Master Informatique Fondamentale (1ère année)
Algorithmes d'approximation

Partiel

23 novembre 2004

Exercice 1 : Représentant de commerce avec des poids 1 et 2 sur les arêtes

On note $\text{TSP}_{\{1,2\}}$ le problème du représentant de commerce restreint aux instances dont la fonction de poids (sur les arêtes) prend ses valeurs dans $\{1, 2\}$ uniquement.

Question 1. *Dans l'hypothèse $P \neq NP$, montrer que, pour tout $\epsilon > 0$, il n'existe pas de $(1 + \frac{1}{n} - \epsilon)$-approximation pour TSP$_{\{1,2\}}$.*

Réponse : On réduit polynomialement une instance I de "Cycle Hamiltonien" à une instance I' de "$TSP_{\{1,2\}}$" :

$$\left[\begin{array}{l} G = (V, E) \text{ un graphe.} \\ \exists\,?\text{ un cycle hamiltonien} \end{array}\right. \longrightarrow \left[\begin{array}{l} G' = (V, E') \text{ complet } (E' = V^2) \\ \text{avec } w(e) = \begin{cases} 1 \text{ si } e \in E \\ 2 \text{ sinon} \end{cases} \\ \exists\,?\ k \text{ tel que } TSP_{\{1,2\}} < k \end{array}\right.$$

Soit $\mathcal{A}$ une $(1 + 1/n - \varepsilon)$-approximation de $TSP_{\{1,2\}}$.
Si I admet un cycle hamiltonien, alors on peut passer par les arêtes de poids 1 donc $OPT(I') = n$. D'où

$$\mathcal{A}(I') \leq (1 + 1/n - \varepsilon)OPT(I') \leq n + 1 - \varepsilon n < n + 1$$

Conclusion : $\mathcal{A}(I') \leq n \Leftrightarrow$ il existe un cycle hamiltonien dans I. □

Exercice 2 : Petit Steiner

Question 2. *Montrer que si l'on donne en entrée les sommets utilisés par un arbre de Steiner optimal, le problème de l'arbre de Steiner devient polynomial.*

Réponse : On connait les arêtes U utilisées par un arbre de Steiner optimal. Une solution optimale est donc parmi les arbres de U : c'est la plus petite. Il suffit de trouver un arbre couvrant de poids minimum (MST), ce qui se fait en temps polynomial. □

Exercice 3 : MAX-DIRECTED-CUT

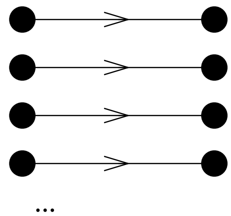
Étant donné un graphe G *orienté* muni d'une fonction de poids positive $w : E \to \mathbb{R}^+$ sur les arcs, trouver une coupe orientée de poids maximum, *i.e.* un ensemble $S \subseteq V$ de sommets qui maximise les poids total des arcs *sortants* de S (*i.e.*, des arcs (u, v) tels que $u \in S$ et $v \notin S$).

Question 3. *Proposer (et analyser) une $\frac{1}{4}$-approximation randomisée de ce problème. Exhiber une famille d'instances critiques.*

Réponse : L'algorithme utilisé est le même que celui vu en cours : pour chaque sommet v_i, on choisit au hasard de le mettre ou pas dans la coupe S.

$$\mathbb{E}(w(S)) = \sum_{e \in S} \mathbb{E}(w(\vec{e})) \qquad \text{par linéarité de l'espérance}$$

Pour une arête $e = (u, v)$, probabilité $1/4$ d'appartenir à la coupe ($1/2$ $u \in S \times 1/2$ $v \notin S$). On obtient une $1/4$-approximation.
Famille d'instance critique :



□

Question 4. *Dérandomiser votre algorithme pour obtenir une $\frac{1}{4}$-approximation déterministe en utilisant la méthode vue en cours. Quelle est la complexité en temps de cet algorithme ? Exhiber une famille d'instances critiques.*

Réponse : Cette algorithme randomisé nécessite seulement des bits 2 à 2 indépendants ; à partir de $\log n$ bits indépendants, on en fabrique $2^{\log n}$ 2 à 2 indépendants. On fait alors une boucle sur toute les chaînes de taille $\log n$ possibles. On garde le maximum, qui est nécessairement supérieur à l'espérance. □

Exercice 4 : Circuits et tournois

Un tournoi est un graphe complet $G = (V, E)$ dont on a orienté chaque arête (*i.e.* tel que, pour toute paire de sommets $(u, v) \in V^2$, $(u, v) \in E \Leftrightarrow (v, u) \notin E$). Étant donné un tournoi G, le problème de l'*élimination de circuits* est de trouver un sous-ensemble de sommets $S \subseteq V$ de taille minimum, tel que le sous-graphe induit par $V \setminus S$ dans G est sans circuit. On dit que S coupe tous les circuits de G.

Question 5. *Montrer qu'un tournoi sans circuit de longueur* 3 *est un tournoi sans circuit.*

Réponse : Par contraposée et par récurrence :
$P(n) =$ "si un tournoi a un circuit de taille n, alors il a un circuit de taille 3"
- $n = 3$: trivial
- $n \to n + 1$
  Si le tournoi possède un circuit de taille $n + 1$: $(x_1, x_2, \ldots, x_n, x_{n+1})$ avec $\forall i \leq n,\ (x_i, x_{i+1}) \in E$ et $(x_{n+1}, x_1) \in E$, alors

  soit $(x_1, x_n) \in E$, alors (x_1, x_n, x_{n+1}) est un circuit de taille 3

  soit $(x_n, x_1) \in E$, alors $(x_1, x_2, \ldots, x_n)$ est un circuit, donc, par induction, il y a un circuit de taille 3.

□

On admet qu'il existe une f-approximation du problème SET-COVER restreint aux instances telles que chaque élément de l'univers est contenu dans au plus f ensembles.

Question 6. *Donner une* 3*-approximation du problème de l'élimination de circuit.*

Réponse : $\{C_i\}$, l'ensemble des circuits de longueur 3, est l'univers.
Les sommets v_i seront les ensembles couvrants, de poids constants 1.
On dira qu'un circuit C_i est couvert par v_j si $v_j \in C_i$.
Le nombre de circuits de longueur 3 est au maximum de n^3 ; leur recheche prend un temps polynomial, la réduction est donc bien polynomiale.
Chaque élément (cycle) est couvert par exactement 3 ensembles (sommets).
Le nombre de sommets qu'il faut retirer au tournoi pour qu'il soit sans circuit est le nombre d'ensembles qui couvrent l'univers (couvrir un cycle, c'est retirer un de ses sommets).

Cette réduction iso-facteur nous permet d'obtenir une 3-approximation de "Elimination de circuits" à partir de la 3-approximation de "SET-COVER". □

Exercice 5 : Chemin du représentant de commerce

Étant donné un graphe non-orienté complet muni d'une fonction de poids métrique sur les arêtes, le problème du chemin du *représentant de commerce* est de trouver un *chemin* de longueur minimum, qui passe une fois et une seule par chaque sommet.

Ce problème se décline en trois variantes selon que 0, 1 ou 2 des extrémités du chemin à trouver sont données en entrée.

Question 7. *Proposer (et analyser) une 2-approximation pour les cas où 0 ou 1 extrémité du chemin est imposée en entrée (i.e. (a) le représentant de commerce peut partir de et arriver à n'importe quel sommet ou (b) le représentant de commerce doit partir d'un sommet fixé donné en entrée, mais peut arriver à n'importe quel autre). Exhiber une famille d'instances critiques.*

Réponse : Algo :

- chercher un MST
- doubler ses arêtes et enlever une arête
- trouver un chemin eulérien dans ce graphe
- simplifier le chemin eulérien en chemin hamiltonien en contractant les fourches (inégalité triangulaire)

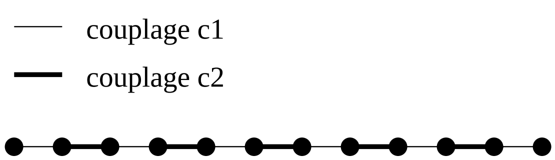
$w(\mathbb{A}(I)) \leq 2\,w(MST(I)) \leq 2\,w(OPT(I))$ □

Question 8. *Proposer (et analyser) une $\frac{3}{2}$-approximation pour les cas où aucune des extrémités du chemin n'est imposée en entrée. Exhiber une famille d'instances critiques.*

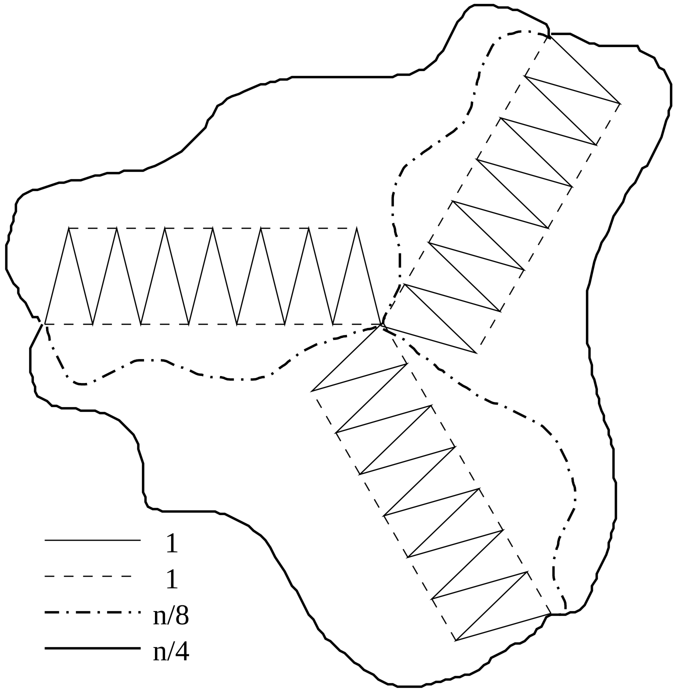
Réponse : Algo :

- chercher un MST
- Pour chaque paire (u, v) de sommets de degré impair dans le MST, chercher un couplage minimum M_i dans les sommets de degrés impairs dans le MST privé de (u, v)
- $M_{min} = \min_i M_i$
- $MST \cup M_{min}$ est un graphe à chemin eulérien
- simplification en chemin hamiltonien

$w(MST(I)) \leq w(OPT(I))$
$w(M_{min}) \leq w(OPT(I))$



C_1 : couplage parfait
C_2 : couplage presque parfait (à une arête près).
Le chemin OPT restreint aux sommets de degré impair du MST
$w(C_1) + w(C_2) \leq w(OPT)$
$w(M_{min}) \leq w(C_1)$
$w(M_{min}) \leq w(C_2)$
Donc $w(M_{min}) \leq 1/2w(OPT(I))$
Instance critique :



□

Exercice 6 : Dérandomisation par la méthode de l'espérance conditionnelle

Question 9. *Soient X et B deux variables aléatoires avec B à valeurs dans un ensemble fini $\mathcal{B}$. Montrer que*

$$\max_{b \in \mathcal{B}} \mathbb{E}(X \mid B = b) \geq \mathbb{E}(X).$$

Réponse : Par l'absurde : Si $\forall b\, \mathbb{E}(X \mid B = b) < \mathbb{E}(X)$

$$\begin{aligned}
\sum_{b \in B} \mathbb{E}(X \mid B = b)\, Pr(B = b) &= \sum_{b \in B} \sum_{a} \frac{a\, Pr(X = a \cap B = b)}{Pr(B = b)} Pr(B = b) \\
&= \sum_{a} a \sum_{b} Pr(X = a \cap B = b) \\
&= \sum_{a} Pr(X = a) \\
&= \mathbb{E}(X)
\end{aligned}$$

alors,

$$\underbrace{\sum_{b \in B} \mathbb{E}(X \mid B = b)\, Pr(B = b)}_{\mathbb{E}(X)} < \sum_{b \in B} \mathbb{E}(X)\, Pr(B = b) = \mathbb{E}(X) \qquad \text{contradiction !}$$

□

Nous allons utiliser ce fait pour démontrer qu'on peut dérandomiser un algorithme de type Monte-Carlo en remplaçant chaque tirage de bits aléatoire par le "meilleur choix aléatoire possible" (le b qui maximise $\mathbb{E}(X|B = b)$ dans la question précédente).

Soit $\mathcal{A}(I, \omega)$ une α-approximation randomisée pour un problème de maximisation Π ($\alpha < 1$), où I est l'instance du problème et $\omega = \omega_1 \omega_2 \cdots$ est la chaîne de bits aléatoires. Nous supposons que l'algorithme $\mathcal{A}(I, \omega)$ est de type Monte-Carlo et que son temps de calcul sur l'instance I est borné (uniformément) par un polynôme $p(|I|)$, indépendamment de la chaîne de bits aléatoires.

Soit I une instance fixée. On note X la variable aléatoire représentant la valeur de la solution (aléatoire) renvoyée par l'algorithme sur l'instance I.

On suppose que, pour tout q et pour toute suite binaire $b_1 \ldots b_q$, on sait *calculer exactement en temps polynomial* les espérances conditionnelles :

$$\mathbb{E}(X \mid \omega_q = b_q, \ldots, \omega_1 = b_1)$$

i.e. l'espérance conditionnelle de la valeur de la solution renvoyée par l'algorithme, sachant que les valeurs des q premiers bits aléatoires sont $b_1, \ldots, b_q$.

Étant données des constantes $b_1, \ldots, b_q \in \{0,1\}$, soit $b_{q+1} \in \{0,1\}$ le bit qui maximise :

$$\mathbb{E}(X \mid \omega_{q+1} = b_{q+1}, \omega_q = b_q, \ldots, \omega_1 = b_1).$$

Remarquez qu'étant donnés $b_1, \ldots, b_q$ fixés, b_{q+1} se calcule en temps polynomial.

Question 10. *Montrer que*

$$\mathbb{E}(X|\omega_{q+1} = b_{q+1}, \omega_q = b_q, \ldots, \omega_1 = b_1) \geq \mathbb{E}(X|\omega_q = b_q, \ldots, \omega_1 = b_1).$$

Réponse : Si les b_i sont définis itérativement comme les bits qui maximisent les éspérances conditionnelles.
$\mathbb{E}(X \mid w_{q+1} = b_{q+1}, w_q = b_q, \ldots, w_1 = b_1) \geq \mathbb{E}(X \mid A)$ □

Le principe de cette dérandomisation est de construire gloutonnement la chaîne de bits aléatoire au fur et à mesure de l'exécution de l'algorithme randomisé. Quand l'algorithme demande à lire le q-ème bit aléatoire, on fait gloutonnement pour b_q le meilleur choix aléatoire possible sachant les valeurs qu'on a déjà choisies pour $b_1, \ldots, b_{q-1}$.

Question 11. *Décrire précisément l'algorithme déterministe glouton ainsi obtenu et démontrer que si $\mathcal{A}$ est une α-approximation (randomisée) du problème Π, alors l'algorithme obtenu est une α-approximation (déterministe) de Π.*

Réponse : $1^{\text{ère}}$ étape, on fabrique, en temps polynomial, les bits q_i itérativement par maximisation de l'espérance conditionnelle.
$2^{\text{ème}}$ étape, on éxécute l'algo randomisé sur la chaîne des b_i obtenue.
On a alors, $w(A_{det}(I) \geq \mathbb{E}(w(A(I)))$ □

On se propose à présent d'appliquer cette méthode au problème MAX-CUT.

Étant donné un graphe G muni d'une fonction de poids w sur les arêtes, le problème MAX-CUT est de trouver un ensemble de sommets S tel que le poids total des arêtes ayant une unique extrémité dans S est maximum. On considère l'algorithme randomisé vu en cours pour le problème MAX-CUT :

```
S_0 ← ∅
Numéroter les sommets : V = {v_1, ..., v_n}
Pour i = 1 à n faire
    Si ω_i = 1 alors S_i ← S_{i-1} ∪ {v_i}
Fin pour
Renvoyer S_n
```

On utilise à partir d'ici les notations générales définies précédemment pour cet algorithme particulier.

Question 12. *Étant données des constantes $b_1, \ldots, b_q \in \{0,1\}$, démontrer qu'on peut calculer en temps polynomial :*

$$\mathbb{E}(X \mid \omega_q = b_q, \ldots, \omega_1 = b_1).$$

Indication : Exprimez $\mathbb{E}(X \mid \omega_q = b_q, \ldots, \omega_1 = b_1)$ en fonction des relations des arêtes avec les deux côtés de la coupe déjà construits.

Réponse : A chaque étape de la déterminisation de b_{q+1}, la question est :

$$\mathbb{E}(X \mid w_{q+1} = 1, w_q = b_q, \dots, w_1 = b_1) \geq \mathbb{E}(X \mid w_{q+1} = 0, w_q = b_q, \dots, w_1 = b_1) \quad ?$$

Les sommets déjà fixés sont $S_Q \subseteq \{x_1, \dots, x_q\}$.

$$\mathbb{E}(X \mid w_{q+1} =, \dots, w_1 = q_1) = \sum_{e \text{ déjà figé}} \mathbb{E}(w(e)) + \sum_{e \text{ non figé}} \mathbb{E}(w(e))$$

Mais on a $\sum_{e \text{ non figé}} \mathbb{E}(w(e)) = 1/2\ w(\{e \text{ non figé}\}$
Il suffit donc de regarder quelle valeur de w_{q+1} maximise la coupe pour les sommets déjà figés (se fait en temps polynomial).

$$\sharp(N(x_{q+1}) \cap S_q) \leq \sharp(N(x_{q+1}) \cap \overline{S_q})$$

□

Question 13. *En déduire (et analyser) une $\frac{1}{2}$-approximation gloutonne déterministe du problème MAX-CUT. Reformuler cet algorithme sous une forme naturelle.*

Réponse :

- $S = \emptyset$, $\bar{S} = \emptyset$,
- Mettre le premier sommet aléatoirement dans S ou $\bar{S}$
- Pour chaque nouveau sommet v, $w(((S \cup \{v\}) \times \bar{S}) \cap V)$ représente la valeur de la coupe en mettant v dans S. Si $w(((S \cup \{v\}) \times \bar{S}) \cap V) \geq w((S \times (\bar{S} \cup \{v\})) \cap V)$, alors mettre v dans S, sinon, mettre v dans $\bar{S}$.

A chaque étape, on a maximisé l'espérance conditionnelle. On est ainsi sur d'obtenir une solution au moins aussi meilleure que l'espérance de la version randomisé, d'où une $1/2$-approximation. □

♣