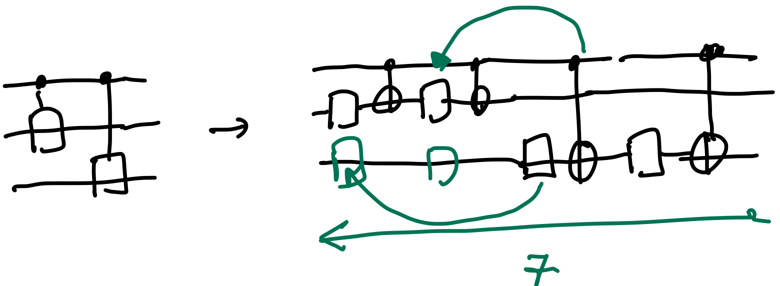
CPEN 400Q Lecture 13
The quantum Fourier transform (QFT)

Monday 26 February 2024

Announcements

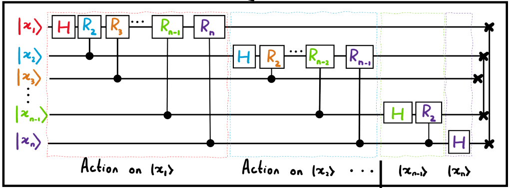
- Quiz 5 today
- Literacy assignment 2 due tonight at 23:59
- Missing project group / stray students, e-mail me ASAP
- First project peer assessment survey this week



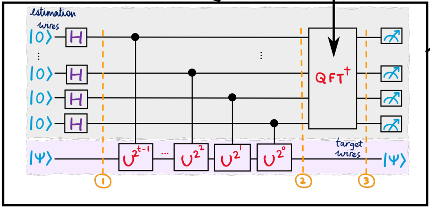
Where are we going?

$O(2^n) \rightarrow O(\sqrt{2^n})$

1. QFT



2. QPE



3. Shor

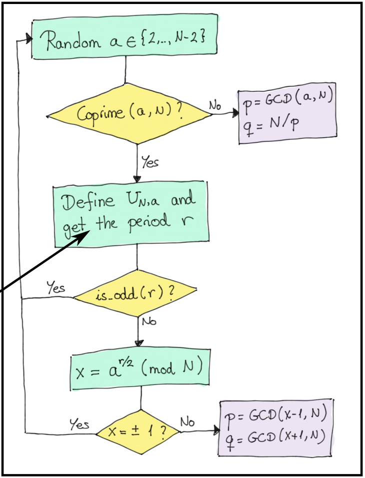


Image credit: Xanadu Quantum Codebook nodes F.3, P.3, S.4

Today

Learning outcomes:

- Express floating-point values in fractional binary representation
- Describe the behaviour of the quantum Fourier transform
- Implement the quantum Fourier transform in PennyLane

From ELEC 221[1]:

$$DFT = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \bar{\omega} & \bar{\omega}^2 & \cdots & \bar{\omega}^{N-1} \\ 1 & \bar{\omega}^2 & \bar{\omega}^4 & \cdots & \bar{\omega}^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \bar{\omega}^{N-1} & \bar{\omega}^{2(N-1)} & \cdots & \bar{\omega}^{(N-1)(N-1)} \end{pmatrix}$$

where $\bar{\omega} = e^{-2\pi i/N}$.

[1] See Lecture 13: `https://github.com/glassnotes/ELEC-221`

The DFT and FFT (which implements it efficiently) convert between time and frequency domains in digital signal processing.

Given a signal $x[n]$, the DFT computes

$$X[k] = \sum_{n=0}^{N-1} e^{-\frac{2\pi i kn}{N}} x[n] = \sum_{n=0}^{N-1} \bar{\omega}^{nk} x[n]$$

$$X[k] = \sum_{n=0}^{N-1} e^{-\frac{2\pi i kn}{N}} x[n] = \sum_{n=0}^{N-1} \bar{\omega}^{nk} x[n] \rightarrow \text{DFT}$$

The inverse DFT computes

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} e^{\frac{2\pi i kn}{N}} X[k] = \frac{1}{N} \sum_{k=0}^{N-1} \omega^{nk} X[k]$$

where $\omega = e^{2\pi i/N} = \bar{\omega}^{-1}$

The DFT is unitary (up to a prefactor).

The quantum Fourier transform (QFT) is the quantum analog of the **inverse DFT**.

$$QFT = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \omega^{jk} |k\rangle\langle j| \quad , \quad \omega = e^{\frac{2\pi i}{N}}$$

$$|k\rangle\langle j| = \begin{pmatrix} 0 & \cdots & & 0 \\ \vdots & & 1 & \vdots \\ 0 & & & 0 \end{pmatrix}$$

$N = 2^n$, n = num qubits

$$|0\rangle\langle 0| = \begin{pmatrix} 1 & 0 & \cdots \\ 0 & & \\ \vdots & & \end{pmatrix}$$

$$|0\rangle\langle 1| = \begin{pmatrix} 0 & 1 & \cdots & 0 \\ 0 & & & \\ \vdots & & & \end{pmatrix}$$

Apply to n-qubit basis state $|x\rangle$

$$QFT|x\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \omega^{jk} |k\rangle\underbrace{\langle j|x\rangle}$$

$$= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} e^{\frac{2\pi i k n}{N}} X[k]$$

As a matrix, it looks a lot like the iDFT:

$$QFT = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{pmatrix}$$

How do we *synthesize* a circuit for it?

$n=1$ $\qquad$ $\omega = e^{\frac{2\pi i}{N}}$, $N=2^n$

QFT_1

$$\frac{1}{\sqrt{N}}\begin{pmatrix} 1 & 1 \\ 1 & \omega \end{pmatrix}$$

Start with special cases $n = 1$ ($N = 2$).

Here, $e^{2\pi i/2} = e^{i\pi} = -1$, so $N = 2^1 = 2$

$$QFT_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = H$$

$n = 2$ ($N = 4$).

$$\mathrm{QFT}_2 = \frac{1}{\sqrt{2^2}} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 \\ 1 & \omega^2 & \omega^4 & \omega^6 \\ 1 & \omega^3 & \omega^6 & \omega^9 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

2 Hadamards

Here $\omega = i$, and $\omega^2 = -1$, $\omega^4 = 1$

i: Y, S, T

If we apply a SWAP, familiar things show up...

$$\frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \longrightarrow \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \\ 1 & -i & -1 & i \end{pmatrix}$$

$$= \frac{1}{2}\begin{pmatrix} \propto H & \propto H \\ \propto HS & \underset{\substack{\uparrow \\ -HS}}{\propto} HS \end{pmatrix}$$

$$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & i \\ 1 & -i \end{pmatrix} = HS$$

Quantum Fourier transform

If we apply a SWAP, familiar things show up...

Top blocks are H, bottom are HS. The following circuit implements this QFT:

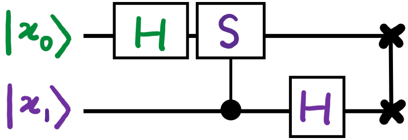
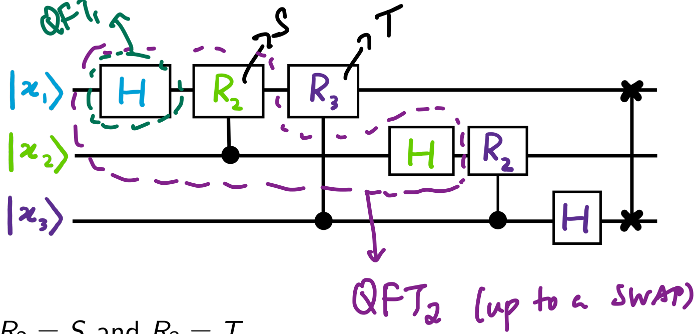


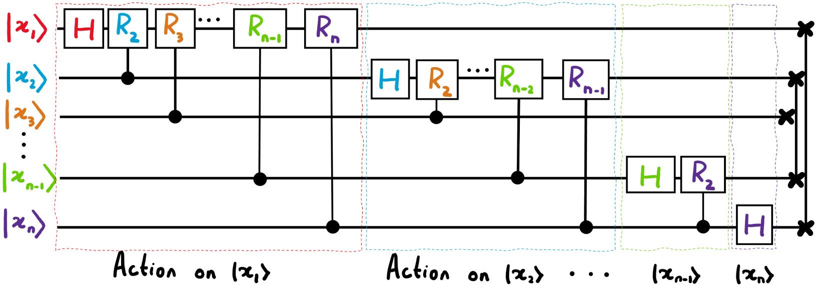
Image credit: Xanadu Quantum Codebook node F.2

Circuit for $n = 3$ ($N = 8$):



Here, $R_2 = S$ and $R_3 = T$.

Image credit: Xanadu Quantum Codebook node F.3



We will derive this by reverse-engineering the analytical definition,

$$|x\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle$$

Image credit: Xanadu Quantum Codebook node F.3

A circuit for the QFT

$$|x\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle$$

Here x and k are integers, which have binary equivalents $|x\rangle = |x_1 \cdots x_n\rangle$, $|k\rangle = |k_1 \cdots k_n\rangle$:

$$x = 2^{n-1} x_1 + 2^{n-2} x_2 + \cdots + 2 x_{n-1} + x_n$$

and similarly for k.

We are working with

$$\omega^{xk} = e^{2\pi i x(k/N)}$$

$$\omega = e^{\frac{2\pi i}{N}}$$

with $N = 2^n$.

We can write a fraction $k/2^n$ in a 'decimal version' of binary:

$$\frac{k}{2^n} = 0.k_1 k_2 \cdots k_n$$

$$= 2^{-1} k_1 + 2^{-2} k_2 + \cdots + 2^{-n} k_n$$

$$= \frac{k_1}{2} + \frac{k_2}{2^2} + \cdots + \frac{k_n}{2^n}$$

$$= \sum_{l=1}^{N} \frac{k_l}{2^l}$$

Exercise: let $k = 0.11010$. What is the numerical value of k?

$$\frac{13}{16} = 0.8125$$

$$k = \frac{k_1}{2} + \frac{k_2}{2^2} + \cdots$$

$$= \frac{1}{2} + \frac{1}{4} + \frac{0}{8} + \frac{1}{16} + \frac{0}{32}$$

$$= \frac{13}{16}$$

We will reexpress k/N in fractional binary notation, then reshuffle and *factor* the output state to uncover the circuit structure.

$$|x\rangle \to \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle \qquad k = k_1 k_2 \cdots k_n$$

$$= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i x \left(\frac{k}{N}\right)} |k\rangle$$

$$= \frac{1}{\sqrt{N}} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} e^{2\pi i x \left(\sum_{l=1}^{n} \frac{k_l}{2^l}\right)} |k_1 k_2 \cdots k_n\rangle$$

(keeping the last equation from the previous slide) $\frac{k_1}{2} + \frac{k_2}{2^2} + \cdots$

$$= \frac{1}{\sqrt{N}} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} e^{2\pi i x \left(\sum_{l=1}^{n} \frac{k_l}{2^l}\right)} |k_1 k_2 \cdots k_n\rangle$$

$$= \frac{1}{\sqrt{N}} \sum_{k_1} \cdots \sum_{k_n} e^{2\pi i x \cdot \frac{k_1}{2}} e^{2\pi i x \frac{k_2}{4}} \cdots e^{2\pi i x \frac{k_n}{2^n}} |k_1 \cdots k_n\rangle$$

N

$$= \frac{1}{\sqrt{N}} \sum_{k_1} \cdots \sum_{k_n} \left(\bigotimes_{l=1}^{n} e^{2\pi i x \cdot \frac{k_l}{2^l}} |k_l\rangle \right)$$

$$= \frac{1}{\sqrt{N}} \bigotimes_{l=1}^{n} \left(\sum_{k_l=0}^{1} e^{2\pi i x \frac{k_l}{2^l}} |k_l\rangle \right)$$

A circuit for the QFT

(keeping the last equation from the previous slide)

$$= \frac{1}{\sqrt{N}} \bigotimes_{l=1}^{n} \left(\sum_{k_l=0}^{1} e^{2\pi i x \frac{k_l}{2^l}} |k_l\rangle \right)$$

$$= \frac{1}{\sqrt{N}} \bigotimes_{l=1}^{n} \left(|0\rangle + e^{2\pi i x \cdot \frac{1}{2^l}} |1\rangle \right)$$

☆ We will start here Weds.

A circuit for the QFT

So...

$$|x\rangle \rightarrow \frac{\left(|0\rangle + e^{2\pi i 0.x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.x_1\cdots x_n}|1\rangle\right)}{\sqrt{N}}$$

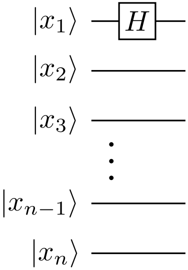
Believe it or not, this form reveals to us how we can design a circuit that creates this state!

A circuit for the QFT

Starting with the state

$$|x\rangle = |x_1 \cdots x_n\rangle,$$

apply a Hadamard to qubit 1:

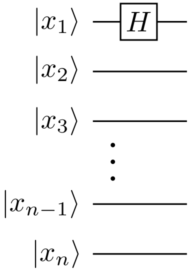


A circuit for the QFT

$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 0.x_1}|1\rangle\right)|x_2 \cdots x_n\rangle$$

If $x_1 = 0$, $e^0 = 1$ and we get $|+\rangle$.

If $x_1 = 1$, $e^{2\pi i(1/2)} = e^{\pi i} = -1$ and we get $|-\rangle$.



A circuit for the QFT

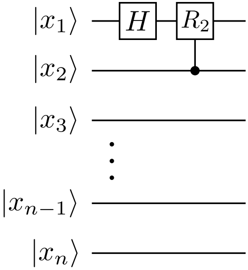
We are trying to make a state that looks like this:

$$|x\rangle \to \frac{\left(|0\rangle + e^{2\pi i 0.x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.x_1\cdots x_n}|1\rangle\right)}{\sqrt{N}}$$

Every qubit has a different *phase* on the $|1\rangle$ state. Define

A circuit for the QFT

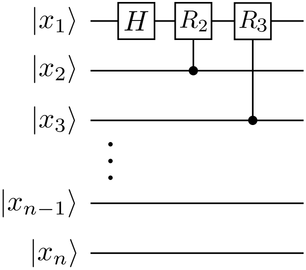
Apply controlled R_2 from qubit $2 \rightarrow 1$



First qubit picks up a phase:

A circuit for the QFT

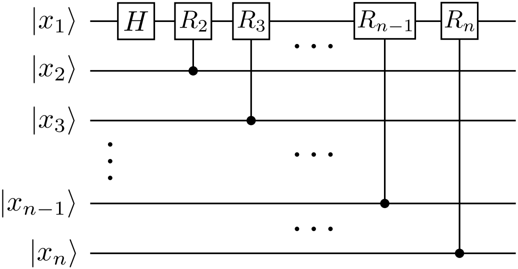
Apply controlled R_3 from qubit $3 \to 1$



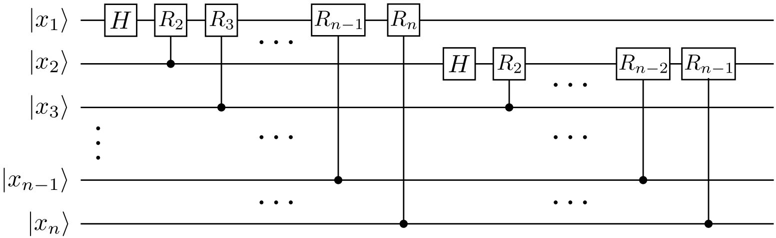
First qubit picks up another phase:

A circuit for the QFT

Apply a controlled R_4 from $4 \to 1$, etc. up to the n-th qubit to get



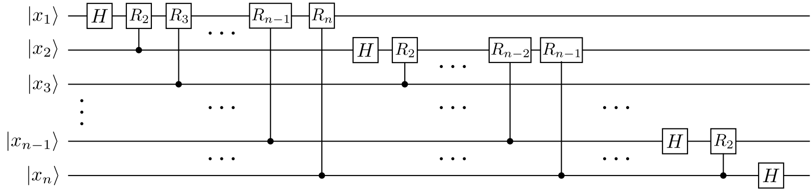
Next, do the same thing with the second qubit: apply *H*, and then controlled rotations from every qubit from 3 to *n* to get



A circuit for the QFT

Do this for all qubits to get that big ugly state from earlier:

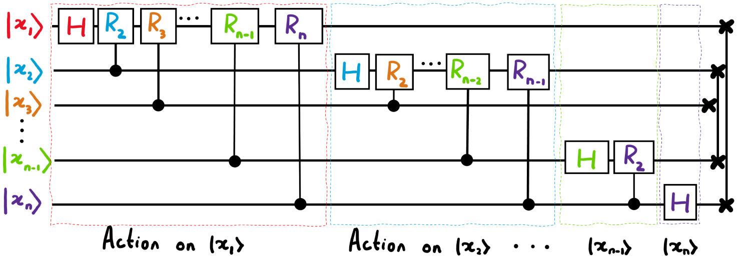
$$|x\rangle \rightarrow \frac{\left(|0\rangle + e^{2\pi i 0.x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.x_1\cdots x_n}|1\rangle\right)}{\sqrt{N}}$$



(though note that the order of the qubits is backwards - this is easily fixed with some SWAP gates)

Gate counts:

- n Hadamard gates
- $n(n-1)/2$ controlled rotations
- $\lfloor n/2 \rfloor$ SWAP gates if you care about the order



The number of gates is *polynomial in n*!

Image credit: Xanadu Quantum Codebook node F.3

Next time

Content:

- Quantum phase estimation

Action items:

1. Finish literacy assignment 2
2. Work on project

Recommended reading:

- Codebook module F
- Nielsen & Chuang 5.1
- Codebook module P