Quantum Computing: An Applied Approach

Chapter 8 Problems:
The Canon: Code Walkthroughs

1. Define the circuit model and query model of quantum computing.

2. Prove the query complexity is a lower bound on circuit (i.e., gate) complexity.

3. How many single qubit gates are needed to implement a Quantum Fourier Transform (QFT) on n qubits? How many single qubits gates are needed?

4. Draw the circuit for the Quantum Fourier Transform on $n = 1, 2$ and 3 qubits.

5. Prove that the Quantum Fourier Transform on the all zero state $|0\rangle^{\otimes n}$ is equivalent to acting with Hadamards. That is, prove that

$$\mathrm{QFT}_n|0\rangle^{\otimes n} = H^{\otimes n}|0\rangle^{\otimes n} \tag{1}$$

Literature questions

6. In their paper, *How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits*, Gidney and Ekera state that:

> The modular exponentiation in Shor's algorithm is performed over a superposition of exponents, meaning a quantum computer is required, and quantum hardware is expected to be many orders of magnitude noisier than classical hardware.

(a) How does the approach of period finding of Ekera and Hastad differ from the original algorithm by Shor and why does this modification improve the implementation?

(b) what is the big-O order of Shor's algorithm and which step of the algorithm dominates?

(c) How does the square and multiply approach convert modular exponentiation to a series of modular multiplications?

(d) How does windowed arithmetic help to reduce the computational load of this implementation of Shor's? Describe the circuit diagram in the figure below and how these sequences of operators realize this implementation.

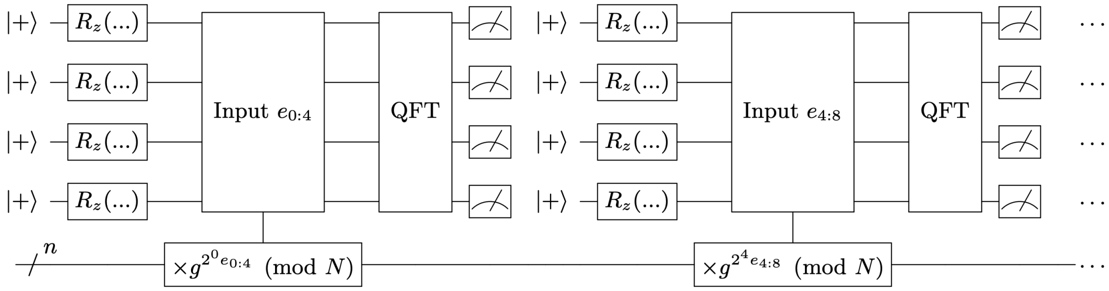
$|+\rangle$
$R_z(...)$
Input $e_{0:4}$
QFT
Input $e_{4:8}$
QFT
n
$\times g^{2^0 e_{0:4}}$ (mod N)
$\times g^{2^4 e_{4:8}}$ (mod N)
$\cdots$

Figure 1: Circuit Diagram. Source: https://arxiv.org/pdf/1905.09749.pdf

(e) What are the implications of this paper as it relates to the number of fault-tolerant qubits needed to factor RSA keys? By what factor does the method described reduce the floor of qubits needed for factoring a 2048-bit key?

7. In https://arxiv.org/pdf/quant-ph/9601018.pdf, the authors present an algorithm for the *approximate* quantum Fourier transform (AQFT). The idea is that the QFT on n qubits requires rotation gates with angles

$$\theta_{jk} := \pi/2^{(j-k)} \tag{2}$$

for integers $j > k = 1, 2, ..., n-1$.

The idea of the AQFT is to *not* implement rotations with very small angles, as these have very small effects on the resultant state. Specifically, we let the m-AQFT on $n > m$ qubits avoid implementing any rotations θ_{jk} where $j - k > m$.

(a) Show that the number of gates for the m-AQFT on n qubits is $O(nm)$. More precisely, show that the number of gates is exactly

$$n + (2n - m)(m - 1)/2$$

.