TD 10 : Algorithmes d'approximation

1 Voyageur de commerce révisé

Exercice 1. *On suppose disposer d'un algorithme de calcul du couplage parfait minimal d'un graphe pondéré.*

1. *Soit G' un multi-graphe connexe. Montrer qu'il existe un cycle empruntant chaque arête exactement une fois si, et seulement si le degré de tout sommet est pair.*
2. *Soit $G = (V, E, c)$ un graphe connexe pondéré vérifiant l'inégalité triangulaire. On note $T = (V, E')$ un arbre couvrant de poids minimal. On note $V_p \subseteq V$ l'ensemble des sommets de degrés impairs dans T. Montrer que $|V_p|$ est pair.*
3. *On considère le sous-graphe complet $G' = (V_p, (V_p)^2)$. Montrer qu'il admet un couplage parfait. On notera M un couplage parfait minimal.*
4. *On considère une solution optimale $\pi = v_1, \sigma_1, v_2, \sigma_2 \ldots v_{2m}, \sigma_{2m}, v_1$ au problème du voyageur de commerce du graphe G, en notant $\{v_1, \ldots, v_{2m}\} = V_p$. Montrer que $c(M) \leq c(\pi)/2$.*
5. *Soit le multigraphe G' constitué de l'union de T et M. Montrer que tout sommet est de degré pair.*
6. *On considère un cycle de G' empruntant toutes les arêtes, montrer que l'on peut en extraire un chemin hamiltonien de coût inférieur à $c(T) + c(M)$.*

2 Bin packing

Une instance du problème BIN-PACKING est donnée par n rationnels $x_1 \ldots x_n$ avec $\forall i \; 0 < x_i \leq 1$. On cherche le plus petit entier m tel que l'on peut ranger tous les éléments en m paquets de taille inférieure à 1, c'est-à-dire tel qu'il existe une fonction $f : [1, n] \to [1, m]$ avec $\forall b \in [1, m] \; \sum_{i \in f^{-1}(b)} x_i \leq 1$.

Exercice 2 (Glouton)**.** *On considère l'algorithme itératif plaçant chaque élément dans le premier paquet disponible dans lequel il peut rentrer (l'algorithme crée un nouveau paquet vide si l'élément ne peut rentrer nulle part). On note m le résultat et f l'assignation correspondante fournie par cet algorithme.*

1. *Montrer qu'au moins $m - 1$ paquets sont remplis (strictement) à plus de la moitié de leur capacité (c'est-à-dire qu'il existe $m - 1$ valeurs de b telles que $\sum_{i \in f^{-1}(b)} x_i > \frac{1}{2}$).*
2. *Montrer que toute solution au problème du bin packing est au moins égale à $S = \sum_i x_i$.*
3. *En déduire que l'algorithme glouton présente une garantie de performance de 2.*

Exercice 3 (FFD)**.** *On considère l'algorithme suivant :*

- *On trie les éléments par ordre décroissant : $x_1 \geq x_2 \geq \ldots \geq x_n$.*
- *On applique l'algorithme glouton précédent.*

Soit $b \leq m$ l'indice d'un conteneur.

1. *Supposons que b contient un élément $x_i > 1/2$ (ie $i \in f^{-1}(b)$). Montrer que la solution optimale m^* vérifie $m^* \geq b$.*

2. *On suppose dans cette question que b ne contient aucun élément de poids supérieur à $1/2$.*

3. (a) *En déduire que les conteneurs $b' \in [b, m-1]$ contiennent au total plus de $2(m-b)$ éléments.*
   (b) *Sous-cas 1: on suppose de plus que $b \leq 2(m-b)$. Montrer que $S > b-1$.*
   (c) *Sous-cas 2: on suppose cette fois $b > 2(m-b)$. Montrer que $S > 2(m-b)$.*

4. *En déduire que l'algorithme possède une garantie de performance égale à $3/2$.*

Exercice 4 (Non-approximabilité)**.** *On suppose dans cet exercice disposer d'un algorithme d'approximation résolvant le problème de bin packing en temps polynomial avec une garantie de performance égale à k.*
On introduit le problème PARTITION défini comme suit :

- *Entrée : $a_1 \dots a_n \in \mathbb{N}$*
- *Sortie : Existe-t-il $A \subseteq [1, n]$ tel que $\sum_{i \in A} a_i = \sum_{i \notin A} a_i$*

On admet que ce problème est NP-complet.

1. *Montrer que PARTITION n'admet pas de solution lorsqu'il existe j tel que $a_i > \sum_{j \neq i} a_j$.*

2. *Convertir toute instance I de PARTITION en une instance du problème de BIN-PACKING I', telle I admet une solution, si et seulement si, I' admet une solution à 2 paquets.*

3. *En déduire que $k \geq \frac{3}{2}$ (à moins que $P = NP$).*

Exercice 5 (Schema d'approximation asymptotique)**.** *Dans cet exercice, nous allons voir qu'il est toujours possible de concevoir un algorithme d'approximation dont la garantie de performance asymptotique reste bonne.*

1. *On fixe $\varepsilon > 0$ et $d \in \mathbb{N}$. Montrer qu'il existe un algorithme polynomial résolvant BIN-PACKING dans le cas d'instances contenant moins de d valeurs différentes toutes supérieures à ε. (Indication: dénombrer le nombre de solutions).*

Pour toute instance I de taille n, on note $m^*(I)$ la valeur optimale de bin packing.

2. *On suppose I triée par poids croissants, tous supérieurs à $\varepsilon > 0$. On fixe $P > 0$ et on note $Q = \left\lfloor \frac{n}{P} \right\rfloor$. On construit les instances H et J de taille n définies par*

$$H = (\underbrace{\min(x_1 \dots x_Q) \dots}_{Q \text{ fois}}, \underbrace{\min(x_{Q+1} \dots x_{2Q}) \dots}_{Q \text{ fois}}, \dots \underbrace{\min(x_{\lfloor (n-1)/Q \rfloor Q+1} \dots x_n) \dots}_{n \mod Q \text{ fois}})$$

$$J = (\underbrace{\max(x_1 \dots x_Q) \dots}_{Q \text{ fois}}, \underbrace{\max(x_{Q+1} \dots x_{2Q}) \dots}_{Q \text{ fois}}, \dots \underbrace{\max(x_{\lfloor (n-1)/Q \rfloor Q+1} \dots x_n) \dots}_{n \mod Q \text{ fois}})$$

Intuitivement, on regroupe I par groupe de taille Q et on donne le même poids à tous les objets d'un même groupe : le poids le plus faible parmis les éléments du groupe (respectivement le plus élevé) dans le cas de H (respectivement J).

   (a) *Montrer que $m^*(H) \leq m^*(I) \leq m^*(J)$*
   (b) *Montrer que $m^*(J) \leq m^*(H) + Q \leq m^*(I) + Q$*
   (c) *En déduire un algorithme d'approximation polynomiale à garantie de performance $1 + \varepsilon$ (dans le cas $\forall i \; x_i \geq \varepsilon > 0$).*

NB : Bien que la meilleure garantie de performance est $\frac{3}{2}$, il existe un *schema d'approximation asymptotiquement polynomial*, c'est-à-dire que pour tout $\varepsilon > 0$, il existe un algorithme polynomial ayant une garantie de performance de $1 + \varepsilon$, pour peu que $m^* \geq 2/\varepsilon$ (instances de grande taille). Ce résultat n'est pas contradictoire avec la preuve d'inaproximabilité précédente, qui se base sur une réduction dans le cas d'un m^* petit (2 ou 3).