CMPT 407/710 - Complexity Theory: Problem Set 4

November 20, 2017

**Note:** Please attempt all problems. You can ask me for hints, but please avoid the temptation to just search the internet for solutions! Working through the problems by yourself will help you understand the material better, and help prepare you for the quizzes.

1. Recall that $L \in \text{PCP}_{c,s}[r(n),q(n)]$ for some constants $0 \leq s < c \leq 1$ and some functions $r(n)$ and $q(n)$ if there is a probabilistic polytime verifier $V$ such that, on input of length $n$, $V$ uses $O(r(n))$ random coin flips, queries an oracle proof in at most $O(q(n))$ locations and has the properties:

   - [completeness] $x \in L \Rightarrow \exists \pi \Pr[V^x(\pi, x) = 1] \geq c$,
   - [soundness] $x \notin L \Rightarrow \forall \pi \Pr[V^x(\pi, x) = 1] \leq s$.

   Below, we shall use $\text{PCP}[r(n),q(n)]$ as a shorthand for $\text{PCP}_{1,3/4}[r(n),q(n)]$.

   (a) Prove that $\text{PCP}[\log n, \text{poly}(n)] \subseteq \text{NP}$.
   (b) Prove that if $\text{NP} \subseteq \text{PCP}[\log n, 0]$, then $\text{NP} = \text{P}$.
   (c) Prove that if $\text{NP} \subseteq \text{PCP}[o(\log n), o(\log n)]$, then $\text{NP} = \text{P}$.

   (Hint: Write the Verifier’s computation as a SAT question of size $n^{o(1)}$. Use this transformation repeatedly for a small number of steps.)

2. Using the PCP Theorem, show that, for any constant $\rho > 0$, it’s NP-hard to $\rho$-approximate MAX-CLIQUE.

   (Hint: First use the fact that, for some constant $\gamma > 0$, it’s NP-hard to $\gamma$-approximate MAX-3SAT, to argue that it’s also NP-hard to $\gamma$-approximate MAX-CLIQUE. Then boost this NP-hardness result to any constant $\rho > 0$ by using graph products defined as follows. Given a graph $G = (V,E)$ on $n$ nodes, define the $k$th power $G^k = (V^k,E')$, where the vertices of $G^k$ are all $k$-tuples $V^k$ of the vertices of $G$, and two $k$-tuples $(u_1,\ldots,u_k)$ and $(v_1,\ldots,v_k)$ in $V^k$ are connected by an edge in $G^k$ iff for every $1 \leq i \leq k$, $(u_i,v_i) \in E$ or $u_i = v_i$.)

3. **Bomb detection.** There is a package which may or may not be a bomb. You interact with the package by sending a bit $b \in \{0,1\}$, where $b = 0$ means you don’t want to perform a test on the package, and $b = 1$ means you want to perform the test. If $b = 0$, then you get back the same bit $b = 0$. If $b = 1$, and the package is not a bomb, you
also get back the same bit $b = 1$. If, however, $b = 1$, and the package is a bomb, then the bomb explodes! Is there a way to check if the package is a bomb without causing the explosion?

It’s not hard to see that classically there is no way to test for a bomb without causing the explosion! (The only way to find out something about the package is to send $b = 1$, and face the consequences.) However, quantumly, we can do much better! Here we can send a *qubit* $b$. The package which is not a bomb will return the same qubit $b$ back. The package that is a bomb will measure $b$, getting a classical bit $a \in \{0, 1\}$, and if the measured bit $a = 0$, the package returns $|0\rangle$ to us (without exploding), but if $a = 1$, then the bomb explodes!

Consider the following Bomb Detection Quantum algorithm, for a parameter $0 < \epsilon < 1$:

1. $b := |0\rangle$;
2. repeat $\pi/(2\epsilon)$ times: { rotate $b$ by angle $\epsilon$, send the resulting qubit to the package, getting back the qubit which we again denote by $b$; }
3. measure $b$ getting a classical bit $c \in \{0, 1\}$;
4. if $c = 0$ then output “BOMB” else output “NO BOMB” endif

Analyze the described quantum algorithm and argue that if the package is not a bomb, we detect this with probability close to 1, and if the package is a bomb, we detect this with probability at least about $1 - \epsilon$, without detonating the bomb.

4. **Quantum teleportation.** Alice has a qubit to send to Bob, but she only has a classical channel (i.e., she can send only classical bits). Can Alice accomplish the task? Yes, assuming that Alice and Bob had shared an EPR pair before they got separated!

Suppose $|AB\rangle$ is an EPR pair of qubits where Alice owns $|A\rangle$ and Bob owns $|B\rangle$. Let $|\Phi\rangle$ be a qubit given to Alice to be teleported to Bob.

**Alice’s algorithm:** Given the quantum register of two qubits $|\Phi A\rangle$, apply the CNOT operation (i.e., $|xy\rangle \rightarrow |x(x \oplus y)\rangle$) to both qubits, and then apply the HADAMARD operation (i.e., $|x\rangle \rightarrow \frac{|0\rangle + (-1)^x|1\rangle}{\sqrt{2}}$) to the first qubit. Finally, measure the quantum register, and send the obtained two classical bits to Bob.

**Bob’s algorithm:** Given the qubit $|B\rangle$, if the bits sent by Alice are 00, do nothing. If the bits are 01, then apply the NOT gate (i.e., $|x\rangle \rightarrow |1 - x\rangle$) to $|B\rangle$. If the bits are 10, then apply the Z gate to $|B\rangle$ (i.e., $|0\rangle \rightarrow |0\rangle$, $|1\rangle \rightarrow -|1\rangle$). Finally, if the bits are 11, then first apply the NOT gate to $|B\rangle$, and then apply the Z gate to the result.

Argue that at the end of Bob’s algorithm, Bob’s qubit is exactly $|\Phi\rangle$.

(Note that the original copy of $|\Phi\rangle$ given to Alice got destroyed in the process, so it’s indeed a teleportation! Also note that to teleport a qubit, Alice needs to send to Bob classical bits of information, and so the described teleportation is *not* faster than the speed of light!)