Modèles de calcul, fiche de TD numéro 8
Pascal Koiran, 17 décembre 2004

Oracles, non-uniform complexity and the polynomial hierarchy.

1. **Sparse sets.** A language $S \subseteq \{0,1\}^*$ is said to be *sparse* ("creux", en français) if there is a polynomial p such that for all n, S has at most $p(n)$ words of length n.

   Show that S is sparse iff there is a polynomial q such that for all n, S has at most $p(n)$ words of length *at most* n.

2. We have seen two characterizations of the complexity class P/poly: in terms of advice functions, and in terms of boolean circuit families. We shall give here a third characterization in terms of sparse oracles.

   First, show that if a language L is recognized in polynomial time by a Turing machine which has access to a sparse oracle then $L \in$ P/poly.

3. Show the converse.

4. **Logarithmic advice.** The complexity class $\mathrm{P}/\log$ is defined in a similar way as P/poly: a language A belongs to this class if there exists a problem $B \in$ P, an "advice function" $f : \mathbb{N} \to \{0,1\}^*$ and a constant c such that:

   (i) $|f(n)| \leq c.\log n$ for all $n \geq 2$
   (ii) For any word x of length n, $x \in A \Leftrightarrow \langle x, f(n)\rangle \in B$.

   It turns out that logarithmic advice cannot help solve NP-complete problems in polynomial time. More precisely, $\mathrm{NP} \subseteq \mathrm{P}/\log$ iff P = NP. Prove this theorem!

   Hint: assuming that $\mathrm{NP} \subseteq \mathrm{P}/\log$, show that the algorithm of Figure 5.8 (from the book *Structural Complexity, vol. I*) solves SAT in polynomial time. In that algorithm, what does c stand for? what properties of SAT have you used?

5. **The Karp-Lipton theorem.** Can polynomial advice help solve NP-complete problems in polynomial time even though logarithmic advice cannot? This is also quite unlikely, as shown by the Karp-Lipton theorem: if NP $\subseteq$ P/poly the polynomial hierarchy collapses at the second level (i.e., $\Sigma_2 = \Pi_2$). The purpose of the next three questions is to prove that theorem.

   We say that a circuit C is s-good if, when given as input a boolean formula of size s, it decides whether that formula is satisfiable. Show that the set of pairs $\langle C, s \rangle$ such that C is s-good is in coNP.

   Hint: use again the self-reducibility of SAT. What hypothesis on the encoding of formulas do you need?

6. Assuming that NP $\subseteq$ P/poly, show that there exists a polynomial p and a circuit family (C_s) such that C_s is s-good and of size at most $p(s)$.

7. Prove the Karp-Lipton theorem. Hint: assume that NP $\subseteq$ P/poly. Since Σ_2 is closed under polynomial time many-one reducibility, it suffices to show that your favourite Π_2-complete problem is in Σ_2.

8. Bonus question: define the complexity class NP/poly, and show that NP $\subseteq$ P/poly iff P/poly = NP/poly.