COMPLEX - Partie 2
2024-2025

Énoncé

Nom ou numéro d'anonymat :

Durée : 2 heures
Les exercices sont indépendants.
Une rédaction claire et concise sera appréciée. Toute affirmation devra être **justifiée**.
Une question non résolue n'empêche pas de faire les suivantes
(dans ce cas indiquez clairement que vous admettez le(s) résultat(s) de la question non faite).

Exercice 1 : Classe de complexité probabiliste $\mathcal{PP}$

Soit Σ un alphabet arbitraire fini (avec $\#\Sigma > 1$). Nous considérons la classe de complexité $\mathcal{PP}$ définie comme étant l'ensemble des langages L définis sur Σ pour lesquels il existe une machine de Turing probabiliste $\mathcal{M}$ telle que :

(1) $\mathcal{M}$ s'arrête sur toute entrée et s'exécute en temps polynomial ;
(2) pour tout $x \in L$, $\Pr[\mathcal{M}(x) = 1] > 1/2$;
(3) pour tout $x \notin L$, $\Pr[\mathcal{M}(x) = 0] > 1/2$;

1.a] Nous considérons la classe de complexité $\mathcal{PP}^{\geq}$ définie comme étant l'ensemble des langages L définis sur Σ pour lesquels il existe une machine de Turing probabiliste $\mathcal{M}$ qui vérifie la propriété (1) précédente et

(2') pour tout $x \in L$, $\Pr[\mathcal{M}(x) = 1] \geq 1/2$;
(3') pour tout $x \notin L$, $\Pr[\mathcal{M}(x) = 0] \geq 1/2$;

Montrer que tout langage défini sur Σ appartient à $\mathcal{PP}^{\geq}$.

1.b] Nous considérons la classe de complexité $\mathcal{PP}_{1/4}$ définie comme étant l'ensemble des langages L définis sur Σ pour lesquels il existe une machine de Turing probabiliste $\mathcal{M}$ qui vérifie la propriété (1) précédente et

(2") pour tout $x \in L$, $\Pr[\mathcal{M}(x) = 1] > 1/4$;

(3") pour tout $x \notin L$, $\Pr[\mathcal{M}(x) = 0] > 3/4$.

Montrer que $\mathcal{PP} \subseteq \mathcal{PP}_{1/4}$

Indication : Pour un langage L de $\mathcal{PP}$, on pourra considérer une machine de Turing probabiliste $\mathcal{M}$ vérifiant les propriétés (1), (2) et (3) et considérer la machine de Turing probabiliste $\mathcal{M}'$ qui

— avec probabilité 1/2, rejette son entrée x (sans faire de calcul) ;
— avec probabilité 1/2, exécute $\mathcal{M}$ sur son son entrée x et retourne la réponse de $\mathcal{M}$.

1.c] Montrer que $\mathcal{PP}_{1/4} \subseteq \mathcal{PP}$ (et donc $\mathcal{PP}_{1/4} = \mathcal{PP}$).

Indication : Pour un langage L de $\mathcal{PP}_{1/4}$, on pourra considérer une machine de Turing probabiliste $\mathcal{M}$ vérifiant les propriétés (1), (2") et (3") et considérer la machine de Turing probabiliste $\mathcal{M}'$ qui

— avec probabilité 3/8, accepte son entrée x (sans faire de calcul) ;
— avec probabilité 1/8, rejette son entrée x (sans faire de calcul) ;
— avec probabilité 1/2, exécute $\mathcal{M}$ sur son son entrée x et retourne la réponse de $\mathcal{M}$.

Exercice 2 : Problème Max2Lin(k)

Dans cet exercice, nous considérons le problème Max2Lin(k) où l'objectif est d'affecter à chaque variable x_i (pour $i \in \{1, \ldots, n\}$) une valeur dans $\{0, 1, \ldots, k-1\}$, de façon à maximiser le *poids total* des contraintes satisfaites où chaque contrainte est de la forme :

$$x_i - x_j \equiv c_{ij} \mod k$$

et est associée à un poids $w_{ij} > 0$. Le poids total de toutes les contraintes est noté W et nous notons m le nombre total de contraintes.

2.a] Considérons une instance avec $n = 3$ variables : x_1, x_2, x_3, $k = 3$ et $m = 4$ contraintes de poids 1 :

- $x_1 - x_2 \equiv 1 \mod 3$,
- $x_2 - x_3 \equiv 2 \mod 3$,
- $x_3 - x_1 \equiv 1 \mod 3$,
- $x_1 - x_2 \equiv 0 \mod 3$.

Donner un exemple d'affectation qui satisfait au moins 2 contraintes et montrer qu'il n'existe pas d'affection qui satisfait 3 contraintes.

2.b] Considérons l'algorithme probabiliste qui étant donné une instance du problème Max2Lin(k) consiste à affecter à chaque variable une valeur tirée uniformément aléatoirement dans $\{0, 1, \ldots, k-1\}$. Notons Z la variable aléatoire représentant le *poids total des contraintes satisfaites* par cette affectation aléatoire. Montrer que $\mathbb{E}[Z] = W/k$.

2.c] En déduire que pour toute instance du problème Max2Lin(k), il existe une affectation des variables qui satisfait un ensemble de contraintes de poids total supérieur ou égal à W/m

2.d] Notons $p := \Pr[Z \geq W/k]$. Montrer que $p \geq 1/k$.

2.e] En déduire, un algorithme probabiliste de type Las Vegas qui étant donné une instance du problème Max2Lin(k) retourne toujours une affectation des variables qui satisfait au moins W/k contraintes. Donner sa complexité temporelle.