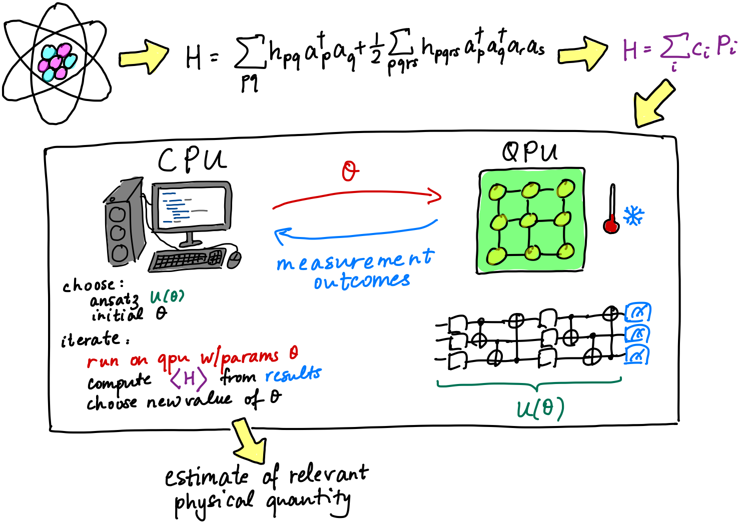
CPEN 400Q Lecture 14
The variational quantum eigensolver; introducing QAOA

Friday 3 March 2023

Announcements

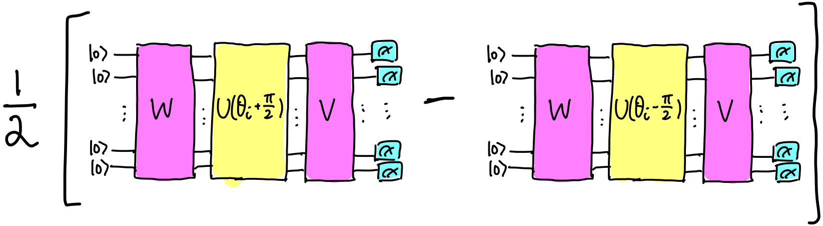
- Assignment 2 due 13 March at 23:59
- Updated class schedule:
  - Monday March 6: Zoom (quiz 6 at end)
  - Friday March 10: Zoom
  - Monday March 13: in person
  - Friday March 17: pre-recorded “infotainment” lecture about compilation
- Meetings *next week* for project prototypes, on Zoom: choose time on Piazza

Variational algorithms


$H = \sum_{pq} h_{pq} a_p^\dagger a_q + \frac{1}{2} \sum_{pqrs} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s$
$H = \sum_i c_i P_i$
CPU
QPU
θ
measurement outcomes
choose:
ansatz $U(\theta)$
initial θ
iterate:
run on qpu w/params θ
compute $\langle H \rangle$ from results
choose new value of θ
$U(\theta)$
estimate of relevant physical quantity

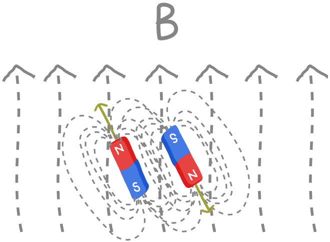
Last time

We applied the parameter-shift rule to compute the gradient of a quantum circuit.



Last time

We introduced the concept of a Hamiltonian:



$$\hat{H} = \hat{H}_0 + \hat{H}_1 + \hat{H}_{01} = -\alpha Z_0 - \alpha Z_1 + \beta(X_0 X_1 + Y_0 Y_1 + Z_0 Z_1)$$

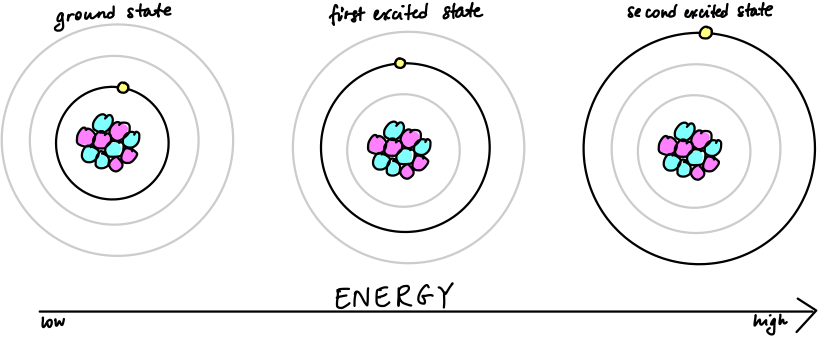
Image credit: Xanadu Quantum Codebook node H.5

Learning outcomes

- Compute the expectation value of a Hamiltonian in PennyLane
- Describe the variational quantum eigensolver algorithm and apply it to compute the ground state energy of a simple physical system
- Describe the underlying ideas of adiabatic quantum computation, and the quantum approximate optimization algorithm

Energy of a physical system

Determining the energy levels of physical systems is relevant for applications of quantum computing like quantum chemistry.

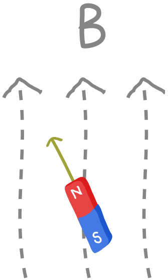


Energy levels allow us to compute useful properties of molecules (e.g., activation energies and reaction rates).

Hamiltonians

In physics, a special operator called a Hamiltonian, $\hat{H}$, describes the energy of a system. Hamiltonians are *Hermitian* operators.

Example: a bar magnet in a magnetic field along the Z direction.

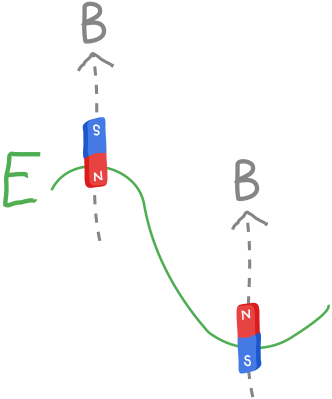


Every orientation of the magnet has an energy associated with it.

Image credit: Xanadu Quantum Codebook node H.3

Mathematical description of energy

Bar magnets like to *align* with the external field.



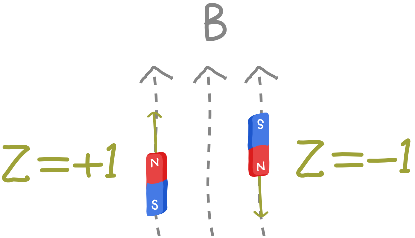
Aligned configuration is the *ground state* (lowest energy state).

Image credit: Xanadu Quantum Codebook node H.3

Hamiltonians

There is a *Hamiltonian* that describes the energy of the system:

where α is a coefficient that depends on some physical values.

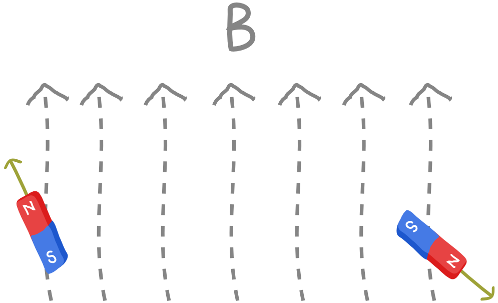


Computing the energy of the system corresponds to *measuring the expectation value of $\hat{H}$*.

Image credit: Xanadu Quantum Codebook node H.3

Hamiltonians

Hamiltonians can be used to describe systems with multiple parts. For example, two magnets that are far away have the Hamiltonian



The energy of this system is .

Image credit: Xanadu Quantum Codebook node H.5

Hamiltonians

Two magnets that are close together may have different terms in the Hamiltonian that describe *how they interact*:

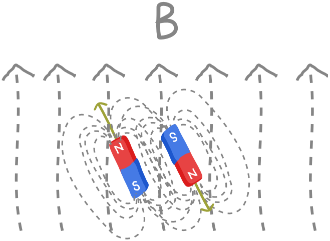


Image credit: Xanadu Quantum Codebook node H.5

Hamiltonians

More generally, any Hermitian Hamiltonian can be expressed as

Then, computation of expectation values is linear:

Computing the energy of a Hamiltonian

Given an arbitrary $\hat{H}$, how can we compute its ground state energy?

Computing the energy of a Hamiltonian

Option: *exact diagonalization* to find its eigenvalues.

$$\hat{H} = \begin{pmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{pmatrix} \xrightarrow{\text{np.linalg.eig}} \left\{ E_i , \, |E_i\rangle \right\}$$

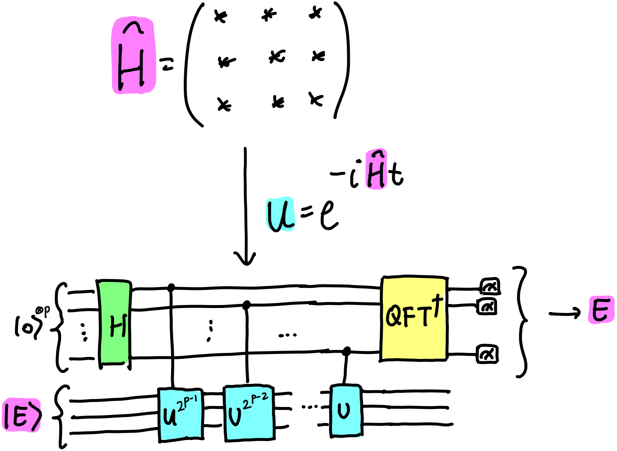
Computing the energy of a Hamiltonian

Let $\hat{H}$ be a Hamiltonian and $|E\rangle$ an eigenstate with energy E. If $U = e^{-i\hat{H}t}$,

So   is an eigenvalue of U... look familiar?

Computing the energy of a Hamiltonian

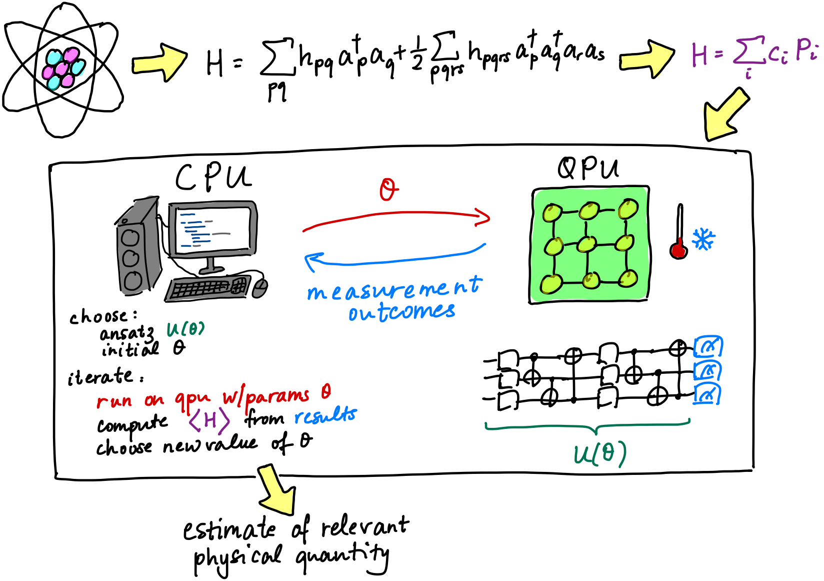
Option: *quantum phase estimation*.



But this is not practical on near-term quantum devices.

Computing the energy of a Hamiltonian

Option: *variational quantum eigensolver* (VQE).



$$H = \sum_{pq} h_{pq} a_p^\dagger a_q + \frac{1}{2}\sum_{pqrs} h_{pqrs} a_p^\dagger a_q^\dagger a_r a_s \quad H = \sum_i c_i P_i$$

```
choose:
    ansatz U(θ)
    initial θ
iterate:
    run on qpu w/params θ
    compute ⟨H⟩ from results
    choose new value of θ
```

estimate of relevant physical quantity

Variational quantum eigensolver (VQE)

The VQE is a variational algorithm that works by:

- training a circuit to prepare the ground state of a system
- measuring the expectation value of the Hamiltonian to compute the energy

The optimization works because of the *variational principle*. If E_g is the ground state energy, then

for *any other state* $|\psi\rangle$.

VQE example

Suppose we want to *learn* the ground state energy of

$$\hat{H} = 0.5 Z \otimes Z + 1.5 X \otimes X$$

Since this is a small system, we can work it out directly.

$$E_0 = -2, \quad E_1 = -1, \quad E_2 = 1, \quad E_3 = 2$$

$$|E_0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ -1 \\ 1 \\ 0 \end{pmatrix}, \quad |E_1\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 0 \\ 0 \\ 1 \end{pmatrix}, \quad |E_2\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ -1 \\ -1 \\ 0 \end{pmatrix}, \quad |E_3\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 0 \\ 0 \\ -1 \end{pmatrix}.$$

VQE example

We are lucky here: we know that the ground state is

So we need a parametrized circuit $U(\theta)$ that can produce

Hands-on with VQE

We can make a very tailored ansatz with just a single parameter here. Let

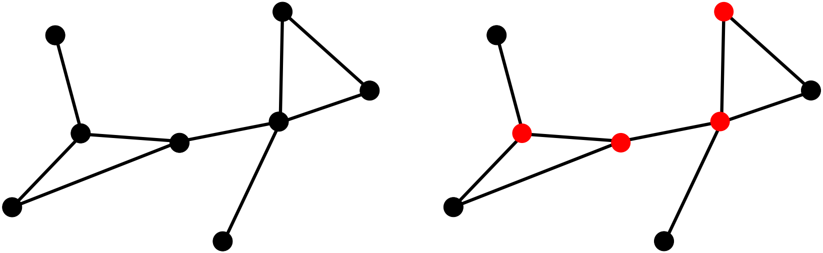
$$U(\theta)|00\rangle = \alpha|01\rangle + \beta|10\rangle$$

Since we know our ground state has real-valued amplitudes, write

What circuit makes states like this?

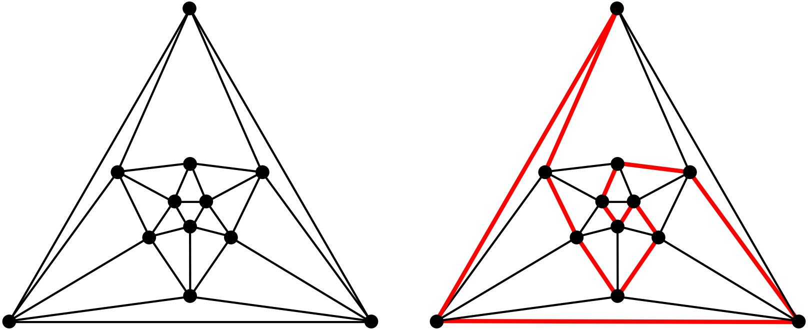
Combinatorial optimization

Example: Given a graph $G = (V, E)$, what is the *smallest number of vertices* you can colour such that every edge in the graph is attached to at least one coloured vertex?



Combinatorial optimization

Example: Given a graph, can we find a path through it that visits every node *exactly once* and returns to the starting point?



(In graph theory terms, can we find a *Hamiltonian cycle*?)

Combinatorial optimization

Example: You have the opportunity to purchase 100 units of stocks from a fixed list of assets. You know the average returns of each stock, and their covariances.

Stock	Avg. return
AAA	3.44 %
BBB	2.21 %
CCC	-0.28 %
⋮	⋮

Cov.	AAA	BBB	⋯
AAA	0.0038	0.002	⋯
BBB	0.002	-0.006	⋯
CCC	0.014	-0.0008	⋯
⋮	⋮	⋮	⋱

Suppose you're restricted to buying no more than 5 of any stock.

Which stocks, and how many of each, should you purchase, to maximize your profits?

Adiabatic quantum computing (AQC)

The structure of a classical optimization problem is something like:

$$\min_{\vec{x}} \ \text{cost}(\vec{x}) \quad \text{subject to constraints}(\vec{x})$$

where $\vec{x}$ is a multi-dimensional vector of parameters in the problem space.

Adiabatic quantum computing (AQC)

The structure of a classical optimization problem is something like:

$$\min_{\vec{x}} \ \text{cost}(\vec{x}) \quad \text{subject to constraints}(\vec{x})$$

where $\vec{x}$ is a multi-dimensional vector of parameters in the problem space.

In a physical context, optimization can be interpreted as an energy minimization problem.

Optimization	**Physical system**
$\vec{x}$	State of the system
cost($\vec{x}$)	Hamiltonian
Optimum $\vec{x^*}$	Ground state
cost($\vec{x^*}$)	Ground state energy

Adiabatic quantum computing (AQC)

Recall that every unitary U is directly related to a Hermitian Hamiltonian H under the correspondence

We know that we can use gate model QC to *simulate* the evolution of a Hamiltonian.

Instead of simulating the Hamiltonians, **adiabatic quantum computing** works with them directly to perform computations. It is generally used to solve **optimization problems**.

Adiabatic quantum computing (AQC)

1. Design a cost Hamiltonian whose ground state represents the solution to our optimization problem
2. Prepare a system in the ground state of an easy-to-prepare mixer Hamiltonian
3. Perform adiabatic evolution to transform the system from the ground state of mixer Hamiltonian to the ground state of the cost Hamiltonian, which is our solution

The adiabatic theorem

Why would we want to do this?

Theorem:

“A physical system remains in its instantaneous eigenstate if a given perturbation is acting on it slowly enough and if there is a gap between the eigenvalue and the rest of the Hamiltonian’s spectrum.”

What we can take from this:

If we initialize a system in the lowest energy state and perturb it slowly enough, it will remain in the lowest energy state (with respect to the changed system)

Adiabatic quantum computing (AQC)

Let H_m be a **mixer Hamiltonian** whose ground state can be easily prepared.

Let H_c be a **cost Hamiltonian** whose ground state represents the solution to a problem of interest.

Adiabatic evolution is expressed mathematically as the function

The parameter s is representative of time; s goes from 0 to 1; $A(s)$ decreases to 0 with time and $B(s)$ increases from 0.

Quantum annealing

D-Wave makes quantum annealers: these are a physical implementation of AQC for a limited set of Hamiltonians.

Image credit:
`www.dwavesys.com/tutorials/background-reading-series/introduction-d-wave-quantum-hardware`

Quantum approximate optimization algorithm (QAOA)

QAOA is a gate-model algorithm that can obtain approximate solutions to combinatorial optimization problems.

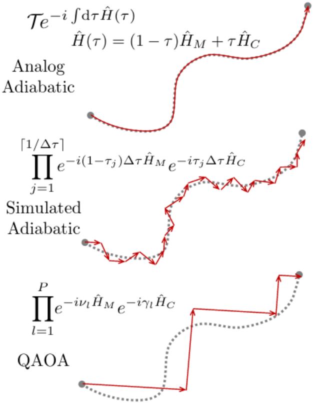
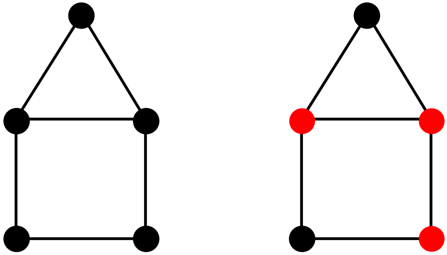

$\mathcal{T}e^{-i\int \mathrm{d}\tau \hat{H}(\tau)}$
$\hat{H}(\tau) = (1-\tau)\hat{H}_M + \tau\hat{H}_C$
Analog
Adiabatic
$\prod_{j=1}^{\lceil 1/\Delta\tau \rceil} e^{-i(1-\tau_j)\Delta\tau\hat{H}_M} e^{-i\tau_j\Delta\tau\hat{H}_C}$
Simulated
Adiabatic
$\prod_{l=1}^{P} e^{-i\nu_l\hat{H}_M} e^{-i\gamma_l\hat{H}_C}$
QAOA

Image credit: G. Verdon, M. Broughton, J. Biamonte. *A quantum algorithm to train neural networks using low-depth circuits.* `https://arxiv.org/abs/1712.05304`

Motivating example: vertex cover

How do we turn an optimization problem for some graph into a Hamiltonian?

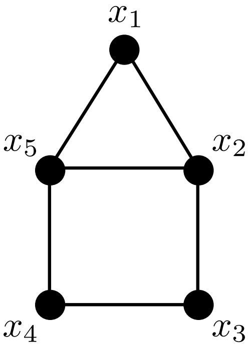
Consider the problem of vertex cover of a graph $G = (V, E)$.



First, define a cost function that is minimized when the colouring is optimal. Then, map it to a Hamiltonian.

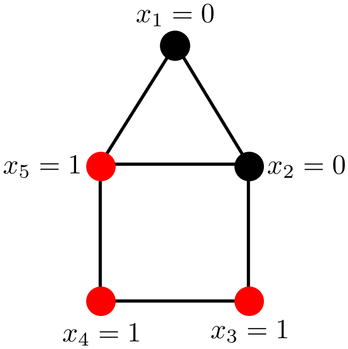
Motivating example: vertex cover

Whether or not a vertex is coloured is a *binary variable*.



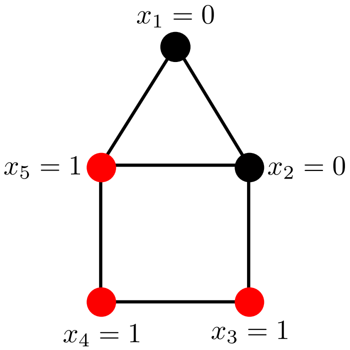
Motivating example: vertex cover

Let's assign coloured vertices to have value 1, and un-coloured 0.



Now that we have our variables, how do we come up with a minimizable cost function that represents the problem?

Motivating example: vertex cover

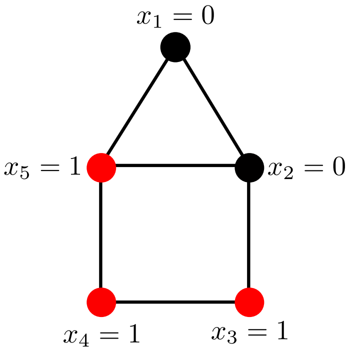


We need every edge to be next to a coloured vertex.

Design a function on two vertices that is minimized when colouring is valid, and maximized if it's not.

$$f(x_i, x_j) = ??$$

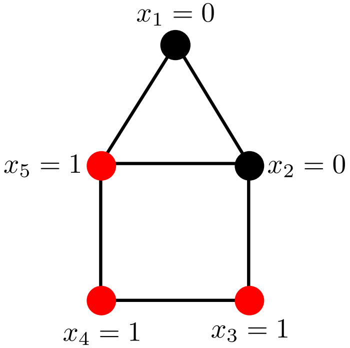
Motivating example: vertex cover



Consider for each edge *ij* the function

The possible values are:

Motivating example: vertex cover

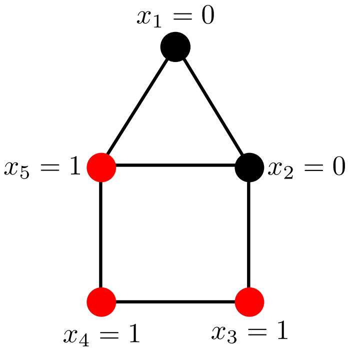


Then in an optimal colouring,

for all edges $ij \in E$.

So we can write

Motivating example: vertex cover



However recall that we also want to colour the fewest vertices. The cost should also depend on the number of coloured vertices.

Solution: add to our cost

Motivating example: vertex cover

The full cost function is then

1. How do we turn this into a Hamiltonian?
2. How do we find its minimum energy / configuration on a quantum computer?

Next time

Content:

- Solving combinatorial optimization problems with QAOA

Action items:

1. Technical assignment 2
2. Work on prototype implementation for project

Recommended reading (PennyLane demos):

- https://pennylane.ai/qml/demos/tutorial_vqe.html
- https://pennylane.ai/qml/demos/tutorial_qaoa_intro.html
- https://pennylane.ai/qml/demos/tutorial_qaoa_maxcut.html