DM14

Un agent secret a pour mission de perturber un réseau de communications ennemi en coupant certains fils. L'agent a pour consigne de couper le moins de fils possibles. Le réseau est modélisé par un *multigraphe non orienté* $G = (S, A)$, où S est l'ensemble des sommets du graphe et A l'ensemble de ses arêtes. Résoudre le problème de l'agent secret, c'est rechercher une coupe minimum dans G.

Un *multigraphe* est un graphe dans lequel des couples de mêmes sommets peuvent être reliés par plus d'une arête. Dans toute la suite, on considère les multigraphes sans arêtes du type (s, s) (boucles).

Une *coupe* d'un multigraphe non orienté $G = (S, A)$ est une partition non triviale (S_1, S_2) de S, c'est-à-dire telle que $S_1 \cup S_2 = S, S_1 \cap S_2 = \varnothing$ et $S_1, S_2 \neq \varnothing$. On dit que les arêtes reliant S_1 à S_2 sont les arêtes de la coupe.

La *taille* d'une coupe (S_1, S_2) est définie par $|(S_1, S_2)| = |\{(s, t) \in A, s \in S_1, t \in S_2\}|$. Autrement dit, c'est le nombre d'arêtes de G qui relient S_1 et S_2.

Une *coupe minimum* est une coupe de taille minimale. Pour trouver une coupe minimum d'un multigraphe $G = (S, A)$, on peut énumérer l'ensemble des coupes possibles (il y en a $2^{|S|}$ ...), ou encore utiliser un algorithme de flot (le plus efficace étant en $O\left(|S|^3\right)$). La suite de cet exercice étudie un *algorithme probabiliste*, appelé algorithme de Karger, basé sur la notion de contraction d'arête.

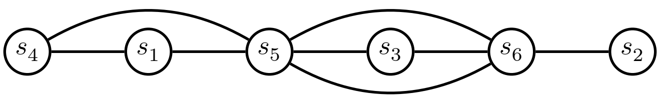
Contraction d'arête

Soient $G = (S, A)$ un multigraphe et $a = (s, t) \in A$ une arête de G de sommets s et t. *Contracter* l'arête a consiste à :

- créer un nouveau sommet $u = (st)$ dans S ; u est un *supersommet* du nouveau graphe ;
- pour toute arête (r, s) ou $(r, t), r \in S$, ajouter une arête (r, u) à A. Dans le cas où (r, s) et (r, t) existent, deux arêtes reliant r à u sont créées.
- Supprimer de A toutes les arêtes ayant s ou t comme extrémité.
- Supprimer s et t de S.

Un supersommet peut donc être vu comme un ensemble de deux sommets du graphe initial, obtenu après une contraction d'arête. Le multigraphe obtenu est appelé *graphe contracté* et est noté G/st. Si plusieurs arêtes relient s à t dans G, contracter l'une d'entre elles supprime toutes les autres du graphe G/st. Dans toute la suite, on notera G_0 le graphe initial, avant transformations par contractions.

Question 1. Donner le graphe obtenu par contraction d'une arête (s_5, s_6). On considère le graphe G_0 de la figure suivante.



Question 2. Après plusieurs contractions, un supersommet u du graphe résultant contient un sous-ensemble $S_u \subseteq S$ de sommets de G. Soit u un supersommet et $s, t \in S_u$. Montrer qu'il existe un chemin Γ entre s et t dans G_0 tel que chaque arête de Γ a été contractée.

Question 3. Montrer qu'en contractant une arête s, t dans un multigraphe G, la taille d'une coupe minimum du graphe contracté G/st est au moins égale à la taille d'une coupe minimum de G.

Question 4. Montrer que la taille d'une coupe minimum de G/st est strictement plus grande que celle d'une coupe minimum de G si et seulement si l'arête (s, t) est une arête de toutes les coupes minimum de G.

Premier algorithme

On construit un premier algorithme qui essaye de trouver une coupe minimum en contractant aléatoirement une arête de G, jusqu'à ce qu'il ne reste que deux sommets (algorithme 3).

Algorithme 1 : algorithme de Karger

fonction *Karger(G, n)*
  Entrée : un multigraphe non orienté $G = (S, A)$, un entier naturel n
  Sortie : une coupe minimum de G
  pour $i = |S|$ *à* n *en décrémentant de* 1 **faire**
    Tirer aléatoirement (loi uniforme) une arête de $a = (s, t) \in A$
    $G \leftarrow G/st$

Question 5. Appliquer l'algorithme 1 au graphe de la figure 2, en contractant successivement (s_5, s_6), $((s_5 s_6), s_2)$, (s_1, s_4) et $((s_5 s_6 s_2), s_3)$. Chaque graphe contracté sera représenté. La coupe calculée est-elle une coupe minimum ?

D'après la question 4, tant que l'on ne contracte pas une arête faisant partie de toutes les coupes minimum de G, alors l'algorithme 1 trouvera une coupe minimum. On cherche alors la probabilité de ne jamais contracter une telle arête.

Question 6. Soit $d(s)$ le degré d'un sommet $s \in S$. Montrer que $\sum_{s\in S} d(s) = 2|A|$ (lemme des poignées de main).

Question 7. En déduire qu'une coupe minimum a une taille d'au plus $2|A|/|S|$.

Question 8. Soit $C = (S_1, S_2)$ une coupe minimum de G. L'objet des questions Q27 et Q28 est de minorer la probabilité que l'algorithme 3 renvoie C. Pour cela, on remarque que cet algorithme ne renvoie pas C si et seulement si lors d'une itération on choisit une arête qui traverse C, c'est-à-dire telle qu'un sommet est dans S_1 et l'autre dans S_2. Quelle est la probabilité maximum de choisir une arête qui traverse C ?

Question 9. En déduire que la probabilité $P_{|S|}$ que l'algorithme 3 renvoie C est supérieure ou égale à $\frac{2}{|S|(|S|-1)}$.

Question 10. Déduire de la question précédente qu'un multigraphe non orienté $G = (S, A)$ possède au plus $\frac{|S|(|S|-1)}{2}$ coupes minimum.

Deuxième algorithme

Le résultat précédent n'est pas satisfaisant d'un point de vue complexité. On peut cependant l'améliorer facilement en utilisant une technique d'amplification : on itère l'algorithme **3** plusieurs fois et on retourne la valeur minimale obtenue (algorithme 4).

Algorithme 2 : algorithme de Karger amplifié

```
fonction KargerAmplifie(G, n)
    Entrée : un multigraphe non orienté G = (S, A), un entier naturel n
    Sortie : une coupe minimum de G
    t ← ∞
    pour i = 1 à n faire
        X ← Karger(G, 2)
        si |X| < t alors
            t ← |X|
            C ← X
    renvoyer C
```

Question 11. Montrer que la probabilité maximale que l'algorithme 2 ne trouve pas une coupe minimum est égale à $\left(1 - \frac{2}{|S|(|S|-1)}\right)^N$.

Question 12. On rappelle que pour tout $x \in \mathbb{R}$, $1 + x \leqslant e^x$. Montrer que la probabilité que l'algorithme 2 renvoie une coupe minimum est supérieure à $1 - e^{2N/|S|(S|-1)}$. En déduire, pour $c > 0$ fixé, la plus petite valeur de N telle que cette probabilité soit supérieure à $1 - \frac{1}{N^c}$.