Supplementary to Lecture 11

Richard Cleve

In Lecture 11, there is a step where it is claimed that the mapping

$$|x, b\rangle \mapsto |x, a^x b \bmod m\rangle \tag{1}$$

can be computed with $O(n^2 \log n)$ elementary gates. The justification is that there is a classical algorithm for efficently computing $a^x b \bmod m$. But there is a loose end.

Based on Lecture 5 (slides 10, 12) and Lecture 8 (slides 15, 16), what we can efficiently compute is the mapping

$$|x, b, c\rangle \mapsto |x, b, (a^x b \bmod m) \oplus c\rangle \tag{2}$$

But the mappings in Eqns. (1) and (2) are not the same.

There are two remedies.

Bypassing

Instead of computing mapping (1) with target state $|x\rangle|00\ldots01\rangle$, we can compute mapping (2) with target state $|x\rangle|00\ldots01\rangle|00\ldots00\rangle$. Note that, for the phase estimation circuit, the first and third resisters are in exactly the same state in both cases (and the second register is in a fixed state so it can be ignored).

This makes the order-finding algorithm work, but then the actual circuit is slightly different from the one in the lecture notes.

Actually compute the multiplicity-controlled-U

This can be done and enables us to use the circuit in the lecture notes. But it needs an additional idea:

> Suppose that $f : \{0,1\}^n \to \{0,1\}^n$ is a bijection and there is an efficient classical algorithm for computing f and there there is also an efficient classical algorithm for computing f^{-1}. Then there is an efficient quantum circuit that computes the mapping $|x\rangle \mapsto |f(z)\rangle$. The method actually uses an ancilla, so it really computes $|x\rangle|00\ldots0\rangle \mapsto |f(z)\rangle|00\ldots0\rangle$ (but that's OK).

This result (which is not proven here) can be applied to efficiently compute the mapping (1). The idea is to set $f(x, b) = (x, a^x b \bmod m)$ (which is a bijection) and note that $f^{-1}(x, b) = (x, a^{-x} b \bmod m)$. Both f and f^{-1} can be computed efficiently by classical algorithms. So this fits the above framework.