FERMIONIC MANY-BODY PROBLEMS ON A QUANTUM COMPUTER

ROK MLINAR VAHTAR

Fakulteta za matematiko in fiziko
Univerza v Ljubljani

Many-body quantum systems represent a formidable computational challenge due to the exponential growth of the Hilbert space. This paper discusses how fermionic many-body problems can be solved using a hybrid quantum-classical approach on contemporary noisy quantum computers. We explain the mapping of fermionic problems onto qubits and qubit operations, followed by a discussion of two methods for evaluating Green's functions. We first use the Variational Quantum Eigensolver for ground-state preparation and then Trotterization and the McLachlan variational principle for real-time evolution.

FERMIONSKI VEČDELČNI PROBLEMI NA KVANTNEM RAČUNALNIKU

Kvantni večdelčni problemi predstavljajo velik izziv za sodobne računske metode, saj se klasični računalniki težko soočajo z eksponentno rastjo Hilbertovega prostora. V članku je predstavljena hibridna kvantno-klasična metoda, ki nam omogoči, da takšne probleme rešujemo na trenutno obstoječih kvantnih računalnikih. Začnemo z razlago preslikave fermionskega problema na kubite in kubitne operatorje, nato pa predstavimo dva pristopa za izračun Greenovih funkcij. Ta sestojita iz metode VQE (angl. Variational Quantum Eigensolver), za pripravo osnovnega stanja in iz Troterizacije in McLachlanovega variacijskega principa za propagacijo stanja v času.

1. Introduction

Interactions between fermions lead to many interesting phenomena such as magnetism, superconductivity, Mott insulating states, and many others. Consequently, the study of many-body fermionic systems is an important yet challenging subject [1]. Since the Hilbert space of such problems grows exponentially with the number of particles, most methods used to solve them either rely on very restrictive approximations or are limited to a small number of particles [2]. This motivates the search for quantum computational methods to solve these problems, especially those that can be used on contemporary noisy quantum computers.

2. Many-body fermionic models

Many-body fermionic models are expressed in terms of a Hamiltonian in the Fock basis, which consists of fermionic creation ($c^\dagger$) and annihilation (c) operators. A good example is the Hubbard model

$$H = -t \sum_{i,\sigma} \left(c^\dagger_{i,\sigma} c_{i+1,\sigma} + c^\dagger_{i+1,\sigma} c_{i,\sigma} \right) + U \sum_i n_{i,\uparrow} n_{i,\downarrow},$$

where $n_{i,\sigma} = c^\dagger_{i,\sigma} c_{i,\sigma}$ is the number operator, j is the index that represents a particle's position and $\sigma \in \{\uparrow, \downarrow\}$ is the particle's spin. It is used to describe the Mott transition [3] of a metal to an insulator, which is a not yet fully understood phenomenon with many potential applications from electrical components to high-temperature superconductors. Even with other more complex models, the Hamiltonian will still be a sum of products of creation and annihilation operators, which permits us to solve them generally using the same method, but before that, we need to establish what solving a model even means in this context.

Ideally, one would desire to obtain a model's eigenstates as an expansion in some basis, as this would give us all the information about it. However, even if the calculation would not take an

impossible amount of time, just to store the expansion coefficients would be a challenge. Thankfully, most of this information is superfluous and most relevant properties of many-body systems can be expressed in terms of correlation functions between creation and annihilation operators called Green's functions. We are going to focus on single-particle Green's functions, defined as

$$G^R_{ab}(t) = -i\theta(t)\left\langle \psi_0 \left| \left\{ c_a(t), c_b^\dagger(0) \right\} \right| \psi_0 \right\rangle ,$$

where $c_a(t) = e^{iHt} c_a e^{-iHt}$ is the Heisenberg picture annihilation operator at time t and ψ_0 is the ground state. These tell us how the system reacts when a particle is added or removed from the ground state as illustrated in Figure 1 and therefore hold all the relevant information about single-particle excitations.

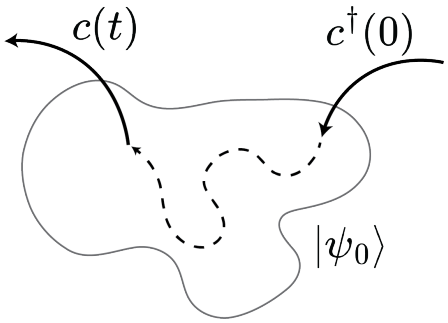

Figure 1. Illustration of how the single-particle Green's function is related to single-particle excitations in the ground state, where the $c_a(t)c_b^\dagger(0)$ term represents particle excitations and $c_b^\dagger(0)c_a(t)$ hole excitations. The arrow pointing into the system represents creation of a particle and the arrow pointing outwards its annihilation.

An example of an observable, linked to a Green's function of this kind, is the spectral function $A(\omega) = -\frac{1}{\pi}\sum_a \mathrm{Im} G^R_{aa}(\omega)$, where $G^R_{aa}(\omega)$ is the Fourier transformed Green's function, which is the many-body generalization of the density of states [4].

Similarly, most macroscopic quantities such as magnetization and conductivity can be connected to a correlation function of some order and can be calculated using the method we are going to describe. This is often done using linear response theory, specifically the Kubo formula [4].

3. Jordan-Wigner transformation

Most current quantum processors natively support only a restricted set of operations. For gate-based quantum computers these usually include the Pauli operators of a spin-$\frac{1}{2}$ system σ^x, σ^y and σ^z. Consequently, fermionic creation and annihilation operators must be expressed in terms of the Pauli operators through a suitable mapping. There exist several ways to do this ranging in complexity and with different advantages [2]. We will only consider the simplest method, the Jordan-Wigner transformation. We are going to use the notation σ^*_n where $* \in \{x, y, z, +, -\}$ for Pauli operators on the n-th qubit.

The obvious choice for mapping creation operators are the raising and lowering operators $\sigma^+ = \frac{1}{2}(\sigma^x + i\sigma^y)$ and $\sigma^- = \frac{1}{2}(\sigma^x - i\sigma^y)$ as they already behave like fermionic creation and annihilation operators on single qubits:

$$\begin{aligned} c|0\rangle &= 0 & \sigma^-|\downarrow\rangle &= 0 \\ c|1\rangle &= |0\rangle & \sigma^-|\uparrow\rangle &= |\downarrow\rangle \\ c^\dagger|0\rangle &= |1\rangle & \sigma^+|\downarrow\rangle &= |\uparrow\rangle \\ c^\dagger|1\rangle &= 0 & \sigma^+|\uparrow\rangle &= 0\,, \end{aligned}$$

if we take the $|\downarrow\rangle$ state to be the empty state and $|\uparrow\rangle$ to represent the state filled with one fermion.

The problem arises when multiple qubits are considered. Pauli operators on different qubits commute $\sigma_i^{\pm}\sigma_j^{\pm} = \sigma_j^{\pm}\sigma_i^{\pm}$, as these are just matrices applied to separate qubits, so order does not matter. The same logic does not apply to fermionic operators, which famously anticommute $c_i c_j = -c_j c_i$. For the mapping to work, additional operators need to be added, to ensure anticommutation is preserved. The Jordan-Wigner transformation uses the fact that $\sigma^{\pm}$ both anticommute with σ^z. Therefore, if the mapping of the creation operators functions in such a way, that when calculating anticommutators, exactly one of the $\sigma^{\pm}$ operators intersects with a σ^z, the whole operators will anticommute.

For an example we can look at the Jordan-Wigner transformation between a 3-site fermionic model and a system of 3 qubits:

$$\begin{aligned} c_1 &= \sigma_1^- \otimes \mathbb{1}_2 \otimes \mathbb{1}_3 \\ c_2 &= \sigma_1^z \otimes \sigma_2^- \otimes \mathbb{1}_3 \\ c_3 &= \sigma_1^z \otimes \sigma_2^z \otimes \sigma_3^-. \end{aligned}$$

For every anticommutator $\{c_i, c_j\}$, operators on two of the qubits commute, namely $[\sigma^-, \mathbb{1}] = [\sigma^z, \mathbb{1}] = [\sigma^z, \sigma^z] = 0$ and anticommute on exactly one with $\{\sigma^z, \sigma^-\} = 0$. Taken as a whole, the mapped creation operators therefore anticommute.

This structure can easily generalize to other sizes of many-site systems, where for readability we drop the $\otimes$ symbol, giving us

$$c_k^\dagger = \sigma_k^+ \prod_{j=1}^{k-1} \sigma_j^z \qquad c_k = \sigma_k^- \prod_{j=1}^{k-1} \sigma_j^z \qquad c_k^\dagger c_k = \sigma_k^+ \sigma_k^-,$$

which is the general formula for the Jordan-Wigner transformation.

4. Ground-state preparation

To run any calculation on contemporary quantum computers, we have to be clever about how we use them. The main issue we have to deal with is decoherence, the constant buildup of noise. For gate-based quantum computers, this means there is an upper limit on the number of gates one can use sequentially. This is a problem, as the direct methods that could be used to prepare the ground state require a lot of gates, rendering them unusable for now.

To circumvent this problem, the idea is to split the calculation into several smaller pieces, so that each piece requires a comparatively small circuit depth. These can then be combined using a classical computer, trading the problem of deep circuits for the problem of combining multiple results, which can be computationally taxing. This, however, is actually more difficult than it might seem at first. One cannot simply split any method into parts and then plug the result of one quantum circuit into another, as this would not decrease the circuit depth. For this reason, the pieces have to be completely independent calculations.

This can be achieved by using variational methods. They work based on the variational principle of quantum mechanics, which, in the context of looking for the ground state, means that it can be found via minimization of the energy functional $\langle\psi|H|\psi\rangle$, as it is, by definition, the state with the lowest energy. The specific method that utilizes this approach is called the Variational Quantum Eigensolver, usually shortened to VQE.

4.1 Variational Quantum Eigensolver (VQE)

The Variational Quantum Eigensolver [5] uses an ansatz circuit $U(\theta_1, \theta_2, \dots)$ to prepare a wide variety of states $|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\varphi_0\rangle$, from a fixed initial state $|\varphi_0\rangle$. The solver functions as a hybrid

quantum-classical algorithm. It finds the optimal parameters $\{\theta_i\}$ of the ansatz by using some classical minimization algorithm such as gradient descent. For each iteration of the minimization algorithm, the energy and gradients are evaluated on a quantum computer and then the result is used by a classical computer to improve the parameters for the next iteration, as illustrated in Figure 2. This approach keeps all the difficult wavefunction operations on the quantum computer and ensures each individual calculation is short and independent of the others.

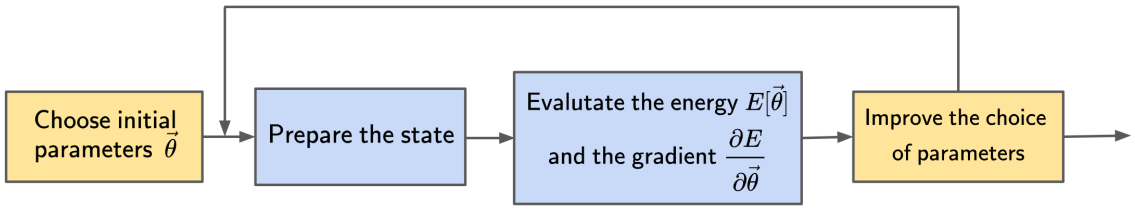

Figure 2. Diagram showing the structure of the VQE algorithm, where the steps in yellow are run on a classical computer and the blue steps on a quantum computer.

For this algorithm to work, we now only need a way to evaluate the energy and its gradient and to see, how an ansatz circuit can be made.

4.2 The ansatz circuit

For the described VQE algorithm to function, the ansatz circuit needs to be able to produce states in the vicinity of the ground state, otherwise the minimization will return the lowest energy state from the states accessible to it, but not the ground state. Since quantum computers have a limited set of operations they can perform, this ansatz is usually constructed from many rotation gates, which rotate one qubit around the x, y or z axis for an angle θ. For example, a rotation for the angle θ around the x-axis is given by

$$R_x(\theta) = e^{-i\frac{\theta}{2}\sigma_x}.$$

But using just single qubit rotations, as seen on Figure 3,

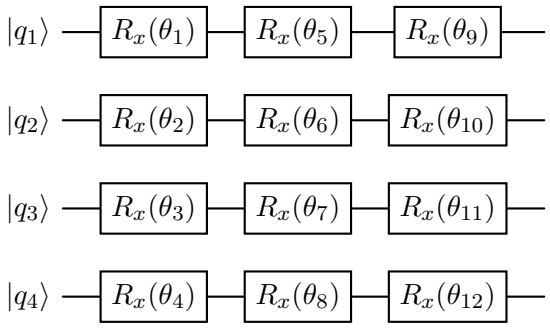

Figure 3. Ansatz circuit using only single qubit rotations.

would not be able to produce entangled states, as none of the lines are connected. For this reason, layers of CNOT gates are added. The CNOT or controlled NOT gate is a two-qubit gate, which flips one qubit, marked by the empty circle, if another qubit, marked by a black dot, is in the state $|1\rangle$. This produces the ansatz like the one in Figure 4,

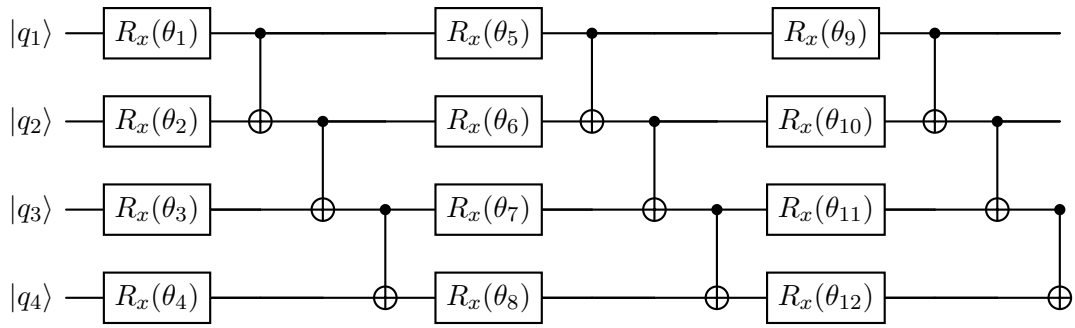

Figure 4. Example of an ansatz circuit built from single qubit rotations and CNOT gates.

which is a simple example of a hardware-efficient ansatz, which can produce a relatively large family of states at a shallow circuit depth.

These kind of generic ansatze have a problem, often referred to as the barren plateau [6]. This refers to the phenomenon where the ansatz produces states that are energetically indistinguishable from the totally mixed state over a large portion of the parameter space. This makes optimization difficult if the starting guess lies in that region, as the gradient will be very close to zero. While there are always starting guesses with non-zero gradients which lead to the ground state, the probability of randomly selecting such a state is exponentially suppressed in the number of qubits. The barren plateau problem is mitigated by tailoring the ansatz to a specific problem. If a model has a symmetry the ansatz can be limited to only produce states respecting that symmetry and if there already exists a solution to a similar problem, the ansatz can be made to produce perturbation on that same solution. In this way the ansatz can be limited to a smaller part of the parameter space, making the minimization process much easier.

Another promising way to construct the ansatz that attempts to avoid the barren plateau is the ADAPT-VQE ansatz [7]. Unlike the previous options the ansatz changes with each iteration of the minimization algorithm. It starts with a shallow circuit, to quickly find the region of the Hilbert space containing the ground state and then iteratively adds additional gates as necessary to locate the ground state with greater precision.

4.3 Hadamard test

There are many methods for evaluating expected values on a quantum computer. A simple one, which we are going to use, is called the Hadamard test (shown in Figure 5).

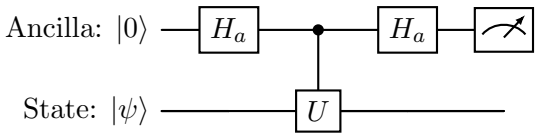

Figure 5. Hadamard test circuit.

We denote H_a as the Hadamard gate and U as an arbitrary unitary operator

$$H_a = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad\qquad U^\dagger U = \mathbb{1}.$$

The black dot above the U gate means that the gate is a controlled gate, meaning it is applied only if the ancilla qubit is in the state $|1\rangle$. It is important to stress that U must be unitary, otherwise, it cannot be a quantum gate, as evolution of a quantum state can only ever be unitary. This circuit can

be used to measure the expected value $\langle\psi|U|\psi\rangle$. To see how, we can look at what each subsequent gate does to the initial state:

$$|0\rangle \otimes |\psi\rangle \xrightarrow{H_a} \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |\psi\rangle \xrightarrow{U} \frac{1}{\sqrt{2}} \left(|0\rangle \otimes |\psi\rangle + |1\rangle \otimes U|\psi\rangle\right)$$
$$\xrightarrow{H_a} |0\rangle \otimes \frac{1}{2}\left(1+U\right)|\psi\rangle + |1\rangle \otimes \frac{1}{2}\left(1-U\right)|\psi\rangle.$$

The measurement of the ancilla qubit at the end gives the result $|0\rangle$ with probability

$$P_{|0\rangle} = \frac{1}{4}\langle\psi|(1+U^\dagger)(1+U)|\psi\rangle = \frac{1}{2}\left(1+\mathrm{Re}\langle\psi|U|\psi\rangle\right),$$

meaning that by running the circuit many times, we can infer $\mathrm{Re}\langle\psi|U|\psi\rangle$ from the number of times we measure $|0\rangle$. If we want to get the imaginary part as well, the initial state $|0\rangle$ just has to be replaced by the state $\frac{1}{\sqrt{2}}\left(|0\rangle - i|1\rangle\right)$ and then $P_{|0\rangle} = \frac{1}{2}\left(1+\mathrm{Im}\langle\psi|U|\psi\rangle\right)$.

To evaluate the energy of a state, the Hadamard test cannot be used directly, as H is hermitian rather than unitary. Thankfully, any qubit operator can be expressed as a linear combination of unitary terms $H = \sum_j \lambda_j P_j$ [2]. The way this can be done for our Hamiltonian is to remember the Jordan-Wigner transformation. The general formula can be rewritten as

$$c_n = \sigma_n^- \prod_{j=1}^{n-1} \sigma_j^z = \frac{1}{2}\sigma_n^x \prod_{j=1}^{n-1} \sigma_j^z - \frac{i}{2}\sigma_n^y \prod_{j=1}^{n-1} \sigma_j^z = \frac{1}{2}P_n^x - \frac{i}{2}P_n^y,$$

where P_n^x and P_n^y are unitary, as they are just products of unitary matrices. Since the hamiltonian consists of products of creation and annihilation operators, it becomes a sum of terms, which are products of P_j^x and P_j^y. In other words, it is a linear combination of unitary terms. The Hadamard test can be used on each term in the sum separately, as seen in Figure 6.

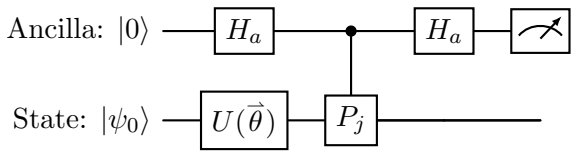

Figure 6. Circuit for evaluating the energy.

Here, $U(\vec{\theta})$ is the ansatz. The results of multiple circuits can then be summed back up to get the expectation value of the energy.

Many minimization algorithms also require the gradient of the function they are minimizing. A more general derivation for this kind of circuit can be found in [8], but we will focus on the previously described hardware-efficient ansatz. The gradient we need is

$$\frac{\partial E}{\partial \theta_i} = \frac{\partial}{\partial \theta_i}\langle\psi(\vec{\theta})|H|\psi(\vec{\theta})\rangle.$$

Letting the derivative act on the states we get

$$\frac{\partial E}{\partial \theta_i} = \frac{\partial\langle\psi(\vec{\theta})|}{\partial \theta_i}H|\psi(\vec{\theta})\rangle + \langle\psi(\vec{\theta})|H\frac{\partial|\psi(\vec{\theta})\rangle}{\partial \theta_i} = \frac{\partial\langle\psi(\vec{\theta})|}{\partial \theta_i}H|\psi(\vec{\theta})\rangle + \overline{\frac{\partial\langle\psi(\vec{\theta})|}{\partial \theta_i}H|\psi(\vec{\theta})\rangle}$$

which can be rewritten as

$$\frac{\partial E}{\partial \theta_i} = 2\cdot\mathrm{Re}\left(\frac{\partial\langle\psi(\vec{\theta})|}{\partial \theta_i}H|\psi(\vec{\theta})\rangle\right) = 2\cdot\mathrm{Re}\left(\langle\psi_0|\frac{\partial U(\vec{\theta})^\dagger}{\partial \theta_i}HU(\vec{\theta})|\psi_0\rangle\right),$$

if we let $U(\vec{\theta})$ be the ansatz. At this point we can use the fact that each parameter corresponds to a single rotation gate $R_x(\theta_j)$ within $U(\vec{\theta})$. This means that if we write the whole ansatz circuit as a product of separate gates

$$U(\vec{\theta}) = \prod_{j=0}^{N} U_j,$$

where U_j are the various rotation and CNOT gates in the ansatz from left to right and N is their total number, the derivative simplifies to

$$\frac{\partial U(\vec{\theta})}{\partial \theta_i} = \prod_{j=0}^{n-1} U_j \cdot \frac{\partial R_x(\theta_i)}{\partial \theta_i} \cdot \prod_{j=n+1}^{N} U_j,$$

where we take the n-th gate to be the rotation gate corresponding to the parameter θ_i. The remaining derivative is then simply

$$\frac{\partial R_x(\theta_i)}{\partial \theta_i} = \frac{\partial}{\partial \theta_i} e^{-i\frac{1}{2}\theta_i \sigma^x} = -\frac{i}{2}\sigma_x R_x(\theta_i).$$

Since $\sigma_x R_x(\theta_i)$ is unitary, the Hadamard test can be used just like before, except that an additional σ^x is inserted into the ansatz on the same qubit as the corresponding rotation gate, as shown in Figure 7.

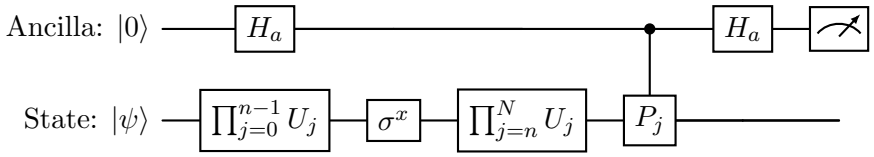

Figure 7. Circuit for evaluating the gradient of the energy.

5. Unitary evolution

Before evaluating the Green's function, we need to be able to express the creation operators $c_a(t)$ at different times. This requires applying the propagator $U(t) = e^{-iHt}$ to the states on the quantum computer. There are many ways to do this, but we will focus on two of them.

5.1 McLachlan variational method

Much like with ground-state preparation, one can use a variational approach. Using the same ansatz $\psi(\boldsymbol{\theta}(t))$ as before, except that the parameters are now functions of time, solving the Schrödinger equation

$$\frac{d}{dt}|\psi(\boldsymbol{\theta}(t))\rangle = -iH|\psi(\boldsymbol{\theta}(t))\rangle,$$

is equivalent to finding the functions $\{\theta_i\}$ which minimize the norm

$$L[\boldsymbol{\theta}] = \left\| \left(\frac{d}{dt} + iH \right) |\psi\rangle \right\|^2$$

within the space of states covered by our chosen ansatz. $L[\boldsymbol{\theta}]$ can be rewritten as

$$L[\boldsymbol{\theta}] = \left\| |\dot{\psi}\rangle + iH|\psi\rangle \right\|^2 = \langle\dot{\psi}|\dot{\psi}\rangle + \langle\psi|H^2|\psi\rangle + \langle\dot{\psi}|iH|\psi\rangle - \langle\psi|iH|\dot{\psi}\rangle.$$

The time derivative of the state can then be expressed using index notation as

$$|\dot{\psi}\rangle = \frac{dU}{dt}|\psi_0\rangle = \frac{\partial U}{\partial \theta_i}\frac{d\theta_i}{dt}|\psi_0\rangle = \frac{\partial|\psi\rangle}{\partial\theta_i}\dot{\theta}_i,$$

giving the final expression

$$L[\boldsymbol{\theta}] = \frac{\partial\langle\psi|}{\partial\theta_i}\frac{d|\psi\rangle}{\partial\theta_j}\dot{\theta}_i\dot{\theta}_j + \langle\psi|H^2|\psi\rangle + i\left(\frac{\partial\langle\psi|}{\partial\theta_i}H|\psi\rangle - \langle\psi|H\frac{d|\psi\rangle}{\partial\theta_i}|\psi\rangle\right)\dot{\theta}_i,$$

$$L[\boldsymbol{\theta}] = \frac{\partial\langle\psi|}{\partial\theta_i}\frac{d|\psi\rangle}{\partial\theta_j}\dot{\theta}_i\dot{\theta}_j + \langle\psi|H^2|\psi\rangle - 2\mathrm{Im}\left(\frac{\partial\langle\psi|}{\partial\theta_i}H|\psi\rangle\right)\dot{\theta}_i.$$

In this form, we can find the minimum using variations. Specifically, the position $\boldsymbol{\theta}$ is considered fixed and we want to know the direction $\dot{\boldsymbol{\theta}}$, which minimizes $L[\boldsymbol{\theta}]$. Varying the velocity gives the expression

$$\delta L = \frac{\partial\langle\psi|}{\partial\theta_i}\frac{d|\psi\rangle}{\partial\theta_j}\left(\delta\dot{\theta}_i\dot{\theta}_j + \dot{\theta}_i\delta\dot{\theta}_j\right) + 0 - 2\mathrm{Im}\left(\frac{\partial\langle\psi|}{\partial\theta_i}H|\psi\rangle\right)\delta\dot{\theta}_i,$$

which can be reduced to

$$\delta L = \left[2\mathrm{Re}\left(\frac{\partial\langle\psi|}{\partial\theta_i}\frac{d|\psi\rangle}{\partial\theta_j}\right)\dot{\theta}_j - 2\mathrm{Im}\left(\frac{\partial\langle\psi|}{\partial\theta_i}H|\psi\rangle\right)\right]\delta\dot{\theta}_i.$$

Setting $\delta L = 0$ gives the final coupled set of differential equations

$$\sum_j M_{ij}\dot{\theta}_j = V_i, \qquad M_{ij}(t) = \mathrm{Re}\left(\frac{\partial\langle\psi|}{\partial\theta_i}\frac{\partial|\psi\rangle}{\partial\theta_j}\right), \qquad V_i(t) = \mathrm{Im}\left(\langle\psi|H\frac{\partial|\psi\rangle}{\partial\theta_i}\right),$$

which at first glance looks very complicated. Thankfully the coefficients can be evaluated on a quantum computer using another variation of the Hadamard test [8].

Any numerical method for solving time-dependent ODEs can then be used to solve the system, where at each step the coefficients $V_i(t)$ and $M_{ij}(t)$ are computed on a quantum computer. As with VQE, the gradients have empirically been shown to be very small in most cases, meaning the coefficients V_i and M_{ij} are often near zero. This is a problem for the numerical stability of ODE-solvers and forces the use of very small time steps, which naturally leads to longer computation times. A more detailed explanation of this kind of methods can be found in [9].

5.2 Trotterization

Since the McLachlan method runs into the same problem of the barren plateau as VQE, another more direct approach is often used.

Trotterization refers to using the Trotter product formula $e^{A+B} = \lim_{n\to\infty}\left(e^{A/n}e^{B/n}\right)^n$ on the propagator e^{-iHt} to split it into many small time steps. The formula is useful because the whole exponent e^{-iHt} usually cannot be directly implemented as a gate, since H tells us about the energy of the whole system and therefore contains a global set of operators. However, when the Hamiltonian is split into a sum its constituent terms, each separate term is usually much simpler to implement. If all the terms commuted with each other, we could then write

$$H = \sum_j \lambda_j P_j \xrightarrow{[P_i,P_j]=0} e^{-iHt} = \prod_j e^{-i\lambda_j P_j t},$$

but since they most often do not, we use the approximation

$$e^{-iHt} = \lim_{n\to\infty}\left(\prod_j e^{-i\lambda_j P_j \frac{t}{n}}\right)^n,$$

where instead of taking the limit, we choose a large enough integer n and we apply the gates $\prod_j e^{-i\lambda_j P_j \frac{t}{n}}$ that many times. After each application the state is propagated forward a fixed time step.

While this method is free of the barren plateau problem, it faces other challenges. Since the time evolution is split into n small steps with duration $\Delta t = t/n$ and each step requires the whole set applied each time, the circuit depth depends on both t and step size Δt. This imposes a limit on the energy scales that can be resolved. Specifically, resolving low energy solutions, which correspond to low frequency, requires a long evolution time t to capture the oscillation, while high energy solutions with high frequencies need a small enough Δt to be resolved. Trotterization is therefore only useful for calculating $G^R_{ab}(\omega)$ within a window of frequencies determined by the maximal circuit depth.

6. Green's function

Using the methods outlined above we can now move onto calculating the Green's function

$$G^R_{ab}(t) = -i\theta(t)\langle\psi_0|c_a(t)c_b^\dagger(0) + c_b^\dagger(0)c_a(t)|\psi_0\rangle.$$

Much like in the previous sections, a version of the Hadamard test can be used, as the product $c_a(t)c_b^\dagger(0)$ can be split into a sum of unitary gates. If we denote a set of Pauli operators $\{Q_j\}$, such that

$$c_a(0) = \sum_j \lambda_j^{(a)} Q_j^{(a)},$$

then calculating the Green's function simplifies to calculating terms of the form

$$\langle\psi_0|U^\dagger(t)Q_i^{(a)}U(t)Q_j^{(b)\dagger}|\psi_0\rangle,$$

where $U(t)$ is the propagator. Since the propagator $U(t)$ is by definition unitary, the whole term is also unitary and the Hadamard test can be used to evaluate it.

In the case of Trotterization, the Hadamard test can be applied directly, as $U(t)$ simply turns into a set of gates, as shown on Figure 8.

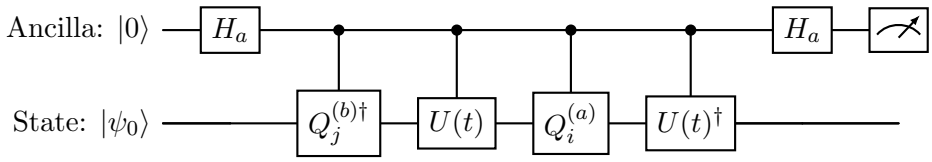

Figure 8. Hadamard test circuit for the Green's function using Trotterization.

This becomes more complicated for the McLachlan approach, as there is no general $U(t)$ circuit. The minimization algorithm only returns the parameters $\{\theta_i(t)\}$ needed to prepare the time-evolved state for a single starting state defined by $\{\theta_i(0)\}$. The Hadamard test circuit therefore needs to use two different propagators, where $U^{(1)}(t)$ is the propagator for the initial state $|\psi_0\rangle$ and $U^{(2)}(t)$ is the propagator for the initial state $Q_i^{(a)}|\psi_0\rangle$, which has to be found by running the minimization algorithm a second time. The resulting circuit is shown in Figure 9.

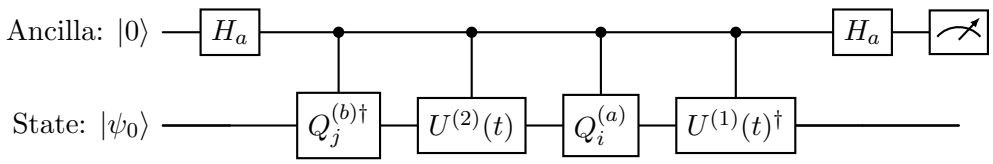

Figure 9. Hadamard test circuit for the Green's function using the McLachlan approach.

Both circuits can be further optimized as seen in [2] and [10], but the core idea remains the same. A more comprehensive look at evaluating Green's functions on contemporary quantum computers along with numerical simulations can be found in [10].

7. Conclusion

In this work, we have explored the process of evaluating quantities on a contemporary noisy quantum computer. Though our focus was the Green's function of many-body fermionic models, we described general methods, which are applicable even outside this particular context. We first described the Jordan-Wigner transformation to allow us to work with fermions on a quantum computer that consists of qubits. We then discussed the VQE algorithm in detail, going through the construction of the ansatz and the evaluation of the energy and its gradient on a quantum computer via the Hadamard test. Afterwards we presented two different methods for propagating a state in the quantum computer forward in time. The McLachlan method shares the same issues as VQE with the barren plateau, while the Trotterization method is simpler and faster, but is limited by the fact that it is a single long calculation and not split into smaller chunks to mitigate noise. In the end we put everything together to build a circuit for calculating the Green's function of a fermionic model.

REFERENCES

[1] C. Kittel and P. McEuen, *Introduction to Solid State Physics*, 8th edition, John Wiley & Sons, 2005.

[2] E. Pavarini, E. Koch, A. Lichtenstein and D. Vollhardt, *Understanding Correlated Materials with DMFT*, vol. correl25, Forschungzentrum Jülich, 2025.

[3] M. Imada, A. Fujimori and Y. Tokura, *Metal-insulator transitions*, Rev. Mod. Phys. **70** (1998), 1039–1263.

[4] P. Coleman, *Introduction to Many-Body Physics*, Cambridge University Press, 2015.

[5] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik and J. L. O'Brien, *A variational eigenvalue solver on a photonic quantum processor*, Nature Communications **5** (2014), 4213.

[6] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush and H. Neven, *Barren plateaus and quantum neural network training landscapes*, Nature Communications **9** (2018), 7090.

[7] H. R. Grimsley, S. E. Economou, E. Barnes and N. J. Mayhall, *An adaptive variational algorithm for exact molecular simulations on a quantum computer*, Nature Communications **10** (2019), 10988.

[8] K. Mitarai and K. Fujii, *Methodology for replacing indirect measurements with direct measurements*, Physical Review Research **1** (2019), 013006.

[9] X. Yuan, S. Endo, Q. Zhao, Y. Li and S. C. Benjamin, *Theory of variational quantum simulation*, Quantum **3** (2019), 191.

[10] S. Endo, I. Kurata and Y. O. Nakagawa, *Calculation of the Green's function on near-term quantum computers*, Phys. Rev. Res. **2** (2020), 033281.