Modeling Electronic Systems on Quantum Computers

University of Florida, October

Recap of last time

- We discussed solving for the ground state $|\Psi_0\rangle$ and ground state energy of a molecule with n electrons,

$$\hat{H} = \sum_{ij} h_1^{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{ijkl} h_2^{ijkl} a_i^\dagger a_j^\dagger a_k a_l$$

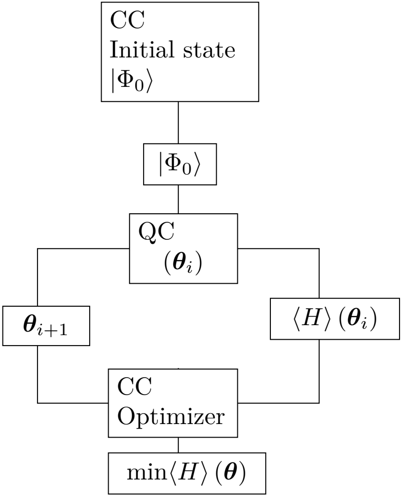
- $a_i^\dagger$ creates a particle in the MO $|\chi_i\rangle$.
- For a calculations on a computer we need to to expand these orbital in some finite set of basis (AO). Let the number of basis be m.
- We use the occupation number representation

$$|n_1, n_2, \ldots, n_{m-1}, n_m\rangle$$

- The Hilbert space dimensions is $\mathcal{O}(e^n)$.
- The Hartree-Fock approximation give us the state with the lowest energy that can also be written as a Slater determinant.

Where quantum computer can help

- Classical computer can do HF efficiently.
- In going beyond HF the bottle necks are:
  1. Sampling the Hilbert space.
  2. Calculating the expectation value of the Hamiltonian.
- A variational quantum eigensolver algorithm (VQE), put the exponentially hard part of the calculation on a quantum computer.



Outline

1. Introduction to circuit model for quantum computing

2. Converting fermionic Hamiltonians to qubit Hamiltonians
   Jordan-Wigner

3. VQE algorithms circuit designs for UCCSD

Outline

1. Introduction to circuit model for quantum computing

2. Converting fermionic Hamiltonians to qubit Hamiltonians
   Jordan-Wigner

3. VQE algorithms circuit designs for UCCSD

Qubits

A qubit lives in a 2 dimensional Hilbert space, $\mathcal{H}^1$. An arbitrary qubit state can be described as,

$$|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle .$$

Qubits

A qubit lives in a 2 dimensional Hilbert space, $\mathcal{H}^1$. An arbitrary qubit state can be described as,

$$|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle .$$

A system of N qubits lives in a 2^N dimensional Hilbert space, $\mathcal{H}^N$ such that,

$$\mathcal{H}^N = \bigotimes_i^N \mathcal{H}_i^1$$

An arbitrary state describing the system of N qubits can be written as,

$$|\Psi\rangle = \sum_{\{s_i\}} c_{1\ldots N} |s_1 \ldots s_N\rangle , \; s_1, \ldots s_N = \{0, \; 1\}$$

Gates

Operators can act on one or two (or more) qubits independently.

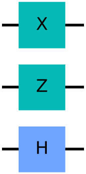
- These operators acting on the qubits are called quantum gates.

Gates

Operators can act on one or two (or more) qubits independently.

- These operators acting on the qubits are called quantum gates.

Examples of 1-qubit gates:



- The X-gate: $X|0\rangle = |1\rangle\,,\; X|1\rangle = |0\rangle$.
- The Z-gate: $Z|0\rangle = |0\rangle\,,\; Z|1\rangle = -|1\rangle$.
- The H-gate:
  $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\,,\; H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.

Gates

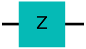
Operators can act on one or two (or more) qubits independently.

- These operators acting on the qubits are called quantum gates.

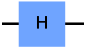
Examples of 1-qubit gates:

- The X-gate: $X|0\rangle = |1\rangle, \; X|1\rangle = |0\rangle$.

- The Z-gate: $Z|0\rangle = |0\rangle, \; Z|1\rangle = -|1\rangle$.

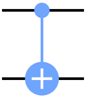


- The H-gate: $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \; H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$.



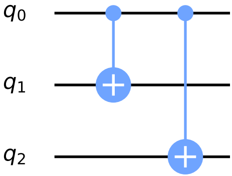
Example of 2-qubit gates:

- Controlled-X gate: $cX|00\rangle = |00\rangle, \; cX|01\rangle = |01\rangle, cX|10\rangle = |11\rangle, \; cX|11\rangle = |10\rangle$. It *adds* the two qubits and puts the result in the second qubit



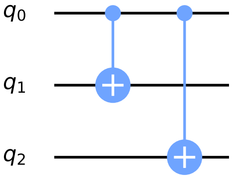
Circuits

Example 1: $|0\rangle \to |000\rangle$, $|1\rangle \to |111\rangle$. (Quantum repetition code.)

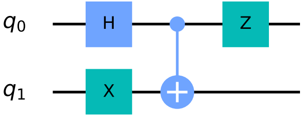


Circuits

Example 1: $|0\rangle \to |000\rangle$, $|1\rangle \to |111\rangle$. (Quantum repetition code.)



Example 2: $|00\rangle \to \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$, (Adds entanglement.)



- The most prevalent model for quantum computations is the circuit model.
- A circuit model is similar to how classical computers operate.

Outline

Fermionic and qubit degrees of freedom

Fermionic system with m orbitals: (2^m dimensional)

$$\{a_i, a_j^\dagger\} = \delta_{ij},\ \{a_i, a_j\} = 0.$$

$$a_i^\dagger \left| n_1 \dots n_i \dots n_m \right\rangle = \begin{cases} \zeta_i \left| n_1 \dots n_i = 1 \dots n_m \right\rangle, & n_i = 0 \\ 0, & n_i = 1 \end{cases}$$

$$\zeta_i = (-1)^{\sum_{j=1}^{i-1} n_j} \quad \text{Parity counting}$$

Fermionic and qubit degrees of freedom

Fermionic system with m orbitals: (2^m dimensional)

$$\{a_i, a_j^\dagger\} = \delta_{ij},\ \{a_i, a_j\} = 0.$$

$$a_i^\dagger |n_1 \dots n_i \dots n_m\rangle = \begin{cases} \zeta_i |n_1 \dots n_i = 1 \dots n_m\rangle, & n_i = 0 \\ 0, & n_i = 1 \end{cases}$$

$$\zeta_i = (-1)^{\sum_{j=1}^{i-1} n_j} \quad \text{Parity counting}$$

Qubits system with m qubits:

$$\sigma^+ = \frac{1}{2}(\sigma_x + i\sigma_y),\ \sigma^- = \frac{1}{2}(\sigma_x - i\sigma_y),$$

$$\{\sigma_i^-, \sigma_j^+\} = 1,\ \{\sigma_i^-, \sigma_j^-\} = 0, \quad i = j,$$

$$[\sigma_i^-, \sigma_j^+] = 0, \quad [\sigma_i^-, \sigma_j^-] = 0, \quad i \neq j$$

$$\sigma_i^+ |s_1 \dots s_i \dots s_m\rangle = \begin{cases} |n_1 \dots s_i = 1 \dots n_m\rangle, & s_i = 0 \\ 0, & s_i = 1 \end{cases}$$

Jordan-Wigner transformation

We need a map from the fermionic operators the qubits operators.

It seems reasonable to try,

$$a_i^\dagger = \sigma_i^+, \quad a_i = \sigma_i^-.$$

This doesn't produce the parity counting (the phase ζ_i) that the fermionic operators generate when acting on basis states.

Jordan-Wigner transformation

We need a map from the fermionic operators the qubits operators.

It seems reasonable to try,

$$a_i^\dagger = \sigma_i^+, \quad a_i = \sigma_i^-.$$

This doesn't produce the parity counting (the phase ζ_i) that the fermionic operators generate when acting on basis states.
Jordan-Wigner transformation:

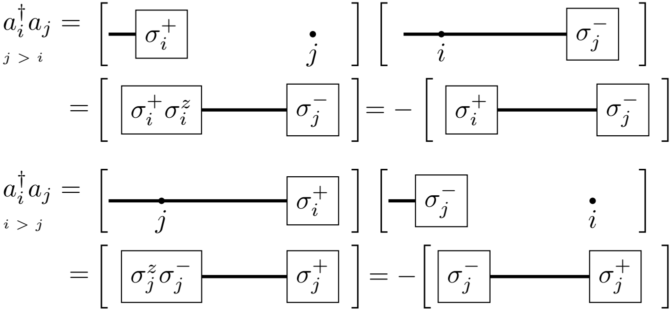
$$a_i^\dagger = \prod_{j=1}^{i-1} \sigma_j^z \ \sigma_i^+ \quad a_i = \prod_{j=1}^{i-1} \sigma_j^z \ \sigma_i^-.$$

$$\prod_{j=1}^{i-1} \sigma_j^z \ \sigma_i^+ \left| s_1 \dots s_i \dots s_m \right\rangle = \begin{cases} \zeta_i \left| n_1 \dots s_i = 1 \dots n_m \right\rangle, & s_i = 0 \\ 0, & s_i = 1 \end{cases}$$

Opertators transformation

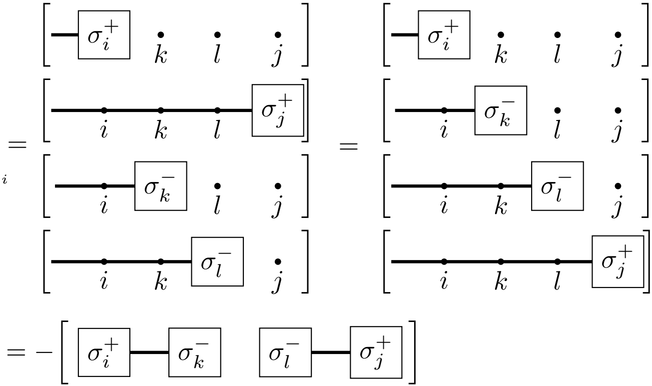
$$a_i^\dagger = \rule[0.5ex]{4em}{1pt}\boxed{\sigma_i^+} \qquad a_i = \rule[0.5ex]{4em}{1pt}\boxed{\sigma_i^-}$$

Opertators transformation



Opertators transformation

$$a_i^\dagger a_j^\dagger a_k a_l =$$
$$_{j > l > k > i}$$



The qubit Hamiltonian

Remember the fermionic Hamiltonian is,

$$\hat{H} = \sum_{ij} h_1^{ij} a_i^\dagger a_j + \frac{1}{2} \sum_{ijkl} h_2^{ijkl} a_i^\dagger a_j^\dagger a_k a_j$$

- For our minimal model for the Hydrogen molecule we have 4 basis, and hence the system will be represented by 4 qubits.
- After Jordan-Wigner transformation the uibit Hamiltonian will look something like this,

$$\hat{H} = \sum_{i,j,k,l=0}^{3} j^{ijkl} \sigma_1^i \sigma_2^j \sigma_3^k \sigma_4^l$$

To measure the $\langle \hat{H} \rangle$, our quantum computer will need to measure $\langle \sigma_1^i \sigma_2^j \sigma_3^k \sigma_4^l \rangle$.

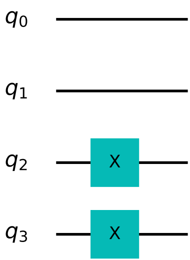
- There are other proposed maps.

Setting up the quantum circuit

For the Hydrogen example.

$$|\Phi_0\rangle = |1100\rangle$$

$$a_i^\dagger a_i = \frac{1}{2}(1 + \sigma_i^z).$$



Outline

1. Introduction to circuit model for quantum computing
2. Converting fermionic Hamiltonians to qubit Hamiltonians
   - Jordan-Wigner
3. VQE algorithms circuit designs for UCCSD

Unitary coupled-cluser (UCC) expansion

The UCC method uses the following ansatz,

$$|\Psi_0\rangle = e^{T(\boldsymbol{\theta})-T^\dagger(\boldsymbol{\theta})}\,|\Phi_0\rangle$$

$$T(\boldsymbol{\theta}) = T_1(\boldsymbol{\theta}) + T_2(\boldsymbol{\theta}) + \dots$$

$$T_1(\boldsymbol{\theta}) = \sum_{\substack{m\in\text{emp}\\ i\in\text{occ}}} \theta^{mi} a_m^\dagger a_i$$

$$T_2(\boldsymbol{\theta}) = \frac{1}{2}\sum_{\substack{m,n\in\text{emp}\\ i,j\in\text{occ}}} \theta^{mnij} a_m^\dagger a_n^\dagger a_i a_j$$

$$\vdots$$

$$e^{T(\boldsymbol{\theta})-T^\dagger(\boldsymbol{\theta})} = e^{\sum_i \theta_i(\tau_i-\tau_i^\dagger)},$$

where τ_i include all kinds of excitations.

Unitary coupled-cluser (UCC) expansion

The UCC method uses the following ansatz,

$$|\Psi_0\rangle = e^{T(\boldsymbol{\theta})-T^\dagger(\boldsymbol{\theta})}|\Phi_0\rangle$$
$$T(\boldsymbol{\theta}) = T_1(\boldsymbol{\theta}) + T_2(\boldsymbol{\theta}) + \dots$$
$$T_1(\boldsymbol{\theta}) = \sum_{\substack{m\in\text{emp}\\ i\in\text{occ}}} \theta^{mi} a_m^\dagger a_i$$
$$T_2(\boldsymbol{\theta}) = \frac{1}{2}\sum_{\substack{m,n\in\text{emp}\\ i,j\in\text{occ}}} \theta^{mnij} a_m^\dagger a_n^\dagger a_i a_j$$
$$\vdots$$

$$e^{T(\boldsymbol{\theta})-T^\dagger(\boldsymbol{\theta})} = e^{\sum_i \theta_i(\tau_i-\tau_i^\dagger)},$$

where τ_i include all kinds of excitations.

- For this ansatz to be computationally efficient we need to include only a few excitation operators.
- UCCSD include single and double excitations.
- Next step would be to design a circuit that implement this unitary operation.

Trotterization

Different terms of τ_i don't necessarily commute which make simulating the exponential with quantum circuits not easy.
We make use of the following identity,

$$e^{\sum_i \theta_i(\tau_i - \tau_i^\dagger)} = \lim_{N\to\infty} \left(\prod_i e^{\frac{\theta_i(\tau_i - \tau_i^\dagger)}{N}} \right)^N$$

Trotterization

Different terms of τ_i don't necessarily commute which make simulating the exponential with quantum circuits not easy.
We make use of the following identity,

$$e^{\sum_i \theta_i(\tau_i - \tau_i^\dagger)} = \lim_{N\to\infty} \left(\prod_i e^{\frac{\theta_i(\tau_i - \tau_i^\dagger)}{N}} \right)^N$$

- Taking N to infinity is not possible for computations. Good results can be achieved by taking just a few trotter steps.

$$e^{\sum_i \theta_i(\tau_i - \tau_i^\dagger)} = \left(\prod_i e^{\frac{\theta_i(\tau_i - \tau_i^\dagger)}{\rho}} \right)^\rho$$

- The smaller the θ_i's are the better the approximation.
- It's better to start with a good initial guess.

Converting to qubits operations

one-body exponentials

One body exponential $e^{\theta(a_m^\dagger a_i - a_i^\dagger a_m)}$. Note: $m > i$.

$$a_m^\dagger a_i - a_i^\dagger a_m = -\sigma_i^- \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^+ + \sigma_i^+ \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^-$$

$$a_m^\dagger a_i - a_i^\dagger a_m = \frac{i}{2} \left[\sigma_i^y \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^x - \sigma_i^x \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^y \right]$$

Note that the two terms commute.

Converting to qubits operations

one-body exponentials

One body exponential $e^{\theta(a_m^\dagger a_i - a_i^\dagger a_m)}$. Note: $m > i$.

$$a_m^\dagger a_i - a_i^\dagger a_m = -\sigma_i^- \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^+ + \sigma_i^+ \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^-$$

$$a_m^\dagger a_i - a_i^\dagger a_m = \frac{i}{2} \left[\sigma_i^y \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^x - \sigma_i^x \left(\prod_{s=i+1}^{m-1} \sigma_s^z \right) \sigma_m^y \right]$$

Note that the two terms commute.
We rotate σ^x and σ^y to σ^z,

$$e^{i\theta\sigma_i^y \left(\prod_{s=i+1}^{m-1} \sigma_s^z\right)\sigma_m^x} = H_m R_i^x(\pi/2) \left(e^{i\theta\sigma_i^z \left(\prod_{s=i+1}^{m} \sigma_s^z\right)} \right) H_m^\dagger R_i^{x\dagger}(\pi/2),$$

using,

$$H\sigma_z H^\dagger = \sigma_x$$

$$R^x(\pi/2)\,\sigma_z R^{x\dagger}(\pi/2) = \sigma_y$$

Converting to qubit operations

one-body exponentials

Need to implement $\exp\left[i\theta\sigma_i^z\left(\prod_{s=i+1}^{m}\sigma_s^z\right)\right]$.

- $\prod_{s=i+1}^{m}\sigma_s^z$ counts the parity between qubits i and m. The answer is plus or minus.
- $e^{\pm i\theta\sigma_i^z}$ makes a rotation around the z-axis on the i-th qubit accordingly.

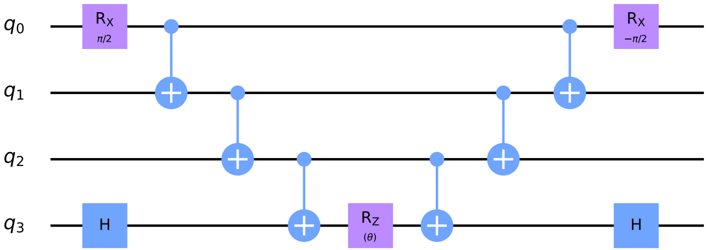
Converting to qubit operations

one-body exponentials

Need to implement $\exp\left[i\theta\sigma_i^z\left(\prod_{s=i+1}^{m}\sigma_s^z\right)\right]$.

- $\prod_{s=i+1}^{m}\sigma_s^z$ counts the parity between qubits i and m. The answer is plus or minus.
- $e^{\pm i\theta\sigma_i^z}$ makes a rotation around the z-axis on the i-th qubit accordingly.

Example: $\exp\left[i\theta\sigma_1^y(\sigma_2^z\sigma_3^z)\sigma_4^x\right]$ for Hydrogen molecule.



Converting to qubit operations

two-body exponentials

Two-body exponential $e^{\theta(a_m^\dagger a_n^\dagger a_j a_i - a_i^\dagger a_j^\dagger a_n a_m)}$, and assume $j > i$, and $m > n$

$$
\begin{aligned}
a_m^\dagger a_n^\dagger a_i a_j - a_i^\dagger a_j^\dagger a_n a_m =& \sigma_i^- \left(\prod \sigma^z\right) \sigma_j^- \ \sigma_n^+ \left(\prod \sigma^z\right) \sigma_m^+ \\
&- \sigma_i^+ \left(\prod \sigma^z\right) \sigma_j^+ \ \sigma_n^- \left(\prod \sigma^z\right) \sigma_m^- \\
=& \frac{i}{4} \sigma_i^x \left(\prod \sigma^z\right) \sigma_j^x \ \sigma_n^x \left(\prod \sigma^z\right) \sigma_m^y \\
&+ \text{terms with odd } \sigma^y
\end{aligned}
$$

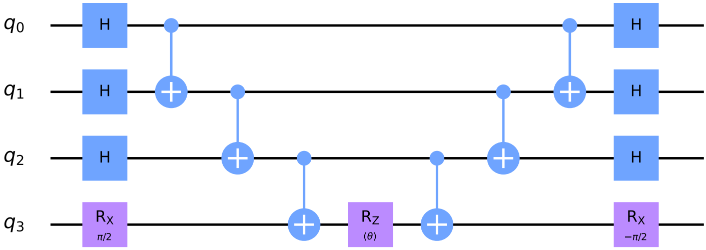
This generate 8 terms, all of which will commute. As before we rotate σ^x and σ^y to σ^z,

$$
H_i H_i H_n Y_m e^{\frac{i\theta}{4}\left(\sigma_i^z \left(\prod_{i+1}^{j} \sigma^z\right) \ \left(\prod_n^m \sigma^z\right)\right)} H_i^\dagger H_i^\dagger H_n^\dagger Y_m^\dagger
$$

Converting to qubit operations

two-body exponentials

Note we only add parity from m to n and from j to $i+1$ then perform the rotation on the i-th qubit.
Example: $\exp\{[i\theta\sigma_1^x\sigma_2^x\sigma_3^x\sigma_4^y\ ]\}$ for the Hydrogen molecule,



Measuerment

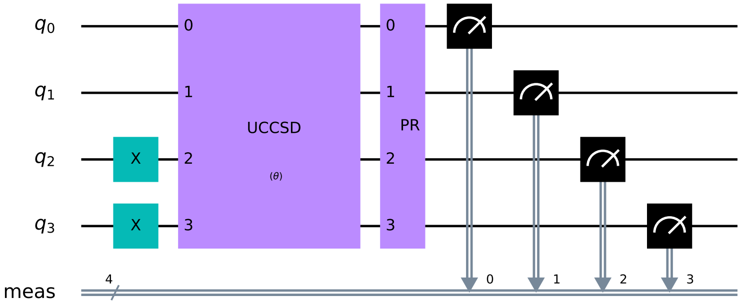
The Hamiltonian expectation value is written as,

$$\langle \hat{H} \rangle = \sum_{i,j,k,l=0}^{3} j^{ijkl} \langle \sigma_1^i \sigma_2^j \sigma_3^k \sigma_4^l \rangle$$

The computational basis is the z-axis. To measure the expectation value a general string of Pauli operators, we need a post-rotation circuit.

- Measuring the qubits gives a string of 0's and 1's, $\{s_i\}$.
- $P(\{s_i\}) = |\langle \{s_i\} | \Psi_0 \rangle|^2$
- To measure the i-th qubit with respect to the x or y-axes we need to rotate our basis for the i-th qubit using H_i or $R_i^x(\pi/2)$ respectively.
- Any Pauli string has eigenvalues of either ± 1.
- $\langle \sigma_1^i \sigma_2^j \sigma_3^k \sigma_4^l \rangle = P(1) - P(-1)$

The end result look something like this:



1. Initialize the qubits
2. Apply the UCCSD gate with parameters $\boldsymbol{\theta}$
3. Apply post-rotations .
4. Measure the qubits.

Comparison:

$E_{\text{HF}} = -1.116$

$E_{\text{UCCSD}} = -1.137$

$E_{exact} = -1.166$

Summary

The VQE method can be summarized as follows:

1. A classical computer calculate the Hamiltonian and come up with an initial state.
2. A quantum computer samples the Hilber space and measure the energy.
3. A classical computer will run an optimization routine to minimize the energy expectation value.