TD Algorithme approché

Un algorithme $^3/_2$ approché pour le TSP métrique

On a vu en cours un algorithme 2-approché pour le voyageur de commerce métrique (poids satisfaisant l'inégalité triangulaire). Ici on présente une amélioration : l'algorithme de Christofides.

On rappelle le problème : étant donné un graphe arête-valué, trouver un cycle passant par tous les sommets, i.e. un cycle hamiltonien, de poids minimum.

Algorithme

1-Calculer T un MST-Arbre couvrant de poids min.

2-Calculer un couplage de poids min entre les sommets de degré impair dans T.

3-En rajoutant à T les arêtes du couplage, on obtient un graphe eulérien (degrés pairs). Calculer un cycle eulérien dans ce graphe (i.e. cycle passant par toutes les arêtes une fois et une seule).

4-« Shunter » les sommets parcourus plus d'une fois

Cet algorithme donne un cycle hamiltonien de poids ne dépassant jamais $^3/_2$ du poids du cycle hamiltonien minimum.

1-Appliquer l'algorithme sur l'instance suivante de 6 sommets numérotés de 1 à 6 :

Poids	1	2	3	4	5	6
1	x	10	10	10	10	10
2		x	20	11	11	11
3			x	20	11	11
4				x	20	11
5					x	20

2-On obtient un cycle de poids 73. Le cycle optimal est 64. Vérifier la garantie.

3-Appliquer l'algorithme 2-approché du cours et vérifier qu'il peut donner un cycle hamiltonien de poids 100.

4-Démonstration de la garantie pour les graphes métriques (inégalité triangulaire vérifiée).

On note C_{opt} le cycle optimal et T l'arbre couvrant de poids minimum.

a-Montrer que le poids de T est inférieur ou égal au poids de C_{opt}.

b-Montrer qu'à partir de C_{opt} on peut construire un cycle hamiltonien C' parcourant uniquement les sommets de degré impairs dans T et de poids inférieur ou égal au poids de C_{opt}.

c-A partir de C' montrer que l'on peut construire un couplage de poids inférieur ou égal à la moitié du poids de C'. En déduire qu'un couplage de poids minimum sur les sommets de degré impair dans T est inférieur ou égal à la moitié du poids de C_{opt}.

d-En déduire que le cycle construit par l'algorithme est de poids inférieur ou égal à $^3/_2$ le poids de C_{opt}.

Préambule : technique pour montrer qu'un algorithme est ρ-approché.

- Problème de maximisation :

z* **= valeur optimale** du problème

$\bar{z}$ une **borne sup.** de la valeur (optimale) z* du problème : $\bar{z} \geq z^*$

z la valeur rendue par un **algorithme approché**.

Si on arrive à montrer que $z \geq \rho\bar{z}$ (0≤ρ≤1) alors on a immédiatement $z \geq \rho z^*$ c'est-à-dire z est une solution ρ-approchée.

Un algorithme ½-approché pour le problème du sac-à-dos

Soit (P) le problème de sac-à-dos :

$$\max z = \sum_{i=1}^{n} c_i x_i$$

Sous contraintes : $$\begin{cases} \sum_{i=1}^{n} a_i x_i \leq b \\ x_i \in \{0,1\}\ i = 1, \dots, n \end{cases}$$

Où c_i, a_i, b sont des entiers naturels positifs : a_i est le poids de l'objet i, c_i sa valeur marchande, b la capacité du sac.

Soit (PR) la relaxation continue de (P) où les variables sont dans l'intervalle [0,1] : $x_i \in [0,1]\ i = 1, \dots, n$.

(PR) se résout en temps polynômial de la façon suivante :

- on trie les ratios $\frac{c_i}{a_i}$ *i*=1,….,*n* dans l'ordre décroissant. Ensuite, selon cet ordre, on met les objets dans le sac jusqu'à ce que l'on dépasse la capacité b. Le premier objet qui dépasse est mis fractionnaire.

- Supposons les objets indexés par ordre des ratios $\frac{c_i}{a_i}$ *i*=1,….,*n* décroissant. Soit i^* tel que $\sum_{i=1}^{i^*-1} a_i \leq b$ et $\sum_{i=1}^{i^*} a_i > b$, on pose :

$$\begin{cases} x_i = 1 \quad i = 1, \dots, i^* - 1 \\ x_{i^*} = \dfrac{b - \sum_{i=1}^{i^*-1} a_i}{a_{i^*}} \\ x_i = 0 \quad i \geq i^* + 1 \end{cases}$$

On a $\bar{z} = \sum_{i=1}^{i^*-1} c_i + c_{i^*} x_{i^*}$ la valeur de la relaxation continue qui est donc une borne sup. de z* la valeur optimale du problème (P) i.e. $\bar{z} \geq z^*$.

1-Résoudre la relaxation continue de l'instance suivante :

objet	1	2	3	4	5
c_i	10	20	30	40	50
a_i	5	4	3	2	1

avec b=7

Note : la valeur optimale en 0-1 est 120.

2-On considère l'instance suivante :

i	1	2
c_i	1	α-1
a_i	1	α

Avec b=α avec $\alpha \geq 2$.

Montrer que si on prend comme solution approchée $z = \sum_{i=1}^{i^*-1} c_i$, il n'existe pas ρ constant tel que $z \geq \rho z^*$.

3-On considère maintenant la solution approchée qui consiste à choisir la meilleure valeur entre « prendre les objets de 1 à i*-1 » et « prendre l'objet de coût maximum ». C'est-à-dire $z = \max \left\{\sum_{i=1}^{i^*-1} c_i \ , c_{i_{max}}\right\}$ où $c_{i_{max}}$ est la valeur la plus élevée.

a-Que deviendrait la solution de l'instance précédente ?

b-Appliquer cet algorithme à l'instance de la question 1 et vérifier que la valeur obtenue dépasse la moitié de la valeur de la relaxation continue.

c-On remarque que $\max \ \{\alpha, \beta\} \geq \frac{1}{2}\alpha + \frac{1}{2}\beta$ et ceci quelque-soit α, β. A l'aide de cette remarque, montrer que la valeur z fournie par cet algorithme est supérieure ou égale à la moitié de $\bar{z}$ la valeur de la relaxation continue. En déduire que cet algorithme est ½-approché.

4-On considère l'instance suivante :

i	1	2	3
c_i	α-1	1	α-1
a_i	α-1	1	α

Avec b=2α-1 avec $\alpha \geq 2$.

Appliquer l'algorithme (question précédente) à cette instance. En déduire que la borne ½ est atteinte asymptotiquement (i.e. quand α devient grand).