Complexité avancée - TD 10

Simon Halfon

November 23, 2016

Exercise 1: NP and BPP
Prove the following:

- if $\mathsf{P} = \mathsf{NP}$ then $\mathsf{BPP} = \mathsf{P}$.
- if $\mathsf{NP} \subseteq \mathsf{BPP}$ then $\mathsf{AM} = \mathsf{MA}$.

Exercise 2: AM with perfect soundness
Define AM_{ps} (resp. ABPP_{ps}) as AM (resp. ABPP) with perfect soundness, that is replace $1 - 1/2^{n^l}$ with 1 in the soundness condition of definiton **??**. Show that $\mathsf{AM}_{ps} = \mathsf{ABPP}_{ps} = \mathcal{C} \subseteq \mathsf{AM}$, where $\mathcal{C}$ is a known complexity class.

Exercise 3: Polynomial identity An n-variable *algebraic circuit* is a directed acyclic graph having exactly one node with out-degree zero, and exactly n nodes with in-degree zero. The latter are called *sources*, and are labelled by variables $x_1, \ldots x_n$; the former is called the *output* of the circuit. Moreover each non-source node is labelled by an operator in the set $\{+, -, \times\}$, and has in-degree two.

An algebraic circuit defines a function from $\mathbb{Z}^n$ to $\mathbb{Z}$, associating to each integer assignment of the sources the value of the output node, computed through the circuit. It is easy to show that this function can be described by a polynomial in the variables $x_1, \ldots x_n$. Algebraic circuits are indeed a form of implicit representation of multivariate polynomials. Nevertheless algebraic circuits are more compact than polynomials.

An algebraic circuit C is said to be *identically zero* if it evaluates to zero for all possible integer assignments of the sources.

The **Polynomial identity** problem is as follows:

- INPUT: An algebraic circuit C
- QUESTION: is C identically zero?

1. Justify the sentence "Algebraic circuits are more compact than polynomials".
2. Show that **Polynomial identity** is in coRP (note that it is not known whether **Polynomial identity** is in P).

   Hint: you may need the following statements

- **Schwartz-Zippel lemma** If $p(x_1, \ldots x_n)$ is a nonzero polynomial with coefficients in $\mathbb{Z}$ and total degree at most d, and $S \subseteq \mathbb{Z}$, then the number of roots of p belonging to S^n is at most $d \cdot |S|^{n-1}$.
- **Prime number theorem** There exists a known integer $X_0 \geq 0$ such that, for all integers $X \geq X_0$, the number of prime numbers in the set $[1..2^X]$ is at least $\frac{2^X}{X}$.

Definition 1 *(Multi-prover interactive protocols) Let $P_1, \ldots, P_k$ be infinitely powerful machines whose output is polynomially bounded. Let V be a probabilistic polynomial-time machine. V is called the* verifier*, and $P_1, \ldots, P_k$ are called the* provers*.*

A round of a multi-prover interactive protocol on input x consists of an exchange of messages (i.e. words over a given alphabet) between the verifier and the provers, and works as follows:

- *The verifier V is executed on an input consisting of x, the history of all previous messages exchanged with all provers (both sent and received messages), and a random tape content of size polynomial in $|x|$. The output of the verifier is computed in time polynomial in $|x|$, and consists of messages to some or all of the provers.*

- *Each message q_i sent from the verifier to prover P_i is followed by an answer a_i, of size polynomial in $|x|$, sent from the prover P_i to the verifier. The answer a_i is computed by P_i on input consisting of x and the history of all messages previously exchanged between the verifier and the prover P_i (and only P_i).*

- *Alternatively the verifier may decide not to produce messages, and terminates the protocol by either accepting or rejecting, based on the input x and the history of all previous messages exchanged with all provers.*

You can view the protocol as executed by the verifier sharing communication tapes with each P_i, where different provers P_i and P_j have no tapes they can both access, besides the input tape. In a round the verifier stores each message q_i to prover P_i on the i-th communication tape, shared between the prover and P_i. The answer of P_i is put on tape i as well. The verifier has access to the input and all communication tapes, while each prover P_i has access only to the input and tape i.

$P_1, \ldots, P_k$ *and V form a* multi-prover interactive protocol *for a language L if the execution of the protocol between V and $P_1, \ldots P_k$ terminates after a polynomial number of rounds (in the size of the input x) and:*

- *if $x \in L$, then $Pr[(V, P_1, \ldots, P_k)$ accepts $x] > 1 - 2^{-q(n)}$;*
- *if $x \notin L$, then for all provers $P'_1, \ldots, P'_k$, $\quad Pr[(V, P'_1, \ldots, P'_k)$ accepts $x] < 2^{-q(n)}$;*

where q is a polynomial and the probability is computed over all possible random choices of V.

In this case, we denote $L \in \mathbf{MIP}_k$. The number of provers k need not be fixed and may be a polynomial in the size of the input x. We say that $L \in \mathbf{MIP}$ if $L \in \mathbf{MIP}_{p(n)}$ for some polynomial p. Clearly $\mathbf{MIP}_1 = \mathbf{IP}$, but allowing more provers makes the interactive protocol model potentially more powerful.

Exercice 3: Characterization of MIP

Prove the following characterizations of the class **MIP**.

1. Let M be a probabilistic polynomial-time Turing machine with access to a function oracle. A language L is accepted by M iff:

   - if $x \in L$, then there exists an oracle O s.t. M^O accepts x with probability greater than $1 - 2^{-q(n)}$;
   - if $x \notin L$, then for any oracle O', $M^{O'}$ accepts x with probability smaller than $2^{-q(n)}$.

   Show that $L \in MIP$ if and only if L is accepted by a probabilistic polynomial time oracle machine.

2. Show that $MIP = MIP_2$.

3. Show that $MIP \subseteq \mathsf{NEXP}$.