Quantum Cryptography (fall 2018) Dominique Unruh

Exercise Sheet 9

Out: 2018-12-07 Due: 2018-12-13

Problem 1: Universal hash functions

(a) Let S be the set of all binary $\ell \times m$-matrices. I.e., $S = \mathbb{F}_2^{\ell \times m}$. Let X be the set of all m-bit vectors. I.e., $X = \mathbb{F}_2^m$. Let $Y = \mathbb{F}_2^\ell$. Let $F : S \times X \to Y$ be defined as $F(s, x) := sx$.

Show that F is a universal hash function.

Note: You may use the fact that for any fixed $z \neq 0$, and uniformly distributed $s \in \mathbb{F}_2^{\ell \times m}$, sz is uniformly distributed on $\mathbb{F}_2^\ell$. (Bonus points if you prove that fact, too.)

Note: This was sketched in the lecture. You only get points if your proof goes beyond the sketch in the lecture in detail/rigor.

(b) **(Bonus problem)** Let $S := X := \mathbb{F}_{2^m}$ be a finite field (encoded in the standard way as an $\mathbb{F}_2$ vector space). Let $\mathit{trunc}_\ell(x)$ denote the first ℓ bits of x. Let $Y := \{0,1\}^\ell$. Let $F : S \times X \to Y$ be defined as $F(s, x) := \mathit{trunc}_\ell(sx)$.

Show that F is a universal hash function.

Note: You may use that $\mathit{trunc}_\ell(a - b) = \mathit{trunc}_\ell(a) - \mathit{trunc}_\ell(b)$. (This is immediate from the encoding of $\mathbb{F}_{2^m}$.)

Problem 2: Concrete parameters (bonus problem)

Consider the QKD scheme described in Definition 45 in the lecture notes. Theorem 5 in the lecture notes shows that the protocol is ε-secure for a certain ε that depends on the protocol parameters.

Suggest a choice of parameters such that $\varepsilon \leq 2^{-80}$ and $\ell = 256$. How many qubits are transmitted for that choice?

Note: The parameter choice should be possible! That is, you need to make sure that there is a universal hash function F and an error correcting code with the right parameters.

Note: For any integers $a, b > 0$ with $b < 2^a - 1$, there exists a so-called Reed-Solomon code with code words of length $a(2^a - 1)$, correcting $\lfloor b/2 \rfloor$ errors, and with syndrome length ab.

Note: You do not need to find an optimal solution.

Problem 3: Discrete Fourier Transform

In this problem, not that the indexes in the definition of the DFT start with 0. I.e., the top-left component of $D_N = N^{-1/2}\,((e^{2i\pi kl/N}))_{kl}$ is $N^{-1/2}\,e^{2i\pi 00/N} = 1$.

(a) Show that the $N \times N$-DFT D_N is unitary.

Hint: Show first that for some $\tilde{\omega} \in \mathbb{C}$ with $\tilde{\omega}^N = 1$ and $\tilde{\omega} \neq 1$, we have $\sum_{k=0}^{N-1} \tilde{\omega}^k = 0$. (What is $\tilde{\omega} \cdot \left(\sum_{k=0}^{N-1} \tilde{\omega}^k\right) = 0$?)

(b) Give a circuit for D_2 using only elementary gates (i.e., only gates given in the lecture notes in Sections 2 and 5).

(c) **(Bonus)** Let $N > 0$ be an integer. Let $r \in \{1, \ldots, N\}$ with $r \mid N$. Let $x_0 \in \{0, \ldots, r-1\}$. Let $|\Psi\rangle := t^{-1/2} \sum_{k=0}^{t-1} |x_0 + kr\rangle$ where δ is a normalization factor and $t := N/r$.

(If $r = \operatorname{ord} a \mid N$ for some group element a, then $|\Psi\rangle$ is the post-measurement state we have in Shor's order-finding algorithm directly before applying the DFT D_N.)

Let D_N be the $N \times N$-DFT. Let $|\Psi'\rangle := D_N|\Psi\rangle$. Consider a measurement on $|\Psi'\rangle$ in the computational basis and let γ denote the outcome. Show that $\Pr[\frac{N}{r} \text{ divides } \gamma] = 1$. (In other words, if $N \nmid \gamma r$ then $|\langle\gamma|\Psi'\rangle|^2 = 0$.)

(That is, at least in the case where $\operatorname{ord} a \mid N$, the order finding algorithm returns a multiple of $N/\operatorname{ord} a$.)

Hint: Show first that for some $\tilde{\omega} \in \mathbb{C}$ and $t \in \mathbb{N}$ with $\tilde{\omega}^t = 1$ and $\tilde{\omega} \neq 1$, we have $\sum_{k=0}^{t-1} \tilde{\omega}^k = 0$.

Note: This was sketched in the lecture. You only get points if your proof goes beyond the sketch in the lecture in detail/rigor.

Problem 4: Inverting cyclic functions

Consider a function $H : [N] \to [N]$ where $[N] := \{0, \ldots, N-1\}$. Let $H^i(x)$ denote $H(H(H(\ldots H(x)\cdots)))$ (applied i times). For the sake of this problem, we call H cyclic if there exists a value p (the period) such that for all x, $H^p(x) = H(x)$.

(a) Let $U_H|x\rangle|i\rangle|0\rangle = |x\rangle|i\rangle|H^i(x)\rangle$. Give a quantum algorithm involving U_H for finding the period of H (assuming that H is cyclic).

Note: You may assume that the DFT D_N can be implemented as a polynomial-time[1] quantum circuit. (This is, in general, not true for all N. But in the general case, you

[1] By polynomial-time, I mean that the size of the circuit is bounded by $p(\log N)$ for some polynomial p.

would be able to use an approximately solution that is only slightly more complicated than the solution needed here.)

Note: "involving U_H" means that you can apply U_H in a single runtime step.

(b) Given $y = H(x)$ and given the period of p, show that you can find x in polynomial-time. (You may still use U_H.)

(c) The following statement is wrong:

> Given a cyclic H and a value $y \in \operatorname{range} H$, using the algorithm from (a), we can find the period p of H, and then using the algorithm from (b), we can compute $H^{-1}(y)$.[2] Moreover, all involved algorithms run in polynomial-time. Hence using quantum computers, cyclic functions can be inverted in polynomial-time.

Why?

[2] Notice that cyclicity implies bijectivity, so H^{-1} is well-defined.