Modèles de calcul, TD numéro 9 : relativisation de $P = NP$ avec des oracles
Natacha Portier
lundi 15 avril 2002

Question 1 : première relativisation en bornant l'oracle

Nous allons montrer que $P = NP$ si et seulement si pour tout oracle T sur un alphabet à une lettre on a $P^T = NP^T$.

Question 1,1

Montrer l'implication dans un sens.

Question 1,2

Soit L un langage et L_n l'ensemble des mots de L de longueur n. Quel est le cardinal maximal de L_n ?

Un langage est *creux* si le cardinal de L_n est borné par un polynôme en n. Montrer qu'un langage sur une seule lettre (*tally*) est creux.

Rappeler comment construire un oracle B tel que $P^B \neq NP^B$. Montrer que B est creux.

Question 1,3

On veut montrer la réciproque. Soit T un oracle sur l'alphabet $\{0\}$ et soit $L \in NP^T$ un langage accepté par la machine de Turing non-déterministe NT utilisant l'oracle T et travaillant en temps borné par le polynôme q. Soit x une entrée. On cherche des ensembles $O(T,x)$ et $N(T,x)$ vérifiant les conditions suivantes : s'il existe un calcul de NT sur l'entrée x qui demande si y appartient à l'oracle alors $y \in T \Rightarrow y \in O(T,x)$ et $y \notin T \Rightarrow y \in N(T,x)$.

Donner un algorithme déterministe polynomial avec oracle T qui sur l'entrée x calcule des ensembles $O(T,x)$ et $N(T,x)$.

Question 1,4

Soit L' l'ensemble des (x,O,N), parmi les (x,O,N) tels que $O \subseteq T$ et $N \subseteq T^c$, tels que la machine NT avec l'oracle O accepte x et si y est une question posée à l'oracle par NT sur un calcul de x alors $y \in O \cup N$. Montrer que L' est NP.

Question 1,5

Montrer le sens que vous n'avez pas encore montré en 1,1.

Question 2 : seconde relativisation en bornant le nombre de questions posées à l'oracle

Soit X un oracle. L'ensemble NP_b^X est la classe des langages qui peuvent être décidés par une machine de Turing non-déterministe travaillant en temps polynomial et telle que pour une entrée donnée le nombre total de questions à l'oracle que la machine peut poser sur tous les calculs possibles est borné polynomialement en fonction de la taille de l'entrée.

Nous allons montrer que $P = NP$ si et seulement si pour tout oracle X on a $P^X = NP_b^X$.

Question 2,1

Montrer l'implication dans un sens. Pour tout langage X on définit le langage $L_X = \{0^n | \exists x[x \in X \wedge |x| = n]\}$. Montrer que $L_X \in NP^X$. Est-ce que $L_X \in NP_b^X$?

Question 2,2

Soit NT une machine de Turing non-déterministe à oracle. Pour tout oracle X et toute entrée x on note $Q(NT, X, x)$ l'ensemble des mots y tels qu'il existe un calcul de $NT^X(x)$ qui pose la question y à l'oracle. Cet ensemble peut-être partitionné en deux sous-ensembles selon la réponse de l'oracle. Pour tout oracle X et toute entrée x on pose $O(NT, X, x) = Q(NT, X, x) \cap X$: c'est l'ensemble des questions auxquelles l'oracle X répond *oui*. De la même manière on définit $N(NT, X, x) = Q(NT, X, x) \cap X^c$.

Étant donnés deux langages finis O et N, un oracle X est dit *compatible* avec O et N si $O \subseteq X$ et $X \cap N = \emptyset$. Intuitivement, un oracle est compatible avec les deux ensembles si ces ensembles peuvent répondre correctement à un ensemble fini de questions.

Soit NT une machine de Turing non-déterministe à oracle et soient $(O, N, \mathcal{C})$ les triplets tels que O et N soient des ensembles finis disjoints et $\mathcal{C}$ une configuration d'un calcul de NT sur une entrée. Considérons l'ensemble L_{NT} des triplets $(O, N, \mathcal{C})$ tels que pour tout oracle X compatible avec O et N il existe un calcul de NT^X partant de la configuration $\mathcal{C}$ et posant à l'oracle une question $y \notin O \cup N$.

Soit NT une machine de Turing non-déterministe à oracle. Montrer que $L_{NT} \in NP$.

Question 2,3

En supposant $P = NP$ donner un algorithme polynomial déterministe utilisant l'oracle X qui calcule $O(NT, X, x)$ et $N(NT, X, x)$.

Question 2,4

Montrer le sens que vous n'avez pas encore montré en 2,1.