour INDEPENDANT, où  $p \in [0, 1]$  :

Algorithme 1

```

 $I \leftarrow$  ensemble obtenu en prenant chaque sommet de  $S$  avec probabilité  $p$ 
Pour  $\{u, v\} \in A$  :
  Si  $u \in I$  et  $v \in I$  :
     $\quad \quad \quad$  Supprimer aléatoirement  $u$  ou  $v$  de  $I$ 
Renvoyer  $I$ 
```

3. Quelle est l'espérance de  $|I|$  juste avant de rentrer dans la boucle Pour ?

Solution : L'espérance de  $|I|$  est  $p|S|$ .

4. Montrer que l'espérance de  $|I|$  renvoyé par l'algorithme 1 est au moins  $p|S| - p^2|A|$ .

Solution : À chaque fois qu'une arête est considérée dans la boucle Pour, il y a une probabilité au plus  $p^2$  que ses deux extrémités soient dans  $I$  (il est possible qu'une de ces deux extrémités ait déjà été supprimée). Donc l'espérance de  $|I|$  est au moins  $p|S| - p^2|A|$ .

5. Montrer que  $\alpha(G) \geq \frac{|S|^2}{4|A|}$ .

Solution : Par étude de fonction, on trouve que  $p|S| - p^2|A|$  est maximum lorsque  $p = \frac{|S|}{2|A|}$  et vaut alors  $\frac{|S|^2}{4|A|}$ . Il existe donc au moins un ensemble indépendant de taille supérieure ou égale à l'espérance  $\frac{|S|^2}{4|A|}$ .

On considère un autre algorithme pour INDEPENDANT, pour un graphe  $G = (S, A)$  :

Algorithme 2

```

 $S \leftarrow$  sommets de  $S$  ordonnés selon un ordre quelconque
 $I \leftarrow \emptyset$ 
Pour  $v \in S$  :
   $\quad \quad \quad$   $I \leftarrow I \cup \{v\}$ 
   $\quad \quad \quad$  Supprimer  $v$  et ses voisins de  $S$ 
Renvoyer  $I$ 
```

6. Montrer que l'algorithme 2 donne une  $\frac{1}{\Delta + 1}$ -approximation de  $\alpha(G)$ , où  $\Delta$  est le degré maximum des sommets de  $G$ .

Solution : À chaque étape, on supprime au plus  $\Delta + 1$  sommets de  $S$  donc il y a au moins  $\left\lceil \frac{|S|}{\Delta + 1} \right\rceil$  étapes.

D'où  $|I| \geq \frac{|S|}{\Delta + 1} \geq \frac{\alpha(G)}{\Delta + 1}$ .

7. On ordonne les sommets de  $S$  suivant une permutation uniformément aléatoire.

Calculer l'espérance de  $|I|$  puis montrer que  $\alpha(G) \geq \sum_{i=1}^{|S|} \frac{1}{d_i + 1}$ , où  $d_i$  est le degré du  $i$ -ème sommet.

Solution : Soit  $X_v$  la variable aléatoire valant 1 si  $v$  est ajouté à  $I$ , 0 sinon.

Un sommet  $v$  est ajouté à  $I$  si et seulement si tous ses voisins apparaissent après lui dans la permutation. Donc

$$\mathbb{E}[X_v] = \mathbb{P}(X_v = 1) = \frac{1}{d_v + 1}.$$

Donc  $\mathbb{E}[|I|] = \sum \mathbb{E}[X_v] = \sum \frac{1}{d_v + 1}$ .

Il existe au moins un ensemble indépendant de taille supérieure à la moyenne.