Variational Quantum Eigensolver (VQE)

Julie Butler

Sidetrack: Implementing QPE for the Two-Particle Two-Hole Case

Pairing Model Review (Two-Particle Two-Hole Case)

$$H_{2p2h} = \begin{bmatrix} \frac{-g}{2} & 0 \\ 0 & 2d - \frac{g}{2} \end{bmatrix}$$

- 2x2 matrix could be an operator on a single qubit
- QPE algorithm can find the eigenvalues of a 2x2 matrix if we know the eigenvectors
  - Eigenvectors of H_{2p2h} are $|\uparrow\rangle$ and $|\downarrow\rangle$

Possible Problem

- Hamiltonian matrices must be
  - Hermitian
- Quantum Gates must be
  - Unitary
- Is the two-particle two-hole Hamiltonian unitary?

Definition

- A matrix is Hermitian if $A = A^\dagger = (A^T)^*$
- A matrix is Unitary if $AA^\dagger = \mathbf{I}$
- Hermitain and Unitary are related, but not the same thing
- Is $H_{2p2h} Hermitian?
- Is $H_{2p2h} Unitary?

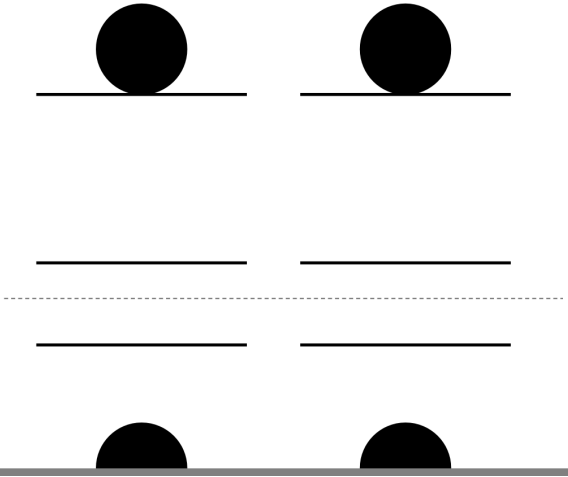
We cannot simply use QPE to find the eigenvalues of the pairing model then. Thus we need to look at other options.

Representing Quantum States as Quantum Circuits

Pairing Model Review (Four-Particle Four Hole Case)

$$H_{4p4h} = \begin{bmatrix} 2d-g & \frac{-g}{2} & \frac{-g}{2} & \frac{-g}{2} & \frac{-g}{2} & 0 \\ \frac{-g}{2} & 4d-g & \frac{-g}{2} & \frac{-g}{2} & 0 & \frac{-g}{2} \\ \frac{-g}{2} & \frac{-g}{2} & 6d-g & 0 & \frac{-g}{2} & \frac{-g}{2} \\ \frac{-g}{2} & \frac{-g}{2} & 0 & 6d-g & \frac{-g}{2} & \frac{-g}{2} \\ \frac{-g}{2} & 0 & \frac{-g}{2} & \frac{-g}{2} & 8d-g & \frac{-g}{2} \\ 0 & \frac{-g}{2} & \frac{-g}{2} & \frac{-g}{2} & \frac{-g}{2} & 10d-g \end{bmatrix}$$

States of the Pairing Model



What are the Eigenvectors?

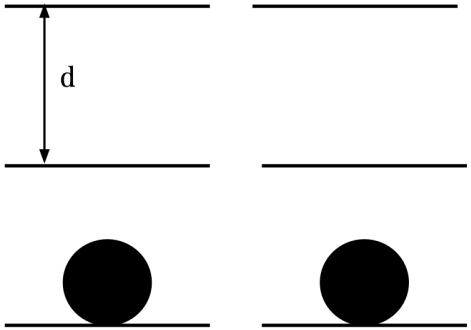
- **To Jupyter!**
- The eigenvectors (or states) of the four-particle four-hole pairing model are not regular vectors. Would be hard to represent with qubits.
- What if there was another way?

Introduction to Second Quantization

- Second quantization is a method of representing the states of a system using a binary representation.
  - Filled states are represented with a 1
  - Empty states are represetned with a 0

Termonology: Fermi Level

- The Fermi level is an imaginary energy level that exists between single particle states that are filled in the ground state and single particle states that are empty in the ground state.



Indexing the Pairing Model

- Consider the ground state of the four-particle four-hole pairing model
  - Single particle states that are filled in the grounds state are indexed with i,j,k, ... OR $i_1, i_2, i_3, ...$
  - Single particle states that are empty in the ground state are indexed with a,b,c,... OR $a_1, a_2, a_3, ...$
  - Any single particle state in the ground state can be indexed with p,q,r,... OR $p_1, p_2, o_3, ...$

Pairing Model States in Second Quantization

- For the four-particle four-hole pairing model, we can represent the six possible states using the following vectors, where the first one is the ground state:

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} \longrightarrow [1//1//0//0], \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}$$

Creation Operators

- A creation operator creates a particle in a state if it does not already contain a particle
  - Applying a creation operator to a state that already has a particle makes the entire equation 0
- Note that for the pairing model we simply things by creating an entire level of particles at a time, but in general a creation operator only creates a single particle.
- We can create the ground state for the four-particle four-hole pairing model using:

$$a_{p_1} a_{p_2} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |\Phi_0\rangle$$

Annihilation Operators

- An annihilation operator removes a particles from a state that already contains a particle
  - Applying an annihilation operator to a state that does not contain a particle makes the entire equation 0
- Note that for the pairing model we simply things by destroying an entire level of particles at a time, but in general a annihilation operator only destroy a single particle.
- We can turn the ground state back into the vacuum state using the following:

$$a^\dagger_{p_1} a^\dagger_{p_2} \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = |0\rangle$$

Creation and Annihilation Operators in Combination

- Creation and annihilation operators are used to convert one state into another state

$$a_{p_2} a_{p_3} a_{p_1}^\dagger a_{p_2}^\dagger \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \end{bmatrix} = |\Phi_5\rangle$$

Representing the States as Quantum Circuits

- **To Jupyter!**

Introduction to the Variational Quantum Eigensolver (VQE) Algortihm

Variational Quantum Eigensolver (VQE)

- Based on the variational method
- Used to find the ground state (or at least an approximation to it) of a quantum system using a hybrid quantum-classical algortihm
- From the ground state can determine the ground state energy: $\langle \Psi_0 | H | \Psi_9 \rangle$

Why is it a hybrid algorithm?

- A quantum computer is good at simulating a quantum systems (because it is a quantum system)
- A quantum computer is not very good at optimization.
- VQE, like the variational method, is an **iterative** algorithm that requires **optimization/minimization**
  - The minimization is done classically and used to update the quantum parameters

Noisy Intermediate Scale Quantum (NISQ) Era

- The VQE is referred to as a noisy untermediate scale quantum (NISQ) algorithm
  - This is the current state/era of quantum computing
- Current quantum processors contain up to 1,000 qubits but are not yet **fault-tolerant** nor large enough for many commerical applications
  - A fault-tolerate quantum computer would be immune, or at least minimally affected, by a noisy environment
- We are not yet capable of performing continuous error correction
- As a rough comparison to classical computing, quantum technology is now where classical computers were in the 1940s and 1950s.

Related Algorithms

Hybrid algorithms that involve simulating a system/problem on a quantum computer and then performing optimization on a classical computer. Most applications are in quantum physics/chemistry, optimization, and machine learning.

- Variational Quantum Eigensolver (VQE)
- Quantum Adiabatic Optimization Algorithm (QAOA)
- Quantum Neural Networks (QNN)
- Quantum Support Vector Machines (QSVM)
- Variational Quantum Classifier (VQC)
- Quantum GAN (QGAN)

Other Applications

- VQE can be thought of as a general minimization algorithm if a problem can be phrased in such a way that finding the lowest eigenvector or eigenvalue if a matrix would provide the answer.

Pros and Cons

- Pro: less gates are needed when compared to QPE and less error prone
- Pro: Can find eigenvalues of non-unitary matrices
- Pro: VQE can run on current hardware
- Pro: VQE can be applied to a wide variety of problems outside of chemistry and physics
- Con: Convergence is not guaranteed and results depend highly on initial guess
- Con: There are many measurements required and the number of measurements scales with the size of the Hamiltonian

VQE Steps

1. **Hamiltonian Preparation:** Define the Hamiltonian of the system then write it in terms of the Pauli operators.
2. **Create an Ansatz**: Create an iniital guess for the ground state of the quantum system, called an ansatz. It must be parameterized by a series of parameters called θ. It can be hardware inspired or physically inspired.
3. **Prepare the Quantum Circuit:** Initialize the quantum system into a simple reference state and then apply the ansatz with specific values for θ. The Hamiltonian is then "added" to the circuit using its Pauli operator form generated in step 1.

VQE Steps Continued

4. **Measure the Expectation Value:** Run the circuit many times and get a measurement for each one. This is an estimation for the energy with the given set of parameters θ.
5. **Classical Optimization**: Using a classical optimization algorithm, minimize the energy with repect to θ.

Repeat 3-5 until the result as converged.

Applying the VQE to an Example System

A Review of Quantum Gates

$$X = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

A Review of Quantum Gates (Part 2)

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

- Note that $X = HZH$

$$S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

- Note that $Y = SXS^\dagger = SHZHS^\dagger$

Quantum Gates with VQE

Any Hamiltonian can be written as a weighted sum of Pauli strings, P_n (c_n are constants):

$$H = \sum_n c_n P_n$$

The Pauli strings are defined as follows, where σ_k is X, Y, Z, or I:

$$P_n = \bigotimes_k \sigma_k$$

The above notation is equivalent to a sum but with tensor products instead of addition

Finding the Energy with Quantum Gates

$$E = \langle H \rangle = \langle \sum_n c_n P_n \rangle = \sum_n c_n \langle P_n \rangle$$

Example System

Consider a quantum mechanical system which can be described by the following Hamiltonian, where A is a constant:

$$H = \frac{A}{2}(X_0 \otimes X_1 + Y_0 \otimes Y_1)$$

Then when we measure the energy of the system, we have:

$$E = \langle E \rangle = \frac{A}{2}(\langle X_0 \otimes X_1 \rangle + \langle Y_0 \otimes Y_1 \rangle)$$

So we need to figure out a way to measure the expectation values in the XX and YY basis. **To Jupyter!**

Back to the Hamiltonian

If we create the Hamiltonian (see Jupyter) we see that it has the following form:

$$H = \frac{A}{2}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Reminder of the One Qubit States

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Reminder of the Two Qubit States

$$|00\rangle = |0\rangle \otimes |0\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad |01\rangle = |0\rangle \otimes |1\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$|10\rangle = |1\rangle \otimes |0\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \quad |11\rangle = |1\rangle \otimes |1\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Developing a Trial Wavefunction

- Note that $H|00\rangle = H|11\rangle = 0$ but $H|01\rangle = \frac{A}{2}|01\rangle$ and $H|10\rangle = \frac{A}{2}|10\rangle$. So our wavefunction is likely made of $|01\rangle$ and $|10\rangle$.
- Consider the following: $|\psi_{trial}\rangle = a|01\rangle + b|10\rangle$, where $|a|^2 + |b|^2 = 1$
- Let's let $a = \frac{1}{\sqrt{2}}$ and $b = \frac{e^{i\theta}}{2}$ (verify for yourself that this is normalized)
  - θ is the parameter to be optimized by the VQE

Hamming Weights and Expectation Values

- For both $\langle XX \rangle$ and $\langle YY \rangle$ we will estimate its value using the same function:

$$\langle P \rangle = \frac{\sum_x h(x) N_x}{N}$$

- In the above equation $\langle P \rangle$ represents any expectation value, N is the total number of measurements made of the circuit, N_x is the total number of measurements made of state $|x\rangle$, and h(x) is the Hamming weight of state $|x\rangle$
  - Hamming weight: +1 if there are an even number of 1's, -1 if there are an odd number of ones
  - $h(|01\rangle) = h(|10\rangle) = -1$ but $h(|00\rangle) = h(|11\rangle) = 1$

Variational Quantum Eigensolver for Example System

To Jupyter!

Applying the VQE to the Pairing Model

Pairing Model Hamiltonian

- The pairing model Hamiltonian (seen previously) is a matrix made up of two terms:

$$H = H_1 + H_2$$

- H_1 determines the energy which comes from each individual particle, assuming that they do not interact with each other
- H_2 determines the energy which comes from each pair of particles, as they interact with each other

Hamiltonians Generalized

- In general a Hamiltonian for a system is a sum of N+1 terms for an N-particle system, where K is the kinetic energy (the energy the particles have from moving) and H_i is the energy which comes from all groups of i particles interacting:

$$H = K + H_1 + H_2 + H_3 + ... + H_N$$

- However, due to model constraints and computational restrictions, typically we will only consider a smaller subset of these terms.

One-Body Term

Two-Body Term

One-Body Term in Second Quantization and Pauli Operators

Two-Body Term in Second Quantization and Pauli Operators

Ansatz

Linear Connectivity vs. All-to-All Connectivitiy

SWAP Network

Summary of VQE Process for Pairing Model

To Jupyter!