Complexité avancée
TD 11

Cristina Sirangelo - LSV, ENS-Cachan

December 10th, 2014

Exercise 1. AM with perfect correctness. We define **AM** with *perfect correctness* just as **AM** except that if a word is not in the language then, for all strategies of Merlin, the probability that Arthur rejects in the end of the game is 1 (instead of at least 2/3). **AM** with *pefect correctness* is clearly included in **AM**. There exists another well known complexity class $\mathcal{C}$ such that

$$\textbf{AM} \text{ with } \textit{perfect correctness } \subseteq \mathcal{C} \subseteq \textbf{AM}$$

Which class is $\mathcal{C}$?

Exercise 2. Collapse of the Arthur-Merlin hierarchy Recall that, for each $\Pi \in \{A, M\}^*$, the class $\mathbf{\Pi}$ is the class of languages recognized by Arthur-Merlin games with protocol Π.

(a) Without using any result about the collapse of the Arthur-Merlin hierarchy, prove that for all $\Pi_0, \Pi_1, \Pi_2 \in \{A, M\}^*$, we have $\mathbf{\Pi_1} \subseteq \mathbf{\Pi_0 \Pi_1 \Pi_2}$.

(b) Now assume the fact that for all $\Pi \in \{A, M\}^*$, one has $\mathbf{\Pi} \subseteq \mathbf{AM}$. Prove the following statement : For all $\Pi \in \{A, M\}^*$ such that Π has a strict alternation of symbols, and $|\Pi| > 2$, we have $\mathbf{\Pi} = \mathbf{AM}$.

Exercise 3. BP · NP and the polynomial hierarchy. We denote by $\mathbf{BP \cdot NP}$ the class of languages L such that there exists a polynomial time randomized Turing machine A and a language $D \in \mathbf{NP}$ such that, for all input x of size n

- If $x \in L$ then $Pr[A(x, r) \in D] \geq 1 - \frac{1}{2^n}$
- If $x \notin L$ then $Pr[A(x, r) \in D] \leq \frac{1}{2^n}$

It is easy to see that $\mathbf{BP \cdot NP} = \mathbf{AM}$. Give a direct proof that $\mathbf{BP \cdot NP} \subseteq \Sigma_3^p$ (i.e. do not use the result $\mathbf{AM} \subseteq \Pi_2^p$).

Definition (PCP). A Turing machine with direct access is a Turing machine with a special state, called the *reading state*, a reading oracle, and two special working tapes, called the *direct access tape*, and the *address tape*, respectively. The machine never reads directly the content of the direct access tape (in the sense that the normal transitions of the machine are independent of the content of the direct access tape). This tape is only accessed via the reading oracle in the following way. When the machine goes in the reading state, the content of the address tape is interpreted as the binary representation of a position i of the direct access tape. The reading oracle then provides in one step, the symbol in position i of the direct access tape. (You can assume this symbol is stored in the control state, or in a special output tape of the reading oracle.)

A $\mathbf{PCP}(R(n), Q(n), T(n))$-*verifier* is a probabilistic Turing machine with direct access to a tape called the *proof tape* over alphabet $\{0, 1\}^*$. On input x of size n and proof tape content π, the machine uses $R(n)$ random bits and works in the following three phases :

1. It first computes $Q(n)$ positions $p_1, \ldots p_{Q(n)}$ (in binary) in polynomial time in n, and with no calls to the reading oracle (i.e. these positions are only a function of x and the random tape content).
2. Then it makes $Q(n)$ calls to the reading oracle, to retrieve the symbols of the proof tape π in positions $p_1, \ldots p_{Q(n)}$.
3. Finally, it computes a boolean value (either accept or reject) in time $T(n)$ and with no calls to the reading oracle (i.e. the answer computed in this phase is only a function of x, the random tape content, and the symbols $\pi[p_1], \ldots \pi[p_{Q(n)}]$).

The class $\mathbf{PCP}(R(n), Q(n), T(n))$ is the set of languages L such that there exists a $\mathbf{PCP}(R(n), Q(n), T(n))$-verifier V such that :

– if $x \in L$, there exists a proof $\pi \in \{0,1\}^*$ such that $Pr_r[V(x, \pi, r) \text{ rejects }] = 0$;
– if $x \notin L$, then for all $\pi \in \{0,1\}^*$ $Pr_r[V(x, \pi, r) \text{ accepts }] \leq 1/2$.

Where the probability is computed over all random tape contents r of size $R(n)$.

Exercise 4. PCP and non-deterministic classes. Prove that, with $R(n) = \Omega(log\ n)$, we have $\mathbf{PCP}(R(n), Q(n), T(n)) \subseteq \mathbf{NTIME}(2^{O(R(n))} \cdot Q(n) \cdot T(n))$.

Exercise 5. PCP witnesses. Let $\mathbf{PCP}'(k_1 \cdot log\ n, Q(n), T(n))$ be defined as $\mathbf{PCP}(k_1 \cdot log\ n, Q(n), T(n))$ except that only proofs π of size $n^{k_1}Q(n)$ are considered, and addresses computed by the verifier have $log(n^{k_1}Q(n))$ bits. Prove that $\mathbf{PCP}(k_1 \cdot log\ n, Q(n), T(n)) = \mathbf{PCP}'(k_1 \cdot log\ n, Q(n), T(n))$.

Exercise 6. Prove the following statements :

$$\bigcup_{R(n),\ T(n) \text{ polynomials}} \mathbf{PCP}(R(n), 0, T(n)) = \mathbf{coRP}$$

$$\bigcup_{Q(n),\ T(n) \text{ polynomials}} \mathbf{PCP}(0, Q(n), T(n)) = \mathbf{NP}$$

$$\bigcup_{c \in \mathbb{N},\ T(n) \text{ a polynomial}} \mathbf{PCP}(0, c \cdot log\ n, T(n)) = \mathbf{P}$$