CPEN 400Q Lecture 14
QFT and quantum phase estimation

Wednesday 28 February 2024

Announcements

- Quiz 6 on Monday
- Assignment 3 coming on Monday (QPE and Shor)
- Fill out peer review survey by Friday (link in Piazza)

We introduced the quantum Fourier transform, and saw how it is the analog of the classical inverse discrete Fourier transform.

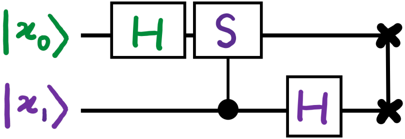
$$QFT|x\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle$$

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

where for n qubits, $N = 2^n$, and $\omega = e^{2\pi i/N}$

Last time

We saw the circuits for some special cases.



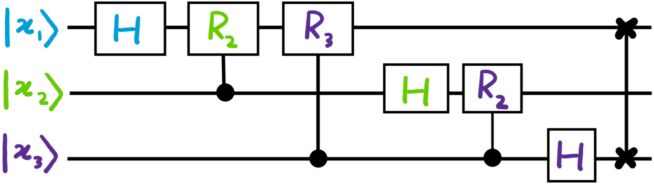


Image credit: Xanadu Quantum Codebook node F.2, F.3

The general form of the circuit is:

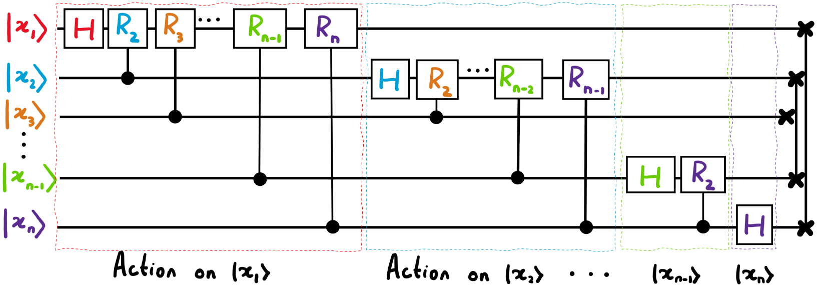


Image credit: Xanadu Quantum Codebook node F.3

Learning outcomes

- Implement the quantum Fourier transform in PennyLane
- Outline the steps of the quantum phase estimation (QPE) subroutine
- Use the QFT to implement QPE

A circuit for the QFT

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

A circuit for the QFT

(keeping the last equation from the previous slide)

A circuit for the QFT

(keeping the last equation from the previous slide)

A circuit for the QFT

So...

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

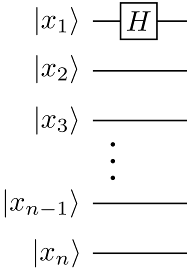
Believe it or not, this form reveals to us how we can design a circuit that creates this state!

A circuit for the QFT

Starting with the state

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

apply a Hadamard to qubit 1:

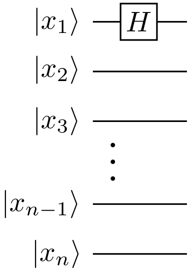


A circuit for the QFT

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

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 $|-\rangle$.

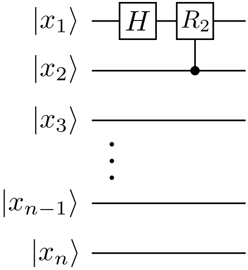


We are trying to make a state that looks like this:

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

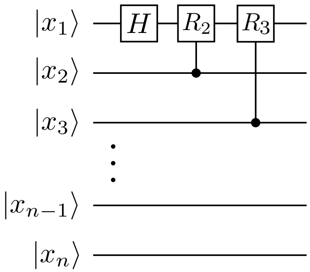
Every qubit has a different *phase* on the $|1\rangle$ state. Define

Apply controlled R_2 from qubit $2 \to 1$



First qubit picks up a phase:

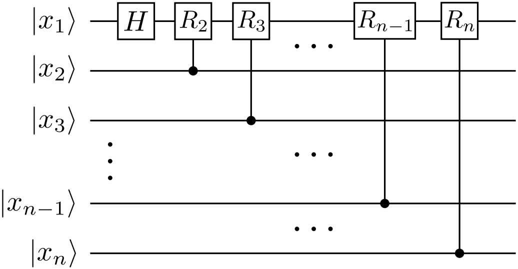
Apply controlled R_3 from qubit $3 \to 1$



First qubit picks up another phase:

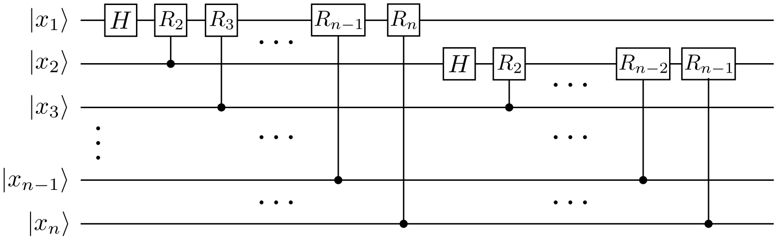
A circuit for the QFT

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



A circuit for the QFT

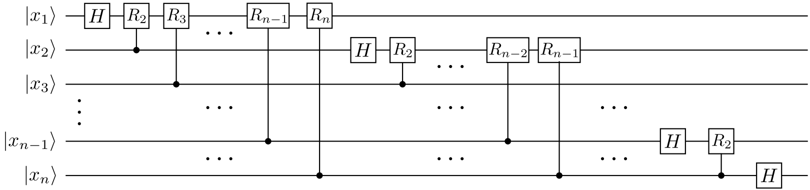
Next, do the same thing with the second qubit: apply H, and then controlled rotations from every qubit from 3 to n to get



A circuit for the QFT

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

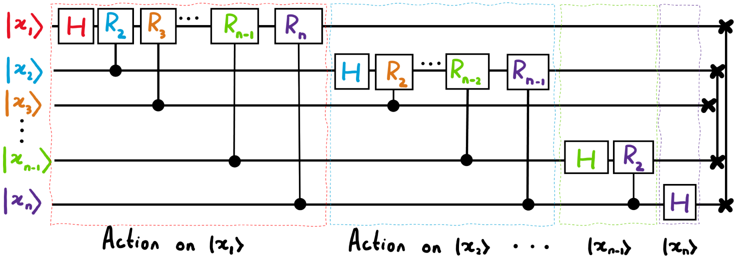
$$|x\rangle \rightarrow \frac{\left(|0\rangle + e^{2\pi i 0.x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.x_1\cdots x_n}|1\rangle\right)}{\sqrt{N}}$$



(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

Reminder: where are we going?

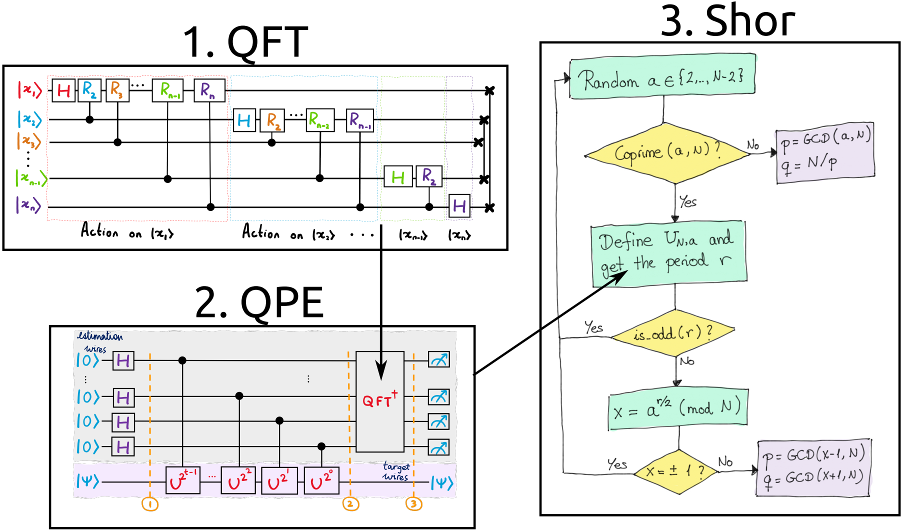


Image credit: Xanadu Quantum Codebook nodes F.3, P.3, S.4

Eigenvalues of unitary matrices

Fun fact: eigenvalues of unitaries are complex numbers with magnitude 1.

Proof:

Eigenvalues of unitary matrices

So we can write

where θ_k is some phase angle such that $|\theta_k| \leq 1$.

What if we want to *learn* an unknown θ_k?

Eigenvalues of unitary matrices

Idea: apply U to the relevant eigenvector, because that's "what makes the phase come out".

...but this is an unobservable *global* phase!

We have to do something different: eigenvalue estimation, or quantum phase estimation (QPE).

Given unitary U and eigenvector $|k\rangle$, estimate θ_k such that

$$U|k\rangle = e^{2\pi i\theta_k}|k\rangle$$

Must determine:

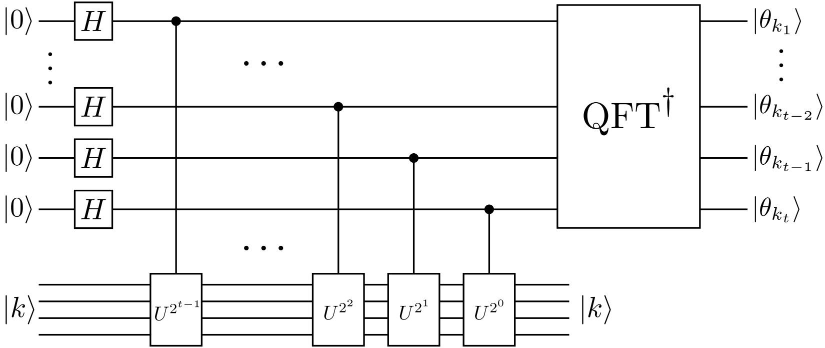
- How to design a circuit that extracts the θ_k
- To what precision can we estimate it
- What to do if we don't know a $|k\rangle$ in advance

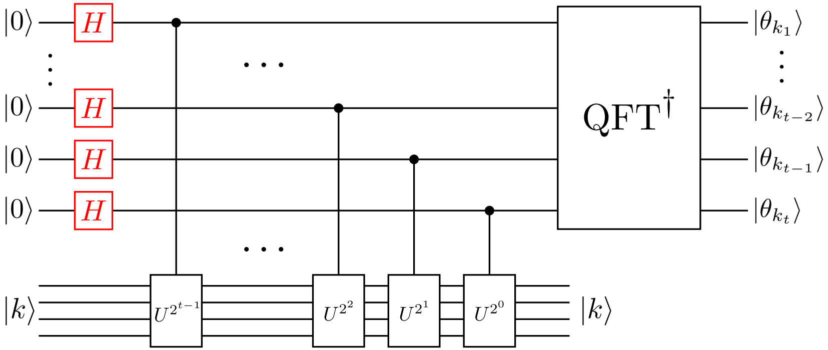
(*You will explore the last two in your homework!*)

Let U be an n-qubit unitary; $|k\rangle$ is an n-qubit eigenstate.

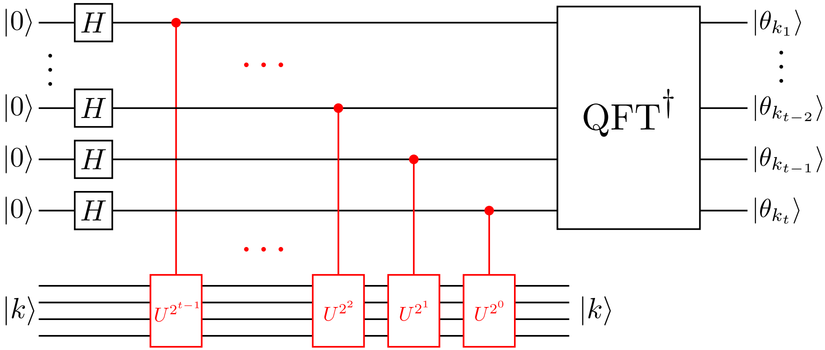
Assume θ_k can be represented *exactly* using t bits:

$$\theta_k = 0.\theta_{k_1} \cdots \theta_{k_t}$$

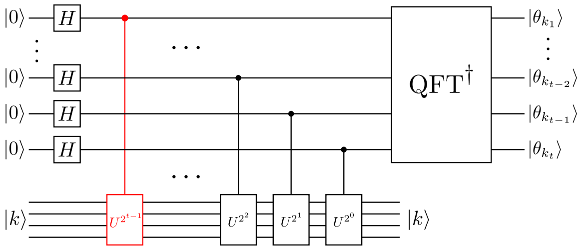



$|0\rangle$
$|0\rangle$
$|0\rangle$
$|0\rangle$
H
H
H
H
$|k\rangle$
$U^{2^{t-1}}$
U^{2^2}
U^{2^1}
U^{2^0}
$|k\rangle$
$\mathrm{QFT}^\dagger$
$|\theta_{k_1}\rangle$
$|\theta_{k_{t-2}}\rangle$
$|\theta_{k_{t-1}}\rangle$
$|\theta_{k_t}\rangle$

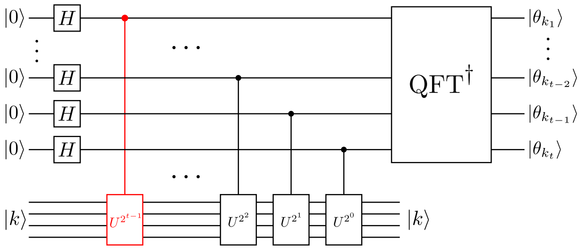
Quantum phase estimation: step 1



Quantum phase estimation: step 2

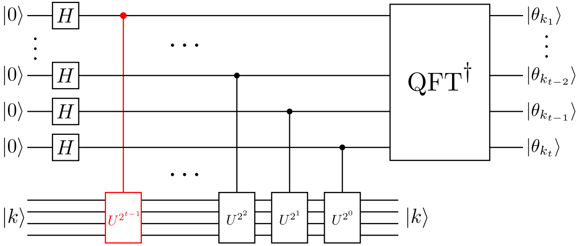


Quantum phase estimation: step 2



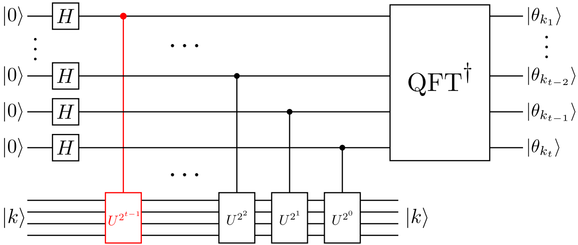
Use phase kickback

Quantum phase estimation: step 2

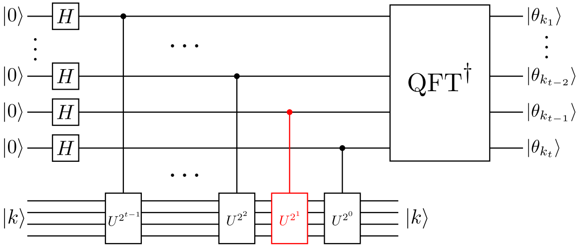


What is happening in the exponent?

Quantum phase estimation: step 2

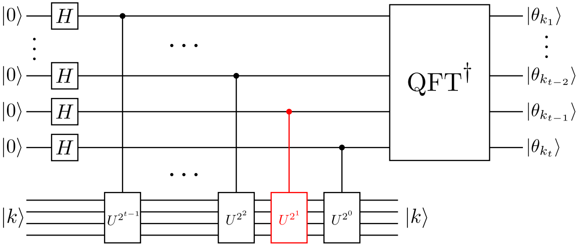


Quantum phase estimation: step 2



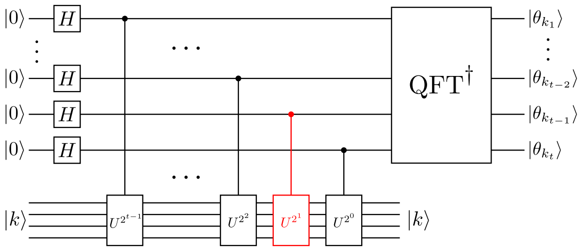
Check second-last qubit (ignore the others)

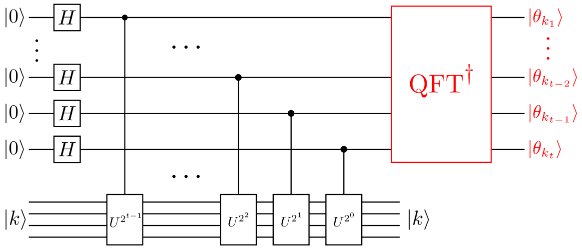
Quantum phase estimation: step 2



Again check the exponent...

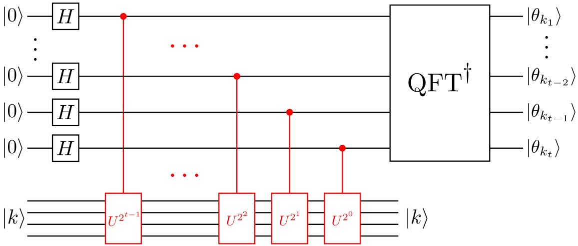
Quantum phase estimation: step 2





Can show in the same way for the last qubit (ignore others)

Quantum phase estimation: step 2

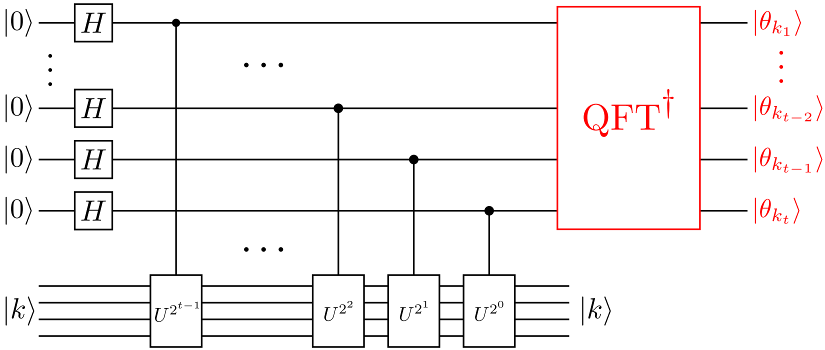


After step 2, we have the state

$$\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0.\theta_{k_t}}|1\rangle)\cdots\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0.\theta_{k_2}\cdots\theta_{k_t}}|1\rangle)\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0.\theta_{k_1}\cdots\theta_{k_t}}|1\rangle)|k\rangle$$

Should look familiar!

Measure to learn the bits of θ_k.



Let's implement it.

Next time

Content:

- Quiz 6 on Monday
- Moving towards Shor's algorithm

Action items:

1. Work on project; midterm checkpoint next week
2. Fill out weekly peer assessment survey by Friday

Recommended reading:

- Codebook nodes F.1-F.3, P.1-P.4
- Nielsen & Chuang 5.1, 5.2