Lecture 6: Quantum Fourier Transform (QFT)

Let's review the Hadamard gate (again!)

\[ H\left|x\right> = \frac{1}{\sqrt{2}} \left( \left|0\right> + (-1)^x \left|1\right> \right) \]

or we can write that as

\[ H\left|x\right> = \frac{1}{\sqrt{2}} \sum_{y=0}^{1} (-1)^{x\cdot y} \left|y\right> \]

Now let's look at a state \( \left|1x\right> \) made out of \( n \) qubits. So you can think of \( \left|1x\right> = \left|x_1, x_2, \ldots, x_n\right> \), then if we apply a Hadamard to all qubits, our state would turn into:

\[ H^{\otimes n} \left|1x\right> = \frac{1}{\sqrt{2^n}} \sum_{y=0}^{1} (-1)^{x\cdot y} \left|y\right> \]

**Fourier Transform:**

One of the most useful tools in mathematics is the Fourier Transform, which allows us to transform problems from one space to another that are easier to solve.

**Ex: Operations on polynomials**

<table>
<thead>
<tr>
<th>operation</th>
<th>representation</th>
<th>samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>evaluation</td>
<td>(O(N))</td>
<td>(O(N^2))</td>
</tr>
<tr>
<td>addition</td>
<td>(O(N))</td>
<td>(O(N))</td>
</tr>
<tr>
<td>multiplica</td>
<td>(O(N^2))</td>
<td>(O(N))</td>
</tr>
</tbody>
</table>

We can transform the problem between coefficient and sample representations using the Fast Fourier Transform (FFT) in \(O(N \log N)\).
The discrete Fourier transform takes a vector of complex numbers, 
\[ x_0, \ldots, x_{N-1} \] of length \( N \) and maps it to a vector of length \( N \).

\[ y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi i j k / N}. \]

The quantum Fourier transform on an orthonormal basis \( \ket{0}, \ldots, \ket{N-1} \) is defined to be a linear operator, and acts as follows:

\[ \ket{j} \rightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i jk / N} \ket{k}. \]

\[ \sum_{j=0}^{N-1} x_j \ket{j} \rightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} y_k \ket{k}. \]

You can think of \( U_{\text{QFT}} \) as \( \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{2\pi i jk / N} \ket{k} \bra{j} \)

better for the general state

\[ \ket{j_1, j_2, \ldots, j_n} \rightarrow \frac{(10 + e^{2\pi i 0} \ket{11}) (10 + e^{2\pi i 0} \ket{11}) \ldots (10 + e^{2\pi i 0} \ket{11})}{2^{n/2}} \]

Here \( 0, j_1, j_2, \ldots, j_n \) is the decimal binary representation.

\[ 0, j_1, j_2, \ldots, j_n = j_{2^0} + j_{2^1} 2 + \ldots + j_{2^n} 2^n \]
We start with state $|j_1 \ldots j_n\rangle$. After we apply $H$ to the first qubit, we get

$$\frac{1}{2^{\frac{1}{2}}} (|0\rangle + e^{2\pi i j_1 11} |1\rangle) |j_2 \ldots j_n\rangle$$

After applying the controlled-$R_2$ gate, the result state would be

$$\frac{1}{2^{\frac{1}{2}}} (|0\rangle + e^{2\pi i j_1 0 \cdot j_2 11} |1\rangle) |j_2 \ldots j_n\rangle$$

we continue by applying $R_3, R_4, \ldots, R_n$ to the first qubit on the output; would be

$$\frac{1}{2^{\frac{1}{2}}} (|0\rangle + e^{2\pi i j_1 0 \cdot j_2 \ldots \cdot j_n 11}) |j_2 \ldots j_n\rangle$$

on the first line of circuit.

After applying the Hadamard and $R_2$ through $R_{n-1}$ to the second qubit we get

$$\frac{1}{2^{\frac{1}{2}}} (|0\rangle + e^{2\pi i j_1 0 \cdot j_2 11} |1\rangle (|0\rangle + e^{2\pi i j_1 0 \cdot j_3 \ldots \cdot j_n 11} |j_3 \ldots j_n\rangle$$

Now it's kinda easy to see how the final result looks like

$$\frac{1}{2^{\frac{1}{2}}} (|0\rangle + e^{2\pi i j_1 0 \cdot j_2 \ldots \cdot j_n 11}) \ldots (|0\rangle + e^{2\pi i j_1 0 \cdot j_n 11})$$

How many gates do we need? $\sum_{k=1}^{n} k = \frac{n(n+1)}{2} = O(n^2)$

QFT runtime: $O(n^2)$

FFT runtime: $O(N \log N) = O(n^2)$ note $N=2^n$
\[ |\mathbf{j} \rangle = \frac{1}{2^{n/2}} \sum_{k_0}^{2^n-1} \sum_{k_1=0}^{2^{n-1}-1} \ldots \sum_{k_{n-1}=0}^{2^{1}-1} e^{2\pi i \mathbf{j} \cdot \mathbf{k}} |\mathbf{k} \rangle \]

\[ = \frac{1}{2^{n/2}} \sum_{k_0}^{2^n-1} \sum_{k_1=0}^{2^{n-1}-1} \ldots \sum_{k_{n-1}=0}^{2^{1}-1} e^{2\pi i \mathbf{j} \cdot \mathbf{k}} |\mathbf{k} \rangle \]

\[ = \frac{1}{2^{n/2}} \sum_{k_0}^{2^n-1} \sum_{k_1=0}^{2^{n-1}-1} \ldots \sum_{k_{n-1}=0}^{2^{1}-1} e^{2\pi i \mathbf{j} \cdot \mathbf{k}} |\mathbf{k} \rangle \]

\[ = \frac{1}{2^{n/2}} \sum_{k_0}^{2^n-1} \sum_{k_1=0}^{2^{n-1}-1} \ldots \sum_{k_{n-1}=0}^{2^{1}-1} e^{2\pi i \mathbf{j} \cdot \mathbf{k}} |\mathbf{k} \rangle \]

\[ = \frac{1}{2^{n/2}} \sum_{k_0}^{2^n-1} \sum_{k_1=0}^{2^{n-1}-1} \ldots \sum_{k_{n-1}=0}^{2^{1}-1} e^{2\pi i \mathbf{j} \cdot \mathbf{k}} |\mathbf{k} \rangle \]

\[ = \frac{1}{2^{n/2}} \sum_{k_0}^{2^n-1} \sum_{k_1=0}^{2^{n-1}-1} \ldots \sum_{k_{n-1}=0}^{2^{1}-1} e^{2\pi i \mathbf{j} \cdot \mathbf{k}} |\mathbf{k} \rangle \]

\[ = \frac{1}{2^{n/2}} \sum_{k_0}^{2^n-1} \sum_{k_1=0}^{2^{n-1}-1} \ldots \sum_{k_{n-1}=0}^{2^{1}-1} e^{2\pi i \mathbf{j} \cdot \mathbf{k}} |\mathbf{k} \rangle \]

Now let's construct a quantum circuit for the QFT.

First, let's construct a gate \( R_k = [1 \quad 0] \quad [0 \quad e^{2\pi i k/2}] \)

\( R_k |0\rangle = |0\rangle \)

\( R_k |1\rangle = e^{2\pi i k/2} |1\rangle \)

The circuit looks like this:
This circuit allows us to estimate $\theta$ to $t$ decimal places
think of $\varphi_1, \varphi_2, \varphi_3, \ldots \varphi_t$

So the output state would be

$$\frac{1}{2^{t/2}} \left( |10\rangle + e^{2\pi i 0 \cdot \varphi_1} |11\rangle \right) \left( |10\rangle + e^{2\pi i 0 \cdot \varphi_2} |11\rangle \right) \ldots \left( |10\rangle + e^{2\pi i 0 \cdot \varphi_t} |11\rangle \right)$$

Quantum

looks familiar? this looks like the result of the Fourier transform
of $|1\varphi_1, \varphi_2, \ldots, \varphi_t\rangle$

So at this point, we just have to apply the inverse QFT to
the output of the circuit. (the first $t$ register)

$$\frac{1}{2^{t/2}} \sum_{j=0}^{2^t} e^{2\pi i j\theta} |j\rangle |u\rangle \rightarrow |1\theta\rangle |u\rangle$$

where $|1\theta\rangle$ is an estimate of $\theta$ to $t$ decimal places.
Simon's Algorithm

Suppose we have a black box for computing 2-to-1 function \( f \):

for all \( x \neq y \), \( f(x) = f(y) \) iff \( x \cdot y = 0 \) or \( y = S \cdot x \)

\[
f(x) = \begin{cases} 
001 & \text{if } x = 000 \text{ or } 011 \\
010 & \text{if } x = 001 \text{ or } 010 \\
100 & \text{if } x = 111 \text{ or } 100 \\
111 & \text{if } x = 110 \text{ or } 101 
\end{cases}
\]

Classically: if we have \( n \) bits, have to check \( 2^n / 2 = O(2^n) \)

Quantum:

Let's define \( U_f \) as the following operator:

\[
\begin{array}{c}
|1x\rangle \\
|0\rangle
\end{array} \xrightarrow{U_f} \begin{array}{c}
|1x\rangle \\
|f(x)\rangle
\end{array}
\]

Step 1: Create a superposition from the first series of qubits

\[
|10\rangle \xrightarrow{H^n} \sqrt{\frac{1}{2^n}} \sum_{x=0}^{2^n-1} |1x\rangle
\]

Then apply \( U_f \):

\[
\sqrt{\frac{1}{2^n}} \sum_{x=0}^{2^n-1} |1x\rangle \xrightarrow{U_f} \sqrt{\frac{1}{2^n}} \sum_{x=0}^{2^n-1} |1f(x)\rangle
\]

Now let's say I make a measurement on the second set of registers and get \( f(x) \) as an output. What are the possible values of \( x \)?

\( x_0 \) and \( x_0 \oplus S \)
So the first register is in an equal superposition of
\[ \frac{1}{\sqrt{2}} |x_0\rangle + \frac{1}{\sqrt{2}} |x_0 \oplus s\rangle \]

Now let's send this through another set of Hadamards

\[
\frac{1}{\sqrt{2}} |x_0\rangle + |x_0 \oplus s\rangle \]

\[
= \frac{1}{\sqrt{2}} \sum_i (-1)^{x_0} (-1)^{x_0 \oplus s} |i\rangle
\]

if \( s \cdot y = 0 \) the coeff infront of \( |y\rangle \) is \( \pm \frac{1}{\sqrt{2}} \)

if \( s \cdot y = 1 \) is 0

we need to find \( n \) of these \( y \)'s to be able to classically solve for \( s \), which takes \( O(2^n) \)