WiDS QRL - Quantum Fourier Transform and Applications

Nilabha Saha

January 2023

1 Quantum Fourier Transform

The discrete Fourier transform takes as input a vector of complex numbers, $x_0, \ldots, x_{N-1}$ where the length N of the vector is a fixed parameter. It outputs the transformed data, a vector of complex numbers $y_0, \ldots, y_{N-1}$, defined by

$$y_k \equiv \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi ijk/N}$$

The quantum Fourier transform on an orthogonal basis $|0\rangle, \ldots, |N-1\rangle$ is defined to be a linear operator with the following action on the basis states,

$$|j\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi ijk/N} |k\rangle$$

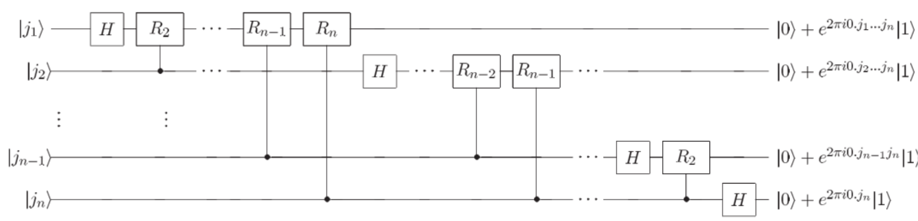
Equivalently, the action on an arbitrary state may be written

$$\sum_{j=0}^{N-1} x_j |j\rangle \rightarrow \sum_{k=0}^{N-1} y_k |k\rangle$$

where the amplitudes y_k are the discrete Fourier transform on the amplitudes x_j. This is a unitary transformation.
If we take $N = 2^n$,where n is some integer, and the basis $|0\rangle, \ldots, |2^n - 1\rangle$ is the computational basis for any n qubit quantum computer, we can write the state $|j\rangle$ using the binary representation $j = j_1 j_2 \ldots j_n$. More formally, $j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0$. We also conveniently adopt the notation $0.j_l j_{l+1} \ldots j_m$ to represent the binary fraction $j_l/2 + j_{l+1}/4 + \cdots + j_m/2^{m-l+1}$. The quantum Fourier transform can be shown to give the following useful product representation:

$$|j_1, \ldots, j_n\rangle \rightarrow \frac{\left(|0\rangle + e^{2\pi i 0.j_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.j_{n-1}j_n}|1\rangle\right) \cdots \left(|0\rangle + e^{2\pi i 0.j_1 j_2 \cdots j_n}|1\rangle\right)}{2^{n/2}}$$

A circuit implementing the quantum Fourier transformation is the following:



2 Quantum Phase Estimation

Inputs:

1. A black box which performs a controlled-U^j operation, for integer j
2. An eigenstate $|u\rangle$ of U with eigenvalue $e^{2\pi i\phi u}$
3. $t = n + \lceil \log{(2 + \frac{1}{2\epsilon})} \rceil$ qubits initialised to $|0\rangle$

Outputs: An n-bit approximation $\tilde{\phi_u}$ to ϕ_u
Runtime: $O(t^2)$ operations and one call to controlled-U^j black box. Succeeds with probability at least $1 - \epsilon$.
Procedure:

- initial state
$$|0\rangle|u\rangle$$
- create superposition
$$\frac{1}{\sqrt{2^t}}\sum_{j=0}^{2^t-1}|j\rangle|u\rangle$$
- apply black box, result of black box
$$\frac{1}{\sqrt{2^t}}\sum_{j=0}^{2^t-1}|j\rangle U^j|u\rangle$$
$$=\frac{1}{\sqrt{2^t}}\sum_{j=0}^{2^t-1}e^{2\pi i j\phi_u}|j\rangle|u\rangle$$
- apply inverse Fourier transform
$$|\tilde{\phi_u}\rangle|u\rangle$$
- measure first register
$$\tilde{\phi_u}$$

3 Quantum Order Finding

Inputs:

- A black box $U_{x,N}$ which performs the transformation $|j\rangle|k\rangle \to |j\rangle|x^j k \pmod N\rangle$, for x co-prime to the L-bit number N
- $t = 2L + 1 + \lceil \log{(2 + \frac{1}{2\epsilon})} \rceil$ qubits intialised to $|0\rangle$
- L qubits initialised to the state $|1\rangle$

Outputs: The least integer $r > 0$ such that $x^r = 1 \pmod N$
Runtime :$O(L^3)$ operations. Succeeds with probability $O(1)$
Procedure:

- initial state

$$|0\rangle|1\rangle$$

- create superposition

$$\frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle|1\rangle$$

- apply $U_{x,N}$

$$\frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle|x^j \pmod N\rangle$$
$$\approx \frac{1}{\sqrt{r2^t}} \sum_{s=0}^{r-1} \sum_{j=0}^{2^t-1} e^{2\pi isj/r} |j\rangle|u_s\rangle$$

- apply inverse Fourier transform to first register

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |\widetilde{s/r}\rangle|u_s\rangle$$

- measure first register

$$\widetilde{s/r}$$

- apply continued fractions algorithm

$$r$$

4 Reduction of Factoring to Order Finding

Inputs: A composite number N
Outputs: A non-trivial factor of N
Runtime: $O((\log N)^3)$ operations. Succeeds with probability $O(1)$
Procedure:

1. If N is even, return the factor 2.
2. Determine whether $N = a^b$ for integer $a \geq 1$ and $b \geq 2$, and if so return the factor a.
3. Randomly choose x in the range 1 to $N-1$. If $\gcd(x, N) > 1$ then return the factor $\gcd(x, N)$.
4. Use the order-finding subroutine to find the order r of x modulo N.
5. If r is even and $x^{r/2} \neq -1 \pmod N$ then compute $\gcd(x^{r/2}-1, N)$ and $\gcd(x^{r/2}+1, N)$, and test to see if one of these is a non-trivial factor, returning that factor if so. Otherwise, the algorithm fails.

5 Period Finding

Inputs:

1. A black box which performs the operation $U|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$
2. A state to store the function evaluation, initialised to $|0\rangle$
3. $t = O(L + \log(1/\epsilon))$ qubits initialised to $|0\rangle$

Outputs: The least integer $r > 0$ such that $f(x+r) = f(x)$
Runtime: One use of U, and $O(L^2)$ operations. Succeeds with probability $O(1)$
Procedure:

1. initial state
$$|0\rangle|0\rangle$$
2. create superposition
$$\frac{1}{\sqrt{2^t}} \sum_{x=0}^{2^t-1} |x\rangle|0\rangle$$
3. apply U
$$\begin{aligned} &\frac{1}{\sqrt{2^t}} \sum_{x=0}^{2^t-1} |x\rangle|f(x)\rangle \\ \approx &\frac{1}{\sqrt{r2^t}} \sum_{l=0}^{r-1} \sum_{x=0}^{2^t-1} e^{2\pi i l x/r} |x\rangle|\hat{f}(l)\rangle \end{aligned}$$

4. apply inverse Fourier transform to first register

$$\frac{1}{\sqrt{r}}\sum_{l=0}^{r-1}|\widetilde{l/r}\rangle|\hat{f}(l)\rangle$$

5. measure first register

$$\widetilde{l/r}$$

6. apply continued fractions algorithm

$$r$$

6 Discrete Logarithm

Inputs:

1. A black box which performs the operation $U|x_1\rangle|x_2\rangle|y\rangle = |x_1\rangle|x_2\rangle|y \oplus f(x_1,x_2)\rangle$, for $f(x_1,x_2) = b^{x_1}a^{x_2}$
2. A state to store the function evaluation, initialised to $|0\rangle$
3. Two $t = O(\lceil \log r \rceil + \log(1/\epsilon))$ qubit registers initialised to $|0\rangle$

Outputs: The least positive integer s such that $a^s = b$
Runtime: One use of U, and $O(\lceil \log r \rceil^2)$ operations. Succeeds with probability $O(1)$
Procedure:

1. initial state

$$|0\rangle|0\rangle|0\rangle$$

2. create superposition

$$\frac{1}{2^t}\sum_{x_1=0}^{2^t-1}\sum_{x_2=0}^{2^t-1}|x_1\rangle|x_2\rangle|0\rangle$$

3. apply U

$$\begin{aligned}
&\frac{1}{2^t}\sum_{x_1=0}^{2^t-1}\sum_{x_2=0}^{2^t-1}|x_1\rangle|x_2\rangle|f(x_1,x_2)\rangle \\
\approx&\frac{1}{2^t\sqrt{r}}\sum_{l_2=0}^{r-1}\sum_{x_1=0}^{2^t-1}\sum_{x_2=0}^{2^t-1}e^{2\pi i(sl_2x_1+l_2x_2)/r}|x_1\rangle|x_2\rangle|\hat{f}(sl_2,l_2)\rangle \\
=&\frac{1}{2^t\sqrt{r}}\sum_{l_2=0}^{r-1}\left[\sum_{x_1=0}^{2^t-1}e^{2\pi i(sl_2x_1)}|x_1\rangle\right]\left[\sum_{x_2=0}^{2^t-1}e^{2\pi i(l_2x_2)/r}|x_2\rangle\right]|\hat{f}(sl_2,l_2)\rangle
\end{aligned}$$

4. apply inverse Fourier transform to first two registers

$$\frac{1}{\sqrt{r}}\sum_{l_2=0}^{r-1}|\widetilde{sl_2/r}\rangle|\widetilde{l_2/r}\rangle|\hat{f}(sl_2,l_2)\rangle$$

5. measure first two registers

$$\left(\widetilde{sl_2/r},\widetilde{l_2/r}\right)$$

6. apply generalised continued fractions algorithm

$$s$$