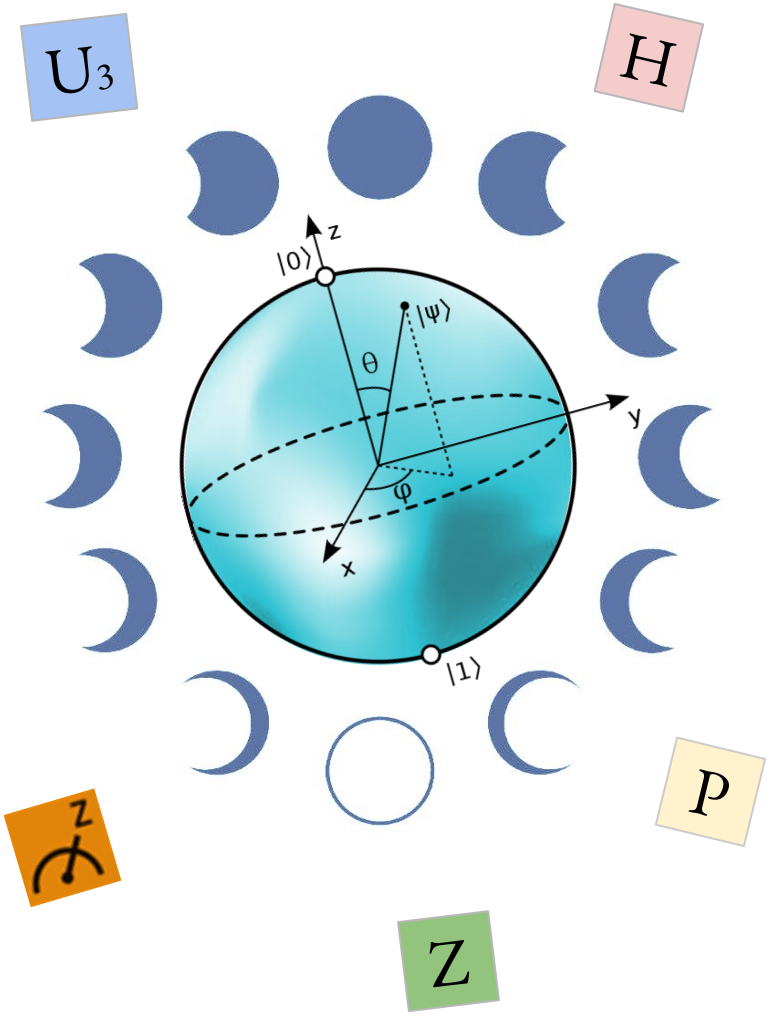


Phase Estimation Algorithm for Quantum Computing

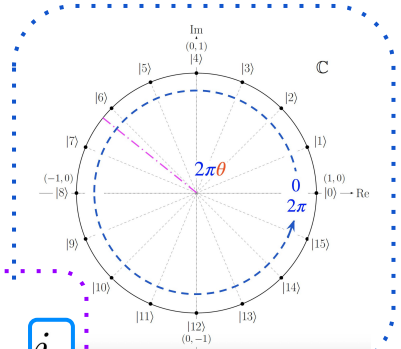
Zhengrong Qian

FRIB

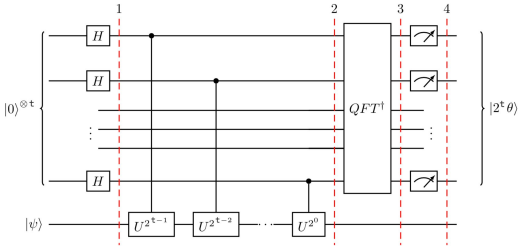
FRIB-TA Summer School 2022

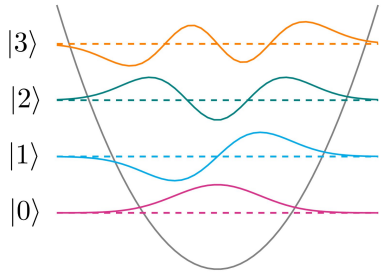
Outline

- *Idea* of quantum phase estimation (QPE)
  - Phase as binary fractions
- How do we *estimate* the phase ?
  - Phase kickback
  - Quantum Fourier Transform
- Quantum circuit for QPE
  - Standard & Iterative
- Application on a physical model

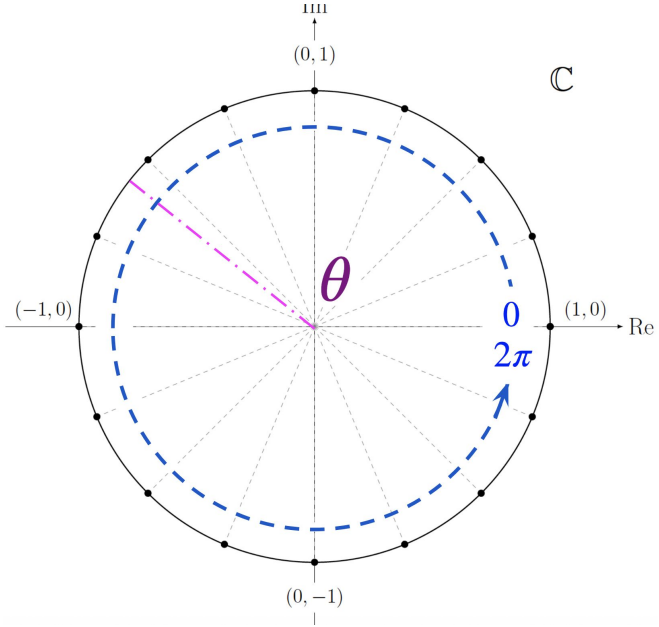


$$\varphi = \sum_{m=1}^{t} j_m 2^{-m} = 0.\, j_1\ j_2\ \dots\ j_t$$





QPE 101: The *Idea*

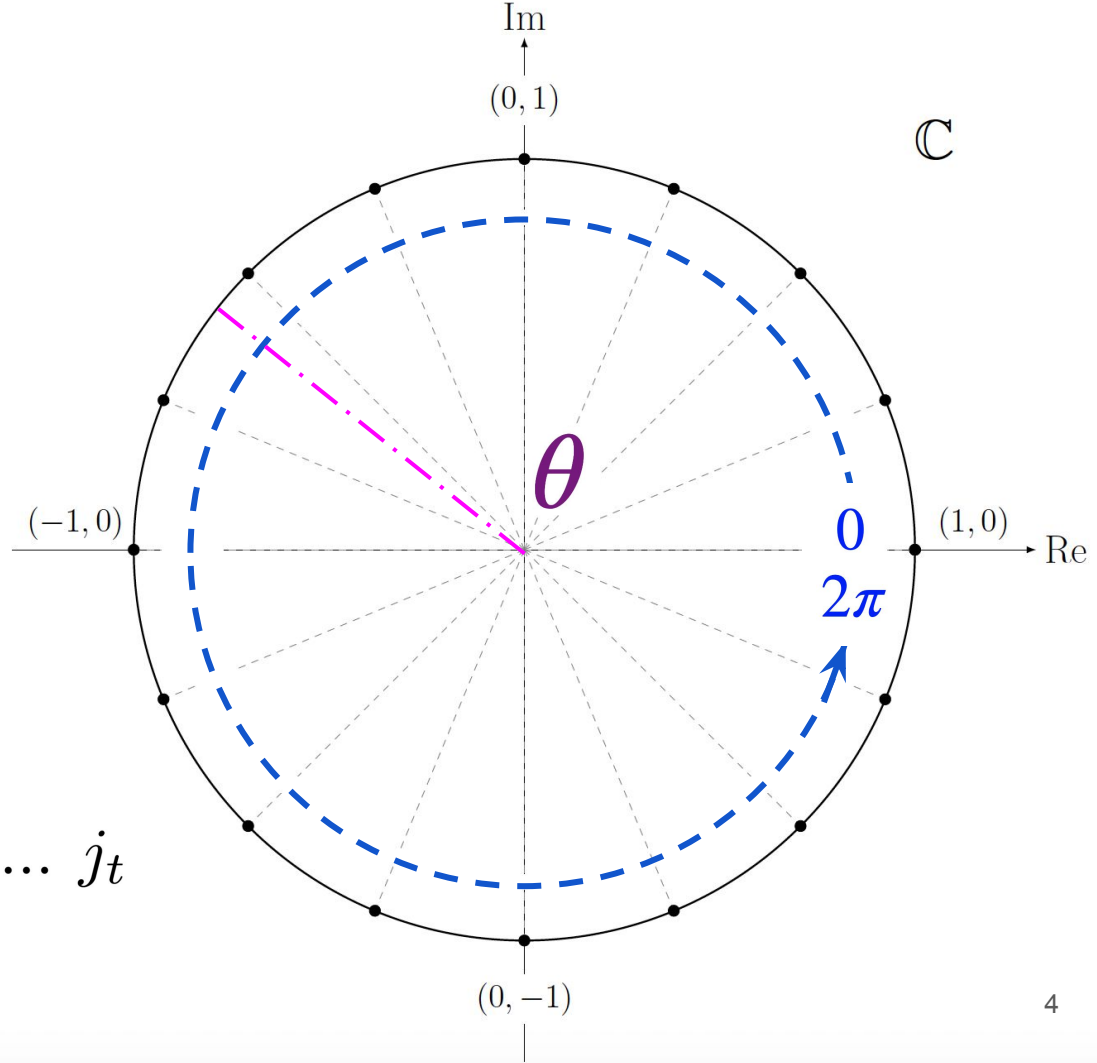


Phase as ***binary*** **fractions**

$$U|\psi\rangle = e^{i\theta}|\psi\rangle$$

$$= e^{2\pi i\varphi}|\psi\rangle$$

$$\varphi = \sum_{m=1}^{t} j_m 2^{-m} = 0.\ j_1\ j_2\ \ldots\ j_t$$



Phase as ***binary*** **fractions**

$$\varphi = \sum_{m=1}^{t} j_m 2^{-m} = 0.\boxed{j_1}\ j_2\ \ldots\boxed{j_t}$$

Most significant bit **Least significant bit**

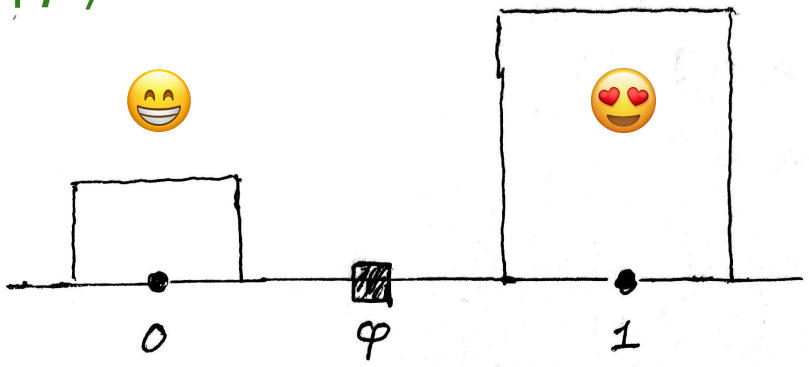
Example:

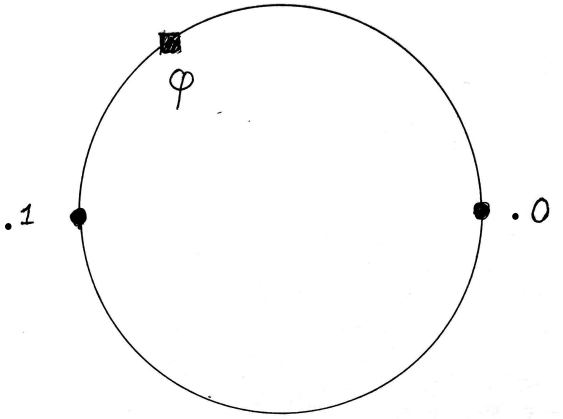
$$\varphi = (\,.\ 1\ 0\ 1\ 1)_2$$

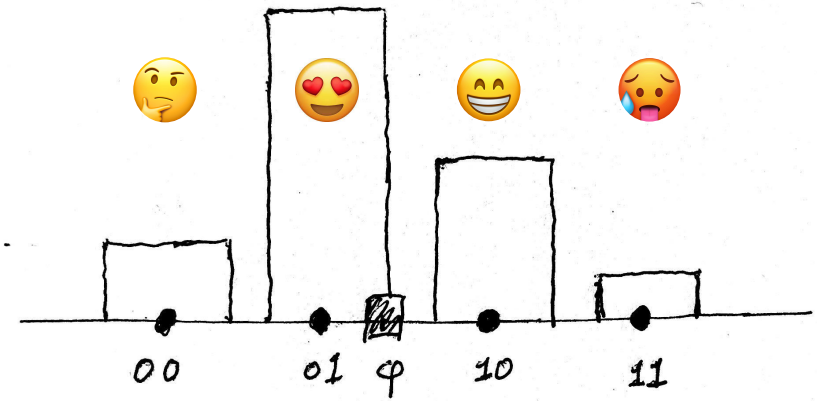
$$\frac{1}{2^1}\cdot 1 + \frac{1}{2^2}\cdot 0 + \frac{1}{2^3}\cdot 1 + \frac{1}{2^4}\cdot 1 = (0.6875)_{10}$$

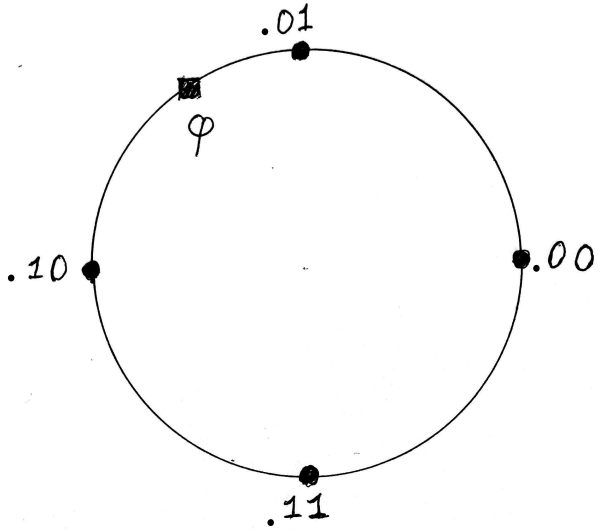
$$U|\psi\rangle = e^{2\pi i\varphi}|\psi\rangle$$

Learn the eigenvalue of your system



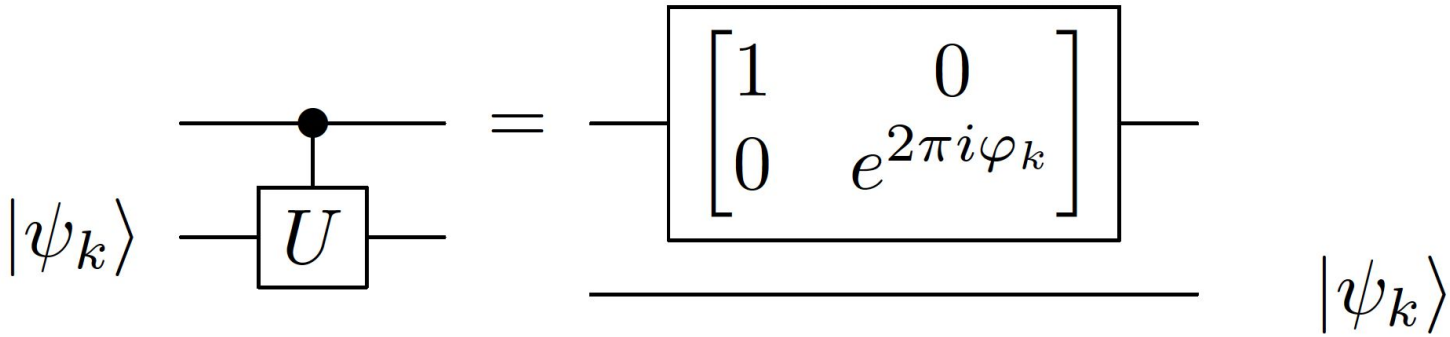






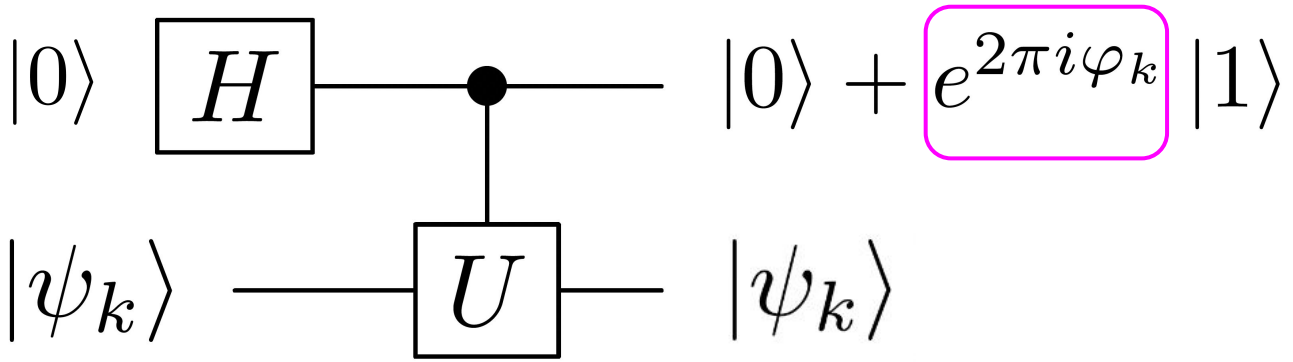
How do we learn the phase using a quantum computer ?

Phase kickback



$$U|\psi\rangle = e^{2\pi i \varphi}|\psi\rangle$$

Phase kickback: Why Hadamard ?

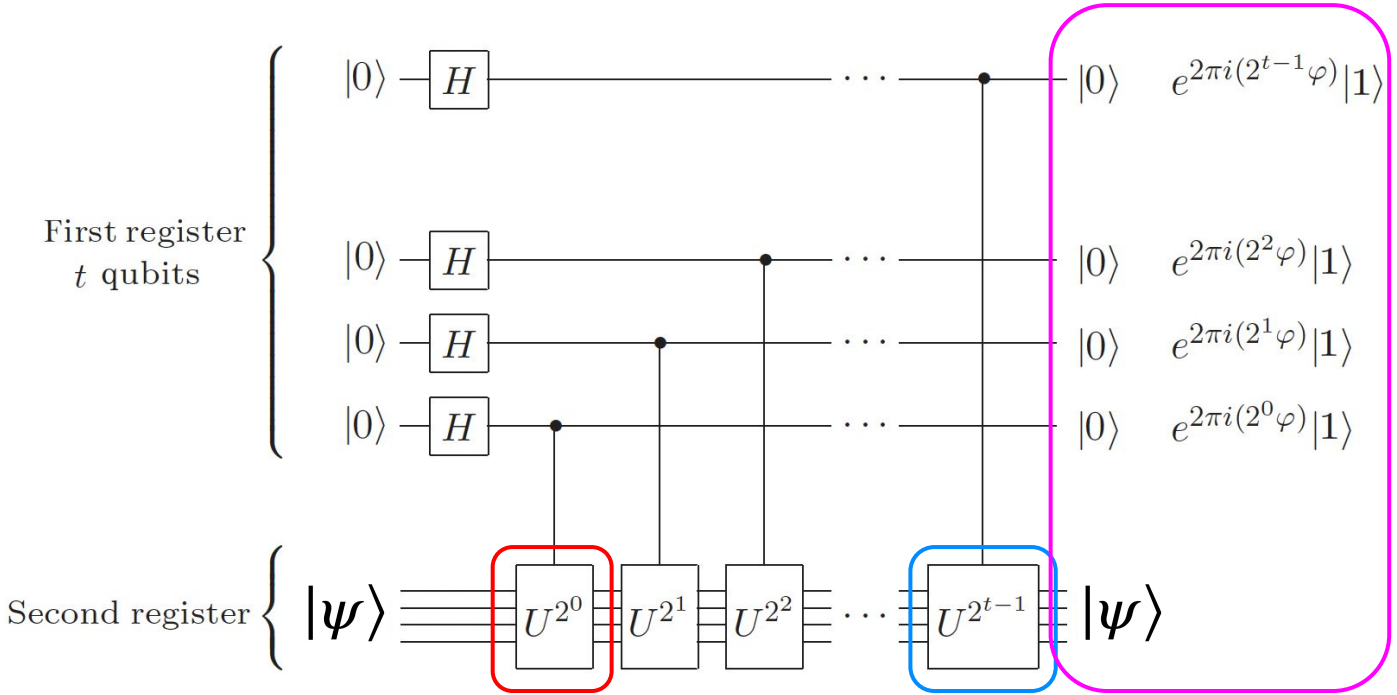


$$U|\psi\rangle = e^{2\pi i\varphi}|\psi\rangle$$

Quantum Fourier Transform

$$\varphi = \sum_{m=1}^{t} j_m 2^{-m} = 0.j_1 j_2 \ldots j_t$$

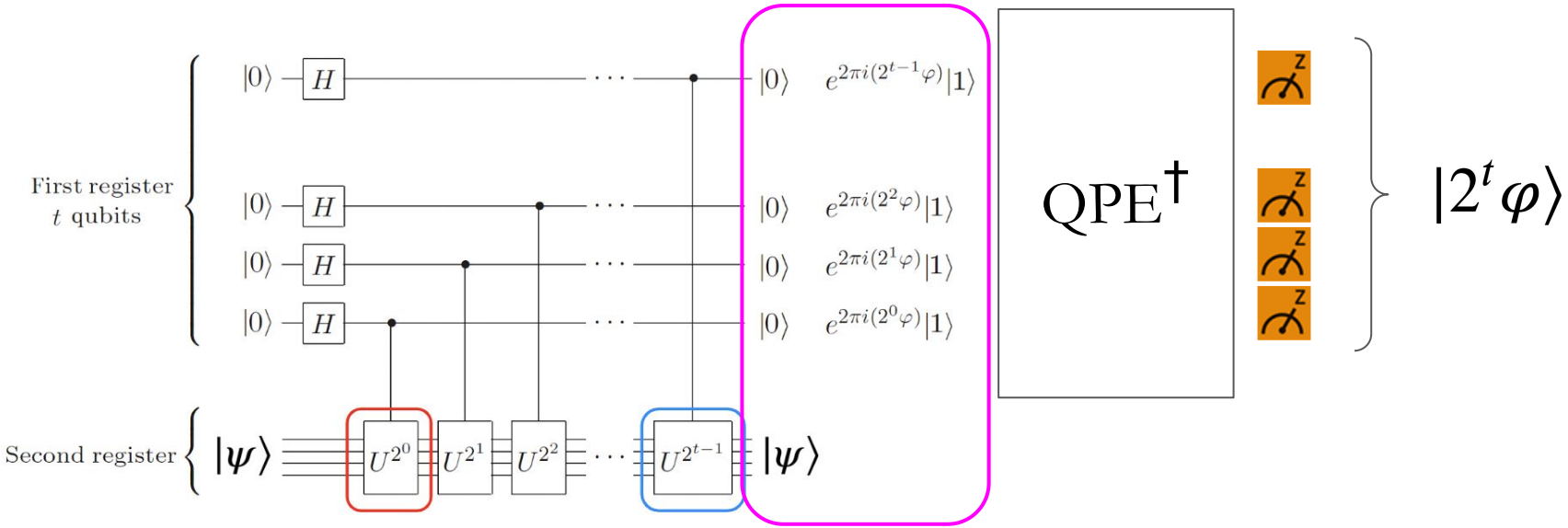
$$\frac{1}{2^{\frac{n}{2}}} \sum_{k=0}^{2^n-1} e^{2\pi i \theta k} |k\rangle \otimes |\psi\rangle$$



Source: Nielsen and Chuang, Ch. 5.2

Standard phase estimation

$$\varphi = \sum_{m=1}^{t} j_m 2^{-m} = 0.\boxed{j_1} j_2 \ \ldots \boxed{j_t}$$



Source: Nielsen and Chuang, Ch. 5.2

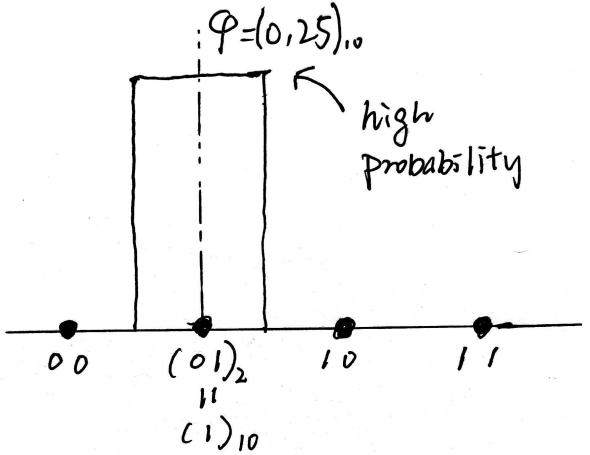
Interpret the phase

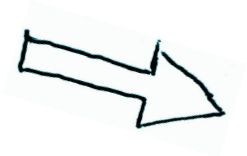
$|2^t\varphi\rangle$

$2^t\varphi$

Is integer

Ex: t = 2,
Phi = 0.25

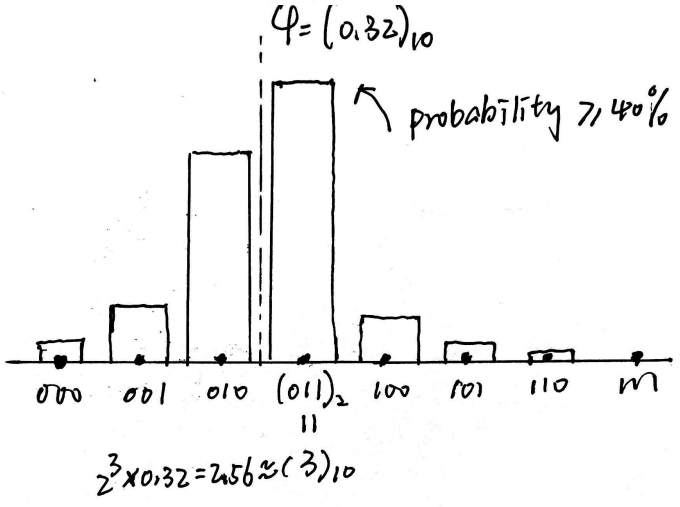




$2^t\varphi$

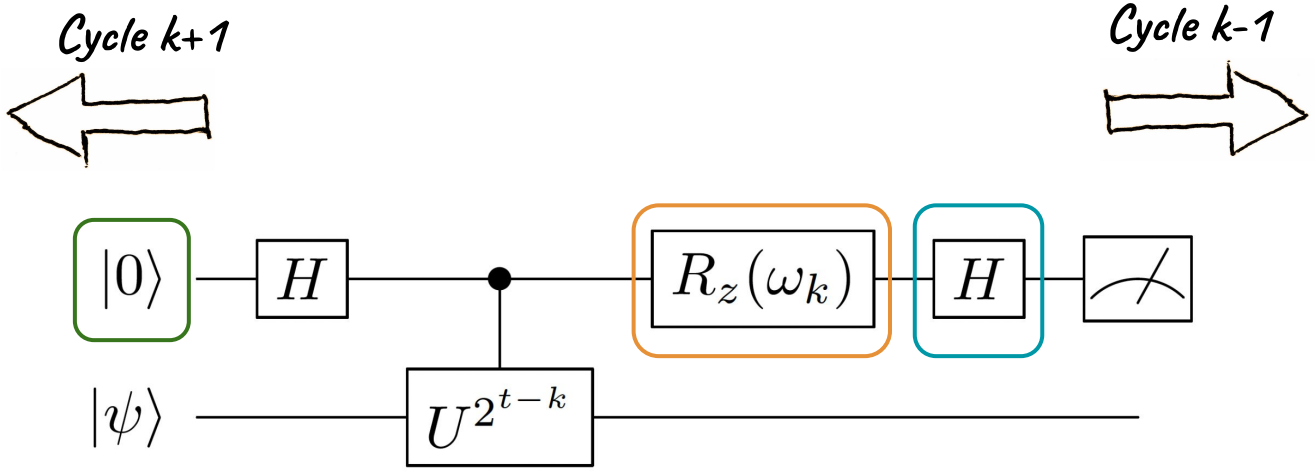
Is not integer

Ex: t = 3,
Phi = 0.32



Iterative phase estimation

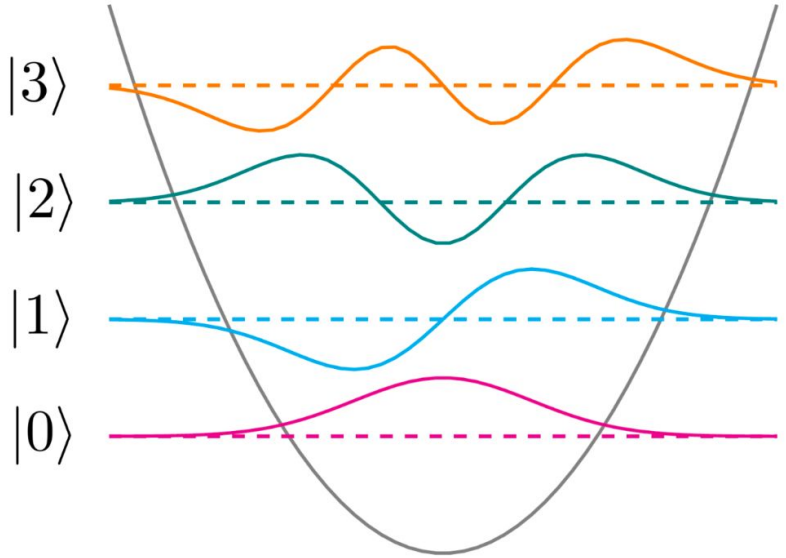
$$\varphi = \sum_{m=1}^{t} j_m 2^{-m} = 0.\ j_1\ j_2\ \dots\ j_k\ \dots\ j_t$$



ω_k Feedback angle, depends on previously measured bit.

Estimate the energy of a number operator

$$\hat{n} = \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & 2 & \\ & & & 3 \end{bmatrix}$$



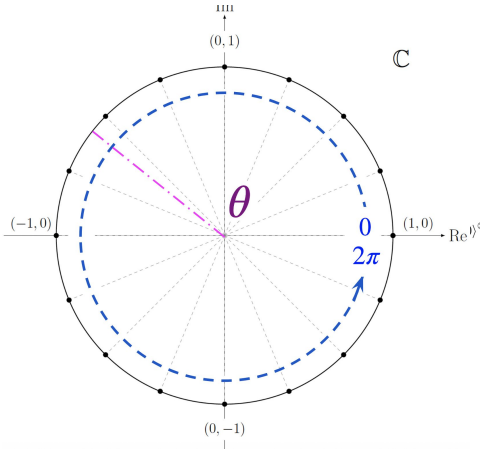
Question: Using the standard phase estimation algorithm, what can we say about the eigenvalue of state $|2\rangle$?

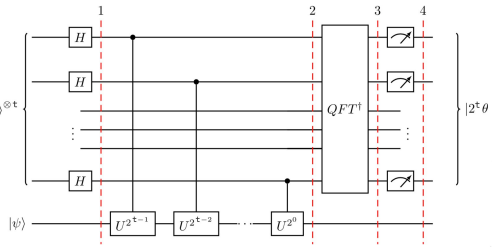
Step 1: 'Wrap' the unit complex circle

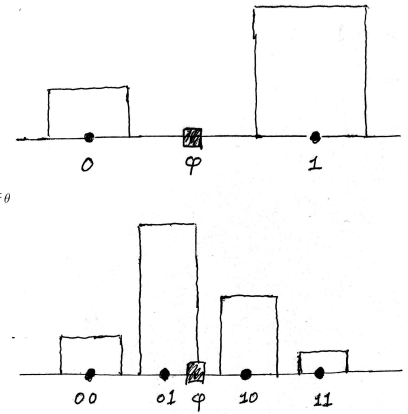
Step 2: Apply the QPE algorithm

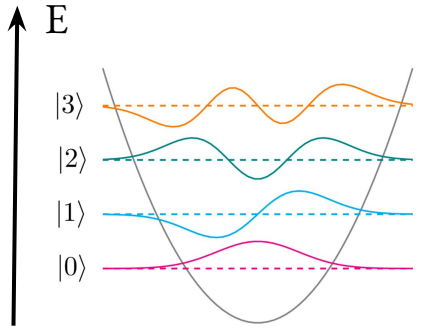
Step 3: Collect data and learn the phase

Step 4: 'Translate' back to the energy spectrum







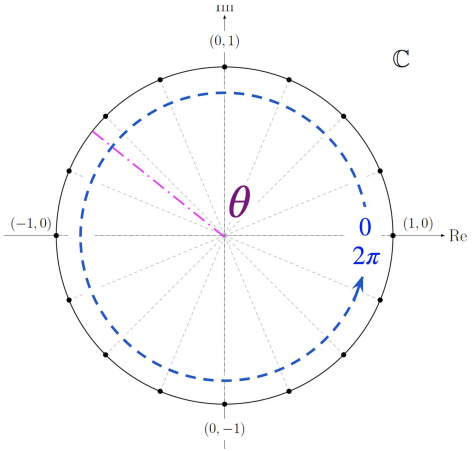


Step 1: 'Wrap' the unit complex circle

Step 2: Apply the QPE algorithm

Step 3: Collect data and learn the phase

Step 4: 'Translate' back to the energy spectrum



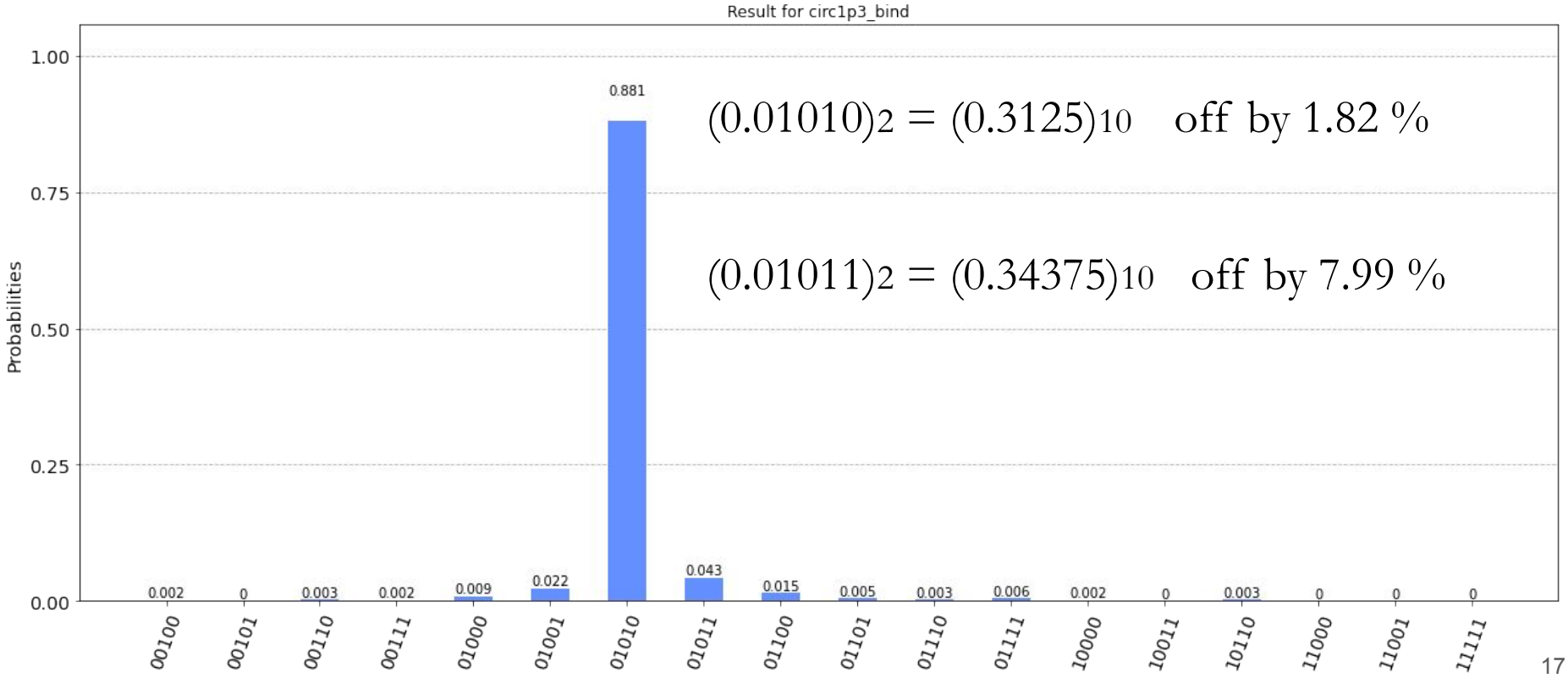
$$U|2\rangle = e^{it2}|10\rangle = e^{2\pi i\varphi}|10\rangle$$

$$t = 1$$

$$\varphi = (0.318309...)_{10} = (0.010100010...)_2$$

$$\varphi = (0.318309...)_{10} = (0.010100010...)_2$$

Step 3



$(0.01010)_2 = (0.3125)_{10}$ off by 1.82 %

$(0.01011)_2 = (0.34375)_{10}$ off by 7.99 %

More Simple Algorithms: Adiabatic Evolution and QAOA

Joey Bonitati (Michigan State University)

FRIB Theory Alliance Summer School
June 20, 2022

FRIB

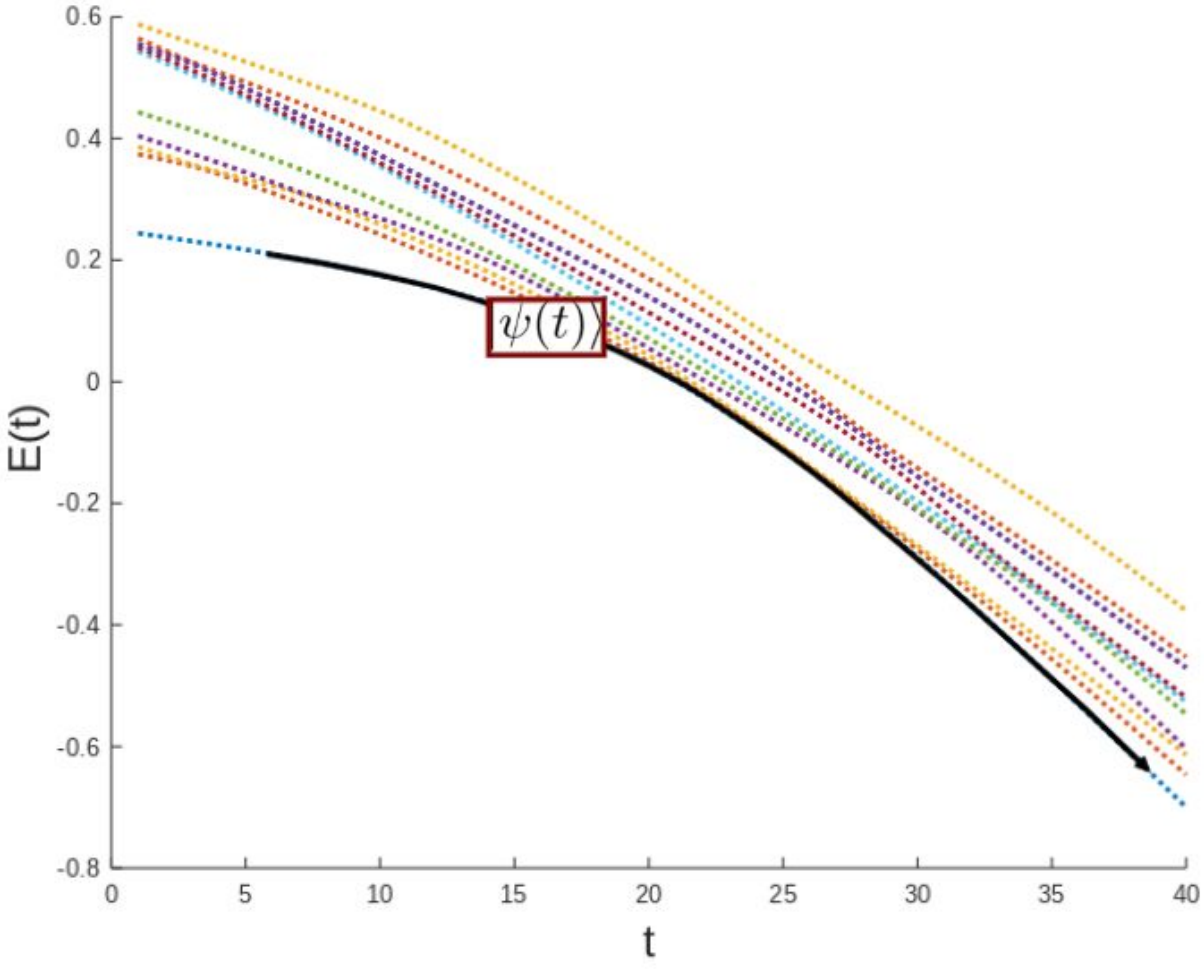
MICHIGAN STATE UNIVERSITY

Adiabatic Theorem

An eigenstate will remain an eigenstate as long as its system is changing slowly

$$H(t)\,|\psi(t)\rangle = E(t)\,|\psi(t)\rangle$$

The smaller the energy gap between states, the slower the changes must be



Adiabatic Evolution

Choose a simple initial Hamiltonian and final target Hamiltonian

$$H(0) = H_I,\ H(t_F) = H_F$$

$$H(t) = \frac{t_F - t}{t_F} H_I + \frac{t}{t_F} H_F$$

Start in an eigenstate of H_I at t = 0. Apply the unitary time evolution operator

$$U(t + dt, t) = e^{-iH(t)dt}$$

$$|\psi(t + dt)\rangle = U(t + dt, t)\,|\psi(t)\rangle$$

Repeat until $t = t_F$. The final state will be an eigenstate of H_F

Adiabatic Evolution Example (two qubits)

Let's choose the initial state to be: $|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

This is the ground state of the Hamiltonian:

$$H_I = -Z_0 \otimes I_1 - I_0 \otimes Z_1 = \begin{bmatrix} -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix}$$

Adiabatic Evolution Example (two qubits)

For the final state, we can choose any 4x4 Hermitian matrix. For example:

$$H_F = H_I + X_0 \otimes X_1 + Y_0 \otimes Y_1 + Z_0 \otimes Z_1 = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 \\ 0 & 2 & -1 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix}$$

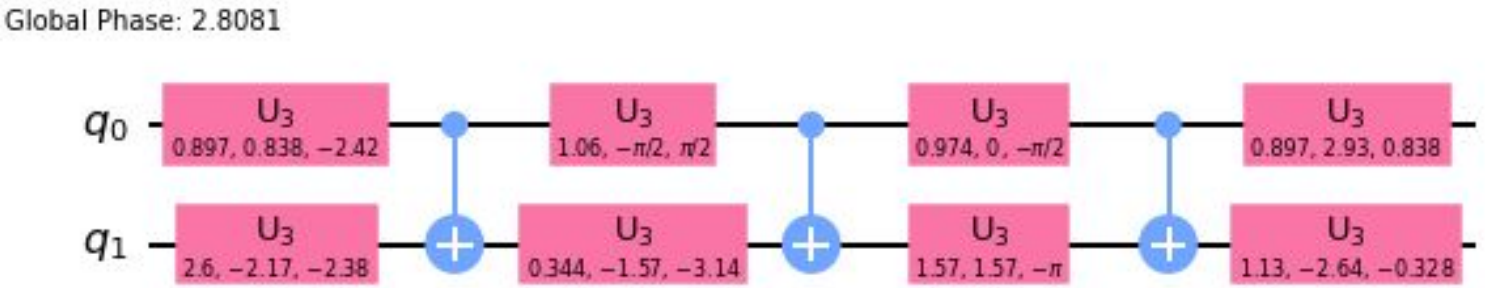
The time-dependent Hamiltonian will then be:

$$H(t) = \frac{t_F - t}{t_F} H_I + \frac{t}{t_F} H_F$$

Adiabatic Evolution Example (two qubits)

Determine the time evolution gates (by hand or using qiskit library functions) and apply them to a quantum circuit in order

```
#one step of adiabatic evolution
evolution_gate = qiskit.extensions.HamiltonianGate(h,dt)
decomposed_gate = qiskit.quantum_info.two_qubit_cnot_decompose(evolution_gate)
circ.append(decomposed_gate, qr)
circ.decompose().draw('mpl')
```

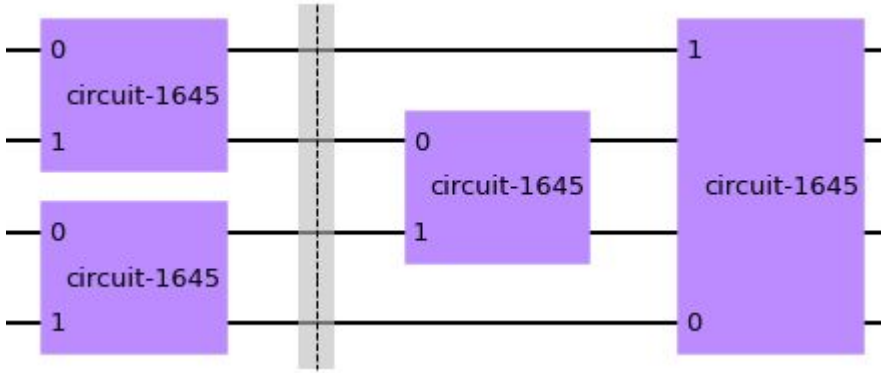


Adiabatic evolution on more than 2 qubits

If a Hamiltonian can be decomposed into a sum of 2-qubit Hamiltonians, then the Trotter approximation can be used to split the 4-qubit gate into four 2-qubit gates.

For problems with more complicated Hamiltonians and problems that require a large number of gates, methods that don't require exact time evolution are used.

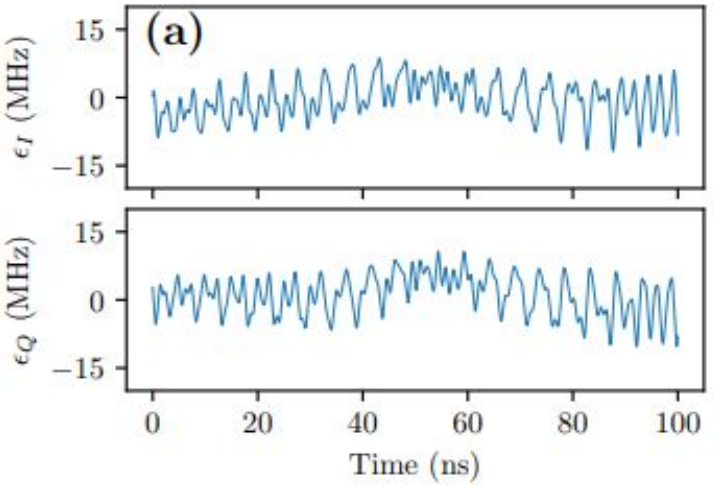
One Trotter-decomposed time evolution step on four qubits



Pulse-level Control for time evolution

Can reduce the number of 2-qubit gates required to only one per time step.

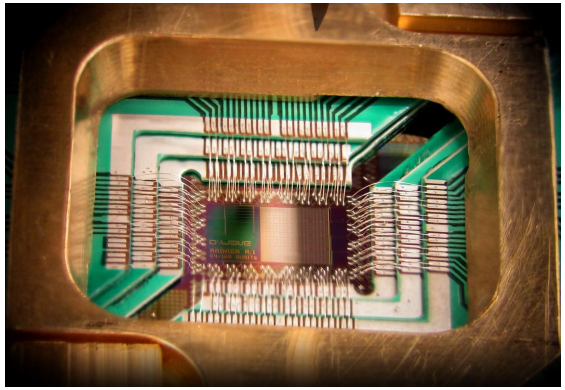
Generate custom gates using classical optimization by simulating a specific quantum device



Optimal Control for the Quantum Simulation of Nuclear Dynamics

Eric T. Holland,[1,*] Kyle A. Wendt,[1,†] Konstantinos Kravvaris,[1] Xian Wu,[1] W. Erich Ormand,[1] Jonathan L DuBois,[1] Sofia Quaglioni,[1] and Francesco Pederiva[2,3]

Quantum Annealing

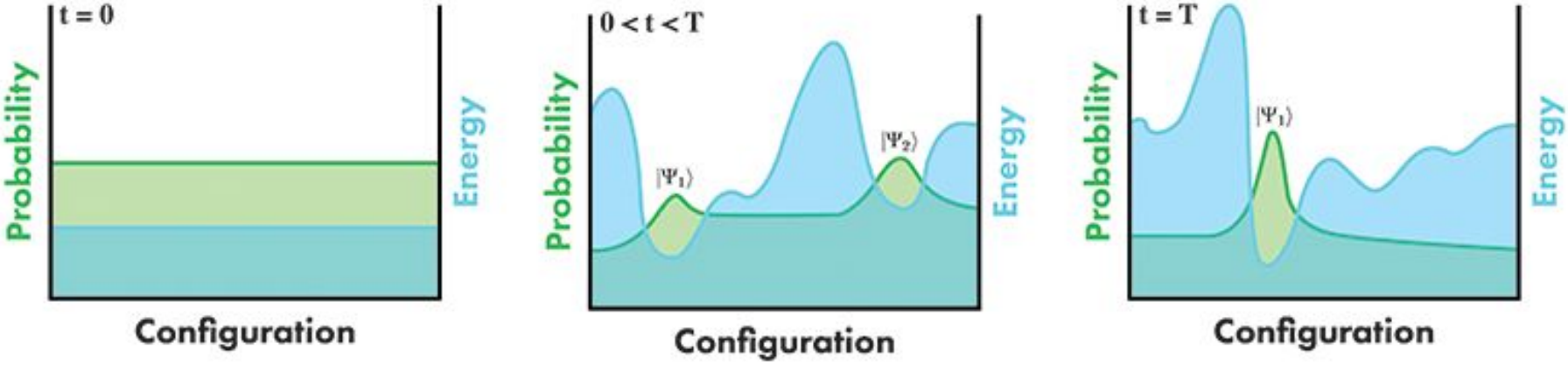


A technique used by D-WAVE to compute ground states.

Adiabatic fluctuations are applied to a quantum system.

Quantum tunneling allows the system to settle into the ground state of a target Hamiltonian.

Currently being used to solve hard combinatorial



Quantum Approximate Optimization Algorithm (QAOA)

Inspired by Adiabatic Evolution

Start with the following ansatz:

$$|\psi(\vec{\beta},\vec{\gamma})\rangle = U(\vec{\beta},\vec{\gamma})\,|\psi_I\rangle$$

$$U(\vec{\beta},\vec{\gamma}) \approx e^{-i\beta_k H_I} e^{-i\gamma_k H_F} \cdots e^{-i\beta_1 H_I} e^{-i\gamma_1 H_F}$$

We use the variational principle,

$$E_F^{\text{ground}} \leq \langle\psi(\vec{\beta},\vec{\gamma})|H_F|\psi(\vec{\beta},\vec{\gamma})\rangle$$

with classical optimization to find an approximate minimum energy.

QAOA can be used to approximately solve some NP-Hard problems efficiently

Questions