TD 12 – Tout en douceur

Exercice 1. *Une petite laine ?*

Nous considérons ici le problème de la couverture par sommets : étant donné un graphe non-orienté $G = (V, E)$ et une fonction de poids sur les sommets $w : V \to \mathbb{Q}+$, trouver $S \subseteq V$ de poids minimum qui couvre les arêtes (c'est à dire $\forall\, e \in E, \exists\, s \in S : s \in e$). En cours, nous avons vu une 2-approximation dans le cas où w est constante. Nous allons étudier ici une 2-approximation dans le cas général. Pour ce faire nous nous intéressons à un type de fonction de poids particulier : les fonctions de poids par degré. w est une fonction de poids par degré si $\exists c > 0, \forall v \in V : w(v) = c \cdot deg(v)$.

1. Soit $\{G = (V, E), w\}$ une instance du problème telle que w est une fonction de poids par degrés. Montrer que $w(V) \leq 2 \cdot OPT$ où $w(V) = \sum_{v \in V} w(v)$.
   On sait donc traiter les instances avec une fonction de poids par degrés. La méthode du mille-feuille consiste alors à décomposer une instance quelconque en une famille d'instances avec fonction de poids par degrés.
2. À chaque étape de la décomposition, on cherche la plus grande fonction de poids par degrés p inférieure à w. Expliquer comment calculer p.

L'algorithme du mille-feuille est le suivant :

Données:
$G = (V, E)$;
une fonction de poids quelconque w;
début
  $t \leftarrow 0$;
  $G_0 \leftarrow G$;
  $w_0 \leftarrow w$;
  tant que G_t *contient une arrête* **faire**
    $D_t \leftarrow u \in V_t : deg_t(u) = 0$;
    $p_t \leftarrow$ plus grande fonction de poids par degrés inférieure à w_t dans G_t;
    $S_t \leftarrow u \in V_t : p_t(u) = w_t(u)$;
    $G_{t+1} \leftarrow G_t \setminus (D_t \cup S_t)$;
    $w_{t+1} \leftarrow w_t - p_t$;
    $t \leftarrow t = 1$;
  retourner $C = \displaystyle\bigcup_{k=0}^{t-1} S_k$

3. Montrer que l'algorithme termine en temps polynomial.
4. Montrer que l'ensemble C de sommets retournés par l'algorithme est bien une couverture.
5. Pour tout $v \in C$, exprimer $w(v)$ en fonction des poids par degrés p_k. Qu'en est-il pour $v \notin C$?
   Remarque : on pourra poser $p_k(u) = 0$ pour tout sommet $u \notin G_t$.

À partir de maintenant, on notera C^* une couverture par sommets optimale pour l'instance de départ $\{G = (V, E), w\}$.

6. Pour toute étape i de l'algorithme, comparer $p_i(C \cap G_i)$ et $p_i(C^* \cap G_i)$.
   Indication : on remarquera que $C \cap G_i$ et $C^* \cap G_i$ sont deux couvertures par sommets de G_i.
7. Montrer que l'algorithme est une 2-approximation du problème de la couverture par sommet minimum avec des poids arbitraires. Trouver un exemple où l'algorithme renvoie effectivement une solution de poids $2.OPT$.

Exercice 2. *Le milieu c'est merveilleux !*

On rappelle quelques définitions :

– Dans un graphe $G = (V, E)$, un *ensemble indépendant* est un sous-ensemble de sommets V' non reliés par des arêtes (si $u \in V'$ et $v \in V'$, alors $(u, v) \notin E$).

– Dans un graphe $G = (V, E)$, un *ensemble dominant* est un sous-ensemble de sommets V' tel que tout sommet de $V \setminus V'$ est adjacent à un sommet de V'. On note $dom(G)$ le cardinal minimal d'un ensemble dominant.

1. Montrer que trouver un ensemble indépendant de cardinal maximal est NP-difficile.

2. Montrer que trouver un ensemble dominant de cardinal minimal $dom(G)$ est NP-difficile.

Soit $G = (V, E)$ un graphe non-orienté, complet, dont les arêtes sont pondérées par une fonction de poids w qui vérifie l'inégalité triangulaire : $w(u, v) \leq w(u, w) + w(w, v)$ pour tout triplet de sommets (u, v, w). Soit aussi un entier $k \geq 1$.

Pour tout $S \subset V$ et tout $v \in V \setminus S$, on définit $connect(v, S)$ comme le poids minimal d'une arête reliant v à un sommet de S : $connect(v, S) = \min_{s \in S}\{w(v, s)\}$. Le problème est de trouver un k-centre, c'est à dire un sous-ensemble S de cardinal k et tel que $center(S) = \max_{v \in V \setminus S}\{connect(v, S)\}$ soit minimal.

3. À quoi peut bien servir de déterminer un k-centre (donner un exemple d'application) ?

4. Montrer que trouver un k-centre est NP-difficile.

On va chercher une 2-approximation, i.e. un S de cardinal k tel que $center(S) \leq 2 \cdot OPT$, où $OPT = \min_{S \subset V, |S| = k}\{center(S)\}$.

On ordonne les arêtes de E par poids croissant : $w(e_1) \leq w(e_2) \leq \ldots \leq w(e_m)$, où $m = |E|$. On pose $G_i = (V, E_i)$ où $E_i = \{e_1, e_2, \ldots, e_i\}$ est l'ensemble des i premières arêtes.

5. Montrer que résoudre le problème du k-centre revient à trouver le plus petit indice i tel que G_i a un ensemble dominant de cardinal au plus k.

Une dernière définition : le carré d'un graphe $G = (V, E)$, noté $G^{(2)} = (V, E^{(2)})$, contient les chemins de longueur au plus deux : $(u, v) \in E^{(2)}$ si $(u, v) \in E$ ou s'il existe $w \in V$ tel que $(u, w) \in E$ et $(w, v) \in E$.

6. Étant donné un graphe H, soit I un ensemble indépendant du graphe carré $H^{(2)}$. Montrer que $|I| \leq dom(H)$.

7. L'algorithme d'approximation du k-centre est le suivant : **début**

   Construire $G_1^{(2)}, G_2^{(2)}, \ldots, G_m^{(2)}$ Trouver de manière gloutonne un ensemble indépendant inextensible (auquel on ne peut pas rajouter des sommets) M_i dans chaque graphe $G_i^{(2)}$ Trouver le plus petit indice i tel que $|M_i| \leq k$, soit j cet indice **retourner** M_j

   (a) Montrer que $w(e_j) \leq OPT$.

   (b) Montrer que l'algorithme est bien une 2-approximation.

8. Montrer que la borne 2 est stricte : donner un exemple de graphe où l'algorithme réalise effectivement une 2-approximation.

9. Montrer que si $P \neq NP$, il n'existe pas de $(2 - \varepsilon)$-approximation au problème du k-centre, pour tout $\varepsilon > 0$.