Partiel du cours d'approximation

{Emmanuelle.Lebhar, Nicolas.Schabanel}@ens-lyon.fr

Lundi 17 novembre de 18h00 à 20h00

Note. Le plus grand soin devra être porté à la rigueur des arguments utilisés.

EXERCICE 1

Question 1 *Démontrez qu'il n'existe pas de $\alpha(n)$-approximation pour le problème du k-centre non-métrique, quelle que soit $\alpha(n)$ calculable en temps polynomial, à moins que P=NP.*

Rappel. Le *problème du k-centre (non-métrique)* est de trouver un sous-ensemble S de k sommets, d'un graphe non-orienté G, muni de poids sur les arêtes $w : E \to \mathbb{Q}^+$, tel que S minimise : $\max_{u \in V-S} \min_{v \in S} w(uv)$.

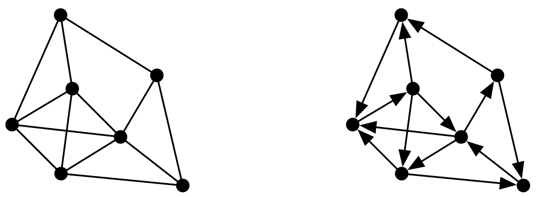
Indication. Utilisez que le problème du dominant minimum d'un graphe non-orienté est NP-difficile.

EXERCICE 2

On étudie le problème suivant :

Problème 1 (3-Cyclicité maximum) *Étant donné un graphe $G = (V, E)$ non-orienté, trouver une orientation des arêtes de G qui maximise le nombre de circuits (cycle orienté) de longueur 3.*

Voici l'exemple d'un graphe et d'une orientation de ses arêtes, qui engendre cinq circuits de longueur 3 :



Question 2 *Proposez une 1/4-approximation randomisée pour ce problème, et prouvez-le. Combien de bits aléatoires utilise votre algorithme ?*

Question 3 *Exhibez une famille d'instances critiques pour votre algorithme.*

Question 4 *Est-il possible d'en déduire une 1/4-approximation déterministe (en temps polynomial) ? Si oui, comment procéderiez-vous ?*

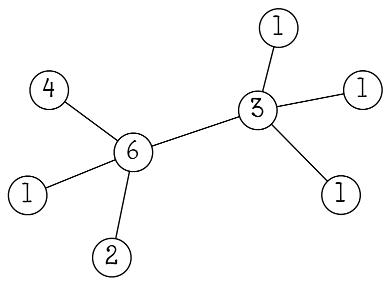
EXERCICE 3 (Couverture par sommets de poids minimum)
Considérons l'algorithme suivant pour le problème de la couverture par sommets avec des poids, où le but est de trouver un ensemble de sommets de poids minimum couvrant toutes les arêtes du graphe G (non-orienté avec des poids sur les sommets).

On associe à chaque sommet w, une variable $t(w)$, initialisée au poids de w. $t(w)$ décroît au cours de l'algorithme. Lorsque $t(w)$ atteint zéro, w est sélectionné. prix(e) est le prix payé pour la couverture de l'arête e.

Algorithme 1

1. Initialisation :
   $C \leftarrow \varnothing$
   $\forall w \in V$, $t(w) \leftarrow \text{poids}(w)$
   $\forall e \in E$, $\text{prix}(e) \leftarrow 0$
2. Tant que C n'est pas une couverture par sommets faire :
   Choisir une arête non-couverte, uv. Poser $m = \min(t(u), t(v))$.
   $t(u) \leftarrow t(u) - m$
   $t(v) \leftarrow t(v) - m$
   $\text{prix}(uv) \leftarrow m$
   Inclure dans C tous les sommets w tels que $t(w) = 0$.
3. Retourner C.

Question 5 *Exécutez l'algorithme sur le graphe suivant, où les poids des sommets sont marqués dans les sommets :*



Question 6 *Démontrez qu'à chaque instant, pour tout sommet u :*

$$t(u) = \text{poids}(u) - \sum_{v\ :\ uv \in E} \text{prix}(uv).$$

Question 7 *Démontrez que le prix total payé pour couvrir les arêtes est un minorant de* OPT.

Question 8 *Démontrez que cet algorithme est une 2-approximation.*

Question 9 *Exhibez une famille d'instances critiques pour cet algorithme.*

EXERCICE 4
Considérons le problème suivant :

Problème 2 (Débordement de coût minimum) *Étant donnés n objets de tailles positives $t_1, \ldots, t_n$, et de coûts positifs $c_1, \ldots, c_n$, et une constante W, trouver un sous-ensemble $S \subset \{1, \ldots, n\}$ de coût minimum tel que $\sum_{i \in S} t_i \geq W$.*

Question 10 *Proposez une réduction polynomiale du problème Partition, NP-difficile, démontrant que le problème du débordement minimum est NP-difficile.*

Rappel. Étant donné n entiers positifs $x_1, \ldots, x_n$, le *problème Partition* consiste à déterminer s'il existe un ensemble $S \subset \{1, \ldots n\}$ tel que : $\sum_{i \in S} x_i = \sum_{i \notin S} x_i$.

On étudie l'algorithme glouton suivant : 1) trier les objets par rapport coût sur taille croissant, quite à renuméroter : $c_1/t_1 \leq \ldots \leq c_n/t_n$; et 2) prendre i, le plus petit indice tel que $\sum_{j \leq i} t_j \geq W$, et poser $S = \{1, \ldots, i\}$.

Question 11 *Démontrez que cet algorithme peut produire des solutions arbitrairement mauvaises, même si tous les objets sont de taille $< W$.*

Question 12 *Posez $C = \max_i c_i$. Donnez un algorithme pseudo-polynomial pour résoudre le problème du débordement minimum.*

Question 13 *Proposez un schéma d'approximation totalement polynomial pour le problème du débordement minimum.*

Indication. Commencez par supposer que $C \leq \text{OPT}$.

EXERCICE 5
On étudie le problème suivant :

Problème 3 (Multi-couverture par ensembles) *Étant donnés $U = \{u_1, \ldots, u_m\}$, une collection $\mathcal{S} = \{S_1, \ldots, S_n\}$ de sous-ensembles de U, une fonction de coût $c : \mathcal{S} \to \mathbb{Q}^+$, et des entiers positifs $r_1, \ldots, r_m \geq 0$, trouver une multi-couverture C de coût minimum qui couvre chaque élément u_i au moins r_i fois.*
Une multi-couverture C est un multi-ensemble où un même S_i peut apparaître plusieurs fois ; le coût de α copies de S_i est $\alpha \cdot c(S_i)$.

Question 14 *Proposez un algorithme glouton qui soit une H_m-approximation pour ce problème, et prouvez-le.*

Indication. On pourra associer à chaque élément d'un ensemble d'une multi-couverture optimale, un événement du type "cet élément est couvert pour la j-ème fois dans mon algorithme".

Question 15 *Exhibez une famille d'instances critiques pour votre algorithme.*
