Complexité avancée - TD 5

Simon Halfon

October 11, 2016

Exercise 1: On the existence of one-way functions A one-way function is a bijection f from k-bit intergers to k-bit intergers such that f is computable in polynomial time, but f^{-1} is not. Prove that if there exists one-way functions, then

$$A = \{(x, y) \mid f^{-1}(x) < y\} \in (\mathsf{NP} \cap \mathsf{coNP}) \backslash \mathsf{P}$$

Exercise 2: Closure under morphisms

Given a finite alphabet Σ, a function $f : \Sigma^* \to \Sigma^*$ is a morphism if $f(\Sigma) \subseteq \Sigma$ and for all $a = a_1 \cdots a_n \in \Sigma^*$, $f(a) = f(a_1) \cdots f(a_n)$ (f is uniquely determined by the value it takes on Σ).

1. Show that NP is closed under morphisms, that is: for any language $L \in \mathsf{NP}$, and any morphism f on the alphabet of L, $f(L) \in \mathsf{NP}$.
2. Show that if P is closed under morphisms, then $\mathsf{P} = \mathsf{NP}$.

Exercise 3: Prime Numbers

1. Show that `UNARY-PRIME` $= \{1^n \mid n$ is a prime number $\}$ is in P.
2. Show that `PRIME` $= \{p | p$ is a prime number encoded in binary $\}$ is in coNP
3. We want to prove that `PRIME` is in NP. Use the following characterization of prime numbers to formulate a non-deterministic algorithm runing in polynomial time.

   A number p is prime if and only if there exists $a \in [2, p-1]$ such that:

   (a) $a^{p-1} \equiv 1[p]$, and

   (b) for all q prime divisor of $p-1$, $a^{\frac{p-1}{q}} \not\equiv 1[p]$

   To prove that your algorithm runs in polynomial time, you can admit that all common arithmetical operations on $\mathbb{Z}/p\mathbb{Z}$ can be performed in polynomial time.

Therefore, `PRIME` is in $\mathsf{NP} \cap \mathsf{coNP}$. Actually, this problem has recently been shown in P (see the AKS algorithm).

Exercise 4: Unary Languages

1. Prove that if a unary language is NP-complete, then $\mathsf{P} = \mathsf{NP}$.
   Hint: consider a reduction from SAT to this unary language and exhibit a polynomial time recursive algorithm for SAT

2. Prove that if every unary language in NP is actually in P, then $\mathsf{EXP} = \mathsf{NEXP}$.
   Hint: remember we can always restrict our attention to Turing machines on alphabet $\{0, 1\}$.

3. Show the converse.

Exercise 5: Planar Circuit Value

Show that `CIRCUIT-VALUE` remains P-complete when the input graph is planar.

Exercise 6: Complete problems for levels of PH

Prove that the following problem Σ_k`QBF` is Σ_k^P-complete (under polynomial time reductions).

- INPUT: A quantified boolean formula $\exists X_1 \forall X_2 \exists ... Q_k X_k \phi$, where $X_1, ..X_k$ are k disjoint sets of variables, Q_k is the quantifier $\forall$ if k is even, and the quantifier $\exists$ if k is odd, ϕ is a boolean formula over variables $\bigcup_{i=1..k} X_i$;
- QUESTION: is the input formula true ?

Define a similar problem Π_k`QBF` such that Π_k`QBF` is Π_k^P-complete.

Exercise 7: Collapse of PH

1. Prove that if $\Sigma_k^P = \Pi_k^P$ for some $k \geq 1$ then $\mathsf{PH} = \Sigma_k^P$.
2. Show that if $\mathsf{P} = \mathsf{NP}$ then $\mathsf{P} = \mathsf{PH}$.
3. Prove that if $\Sigma_k^P = \Sigma_k^P + 1$ for some $k \geq 0$ then $\mathsf{PH} = \Sigma_k^P$.
4. Show that if $\mathsf{PH} = \mathsf{PSPACE}$ then PH collapses.
5. Do you think there is a polynomial time procedure to convert any QBF formula into a QBF formula with at most 10 variables ?