CS 221: Computational Complexity **Prof. Salil Vadhan**

Problem Set 5

Assigned: Wed. Apr. 7, 2010 Due: Thu. Apr. 22, 2010 (5 PM sharp)

- You must *type* your solutions. LaTeX, Microsoft Word, and plain ascii are all acceptable. Submit your solutions *via email* to `cs221-hw@seas.harvard.edu`. If you use LaTeX, please submit both the compiled file (`.pdf`) and the source (`.tex`). Please name your files `PS5-yourlastname.*`.
- Strive for clarity and conciseness in your solutions, emphasizing the main ideas over low-level details. Do not despair if you cannot solve all the problems! Difficult problems are included to stimulate your thinking and for your enjoyment, not to overwork you. *'ed problems are extra credit.

Problem 1. (Quadratic Residuosity) For a number n, the group of units modulo n is $\mathbb{Z}_n^* = \{m \in \{1, \ldots, n-1\} : \gcd(m, n) = 1\}$. The group of *quadratic residues* modulo n is $Q_n = \{m^2 \bmod n : m \in \mathbb{Z}_n^*\}$. Quadratic Residuosity is the language $\mathrm{QR} = \{(n, m) : m \in \mathbb{Q}_n\}$. There are no known polynomial-time algorithms for this problem, and indeed there are cryptographic algorithms based on its conjectured hardness.

1. Show that the following protocol is an interactive proof for Quadratic Residuosity.
   Protocol $(P, V)(n, m)$:
   (a) P finds (or gets as an auxiliary input) a number $k \in \mathbb{Z}_n^*$ such that $k^2 \bmod n = m$,
   (b) P chooses a random element $r \stackrel{R}{\leftarrow} \mathbb{Z}_n^*$, sets $s = m \cdot r^2 \bmod n$, and sends s to V.
   (c) V flips a coin $b \stackrel{R}{\leftarrow} \{0, 1\}$, and sends b to P.
   (d) If $b = 0$, P sends $t = r$ to V. If $b = 1$, P sends $t = kr$ to V.
   (e) If $b = 0$, V accepts if $(t^2 \cdot m) \bmod n = s$. If $b = 1$, V accepts if $t^2 \bmod n = s$.
2. Show that the above protocol is *zero knowledge* in the sense that when $(n, m) \in \mathrm{QR}$, everything V sees, it could have generated efficiently on its own. That is, there is a probabilistic polynomial-time "simulator" S such that when $(n, m) \in \mathrm{QR}$, the output distribution $S(n, m)$ is identical to the distribution of V's view of the protocol $(P, V)(n, m)$ (namely the triple (s, b, t)).

Problem 2. (Randomness in interactive proofs) Unlike Arora–Barak, in our definition of **IP** we allowed the prover to be randomized.

1. (The verifier's randomness is essential) Show that **IP** with deterministic verifiers collapses to **NP**. (This is shown in Arora-Barak for the case where the prover is deterministic.)

2. (The prover's randomness is inessential) Show that for every interactive proof, there is a deterministic prover strategy that is "optimal" (i.e. maximizes the verifier's acceptance probability), and in fact this strategy can be computed in polynomial space. Conclude that **IP** $\subseteq$ **PSPACE**.

Problem 3. (Random self-reducibility) A function $f : \{0,1\}^* \to \{0,1\}^*$ is *random self-reducible* under a sequence D_n of distributions (where D_n is a distribution on $\{0,1\}^n$) if there is a probabilistic polynomial-time oracle algorithm M such that for every n and every $x \in \{0,1\}^n$,

1. $M^f(x) = f(x)$, and
2. The oracle queries made by $M^f(x)$ are each distributed according to D_n.

If in addition M's oracle calls are nonadaptive, we say that f is *nonadaptively random self-reducible.*

1. Show that if f is random self-reducible under D_n and $f \notin$ **BPP**, then there is a polynomial $p(n)$ such that f is not $(1 - 1/p(n))$-easy under D_n.
2. Explain why there are #**P**-complete, **PSPACE**-complete, and **EXP**-complete problems that are randomly self-reducible under the uniform distribution U_n.
3. Show that if there were a nonadaptively random self-reducible **NP**-complete problem (under any distribution D_n), then **coNP** $\subseteq$ **prAM/poly**. The latter class is **prAM** with polynomial advice. We use the promise class rather than the language class for technical reasons that you need not worry about. (Hint: run M many times, take as advice the quantity $\Pr[D_n \in L]$.)
4. (*) Show that if **coNP** $\subseteq$ **prAM/poly**, then the **PH** collapses. Hence **NP**-complete problems cannot be random self-reducible unless **PH** collapses.

Problem 4. (Collapse of the AM hierarchy)

1. For a class **C** of promise problems, we define $\mathbf{pr\Sigma} \cdot \mathbf{C}$ to be the class of promise problems Π such that there exists a promise problem $\Pi' \in \mathbf{C}$ and a polynomial p for which

$$\begin{aligned} x \in \Pi_Y &\Rightarrow \exists y \in \{0,1\}^{p(n)} (x,y) \in \Pi'_Y \\ x \in \Pi_N &\Rightarrow \forall y \in \{0,1\}^{p(n)} (x,y) \in \Pi'_N \end{aligned}$$

   Similarly, we define $\mathbf{prBP} \cdot \mathbf{C}$ to be the class of promise problems Π such that there exists a promise problem $\Pi' \in \mathbf{C}$ and a polynomial p for which

$$\begin{aligned} x \in \Pi_Y &\Rightarrow \Pr_{y \in \{0,1\}^{p(n)}} [(x,y) \in \Pi'_Y] \geq 2/3 \\ x \in \Pi_N &\Rightarrow \Pr_{y \in \{0,1\}^{p(n)}} [(x,y) \in \Pi'_N] \geq 2/3 \end{aligned}$$

   Show that for every integer $k \geq 1$, $\mathbf{prMA}[k] = \mathbf{pr\Sigma} \cdot \mathbf{prAM}[k-1]$ and $\mathbf{prAM}[k] = \mathbf{prBP} \cdot \mathbf{prMA}[k-1]$, where $\mathbf{prMA}[0] = \mathbf{prAM}[0] = \mathbf{prP}$ (by definition).
2. Prove that **prMA** $\subseteq$ **prAM**. (Hint: First do error-reduction.)
3. Prove that for all $k \geq 2$, $\mathbf{prAM}[k] = \mathbf{prAM}$. Conclude that $\mathbf{AM}[k] = \mathbf{AM}$.
4. Where in the above parts was it important that we were working with promise problems?