Resource Estimation for Quantum Variational Simulations of the Hubbard Model: The Advantage of Multi-core NISQ Processing

Zhenyu Cai\textsuperscript{1,2,*}

\textsuperscript{1}Department of Materials, University of Oxford, Oxford, OX1 3PH, United Kingdom
\textsuperscript{2}Quantum Motion Technologies Ltd, Nexus, Discovery Way, Leeds, West Yorkshire, LS2 3AA, United Kingdom

(Dated: October 8, 2019)

As the advances in quantum hardware bring us into the noisy intermediate-scale quantum (NISQ) era, one possible task we can perform without quantum error correction using NISQ machines is the variational quantum eigensolver (VQE) due to its shallow depth. A specific problem that we can tackle is the strongly interacting Fermi-Hubbard model, which is classically intractable and has practical implications in areas like superconductivity. In this Article, we will perform resource estimation on Hubbard VQE using silicon spin qubits as our example platform, though many of our arguments can be extended to other hardware platforms like superconducting qubits, trapped-ion qubits, etc. We outline the details about the gate sequence, the measurement scheme and the relevant error mitigation techniques for the implementation of the Hubbard VQE on a NISQ platform. For a 50-qubit simulation of the Fermi-Hubbard model, which cannot be solved exactly via classical means, we need \(\sim 40000\) gates, which means that we need a gate infidelity of \(10^{-4}\) to obtain meaningful results with the aid of error mitigations. When searching for the ground state presuming a \(\mu s\) gate time, the time required for one time step in gradient descent is \(\sim 6\) days, which can be reduced to less than 10 minutes if we distribute our task among 1000 quantum processors. Hence, implementing a 50-qubit Hubbard model VQE on a NISQ machine can be on the brink of being feasible in near term, but further optimisation of our simulation scheme, improvements in the gate fidelity, improvements in the optimisation scheme and advances in the error mitigation techniques are needed to overcome the remaining obstacles. The scalability of the hardware platform is also essential to overcome the runtime issue via parallelisation.

I. INTRODUCTION

The difficulty of simulating large quantum systems was an inspiration for the idea of quantum computation [1], thus naturally, we look to quantum system simulation as one of the first applications of quantum computers. Fault-tolerant simulation of a non-trivial system requires an integrated quantum device with at least hundreds of thousands of qubits given a gate infidelity of \(10^{-3}\) or tens of thousands of qubits given a gate infidelity of \(10^{-4}\) [2]. This is still out of reach with the current technology.

With the advance of qubit counts and gate error rates bringing us into the noisy intermediate-scale quantum (NISQ) era, one must wonder if there is a classically intractable and physically meaningful task such NISQ hardware can tackle without quantum error correction. One of the most promising candidates is the variational quantum eigensolver (VQE) due to its shallow depth. In VQE, we aim to use a parametrised quantum circuit to prepare the eigenstate (usually the ground state) of a given Hamiltonian. The task is carried out through measuring observables using the quantum circuit and optimising the parameters using a classical computer. As we will see later, one of the lowest hanging fruits in this area is the preparation of the ground state of the strongly interacting Hubbard model, which is what we will be focusing on in this Article.

A huge number of circuit runs are needed to run VQE [3]. Luckily, the task is highly parallelisable and thus can be distributed to many quantum processors. Silicon quantum dot spin qubits appear to be a good candidate for such a task due to its high qubit density and the possibility of leveraging the commercial fabrication technology [4], which enable us to put multiple quantum processors onto a single chip for such a parallelisable task. Furthermore, as we presently explain that the ansatz circuit of the Hubbard model that we are interested in can be naturally broken into the native gates of silicon qubits: \(Z\) rotation and partial swaps. Both of these gates can operate on a sub-microsecond scale in silicon spin qubits, which ensures short circuit runtime on top of the parallelisability. Thus in this Article, we will use silicon spin qubits as our example platform to carry out our gate count and runtime analysis, though many of our arguments are also valid for the other qubit platforms.

In this Article, we will first introduce the concept of VQE and why we choose to simulate the Hubbard model in Section II. Then we will outline the ansatz circuit we use and its gate counts in Section III. Following that, in Section IV we will introduce more details about our implementation and the estimation of the algorithm runtime. This is followed by our conclusion in Section V.
II. VARIATIONAL QUANTUM EIGENSOLVER

A. Background

For a given Hamiltonian $H$, we want to find a circuit that can prepare its ground state $|\psi_0\rangle$ with the associated ground state energy $E_0$. We try to achieve this with a circuit with tunable gates that have $M$ parameters $\theta = \{\theta_1, \theta_2, \cdots, \theta_M\}$ that we can control, which will produce an output state $|\psi(\theta)\rangle$. In VQE, our goal is to obtain the set of optimal parameters $\theta_{\text{opt}}$ such that the state prepared by the circuit can well approximate the ground state we want $|\psi(\theta_{\text{opt}})\rangle \approx |\psi_0\rangle$. Then we can measure various properties of the ground state using $|\psi(\theta_{\text{opt}})\rangle$. Refs [5] and [6] provide a comprehensive overview on variational algorithms and more generally the field of quantum computational chemistry.

B. Choosing the simulation problem and its corresponding ansatz

The parametrised circuit in VQE is called an ansatz. It determines the quantum subspace that our output state can reach, hence it is the key to the success of our algorithm. An example of a simple ansatz is called the hardware-efficient ansatz (HEA) [7], which just uses gates available to the physical qubit systems to create many entangling blocks along the circuit. However, it is difficult to obtain an analytical or even a heuristic estimate of the number of gates required in an HEA for the success of our algorithm. A better approach will be using an ansatz inspired by the problem itself. For example, unitary-coupled-cluster ansatz (UCCA) for quantum chemistry [8], low depth circuit ansatz [9] (LDCA) and Hamiltonian ansatz (HA) [3] for more general simulations of closed quantum systems.

The number of gates needed by HA scales as $O(N_{\text{blk}} R)$ where $R$ is the number of subterms in the Hamiltonian of our simulation and $N_{\text{blk}}$ is the number of repeating ansatz blocks. $R \sim O(N^4)$ for a general chemistry Hamiltonian [3], and $R \sim O(N^2)$ for simulations of electron structure of periodic systems [10]. One of the systems that has the most favourable scaling while maintaining great physical interest is the 2D Hubbard model, with $R \sim O(N)$. Using HA to simulate the Hubbard model, the gates we apply are dependent on the qubit encoding we use. Using the Jordan-Wigner encoding, some of the gates will be non-local, which can be overcome by using $O(N^{\frac{3}{2}})$ additional gates [11]. Other encodings like Verstraete-Cirac encoding [12] or superfast encoding [13] will ensure locality, but require at least doubling the number of qubits. Since we are focusing on near-term devices in which qubit resources can be limited, we will only consider the Jordan-Wigner encoding in this Article. In such case, the number of gates needed to simulate the 2D Hubbard Model using HA scales as $N_{\text{gates}} \sim O(N_{\text{blk}} N^{\frac{3}{2}})$. This is a factor of $N^{\frac{3}{2}}$ better than LDCA assuming the same number of blocks $N_{\text{blk}}$ in both ansatze. To achieve results of sufficient precision, we likely need to go up to double excitation for UCCA, which means a gate scaling of $N_{\text{gates}} \sim O(N^3)$ (assuming Jordan-Wigner encoding, see Ref [6]). To achieve similar precision, $N_{\text{blk}}$ for HA is likely to scale better than $O(N^{\frac{3}{2}})$ based on the previous Hubbard model numerical simulation results [3, 14], thus HA should have a scaling advantage over UCCA in simulating the Hubbard model.

Hence, in the rest of this Article, we will be focusing on the ground state preparation of the Hubbard model using Hamiltonian Ansatz (HA) under the Jordan-Wigner encoding.

Note that for the NISQ regime that we are interested in, the constants we ignored in the above scaling analysis can make a very significant difference. Thus there might exist other problems with a different ansatz implementation that are more practical than the one we are going to consider when we dig into the exact implementation details.

III. ANSATZ CIRCUIT

A. 2D FERMI-HUBBARD MODEL

The Hamiltonian of the 2D Fermi-Hubbard model is:

$$H = -t \sum_{\sigma, (v, w)} (a^\dagger_{v, \sigma} a_{w, \sigma} + a^\dagger_{w, \sigma} a_{v, \sigma}) + U \sum_v n_{v, \uparrow} n_{v, \downarrow}$$

where $a^\dagger_{v, \sigma} / a_{v, \sigma}$ are the creation/annihilation operators of site $v$ with spin $\sigma$. The first term represents the nearest neighbour hopping interactions, with $t$ being the tunnelling energy and $(v, w)$ representing summing up sites $v$ and $w$ that are adjacent to each other in a 2D geometry. The second term is the on-site repulsion energy.

Extracting the ground state properties of the Hubbard model in the parameter regime of e.g. $U/T \approx 4 \rightarrow 8$ at close to half-filling is believed to be relevant to the understanding of high-$T_c$ cuprate superconductors [15, 16]. However, solutions using classical methods have high degree of uncertainties in this parameter regime [15], leading to our attempt here using quantum algorithms. In this Article, for simplicity we will be considering the half-filling case, i.e. for $V$ sites we have $V$ electrons, but most of our arguments can be generalised to any number of electrons.

Using the Jordan-Wigner encoding, each qubit will encode one orbital. There are two spin orbitals to each site, hence the number of qubits ($N$) needed is twice the number of sites ($V$). For a classically intractable 50-qubit problem, we will be looking at $V = 25$, e.g. a $5 \times 5$ Hubbard model. It is important to note that even though a Hubbard Hamiltonian of this size cannot be solved exactly classically, there are various approximation methods available [15].
In the Jordan-Wigner encoding, the Fermionic creation and annihilation operators become:

\[
\begin{align*}
    a_i^\dagger &\rightarrow Z_i A_i^\dagger \\
    a_i &\rightarrow Z_i A_i
\end{align*}
\]

where \( i \) denotes the index of the canonical ordering of the orbitals (including both spins and sites). \( A = |0\rangle \langle 1| = \frac{X+Z}{\sqrt{2}} \) is the qubit lowering operator and \( A^\dagger \) is the qubit raising operator. \( Z_i = \prod_{j=1}^{i-1} Z_j \) is the \( Z \) operator string that maintains the Fermionic commutation relationship.

The on-site repulsion term and the hopping term of the Hubbard Hamiltonian thus becomes:

\[
\begin{align*}
    n_i n_j &= a_i^\dagger a_i a_j^\dagger a_j \rightarrow \frac{1}{4} (I - Z_i)(I - Z_j) \\
    a_i^\dagger a_j + a_j^\dagger a_i &\rightarrow \frac{1}{2} (X_i X_j + Y_i Y_j) Z_{i,j}
\end{align*}
\]

(1)

where we have without loss of generality assumed \( j > i \) and defined \( Z_{i,j} = \prod_{k=i+1}^{j-1} Z_k \).

**B. Hamiltonian Ansatz**

The Hamiltonian ansatz was proposed by Wecker et al. [3]. Its circuit is essentially a Trotterised variational annealing path, which is inspired by both adiabatic state preparation and the quantum approximate optimisation algorithm [17, 18].

For a Hamiltonian of the form:

\[
H = \sum_i \lambda_i h_i
\]

where \( h_i \) are different interaction terms, a block of the Hamiltonian ansatz is just:

\[
\prod_i e^{-i\theta_i h_i}
\]

i.e. we implement a parametrised unitary gate corresponding to every interaction term. We will implement \( N_{blk} \) of such blocks in sequence with different sets of parameters. The Hamiltonian ansatz, though inspired by Trotterised annealing, is not equivalent to Trotterisation. Thus a Hamiltonian ansatz using higher-order Trotterisation does not necessarily have lower algorithmic errors than a Hamiltonian ansatz using lower-order Trotterisation. In NSIQ devices, we might want to avoid using higher-order Trotterisation in the Hamiltonian ansatz due to higher gate counts and deeper circuits. Hence, here we have used the first-order Trotter formula for the Hamiltonian ansatz.

From (1), we see that the gates corresponding to the repulsion terms are all local, while the gates that correspond to hopping will only be local if the two orbitals involved are close to each other in the canonical ordering due to the trailing \( Z \) string.

Now if we choose a canonical ordering such that the two spin orbitals of the same site are always adjacent to each other, while the orbitals of different sites are ordered in a folding pattern running along the horizontal direction as shown in Figure 1, then all the horizontal hopping terms will be local while some of the vertical hopping terms will not be. The non-local vertical hopping can be made local by using additional Fermionic swap (fswap) gates to swap the canonical order of the orbitals [19]:

\[
f_{\text{swap}}^{i,i+1} \rightarrow \frac{1}{2} [(X_i X_{i+1} + Y_i Y_{i+1}) + Z_i + Z_{i+1}].
\]

Kivlichan et al. [11] outline a way to implement a block of the Hamiltonian ansatz for the open boundary 2D Hubbard model using only local gates with \( O(N^{3/2}) \) depth using a Fermionic swap network. The exact structure of this ansatz is recapped in Appendix A 1. In Appendix A, we have decomposed the gates corresponding to the repulsion terms, the hopping terms and the Fermionic swap into the native gates of silicon spin qubits, and found that we only need single-qubit \( Z \) rotation and partial swaps. Then we applied the gate decompositions to the Hamiltonian ansatz, optimised the circuit using the qubit exchange symmetry of the gates, and obtained the number of single-qubit gates and two-qubit gates needed for one block of Hamiltonian ansatz for \( V \) sites to be:

\[
N_{1g,ha} = 4V^{3/2} + 7V - 4\sqrt{V}
\]

\[
N_{2g,ha} = 8V^{3/2} + V - 4\sqrt{V}
\]
For $V = 25$, we have
\[
N_{1q,ha} \approx 650 \\
N_{2q,ha} \approx 1000
\]

C. Full Ansatz Circuit

To produce a good approximation to the Hubbard ground state, we need a good starting state for the Hamiltonian ansatz. The starting state that we choose is the ground state of the non-interacting Hubbard Hamiltonian (i.e. $U = 0$), which can be solved classically and can be efficiently prepared using a quantum circuit since it is a single Slater determinant \[11, 16, 20\]. In Appendix B we have recapped the details about the Slater determinant preparation circuit using Givens rotation. We decomposed the Givens rotation into the native gates of silicon spin qubits and again found that we only need $Z$ rotation and partial swaps. Then we applied the gate decompositions to the Slater determinant preparation circuit, optimised the circuit and obtained the number of single-qubit gates and two-qubit gates needed for $V$ sites to be:
\[
N_{1q,prep} = 2V^2 \\
N_{2q,prep} = 2V^2
\]

Since the two spin subspaces of the non-interacting Hubbard Hamiltonian are decoupled, we can prepare the Slater determinant separately in the two spin subspaces, which is also discussed in Appendix B. In this case, we need to start in an orbital ordering in which the spin-up and spin-down are separated, and end in a orbital ordering in which the spin-up and spin-down are interleaved for inputting into the Hamiltonian ansatz. We found that the saving in gate counts is limited due to the need to rearrange the orbital ordering while the runtime needed is longer. Hence, here we will stick with the simple Slater determinant preparation scheme in which we do not consider the two spin subspaces separately.

The total number of gates needed for the whole ansatz circuit is:
\[
N_{1q} = \frac{2V^2}{N_{1q,prep}} + \left(4\sqrt{\frac{V}{2} + \frac{7V - 4\sqrt{V}}{N_{1q,ha}}} \right) N_{blk} \\
N_{2q} = \frac{2V^2}{N_{2q,prep}} + \left(8\sqrt{\frac{2V}{2} + \frac{V - 4\sqrt{V}}{N_{2q,ha}}} \right) N_{blk}
\]

Hence, if $N_{blk}$ scale better than $O(\sqrt{V})$, then the Slater determinant preparation will dominate the gate count at large $V$, otherwise the Hamiltonian ansatz will dominate the gate count at large $V$. In the previous simulation of the Hubbard model in a ladder grid structure with periodic boundaries \[3\], $N_{blk}$ scale super-linear to $V$, while Ref \[14\] also shows similar results for the open-boundary Hubbard model. Hence, we will expect the gate cost due to the Slater determinant preparation part of the circuit to be negligible at large $V$.

If we take the optimistic assumption of $N_{blk} \sim V$, then the number of gates needed in the ansatz circuit for $V = 25$ will be:
\[
N_{1q} \approx 17000 \\
N_{2q} \approx 26000
\]

Now even if we assume an optimistic gate error rate of $\epsilon_{gate} = 10^{-4}$ for both single- and two-qubit gates, the first order circuit error rate is still non-negligible: $\epsilon_{circ} = N_{gate}\epsilon_{gate} \sim O(1)$. Hence, to obtain a meaningful result out of our circuit, we must apply error mitigations as will be discussed in Section IV B.

IV. VQE IMPLEMENTATION

A. Parametrising the Hamiltonian Ansatz

We have outlined the structure of the ansatz circuit in the last section, now we will turn to the way we parametrise the gates in the circuit. In the simplest scheme, we can assign a different parameter to each parametrised gate to allow an unconstrained optimisation of the ansatz circuit. However, a large number of parameters will mean a large number of dimensions in the parameter space. This can lead to difficulties in optimisation and may lead to long runtime since we need to probe more directions to obtain the gradient vector. Hence, here we will try to reduce the number of parameters by using the symmetry of the site layout.

For the open-boundary Hubbard model, the site layout has mirror symmetries along the horizontal and vertical direction. Thus the 2D Hubbard grid can be sliced into four equivalent partitions: $N_{eq} = 4$. On top of that if we have the same number of rows and columns, then the site layout also has an additional diagonal mirror symmetry, which gives $N_{eq} = 8$. The ground state of the Hubbard model is expected to follow the same symmetry.

The input Slater determinant, which is the ground state of the non-interacting version of the Hamiltonian, follows the same layout symmetry, hence so does the ansatz parametrisation. That is, the parametrised gates that represent the corresponding interaction terms in different equivalent partitions, which can be mapped to each other via layout-symmetry transformations, will share the same parameters.

Hence, the number of parameters in our ansatz is:
\[
N_{para} \approx \frac{N_{site}}{N_{eq}} N_{blk}.
\]

where $N_{para}^{site}$ is the number of parameters per site.

Ignoring the boundary case, there will be 5 interaction terms associated with each site: the repulsion term and the horizontal and vertical hopping terms of the two
spins. The input Slater determinant and the output ground state must have the same spin symmetry since our ansatz preserves spins. Hence, the ansatz parametrization will also have the same spin symmetry. Without spin-flip symmetry, the parameters for the spin-up and spin-down hopping terms will be different, thus 5 interaction terms means 5 parameters: \( N_{\text{site}}^{\text{para}} = 5 \). With spin-flip symmetry, the gates for the spin-up and spin-down hopping terms can share the same set of parameters, thus \( N_{\text{site}}^{\text{para}} = 3 \). If we are considering the half-filling ground state with the smallest total spin, then:

- Odd number of sites (electrons): different numbers of spin-up and spin-down electrons, which means no spin-flip symmetry and \( N_{\text{para}}^{\text{site}} = 5 \).
- Even number of sites (electrons): same number of spin-up and spin-down electrons, which means spin-flip symmetry and \( N_{\text{para}}^{\text{site}} = 3 \).

For the Hubbard model with periodic boundaries along the horizontal direction, we have translational symmetry of the sites along the horizontal direction, which means that all columns are equivalent on top of the vertical mirror symmetry we have, giving \( N_{\text{eq}} = 2N_{\text{col}} \). This can lead to fewer parameters. However, as we will discuss in Section IV C, the vertical interaction terms cannot be measured locally and efficiently in this case, thus we will not consider such a boundary condition.

For the Hubbard model with periodic boundaries in both directions, we have complete translational symmetry and thus every site is equivalent. However, we will also not consider this case since there is not yet an efficient Hamiltonian ansatz circuit for the periodic Hubbard model with a gate count and scaling as well as the one we have adopted for the open-boundary case.

**B. Error Mitigation**

VQE is inherently robust against local coherent errors since they can be offset by shifts in the variational parameters [21], which was observed in experiments [22]. For the other error components, we can further mitigate them via error extrapolation and symmetry verification.

1. **Error extrapolation**

We will write the average value of the observable \( O \) obtained from a quantum circuit with component error rate \( \epsilon \) as \( \overline{O}_\epsilon \). To the first order approximation, \( \overline{O}_\epsilon \) will be a linear function in \( \epsilon \) for small \( \epsilon \). Hence, if we run the circuit with the minimal error rate \( \epsilon \) to obtain \( \overline{O}_\epsilon \), and we boost the error rate by a factor of \( \lambda \) to obtain \( \overline{O}_{\lambda \epsilon} \), then an estimate of the error-free observable expectation value \( \overline{O}_0 \) can be obtained via linear extrapolation [7, 23]:

\[
\overline{O}_{\text{ext}} = \frac{\lambda \overline{O}_\epsilon - \overline{O}_{\lambda \epsilon}}{\lambda - 1}
\]  

Assuming \( \text{Var}[O] = \text{Var}[O_{\lambda \epsilon}] \), then the variance in our error-free estimate \( \overline{O}_{\text{ext}} \) is

\[
\text{Var}[\overline{O}_{\text{ext}}] = \frac{\lambda^2 + 1}{(\lambda - 1)^2} \text{Var}[O_\epsilon]
\]

To keep \( \text{Var}[\overline{O}_{\text{ext}}] \) at a target precision level, we thus need \( C_{\text{ext}} \) times more circuit runs as compared to merely keeping \( \text{Var}[O_\epsilon] \) at the same precision level, in which

\[
C_{\text{ext}} = 2 \left( \frac{\lambda^2 + 1}{(\lambda - 1)^2} \right).
\]  

The factor of 2 is due to the fact that we need to sample both \( O_\epsilon \) and \( O_{\lambda \epsilon} \).

For \( \lambda = 2 \), we have:

\[
C_{\text{ext}} = 10.
\]

i.e. if we want to perform two-point linear extrapolation with \( \lambda = 2 \), we need 10 times more circuit runs to obtain an estimated observable value to the same precision.

For large quantum circuits, it is believed to be more appropriate to extrapolate along an exponential curve instead [24]. The factor of extra samples needed in this case is lower-bounded by \( C_{\text{ext}} \) as proven in Appendix E, and will be similar to \( C_{\text{ext}} \) at a small error rate. Thus we will use \( C_{\text{ext}} \) as the cost factor of two-point extrapolation in this Article.

2. **Symmetry verification**

It is often the case that there are some symmetries that the resultant state must follow. In the case of the Hubbard model ground state, such a symmetry can be the parity of the number of electrons. For example, for the 5 x 5 Fermi-Hubbard model, we know that the parity of the number of electrons number must be odd since we are considering the half-filling case. Such a parity can be obtained by measuring the symmetry operator \( Z^\otimes N \) at the end of the state preparation. In such a way we can detect the erroneous states and discard them in our expectation value calculation [25, 26]. Alternatively, suppose we want to obtain the expectation value of an observable \( O \) using the output state, and we know that our output state must follow the symmetry \( S \) with the eigenvalue \( s = \pm 1 \). Then the symmetry-verified expectation value \( \overline{O}_{\text{sym}} \) can be obtained using [25]:

\[
\overline{O}_{\text{sym}} = \frac{\overline{O} + sOS}{1 + sS}.
\]  

Here we have assumed the observable \( O \) commute with \( S \), which is often the case when \( O \) is a term in the Hamiltonian since many symmetries of the ground state follows from the symmetries of the Hamiltonian.
3. Combination

All the average values $\overline{O}$, $\overline{OS}$ and $\overline{S}$ in (6) are obtained from noisy circuits. We can substitute them with the extrapolated value obtained from (4): $\overline{O}_{\text{est}}$, $\overline{OS}_{\text{est}}$, $\overline{S}_{\text{est}}$ to combine error mitigation with symmetry verification for maximal error suppression. Combination of symmetry verification and error mitigation was proven to be very effective in numerical simulations [26].

As mentioned in Section III C, for $V = 25$, our circuit has a large number of gates $N_{\text{gates}} \sim 50000$ and if we assume the gate error rate of $10^{-4}$, we have a first-order circuit error rate $\epsilon_{\text{circ}} \sim O(1)$. Provided that the overall circuit error rate is of order unity or lower, ref. [24] has shown through numerical simulation that the bias in the measurement results can be efficiently removed using exponential extrapolation using Poisson approximation of the noise events. And if we add the robustness of the variational circuit against coherent errors and symmetry verification on top of it, we should be able to obtain meaningful result out of our quantum circuit.

C. Measurement

The measurement of energy is carried out by measuring the individual Pauli components of the Hamiltonian. The Pauli components within an individual repulsion term or a hopping term commute with each other. Thus with the availability of non-demolishing measurements, ideally we should be able to measure all the commuting interaction terms at one go. For the Hubbard model, there are five commuting subsets as shown in Figure 1: repulsion terms, even horizontal hopping terms, odd horizontal hopping terms, even vertical hopping terms and odd vertical hopping terms. Thus we should be able to obtain one sample of each interaction term in five circuit runs. However, direct measurements of the vertical hopping terms can be costly due to their non-locality. We can do local Pauli measurements and obtain the non-local measurement results via post-processing. However, local measurements can break the commutativity of the vertical hopping terms such that we cannot measure them in parallel.

For the case of the open-boundary Hubbard model, we can tackle this by switching the canonical ordering of orbitals from running horizontally in the 2D grid to running vertically when we try to measure the vertical hopping terms. In such a way, just like the horizontal hopping terms are local in the horizontal-running canonical order, the vertical hopping terms will also be local in the vertical-running canonical order. Note that switching the orbital canonical order will require us to switch the ansatz accordingly, but the same parameters will be used for the parametrised gates that correspond to the same interaction terms. In such a way, we can still obtain one sample of all interactions terms in five circuit runs.

The same measurement scheme can be used to measure the energy gradients since it involves the same Pauli measurements with some small modifications to the ansatz circuit (see Appendix C 2).

Now on top of obtaining the energy or energy gradient by measuring the Pauli components $G_j$ of the Hubbard Hamiltonian, we also want to apply symmetry verification using the symmetry operator $S$ by measuring $G_j S$ and $S$ according to (6). It is often the case that the symmetry operator $S$ is not local, making $G_j S$ non-local. Hence, to measure $G_j S$ using local measurement, we need to rely on post-processing. However, this will break the commutativity between $G_j S$ of different $j$ so that we cannot measure them in parallel. Instead, we can opt to measure $G_j$ in every run and then at the end of that run measure $S$ using local measurements. In that way, we can obtain both $S$ and $G_j S$ via post-processing.

In our example, however, there is no need to measure the electron number parity symmetry, it can be obtained by composing the measurement results of the interaction terms. When measuring the repulsion terms, we will be performing single-qubit $Z$ measurement for every qubit. The repulsion terms and the electron number parity can both be obtained via post-processing.

In the case of measuring the hopping terms, we will be measuring $XX$ and $YY$ for the hopping pairs adjacent to each other in the canonical order. We can obtain the results of $ZZ$ measurements by composing $XX$ and $YY$, composing with the $Z$ measurements of the qubits not included in the hopping pairs, we can then obtain the electron number parity via post-processing.

It is worth noting that the Hubbard model simulation is quite friendly for efficient local measurements of the Hamiltonian terms. For more general problems, one might need to turn to more sophisticated measurement schemes [27–29].

In the above scheme, we have assumed we can carry out non-demolishing measurements, which can be carried out in silicon using ancilla qubits. In Appendix F, we outline a possible quantum dot layout that enables an efficient implementation of our measurement scheme.

D. Optimisation Method

We need to employ classical optimisation algorithms to obtain the optimal set of parameters for our parametrised quantum circuit. There are two general approaches, direct search and gradient-based. Direct search involves evaluating the cost function at different points and choosing the next set of points to evaluate based on the known points of the cost function, while gradient-based methods make use of the gradient of the cost function. In our case, since we are searching for the ground state, the cost function that we want to minimise is the energy of the state produced by our quantum circuit.

The energy can be straightforwardly evaluated by measuring the Pauli components of the Hamiltonian as mentioned in Section IV C. As discussed in [30], to compete
with the best classical algorithm for Hubbard model simulation, we need to estimate the energy per site to the precision $\epsilon_{E,\text{site}} = 10^{-3}t$. In Appendix D, we have translated this precision requirement into the precision requirement on the estimates of each Hamiltonian Pauli term. For the $5 \times 5$ Hubbard model, the number of circuit runs needed to estimate the energy to the required precision is:

$$M_E \approx 4 \times 10^5$$

as derived in Appendix D.

The gradient vector can be evaluated using finite difference, which involves evaluating the energy at two neighbouring points. In the simplest gradient descent scheme, by evaluating the energy points used in finite difference to the precision $\epsilon_{E,\text{site}}$ and also choose the terminating threshold of the change in energy to be $\epsilon_{E,\text{site}}$, we will be able to find the local energy minimum with the required precision $\epsilon_{E,\text{site}}$ given the right gradient-descent step size. In such a case, we will need twice the number of measurements compared to energy estimation to evaluate the gradient in one direction (since we need to evaluate two energy points). We have $N_{\text{para}}$ directions to probe, thus the number of circuit runs needed to evaluate the full energy gradient vector using finite difference is:

$$M_{\text{grad}}^f = 2N_{\text{para}}M_E$$

Using (3) and assuming $N_{\text{blk}} \sim V$, for $5 \times 5$ Hubbard model, the number of circuit runs needed to estimate the full gradient vector using finite difference is:

$$M_{\text{grad}}^f = 2N_{\text{para}}M_E \approx 3.1 \times 10^8$$

The precision of the gradient vector obtained here is dependent on the finite difference step size we choose. In Appendix D, we have outlined how to obtain the optimal step size and the gradient precision $\epsilon_{\text{grad}}$ that we can achieve using this optimal step size.

The gradient vector can also be evaluated using direct measurements of a modified ansatz circuit. The two approaches to obtain the gradient were compared in [8, 31], in which they explain that more measurements are needed in finite difference to overcome the finite step size approximation it makes. We further compare them for our implementation in Appendix D, taking into account the fact that we have the many parametrised gates share the same parameters due to the symmetries in the site layout. We found that direct measurements require fewer samples compared to finite difference as the number of gates with shared parameters increases. In Appendix D, we found that to achieve the same gradient precision $\epsilon_{\text{grad}}$ achieved above using finite difference (with the optimal step size), the number of circuit runs required using direct measurements is:

$$M_{\text{grad}} \approx 2.5 \times 10^7$$

which is an order of magnitude better than finite difference in this case.

Direct search methods are generally more effective in noisy and non-smooth problems while for gradient-based methods, the number of function calls needed usually scales better in higher dimensional problems [32]. We have also proven above that evaluating the full gradient vector is usually much more costly than evaluating an energy point in our implementation. We can see that neither of the approaches are clearly preferred. Various direct search optimisation methods like Nelder-Mead simplex have been successfully implemented experimentally [33–39] for small-size problems due to the robustness of direct search against noise, while gradient-based method like SPSA, which requires a smaller number of samples than simple gradient descent due to its stochastic nature, have also found success in the simulations of small molecules [7, 40, 41]. With improvements in the quantum hardware noise rate, we will expect gradient-based methods to play a more and more important role in the experimental realisation of VQE, especially considering the success of advanced gradient-based methods like Adam and Adagrad in high dimension noisy optimisation problems in machine learning. There are also investigations into using machine learning for optimisation [42, 43], which might have faster convergence rate and higher robustness to noise. In the end, the optimisation scheme is likely to involve a combination of various methods, with the aid of techniques like block-by-block optimisation [3] and sequential optimisation [44].

### E. Algorithm Runtime

Here we will estimate the algorithm runtime needed for running the VQE for the $5 \times 5$ open-boundary Fermi-Hubbard model.

From Appendix A and B, we know the runtime $T_{\text{circ}}$ needed for the ansatz circuit with Slater determinant preparation is:

$$T_{\text{circ}} \approx (49 + 45N_{\text{blk}})\tau_1 + (196 + 80N_{\text{blk}})\tau_2 + \tau_{\text{in}} + \tau_m$$

where $\tau_1$, $\tau_2$, $\tau_{\text{in}}$, and $\tau_m$ are the time required for qubit initialisation and measurements.

In silicon quantum dot spin qubits, the $Z$ rotations can be implemented using Stark shift at the speed $\tau_1 \approx 0.1\mu s$ [45]. Partial swaps can be implemented using direct exchange at ns scale, however the gate fidelity at such a short time scale may be limited by the finite voltage rise time [46]. Hence, we instead assume here the partial swaps are implemented at a lower exchange energy scale and thus at a slower speed of $\tau_2 \approx 1\mu s$, to achieve the high two-qubit gate fidelity demonstrated in Ref [47]. For readout, a scheme that can achieve more than 98% fidelity in under $6\mu s$ has been demonstrated [48], and a sub-$\mu s$ scheme with 99.7% fidelity has been proposed [49]. For initialisation, the simplest way is via spin relaxation,
which will be on the \( ms \) timescale. Faster initialisation can be achieved via spin-selective tunnelling from charge reservoirs \([50]\) or electron shuttling and ‘hotspot reset’ \([51–54]\). Initialisations at the \( \mu s \) scale have been achieved in silicon donor qubits \([55]\) and other semiconductor quantum dot qubits \([56]\). Thus here we will assume the time needed for initialisation plus readout can be reduced to below \( 100\mu s \). Hence, for the \( 5 \times 5 \) Hubbard model with \( N_{blk} = V = 25 \), the runtime needed for each circuit run is around

\[
T_{\text{circ}} \sim 2ms.
\]

In Section IV D, we have obtained the number of circuit runs needed for estimating the energy and the energy gradient vector. However, as mentioned in Section IV B, due to the high circuit error rate, we need to apply error mitigation. Two point extrapolation will need around 10 times more circuit runs, while symmetry verification can be applied via post-processing in our case. Thus the number of circuit runs needed to estimate the energy and the energy gradient vector with error mitigation is:

\[
M_E^* = 10M_E \approx 4 \times 10^6
\]

\[
M_{\text{grad}}^* = 10M_{\text{grad}} \approx 2.5 \times 10^8
\]

Thus the time needed to evaluate the error-mitigated energy and energy gradient are:

\[
T_E = T_{\text{circ}}M_E^* = 8000 \text{ s} \approx 2 \text{ hrs}
\]

\[
T_{\text{grad}} = T_{\text{circ}}M_{\text{grad}}^* = 5 \times 10^5 \text{ s} \approx 6 \text{ days}
\]

Using the simplest optimisation scheme: gradient descent, each iteration step then involves the evaluation of one gradient vector, which require 6 days. Such a time requirement per iteration step is hardly practical. However, the time cost is mostly due to the large number of samples needed, thus can be easily solved by running many circuits in parallel. For our problem, 1000 processors will mean 50000 qubits, which can be easily fitted onto a single silicon chip along with the classical controls and measurement devices required. We stress that these processors would be independent of one another. With 1000 processors, the time required for each gradient descent step will be reduced to less than 10 minute, thus making gradient descent feasible runtime-wise even if we require ten thousand steps for convergence. Of course, we do not expect that simple gradient descent would necessarily be able to find the solution for the problem size we are considering. However, its runtime feasibility should be indicative of the runtime of other optimisation algorithms, especially of other more advanced gradient-based methods.

V. CONCLUSION

In this article, we have investigated the resource requirements on obtaining the ground state of a Hamiltonian in a quantum computer using VQE, in which the Hamiltonian cannot be solved exactly classically. The Hamiltonian we have chosen is the \( 5 \times 5 \) open-boundary Fermi-Hubbard model due to its favourable scaling in both circuit size and depth. We have chosen silicon spin qubits as our example hardware platform for the resource estimation due to its scalability, relatively fast quantum control and the fact that its native gate set is well-suited for our problem. Our ansatz circuit makes use of one of the latest schemes for the input Slater determinant preparation \([20]\) and for the Hubbard Hamiltonian ansatz implementation \([11]\), which translates into 17000 single-qubit gates (all are \( Z \) rotations) and 26000 two-qubit gates (all are partial swaps) assuming the number of Hamiltonian blocks in the ansatz is equal to the number of sites. Such gate counts will require a gate error rate on the order of \( 10^{-4} \) to bring the circuit error rate down to \( O(1) \). To obtain meaningful results with such a circuit error rate, we must incorporate error mitigation techniques like extrapolation \([7, 23, 24]\) and symmetry verification \([25, 26]\). We have devised a measurement scheme that allows us to estimate various terms in the Hamiltonian in parallel and to apply symmetry verification via post-processing. Bringing all these together, we have estimated the runtime needed for one circuit execution to be around 2 ms and thus one iteration in a simple gradient descent optimisation is around 6 days. Hence, we have to run VQE in parallel in multiple quantum processors to reduce the runtime to a feasible level. 1000 quantum processors with 50 data qubits each, which can be easily fitted onto a single silicon chip, can reduce the time needed for one step in the gradient descent to less than 10 minutes.

Many of our arguments and results can be readily extended beyond silicon platforms. We can see that implementing a 50-qubit Hubbard model VQE on a NISQ machine sits right at the boundary of being practical in terms of gate counts and circuit error rate. Hence, even a constant factor improvement in the circuit error rate can have big impacts on bringing such an application of NISQ machines into reality, which can be brought about via further optimisation of our simulation schemes, improvements in the gate fidelity, improvements in the optimisation scheme and advances in the error mitigation techniques. We also need to rely on parallelisation to tackle the runtime issue. It is worth noting that the number of qubits required in noisy VQE can be comparable to the fault-tolerant implementation, even though in VQE we can reduce the number of quibits needed by allowing longer runtime. In the case of VQE, what we need is a lot of independent small units for parallelisation instead of a single integrated device, which should massively reduce the difficulties in manufacturing and calibration even if the total number of qubits is similar. However, this also place great emphasis on the ability of the hardware platform to reproduce many copies of the quantum processor once we manage to manufacture a good one. The silicon platform has advantages in this respect since the existing semiconductor fabrication tech-
tologies can provide a high-precision, relatively low-cost and highly reproducible manufacturing process.

Due to the large number of hyper-parameters in VQE, our gate and runtime estimates are only indicative of the canonical case. There can be variability in the estimates when we choose a different set of hyper-parameters. One important assumption we made is the number of ansatz blocks in the circuit is equal to the number of sites: \( N_{blk} = V \), which is an optimistic assumption given what we have observed in the numerical simulations for small size system [3, 14]. An increase in \( N_{blk} \) will lead to a linear increase in the gate counts and a quadratic increase in the runtime (due to the increase in both the gate counts and the parameter counts).

Any increase in the gate counts will lead to an increase in the circuit error rate, which needs to be suppressed using stronger error mitigation. For example, instead of two-point error extrapolation, we can sample at more error rates, which will increase the number of circuit runs. We can also try to verify for additional symmetries. However, when we have multiple symmetries, it might not be possible to measure the energy terms along with all the symmetries using local measurements in a single circuit run any more, which may lead to runtime overhead. Of course, we can also try to develop new error mitigation techniques. One possible avenue could be tailoring the noise in real machines using simple gates to increase the sensitivity of our symmetry verification against the noise. The effectiveness of a similar idea in the context of quantum error correction code has been shown [57].

Any increase in the algorithm runtime can be mitigated via further parallelisation by adding more quantum processors. At the earlier stage of optimisation, we may only need to sample the energy or the energy gradient at lower precision, which can further reduce the algorithm runtime. Without any sudden large changes in the parameters during the optimisation, we should expect the energy and the energy gradient change at a relatively smooth manner as the optimisation progress. Hence, we can use the energy and the energy gradients we obtained in the previous steps as the prior for the estimation of the new energy and the energy gradients in a Bayesian manner [58]. This should enable us to achieve the same precision using much fewer samples and thus much shorter algorithm runtime. Other important factors that can influence the algorithm runtime include the way we parametrised our circuit and the exact optimisation algorithm we choose, both worth further explorations.

Since our ansatz circuit has a relatively short depth \((O(N^2))\), the main limiting factor preventing us from simulating the Hubbard model much beyond the size 5\(\times\)5 is due to the increased gate counts and the resultant increased circuit error rate. Stronger error mitigation or even new error mitigation can only alleviate this problem. To fully tackle this, one possibility is to switch to a different kind of qubit encoding: the Majorana loop stabiliser code [59]. This encoding allows local vertical hopping terms using mediator qubits, thus no Fermionic swap network is needed and we can achieve a Hamiltonian ansatz block and a Slater determinant preparation circuit with depth \(O(1)\). Furthermore, it can detect and correct single-qubit errors. Hence, it can potentially suppress circuit errors and take us much beyond our current problem size. However, more qubits are needed for the encoding and we need to implement higher-weight operators for the interaction terms. In addition, when considering the stabiliser checks, we need to consider the errors introduced in implementing the check circuits and the connectivity requirement on the hardware. On the other hand, we may implement these stabiliser checks in a post-processing way similar to symmetry verification [60], but many more circuit runs will be needed.

ACKNOWLEDGEMENTS

The author would like to thank Sam McArdle, Yuan Xiao and Simon Benjamin for reading through the manuscript and providing valuable insights. The author would also like to thank Sofia Patomäki, Michael Fogarty and John Morton for discussions on the hardware implementation.

The author acknowledges support from Quantum Motion Technologies Ltd.
Appendix A: Hubbard Model Hamiltonian Ansatz

1. Simulation scheme

Here we recap the scheme to implement the 2D open-boundary Hubbard model Hamiltonian ansatz as described in [11]. We will be considering a 2D Hubbard models of $V$ sites, with the starting canonical ordering of the orbitals as shown in Figure 2.

Gates will be local if they are applying to the orbitals adjacent to each other in the canonical ordering. In the canonical ordering shown in Figure 2, the orbitals of the same site and different spins are adjacent to each other, enabling the application of local parametrised on-site repulsion gates. To apply local hopping gates, we need to apply fermionic swaps to the orbitals to move them around in the canonical ordering. There are two types of fermionic swaps that we can apply: swaps between or within spins, which correspond to swaps between or within rows for the orbital layout in Figure 2. The swap scheme in [11] involving alternating swaps within and between spins. Now suppose we focus only on the spin-down orbitals. The spin-down orbitals start in row 2 in which we can only perform local swaps and local hopping interactions between the even pairs of orbitals. Then we swap between spins, moving spin down to row 1, and now we can do local swaps and local hopping interactions between the odd pairs of orbitals. Repeating these steps will enable us to alternate between odd- and even-pair swaps and hopping interactions within the spin-down orbitals, which are interleaved with swaps between the two spins. Such a scheme can apply all relevant hopping interactions in a local manner as shown in Figure 3.

We will define an edge pair as a pair of sites that are adjacent to each other in the canonical ordering and are vertical neighbours in the site layout. For example, in Figure 2, the edge pairs are (3, 4) and (6, 7). The way we perform one block of Hamiltonian ansatz is by repeating $2 \times N_{\text{row}}$ rounds of the following two steps:

1. Swap between spins: swapping orbitals of the same sites but different spins.
2. Swap within the same spins, except for edge pairs on which we perform hopping interaction instead.

This will return all the orbitals to their original positions. Within the process, we need to

- Substitute step 1 in the first round with repulsion interactions and swaps between spins.
- Substitute step 2 in the first and last round with hopping interactions and swaps interaction within same spins, except for edge pairs on which we only perform hopping interactions, no swaps.

We can also implement the periodic boundary condition in the horizontal direction by adding a pair of hopping interaction into certain rounds of step 2.

2. Gate Count Analysis for Ansatz

The gates in the Hubbard model simulation scheme are parametrised gates for on-site repulsion, adjacent-site hopping, fermionic swaps and combinations of them. Here we will decompose them into single-qubit rotations and partial swaps, which form one of the basic gate sets in silicon qubits.
We will find the hopping gates, the fermionic swaps, and the fswap+hopping all starts with $Z_{\pi/2}$ and end with $Z_{-\pi/2}$ on one of the qubits. For on-site repulsion and fswap+repulsion, we have $Z_{-\pi/2}$ at the end and we can easily add an $Z_{\pi/2}$ in front by adding a $Z$ rotation pair $Z_{\pi/2}Z_{-\pi/2}$. All of these gates are symmetric under the exchange of qubits, thus we can choose which qubit to place the $Z$ rotations.

We will choose to place these $Z$ rotation on the odd qubits in the spin-up space and on the even qubits in the spin-down space. In this way, these $Z$ rotation will cancel \(^1\). For the hopping term, the fermionic swap and the fswap+hopping, this means removing two single-qubit rotations at the beginning and the end of the circuit are not cancelled. However, when estimating the number of gates needed, for a large number of sites (hence large number of rounds of iterations), we will assume such boundary effects are negligible.

\(^1\) Note that in each iteration, we will have $\sqrt{N} - 1$ interactions within the same spin, while we have $\sqrt{N}$ interaction in between different spins, hence, the $Z$ gate on one of the dot will not be cancelled, this will be the last dot in either the spin up space or the spin-down space. Also, the $Z$ rotations at the beginning and the end of the circuit are not cancelled. However, when estimating the number of gates needed, for a large number of sites (hence large number of rounds of iterations), we will assume such boundary effects are negligible.
For $V = 25$, we have:

\[ N_{1q} \approx 650 \]
\[ N_{2q} \approx 1000 \]
\[ T \approx 45\tau_{1q} + 80\tau_{2q} \]

**Appendix B: Slater Determinant Preparation**

Here we recap the Slater determinant preparation scheme outlined in [11, 20]. Note that the input Slater determinant we choose to prepare in this Article will follow the same spin and site-layout symmetry as the output ground state since the ansatz we choose preserves these two symmetries.

### 1. Background

We will use $N_{\text{orb}}$ to denote the total number of orbitals that we are considering while $N_e$ will be the number of electrons (i.e. the number of occupied orbitals).

We start with the qubits representing the eigenorbitals of the non-interacting Hubbard Hamiltonian. The initial state is the ground state of the non-interacting Hubbard Hamiltonian with the first $N_e$ orbitals being occupied (i.e. the first $N_e$ qubits are initialised to 1 while the rest are initialised to 0). Now the role of the state preparation circuit is to transform our qubits from representing the eigenstates of the non-interacting Hubbard Hamiltonian (with orbital creation operators $\{a_i^{\dagger}\}$) to the orbitals that can have a compact description of our interacting Hamiltonian, i.e. to the site orbital basis (with orbital creation operators $\{b_i^{\dagger}\}$). This basis transformation can be described by the transformation matrix $Q^\dagger$ (also called the Slater determinant) of the shape $N_{\text{orb}} \times N_e$:

\[
\bar{b}^\dagger = Q^\dagger \bar{a}^\dagger \\
= U^\dagger \Lambda W \bar{a}^\dagger \\
= U^\dagger \Lambda \bar{a}^\dagger
\]

Here we have carried out singular value decomposition of $Q^\dagger$. $W$ is a rotation within the filled-orbital subspace to find a new set of basis $\{a_i^{\dagger}\}$ other than the eigenstate of the non-interacting Hubbard Hamiltonian $\{a_i\}$. The qubit ground state in basis $\{a_i^{\dagger}\}$ is the same as basis $\{a_i\}$ with all the orbitals filled. Hence, we do not need to carry out the transformation $W$ explicitly, we only need to keep in mind that now we are working in this new basis for the input state instead of the non-interacting basis [20]. $A$ is a rectangular matrix of the $N_{\text{orb}} \times N_e$ with ones at the diagonal and zeros elsewhere. This is just an isometry to expand our space from the filled orbital subspace to the full orbital space by attaching $N_{\text{orb}} - N_e$ empty orbitals. Then the transformation $U^\dagger$ on the full orbital space will complete our transformation of basis. The transformation $U^\dagger$ will be implemented as compositions of Givens rotations in the quantum circuit.

### 2. Givens rotation

A Givens rotation is just a general rotation operation within a 2D complex subspace. There are two parts to a Givens rotation, one is the rotation to change the amplitude, and the other is phase operator to change the relative phase between the two basis in the subspace [20]:

\[
G(\theta, \phi) = \begin{pmatrix}
\cos \theta & -\sin \theta \\
\sin \theta & \cos \theta
\end{pmatrix}
\begin{pmatrix}
1 & 0 \\
0 & e^{i\phi}
\end{pmatrix}
\]

We will be mostly dealing with real Slater determinant, so let us ignore the phase part here. The rotation part can be carried out using the operator:

\[
R_{ij}(\theta) = e^{\frac{i\theta}{2}(a_i^{\dagger}a_j - a_j^{\dagger}a_i)}
\]

JW transform the neighbouring orbital case, we have

\[
\left( a_i^{\dagger}a_{i+1} - (a_i^{\dagger}a_{i+1})^\dagger \right) = 2i \text{Im}\{a_i^{\dagger}a_{i+1}\} \\
\Rightarrow 2i \text{Im}\{(X - iY)(X + iY)\} = 2i(YX - YX)
\]

Hence, the given rotation for adjacent orbitals are:

\[
R(\theta) = e^{\frac{i\theta}{2}(XY - YX)} = e^{-\frac{i\theta}{2}(YX - XY)}
\]

which translate into the following circuit using partial swap and $Z$ rotations.

Using the notation in Appendix A 2, we have:

Gate counts: $G_{1q,G} = 2$, $G_{2q,G} = 2$
Time needed: $\tau_G = \tau_{1q} + 4\tau_{2q}$

### 3. Gate count for Givens rotation

#### a. Simple scheme

The transformation from the non-interacting Hubbard model eigenbasis $\{a_i^{\dagger}\}$ to the site orbital basis $\{b_i^{\dagger}\}$ can be viewed as a process of trying to diagonalise the transformation matrix (Slater determinant) $Q$. The transformation $W$ can zero out a triangle of entries along the $N_e$ dimension. We need one Givens rotation to zero out each
remaining non-zero off-diagonal elements. The number of Givens rotation needed is [20]:

\[ N_e \left( \frac{N_{\text{orb}}}{2} - \frac{N_e}{2} \right) \]

For the half-filling Hubbard model with \( V \) sites, we have \( N_{\text{orb}} = 2V \) and \( N_e = V \). Hence, we have \( V^2 \) off-diagonal elements to be zeroed out. Each Givens rotation is decomposed into two partial-swaps and two \( Z \) rotations, thus the number of gates we needed are (following the notation in Appendix A 2)

\[ N_{1q} = 2V^2 \]
\[ N_{2q} = 2V^2 \]

Using the parallel scheme suggested in [11], we have a circuit of depth \( (N_{\text{orb}} - 1 - (N_e - 1)) + (N_e - 1) = N_{\text{orb}} - 1 = 2V - 1 \) which translate into the circuit runtime of

\[ T = (2V - 1) (\tau_{1q} + 4\tau_{2q}) \]

For \( V = 25 \), we have:

\[ N_{1q} = 1250 \]
\[ N_{2q} = 1250 \]
\[ T = 49\tau_{1q} + 196\tau_{2q} \]

b. Using spin conservation

For the Hubbard Hamiltonian that we are considering, the two spin spaces are decoupled. Hence, we can consider the Slater determinants of the two spin subspace separately, each of the shape \( \frac{N_e}{2} \times \frac{N_{\text{orb}}}{2} \). Hence, the number of Givens rotation needed in each spin subspace is

\[ \frac{N_e}{2} \left( \frac{N_{\text{orb}}}{2} - \frac{N_e}{2} \right) = \frac{V^2}{4}. \]

To keep the applications of the Givens rotations within the spin subspace on only adjacent orbitals, we need to start in an orbital ordering with the first \( N_{\text{orb}}/2 \) being the spin-up orbitals and the next \( N_{\text{orb}}/2 \) being the spin-down orbitals. Hence, we need to carry out orbital rearrangement after the Givens rotations to restore the spin-orbital ordering for the Hamiltonian ansatz (Appendix A 1).

We can arrange the Givens rotations such that all the Givens rotation on the last spin-up orbital (the \( \frac{N_{\text{orb}}}{2} \)-th orbital) and the first spin-down orbital (the \( \frac{N_{\text{orb}}}{2} + 1 \)-th orbital) finished first. In the next time step, while we are carrying out other Givens rotations, we can start performing fswap on the two orbitals that have finished Givens rotation. In the next step, two more orbitals will finish their Givens rotation (orbitals \( \frac{N_{\text{orb}}}{2} - 1 \) and \( \frac{N_{\text{orb}}}{2} + 2 \)), thus now we can perform fswap on orbitals (\( \frac{N_{\text{orb}}}{2} - 1, \frac{N_{\text{orb}}}{2} \)) and (\( \frac{N_{\text{orb}}}{2} + 1, \frac{N_{\text{orb}}}{2} + 2 \)). Carry on we will have more and more orbitals finishing their Givens rotations, and in each time step we will perform fswap on all orbitals that finished Givens rotations, alternating between the odd pair of orbitals and even pair of orbitals. After \( \frac{N_{\text{orb}}}{2} - 2 \) layer of swap, we will have alternating up and down orbitals. One more layer of swaps between all orbitals \( 4n - 1 \) and \( 4n \), will give us the \( \uparrow \downarrow \uparrow \downarrow \cdots \) order we used in the Hamiltonian ansatz (Appendix A 1). Note here we only talk about how to arrange the orbitals to have the right spin ordering, for the ordering of the orbitals within the same spin, it is determined by order of the rows and columns of the Slater determinants that we wrote down. In total, we need

\[ \frac{(N_{\text{orb}}/2 - 2 + 1) (N_{\text{orb}}/2 - 2)}{2} + \left( \frac{N_{\text{orb}}}{4} \right) \]
\[ \approx \frac{N_{\text{orb}}^2}{8} - \frac{N_{\text{orb}}}{2} \]
\[ = \frac{V^2}{2} - V \]

fswaps to achieve the desired orbital order.

For the fswap gates, similar to Appendix A 2, we can arrange the gates such that the \( \sqrt{Z} \) only acts on the odd orbitals, which will enable the cancellation of all the \( \sqrt{Z} \) other than those at the boundary. Hence, we can ignore the \( \sqrt{Z} \) gates required by the fswaps. Our scheme need \( V^2 \times 2 \) Givens rotations and \( V^2 - V \) fswaps. When decomposed into partial swap and \( Z \) rotation, the number of one-qubit gates and two-qubit gates needed are

\[ N_{1q} = 2 \left( \frac{V^2}{2} \right) + \frac{V^2}{2} - V = \frac{3}{2} V^2 - V \]
\[ N_{2q} = 2 \left( \frac{V^2}{2} \right) + 2 \left( \frac{V^2}{2} - V \right) = 2V^2 - 2V \]

The depth of the circuit before finishing the first Givens rotation is \( N_{\text{orb}} - N_e = V \). This section of the circuit consist of only Givens rotation, hence require runtime:

\[ T_a = V (\tau_{1q} + 4\tau_{2q}) \]

After this, we have the fswap network with Givens rotation happening concurrently, the depth of the circuit here is \( V - 1 \), the runtime is limited by the fswap instead of Givens rotation since fswap contains \( \pi \)-rotations of \( Z \). Hence, the runtime needed for the fswap networks is:

\[ T_b = (V - 1) (2\tau_{1q} + 4\tau_{2q}) \]

Hence, the total runtime needed for the state preparation circuit is:

\[ T = T_a + T_b = (3V - 2) \tau_{1q} + (8V - 4) \tau_{2q} \]

For \( V = 25 \), we have:

\[ N_{1q} = 910 \]
\[ N_{2q} = 1250 \]
\[ T = 73\tau_{1q} + 196\tau_{2q} \]
Hence, the spin subspace scheme leads to some reduction in the number of one-qubit gate. However, it also leads to longer circuit runtime.

c. Comparison to Ansatz circuit

When compared to Appendix A 2, the number of one-qubit gates of the Slater determinant preparation circuit is twice of that of one layer of the Hamiltonian ansatz, the number of two-qubit gates and the runtime are comparable.

Do note that the number of gates of the Slater determinant preparation circuit scale as $O(V^2)$, which is worse than the $V^{3/2}$ ansatz scaling. The depth of the Slater determinant preparation circuit scale as $O(V)$, which is also worse than the ansatz scaling $O(\sqrt{V})$ as well. However, we need to note that we did not take into account of the number of blocks needed in the ansatz scaling, and the number of blocks needed is very likely to scale worse than $\sqrt{V}$, which means that the state preparation should have a better scaling than the ansatz if we take that into account.

Appendix C: Obtaining Energy Gradient in Quantum Computers

1. Background

As mentioned in Section II. For a given Hamiltonian $H$, we want to find the set of optimal parameters $\vec{\theta}$ for an ansatz circuit such that the state $\lvert \psi(\vec{\theta}) \rangle$ it produce is as close to the ground state of the Hamiltonian as possible, i.e. we want to find $\vec{\theta}$ such that $E_{\text{tot}}(\vec{\theta}) = \langle \psi(\vec{\theta}) \lvert H \lvert \psi(\vec{\theta}) \rangle$ is minimised.

However, since the energy $E_{\text{tot}}(\vec{\theta})$ cannot be directly measured, we need to rewrite our Hamiltonian in terms of its Pauli components:

$$H = \sum_j \lambda_j G_j$$

$$\langle \psi(\vec{\theta}) \lvert H \lvert \psi(\vec{\theta}) \rangle = \sum_j \lambda_j \langle \psi(\vec{\theta}) \lvert G_j \lvert \psi(\vec{\theta}) \rangle.$$ 

Hence,

$$E_{\text{tot}}(\vec{\theta}) = \sum_j \lambda_j E_j(\vec{\theta})$$

where $E_j(\vec{\theta}) = \langle \psi(\vec{\theta}) \lvert G_j \lvert \psi(\vec{\theta}) \rangle$, which is just the expectation value of a Pauli observable, hence can be directly obtained from the circuit.

To find the ground state, we can use gradient-based optimisation methods like gradient decent and L-BFGS-B, etc. This require us to obtain $\frac{\partial E_{\text{tot}}(\vec{\theta})}{\partial \theta_m}$, which in turns means that we need to measure $\frac{\partial E_j(\vec{\theta})}{\partial \theta_m}$ for every Hamiltonian component $j$ and every parameter $m$.

We can obtain $\frac{\partial E_j(\vec{\theta})}{\partial \theta_m}$ simply using finite difference, in which we will measure $E_j(\theta_1, \theta_2, \ldots, \theta_m)$ and $E_j(\theta_1, \theta_2, \ldots, \theta_m + \delta \theta_m)$, and take their difference divided by $\delta \theta_m$ to approximate the gradient.

2. The Circuit Approach to Obtain Gradients

The exact gradient $\frac{\partial E_j(\vec{\theta})}{\partial \theta_m}$ can be obtained using circuit measurement instead of using finite difference approximation. Suppose our ansatz consist of $N$ blocks $R_n$ of the form

$$R_n(\theta_n) = e^{-i\theta_n F_n}$$

where $F_n$ are both Hermitian and unitary, e.g. Pauli or swaps.

Then our parametrised circuit can be written as:

$$\lvert \psi(\vec{\theta}) \rangle = \prod_{n=N}^1 R_n(\theta_n) \lvert 0 \rangle = R(\vec{\theta}) \lvert 0 \rangle$$

where we have denote the whole circuit using $R(\vec{\theta})$.

A string of parametrised rotation can be denoted as:

$$\prod_{n=a}^b R_n(\theta_n) = R_{a:b}$$

which means that $R(\vec{\theta}) = R_{N:1}$

Hence,

$$\frac{\partial R(\vec{\theta})}{\partial \theta_m} = R_{N,m+1} \frac{\partial R_{m}(\theta_m)}{\partial \theta_m} R_{m-1:1} = -i R_{N,m+1} F_m R_{m:1}$$

Hence we have

$$\frac{\partial E_j(\vec{\theta})}{\partial \theta_m} = 2 \text{Re} \left\{ \left( \langle \psi(\vec{\theta}) \lvert G_j \lvert \theta_m, \psi(\vec{\theta}) \rangle \right) \right\}$$

$$= 2 \text{Re} \left\{ \left( \langle 0 \lvert R(\vec{\theta}) \lvert G_j \lvert R_{N,1} \lvert 0 \rangle \right) \right\}$$

$$= 2 \text{Im} \left\{ \left( \langle 0 \lvert (R_{N,1})^\dagger G_j R_{N,m+1} F_m R_{m:1} \lvert 0 \rangle \right) \right\}$$

This can be measured via two kind of circuits:

- Indirect measurement [23]:

  The circuit is shown in Figure 4 (a) will measure $\frac{1}{2} \frac{\partial E_j(\vec{\theta})}{\partial \theta_m}$. Here we use an ancilla which probes the main ansatz with a control unitary. The advantage of such a scheme is that we can obtain the exact gradient of the circuit via measurement of only one qubit. However, it requires an extra ancilla qubit.
The need for performing the control unitary between the ancilla and any other qubits also lead to connectivity challenges. Control unitaries like control swap are also not straightforward to implement in lots of architectures.

- Direct measurement [61]:

The circuit is shown in Figure 4 (b). Denoting $|\phi\rangle = R_{m,1} |\emptyset\rangle$, $W = R_{N,m+1}$, the circuit will measure:

$$A_{jm,\pm} = \langle \phi | e^{\pm iF_m} W^\dagger G_j W e^{\mp iF_m} | \phi \rangle$$

$$= \frac{1}{2} \langle \phi | (1 \mp iF_m) W^\dagger G_j W (1 \mp iF_m) | \phi \rangle$$

$$= \frac{1}{2} \langle \phi | W^\dagger G_j W | \phi \rangle \mp \frac{1}{2} \langle \phi | F_m W^\dagger G_j W F_m | \phi \rangle$$

$$\pm \frac{1}{2} \left( \langle \phi | F_m W^\dagger G_j W | \phi \rangle - \langle \phi | W^\dagger G_j W F_m | \phi \rangle \right)$$

Hence, the gradient $\frac{\partial E_j(\bar{\theta})}{\partial \theta_m}$ can be obtained via

$$A_{jm,+} - A_{jm,-}$$

$$= i \left( \langle \phi | F_m W^\dagger G_j W | \phi \rangle - \langle \phi | W^\dagger G_j W F_m | \phi \rangle \right)$$

$$= 2 \text{Im} \{ \langle \phi | W^\dagger G_j W F_m | \phi \rangle \}$$

$$= \frac{\partial E_j(\bar{\theta})}{\partial \theta_m}$$

For this method, we do not require any extra ancilla or control unitaries. However, we need to measure two expectation values $A_{jm,+}$ and $A_{jm,-}$ for the estimation of the gradient $\frac{\partial E_j}{\partial \theta_m}$ instead of one in the case of indirect measurement.

3. Shared Parameters

For the case where there are parametrised gates with shared parameters, we can still obtain the gradients using finite difference in a similar way.

When using gradient circuits, the story is more complicated. Firstly, we add a scaling factor $\beta_m$ to each parameters: $\theta_m \rightarrow \beta_m \theta_m$:

$$\frac{\partial R(\theta_n)}{\partial \theta_m} = -i R_{N,m+1} F_m R_{m,1}$$

$$\Rightarrow \frac{\partial R(\beta_m \theta_n)}{\partial \beta_m \theta_m} = -i R_{N,m+1} F_m R_{m,1}$$

$$\frac{\partial R(\beta_m \theta_n)}{\partial \theta_m} = -i \beta_m R_{N,m+1} F_m R_{m,1}$$

Now we change the labelling of the parametrised gates: $m \rightarrow m, v$, for which gates with the same $m$ will share the same parameters, and $v$ labels the different parametrised gates that share the same parameter:

$$\theta_{m,v} = \theta_m \ \forall v$$

In such case, we have

$$\frac{\partial R(\beta_{m,w} \theta_{n,w})}{\partial \theta_m} = \sum_v \frac{\partial R(\beta_{n,w} \theta_{n,w})}{\partial \theta_{m,v}}$$

i.e. the gradient w.r.t. to a given parameter $\theta_m$ is the sum of all the gradient w.r.t. the parameter of each parametrised gate that shared the parameter value. Hence, when trying to obtain the gradient using circuit measurements, we still need to treat the parameters in each gate as independent, and then sum those gradient up based on which gates have shared parameters.

Appendix D: Number of Samples Needed for Energy and Energy Gradient

1. Number of samples in finite difference

a. Gradient Precision

The equation for the estimation of the gradient of the $j^{th}$ Pauli term in the Hamiltonian using finite difference is:

$$\frac{\partial E_j(\bar{\theta})}{\partial \theta_m} = \frac{E_j(\bar{\theta} + \frac{\delta_m}{2}) - E_j(\bar{\theta} - \frac{\delta_m}{2})}{\delta}$$

(D1)

where $\delta_m$ is a vector with the $m^{th}$ parameter set to $\delta$ and all other parameters set to 0. $E_j$ denote the sampling average of $E_j$.

Hence, we have

$$\text{Var} [\theta_m E_j] = \frac{2}{\delta^2} \text{Var} [\overline{E_j}]$$

(D2)
i.e. for larger $\delta$, we can achieve smaller variance in gradient for a fixed variance in energy. However, we cannot increase $\delta$ indefinitely because there is an error associated with the finite step size when using finite difference, which has the magnitude of:

$$
\frac{\delta^2}{24} \sum_{u,v,w=1}^{N_{sh}} \frac{\partial^3 E_j(\tilde{\theta})}{\partial \theta_m \partial \theta_m \partial \theta_m} \approx N_{sh} \frac{\delta^2}{24} \partial^3 m E_j
$$

where number of parametrised gates share parameter $\theta_m$ is $N_{sh}$. The sum over $v,u,w$ are the sum over all the parametrised gates that share the same parameter $\theta_m$ as discussed in Appendix C 2. Hence, the magnitude of

$$\partial^3 E_j(\tilde{\theta})$$

is:

$$\sqrt{\frac{2}{\delta^2} \text{Var} [E_j]} = N_{sh} \frac{\delta^2}{24} \partial^3 m E_j$$

Note that

$$\left( \frac{\partial^3 E_j(\tilde{\theta})}{\partial \theta_m^3} \right) = 2 \text{Re} \left\{ \left( \psi(\tilde{\theta}) \right| G_j \left( \partial^3_m \psi(\tilde{\theta}) \right) \right\}$$

and

$$\left( \frac{\partial^3 E_j(\tilde{\theta})}{\partial \theta_m^3} \right) = 6 \text{Re} \left\{ \left( \partial_m \left( \psi(\tilde{\theta}) \right) \right| G_j \left( \partial^2_m \psi(\tilde{\theta}) \right) \right\}$$

where $\text{Re} \left\{ \left( \partial_m \left( \psi(\tilde{\theta}) \right) \right| G_j \left( \partial^2_m \psi(\tilde{\theta}) \right) \right\}$ can be measured using circuit similar to the first order derivative in Appendix C 2. Hence, the magnitude of $\frac{\partial^3 E_j(\tilde{\theta})}{\partial \theta_m^3}$ is around $\sqrt{22^2 + 6^2} \approx 6$.

Thus we have:

$$\delta = \left( \frac{24 \sqrt{2 \text{Var} [E_j]}}{6 N_{sh}^3} \right) \frac{1}{2}$$

$$\delta = \frac{1.78}{N_{sh}} \sqrt{\text{Var} [E_j]}$$

Substituting into (D2), we have:

$$\text{Var} \left[ \partial_m E_j \right] = 0.63 N_{sh}^2 \sqrt{\text{Var} [E_j]}$$

$$\text{Var} [E_j] = \frac{2}{N_{sh}^3} \text{Var} \left[ \partial_m E_j \right]^2$$

This is the smallest variance in the gradient that we can achieve for a given variance in the energy estimation by choosing the optimal step size $\delta$.

b. Number of samples needed

Assuming $\text{Var} [E_j] \sim O(1)$, then the number of samples needed to achieve the sample average variance $\text{Var} [\bar{E}_j]$ is

$$\frac{\text{Var} [E_j]}{\text{Var} [\bar{E}_j]} \sim \frac{1}{\text{Var} [\bar{E}_j]}$$

Since we need to evaluate sample average $E_j$ at two points in finite difference, the total number of samples needed is:

$$M_{fd} \sim \frac{2}{\text{Var} [E_j]} \frac{N_{sh}^3}{\text{Var} [\bar{E}_j]^2}$$

(D4)

2. Number of samples in direct measurement

a. Gradient precision

The equation of estimation of the gradient of a the $j^{th}$ Pauli term in the Hamiltonian using direct measurement is:

$$\frac{\partial E_j(\tilde{\theta})}{\partial \theta_m} = \sum_{v=1}^{N_{sh}} \left[ \frac{\partial E_j(\tilde{\theta})}{\partial \theta_m} \right] \theta_m \theta_{m,v} = \theta_m \forall v$$

$$\sum_{v=1}^{N_{sh}} (A_{jm,v,+} - A_{jm,v,-})$$

Here the sum over $v$ is the sum over all the parametrised gates that share the same parameter $\theta_m$ as discussed in Appendix C 3. Hence, we have:

$$\text{Var} \left[ \partial_m E_j \right] = 2 N_{sh} \text{Var} [\bar{A}]$$

b. Number of samples needed

Assuming $\text{Var} [A] \sim O(1)$, then the number of samples needed to achieve sample average variance $\text{Var} [\bar{A}]$ is

$$\frac{\text{Var} [A]}{\text{Var} [\bar{A}]} \sim \frac{1}{\text{Var} [\bar{A}]}$$

Since we need to evaluate sample average $\bar{A}$ at two $N_{sh}$ points in direct measurement, the total number of samples needed is:

$$M_{dm} \sim \frac{2 N_{sh}}{\text{Var} [\bar{A}]} = \frac{(2 N_{sh})^2}{\text{Var} [\bar{A}]}$$

(D5)

When compared to (D4) of finite difference, we can see that direct measurement has better scaling in terms of both the number of shared of parameters $N_{sh}$ and the target gradient precision $\text{Var} [\partial_m E_j]$, thus direct measurement is preferred.
3. Comparison between Finite Difference and Direct Measurement

To compare the two methods, we use (D4) and (D5) to study

\[ \frac{M_{dm}}{M_{fd}} = 4 \sqrt{\text{Var}[\partial_mE_j]} / N_{sh} \]

If we define the breaking point of the gradient precision as:

\[ \epsilon^{*}_{\text{grad}} = \frac{N_{sh}}{4} \]  \hspace{1cm} (D6)

then we have:

- \( \sqrt{\text{Var}[\partial_mE_j]} \geq \epsilon^{*}_{\text{grad}} \Rightarrow \frac{M_{dm}}{M_{fd}} \geq 1 \): Finite difference need less samples to achieve the given precision.

- \( \sqrt{\text{Var}[\partial_mE_j]} < \epsilon^{*}_{\text{grad}} \Rightarrow \frac{M_{dm}}{M_{fd}} < 1 \): Direct measurement need less samples to achieve the given precision.

\( N_{sh} = 4N_{eq} \) for hopping term (if we assume spin symmetry) and \( N_{sh} = 3N_{eq} \) for repulsion term where \( N_{eq} \) is the number of equivalent partition in the site layout due to symmetry. Here we will take the approximation that all \( N_{sh} = 4N_{eq} \) since there are more hopping term than repulsion terms. Thus we have:

\[ \text{Var}[\partial_mE_j] = 8N_{eq}\text{Var}[A] \]  \hspace{1cm} (D7)

Using the parametrisation discuss in Section IV A, for open boundary Hubbard model the breaking point of the gradient precision \( \epsilon^{*}_{\text{grad}} \) is:

- Square site layout:
  \( N_{eq} = 8 \Rightarrow \epsilon^{*}_{\text{grad,eq}} = 2.5 \times 10^{-4} \)

- Rectangular site layout:
  \( N_{eq} = 4 \Rightarrow \epsilon^{*}_{\text{grad,rt}} = 1 \times 10^{-3} \)

4. Number of samples needed for energy

As discussed in [30], to compete with the best classical algorithm we need to estimate the energy per site \( E_{\text{site}} \) to \( 10^{-3}t \) precision.

We can decompose the total energy into its subterms \( E_j \):

\[ E_{\text{tot}} \approx \sum_{j=1}^{J} h_j E_j \]

here \( h_j \) is the coefficient of the Pauli decomposition of the Hamiltonian \( H = \sum_{j=1}^{J} h_j G_j \).

The repulsion terms and hopping terms in the Hamiltonian can be decomposed into their Pauli components:

\[ E_{\text{rep}} = \frac{1}{4} + \sum_{j=1}^{3} \frac{1}{4} E_j \]

\[ E_{\text{hop}} = \sum_{j=1}^{2} \frac{1}{2} E_j \]

Hence, the variance in their sampling average are:

\[ \text{Var}[E_{\text{rep}}] = \frac{3}{4^2} \text{Var}[E_j] \]

\[ \text{Var}[E_{\text{hop}}] = \frac{1}{2} \text{Var}[E_j] \]

There are \( V \) repulsion terms and \( 4V \) hopping terms, hence the total energy is:

\[ \sum_{k=1}^{V} E_{\text{rep},k} + \sum_{k=1}^{4V} E_{\text{hop},k} = V E_{\text{site}} \]

which translate into the following equation for variance:

\[ V\text{Var}[E_{\text{rep}}] + 4V\text{Var}[E_{\text{hop}}] = V^2\text{Var}[E_{\text{site}}] \]

\[ \text{Var}[E_j] = \frac{16}{35} V \times 10^{-6} \]

As mentioned above, we want to achieve \( \text{Var}[E_{\text{site}}] = (10^{-3}t)^2 \), assuming \( t \sim O(1) \), we have:

\[ \text{Var}[E_j] = \frac{35}{16V} V \times 10^{-6} \]  \hspace{1cm} (D8)

Assuming \( \text{Var}[E_j] \sim O(1) \), then the number of samples needed to achieve sample average variance \( \text{Var}[E_j] \) is:

\[ M_{E_j} \sim \frac{\text{Var}[E_j]}{\text{Var}[E_j]} \sim \frac{1}{\text{Var}[E_j]} = \frac{35}{16V} \times 10^6. \]

As mentioned in Section IV C, we need 5 circuit runs to evaluate all energy subterms. Thus, the total number of circuit runs needed to evaluate the energy to the required precision is:

\[ M_E = 5M_{E_j} = \frac{1.1 \times 10^7}{V} \]

Thus for \( V = 25 \), we have:

\[ M_E \approx 4 \times 10^5 \]

5. Number of samples needed for energy gradients

First we need to decide what precision of the energy gradient is needed. In Appendix D 4, we have obtained the precision of the energy subterms that we want to
achieve. If we are using finite difference method, in order to achieve such a precision in the final result energy subterms, we can evaluate the energy points in the gradient estimation to the same precision, and our terminating threshold of change in the estimated energy subterms can be set to the same precision. In such case, the gradient precision that we can achieve can be obtained using (D3) and ((D8)):

\[
\text{Var} [\partial_mE_j] = 0.63N_{sh}^2\text{Var} [E_j]^{\frac{5}{2}} = 3.7N_{sh}^2V^{\frac{5}{2}} \times 10^{-5}
\]  
(D9)

To achieve the same precision using direct measurement, the number of sample needed can be obtained by substituting this into (D5):

\[
M_{dm} \approx \frac{(2N_{sh})^2}{\text{Var} [\partial_mE_j]} = 1.1V^{-\frac{5}{2}} \times 10^5.
\]  
(D10)

Note that this is just the number of samples needed to obtain the gradient of an energy subterm w.r.t. one parameter. To obtain the full gradient vector, we need to iterate over all parameters. Using (3), the number of circuit runs needed to evaluate \(\frac{\partial E}{\partial \theta_m}\) for all parameters \(m\) using direct measurement is

\[
M_{\text{grad.,j}} = M_{dm}N_{\text{para}} \approx \frac{1.1N_{\text{site}}N_{\text{blk}}V^{\frac{5}{2}}}{N_{eq}} \times 10^5
\]

As mentioned in Section IV C, we need 5 circuit runs to measure all energy subterms. Thus, the total number of circuit runs needed to evaluate the energy gradient vector to the required precision is:

\[
M_{\text{grad}} = 5M_{\text{grad.,j}} = \frac{5.5N_{\text{site}}N_{\text{blk}}V^{\frac{5}{2}}}{N_{eq}} \times 10^5
\]

For 5 × 5 Hubbard model, we have \(V = 25, N_{eq} = 8\) and \(N_{\text{site}} = 5\). Assuming \(N_{\text{blk}} = V\), the number of circuit runs needed is:

\[
M_{\text{grad}} \approx 2.5 \times 10^7
\]

Appendix E: Number of Circuit Runs for Two-point Exponential Extrapolation

The equation for two-point exponential extrapolation along the curve \(\langle O(\epsilon) \rangle = Ae^{-B\epsilon} \) is:

\[
\overline{O}_{\text{est}} = \left( \frac{\overline{O}_{\epsilon}}{\overline{O}_{\lambda \epsilon}} \right)^{\frac{1}{2}}
\]  
(E1)

Assuming \(\text{Var}[O_{\epsilon}] = \text{Var}[O_{\lambda \epsilon}]\), then the variance in our estimated observable \(O_{\text{est}}\) is

\[
\text{Var}[O_{\text{est}}] = \frac{(\lambda \overline{O}_{\text{est}})^2 + (\overline{O}_{\text{est}})^2}{(\lambda - 1)^2} \text{Var}[O_{\epsilon}]
\]

To keep the \(\text{Var}[O_{\text{est}}]\) at a target precision level, we thus need \(C_{\text{exp}}\) times more circuit runs than keeping \(\text{Var}[O_{\epsilon}]\) at the same precision level, in which

\[
C_{\text{exp}} = 2\frac{(\lambda \overline{O}_{\text{est}})^2 + (\overline{O}_{\text{est}})^2}{(\lambda - 1)^2} = C_{\text{ext}}
\]

The factor of 2 is due to the fact that we need to sample both \(O_{\epsilon}\) and \(O_{\lambda \epsilon}\).

Under Markovian noise, we expect \(\overline{O}_{\epsilon} \leq \overline{O}_{\text{ext}}\), because noise will bring in some mixed state characters which will lower the expectation value [24]. Hence,

\[
C_{\text{exp}} \geq 2\frac{\lambda^2 + 1^2}{(\lambda - 1)^2} = C_{\text{ext}}
\]

where \(C_{\text{ext}}\) is the extra factor of circuit runs needed for two-point linear extrapolation as discussed in Section IV B 1.

For small amount of noise (large probability of a perfect circuit run), we should expect \(\overline{O}_{\epsilon} \sim \overline{O}_{\lambda \epsilon} \sim \overline{O}_{\text{ext}}\), and hence \(C_{\text{exp}} \sim C_{\text{ext}}\).

Appendix F: Quantum dot layout

Figure 5, 6 shows how we can perform the non-demolishing measurements required by our problem using three lines of quantum dot, one line for data and two lines for ancilla. The ancilla here come in the form of double dot, which will be initialised in singlet and will be readout using pauli spin blockade which enable us to distinguish singlet and triplet state. After that, we can compose XX and YY of the hopping terms. Assume that, we can compose XX and YY to obtain ZZ which in terms can give us electron number parity for symmetry verification. Alternatively, we can also measure XX and ZZ, and obtain YY via post-processing since control-Z gate can be easier to implement in silicon qubits. Figure 5 shows how we perform Z-measurement for every data dots to obtain the repulsion terms. The electron number parity in symmetry verification can again be obtained via composing our measurement results. Our measurement can also be carried out using only one row of ancilla. In such a case, we essentially using one row of ancilla to carry out two rounds of measurements with ancilla reinitialisation in between. Other architectures that incorporating spin-to-charge readout using reservoirs are also possible.
FIG. 5. The measurements of the hopping terms. The blue interaction here are control-X gate from the ancilla to the data for the measurement of $XX$ in the hopping term. The red interaction here are control-Y gate from the ancilla to the data for the measurement of $YY$ in the hopping term.

FIG. 6. The measurements of the repulsion terms. All the interaction here are control-Z gates. One row of ancilla is in charge of $Z$ measurements of the odd data dots while the other row of ancilla is in charge of $Z$ measurements of the even data dots.

[1] Richard P. Feynman, “Simulating physics with computers,” International Journal of Theoretical Physics 21, 467–488 (1982).
[2] Ian D. Kivlichan, Craig Gidney, Dominic W. Berry, Nathan Wiebe, Jarrod McClean, Wei Sun, Zhang Jiang, Nicholas Rubin, Austin Fowler, Alán Aspuru-Guzik, Ryan Babbush, and Hartmut Neven, “Improved Fault-Tolerant Quantum Simulation of Condensed-Phase Correlated Electrons via Trotterization,” arXiv:1902.10673 [physics, physics:quant-ph] (2019).
[3] Dave Wecker, Matthew B. Hastings, and Matthias Troyer, “Progress towards practical quantum variational algorithms,” Physical Review A 92 (2015).
[4] L. M. K. Vandersypen, H. Bluhm, J. S. Clarke, A. S. Dzurak, R. Ishihara, A. Morello, D. J. Reilly, L. R. Schreiber, and M. Veldhorst, “Interfacing spin qubits in quantum dots and donors—hot, dense, and coherent,” npj Quantum Information 3 (2017).
[5] Sam McArdle, Suguru Endo, Alán Aspuru-Guzik, Simon Benjamin, and Xiao Yuan, “Quantum computational chemistry,” arXiv:1808.10402 [quant-ph] (2018).
[6] Yudong Cao, Jonathan Romero, Jonathan P. Olson, Matthias Degroote, Peter D. Johnson, Mária Kieferová, Ian D. Kivlichan, Tim Menke, Borja Peropadre, Nicolas P. D. Sawaya, Sukin Sim, Libor Veis, and Alán Aspuru-Guzik, “Quantum Chemistry in the Age of Quantum Computing,” arXiv:1812.09976 [quant-ph] (2018).
[7] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta, “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets,” Nature 549, 242–246 (2017).
[8] Jonathan Romero, Ryan Babbush, Jarrod R. McClean, Cornelius Hempel, Peter J. Love, and Alán Aspuru-Guzik, “Strategies for quantum computing molecular energies using the unitary coupled cluster ansatz,” Quantum Science and Technology 4, 014008 (2018).
[9] Pierre-Luc Dallaire-Demers, Jonathan Romero, Libor Veis, Sukin Sim, and Alán Aspuru-Guzik, “Low-depth circuit ansatz for preparing correlated fermionic states on a quantum computer,” arXiv:1801.01053 [quant-ph] (2018).
[10] Ryan Babbush, Nathan Wiebe, Jarrod McClean, James McClain, Hartmut Neven, and Garnet Kin-Lic Chan, “Low-Depth Quantum Simulation of Materials,” Physical Review X 8, 011044 (2018).
[11] Ian D. Kivlichan, Jarrod McClean, Nathan Wiebe, Craig Gidney, Alán Aspuru-Guzik, Garnet Kin-Lic Chan, and Ryan Babbush, “Quantum Simulation of Electronic Structure with Linear Depth and Connectivity,” Physical Review Letters 120, 110501 (2018).
[12] F Verstraete and J I Cirac, “Mapping local Hamiltonians of fermions to local Hamiltonians of spins,” Journal of Statistical Mechanics: Theory and Experiment 2005, P09012–P09012 (2005).
[13] Sergey Bravyi and Alexei Kitaev, “Fermionic quantum computation,” Annals of Physics 298, 210–226 (2002).
M. Ganzhorn, D. J. Egger, P. Barkoutsos, P. Ollitrault, G. Salis, N. Moll, M. Roth, A. Fuhrer, P. Mueller, S. Werner, I. Tavernelli, and S. Filipp, “Gate-Efficient Simulation of Molecular Eigenstates on a Quantum Computer,” Physical Review Applied 11, 044092 (2019).

Guillaume Verdon, Michael Broughton, Jarrod R. McClean, Kevin J. Sung, Ryan Babbush, Zhang Jiang, Harmut Neven, and Masoud Mohseni, “Learning to learn with quantum neural networks via classical neural networks,” arXiv:1907.05415 [quant-ph] (2019).

Max Wilson, Sam Stromswold, Filip Wudarski, Stuart Hadfield, Norm M. Tubman, and Eleanor Riefel, “Optimizing quantum heuristics with meta-learning,” arXiv:1908.03185 [quant-ph] (2019).

Ken M. Nakanishi, Keisuke Fujii, and Synge Todo, “Sequential minimal optimization for quantum-classical hybrid algorithms,” arXiv:1903.12166 [physics, physics:quant-ph] (2019).

J. C. C. Hwang, C. H. Yang, M. Veldhorst, N. Hendrickx, M. A. Fogarty, W. Huang, F. E. Hudson, A. Morello, and A. S. Dzurak, “Impact of $\delta g$-factors and valleys on spin qubits in a silicon double quantum dot,” Physical Review B 96, 045302 (2017).

K. C. Nowack, M. Shafiei, M. Laforest, G. E. D. K. Prawiroatmodjo, L. R. Schreiber, C. Reichl, W. Wegscheider, and L. M. K. Vandersypen, “Single-Shot Correlations and Two-Qubit Gate of Solid-State Spins,” Science 333, 1269–1272 (2011).

W. Huang, C. H. Yang, K. W. Chan, T. Tanttu, B. Rensen, R. C. C. Leon, M. A. Fogarty, J. C. C. Hwang, F. E. Hudson, K. M. Itoh, A. Morello, A. Laucht, and A. S. Dzurak, “Fidelity benchmarks for two-qubit gates in silicon,” Nature 569, 532 (2019).

Guoji Zheng, Nodar Samkharadze, Marc L. Noordam, Nima Kalhor, Delphine Brousse, Amir Sammak, Giordano Scappucci, and Lieven M. K. Vandersypen, “Rapid gate-based spin read-out in silicon using an on-chip resonator,” Nature Nanotechnology 14, 742–746 (2019).

S. Schaal, I. Ahmed, J. A. Haigh, L. Huttin, B. Bertrand, S. Barraud, M. Vinet, C.-M. Lee, N. Stelsmashenko, J. W. A. Robinson, J. Y. Qiu, S. Hacohen-Gourgy, I. Siddiqi, M. F. Gonzalez-Zalba, and J. J. L. Morton, “Fast gate-based readout of silicon quantum dots using Josephson parametric amplification,” arXiv:1907.09429 [cond-mat, physics:quant-ph] (2019).

J. M. Elzerman, R. Hanson, L. H. Willems van Beveren, B. Witkamp, L. M. K. Vandersypen, and L. P. Kouwenhoven, “Single-shot read-out of an individual electron spin in a quantum dot,” Nature 430, 431–435 (2004).

V. Srinivasa, K. C. Nowack, M. Shafiei, L. M. K. Vandersypen, and J. M. Taylor, “Simultaneous Spin-Charge Relaxation in Double Quantum Dots,” Physical Review Letters 110, 196803 (2013).

Benoit Bertrand, Hanno Flentje, Shintaro Takada, Michi-hisa Yamamoto, Seigo Tarucha, Arne Ludwig, Andreas D. Wieck, Christopher Bäuerle, and Tristan Meunier, “Quantum Manipulation of Two-Electron Spin States in Isolated Double Quantum Dots,” Physical Review Letters 115, 096801 (2015).

M. A. Fogarty, K. W. Chan, B. Hensen, W. Huang, T. Tanttu, C. H. Yang, A. Laucht, M. Veldhorst, F. E. Hudson, K. M. Itoh, D. Culcer, T. D. Ladd, A. Morello, and A. S. Dzurak, “Integrated silicon qubit platform with single-spin addressability, exchange control and single-shot singlet-triplet readout,” Nature Communications 9, 4370 (2018).

C. H. Yang, R. C. C. Leon, J. C. C. Hwang, A. Saraiva, T. Tanttu, W. Huang, J. Camirand Lemyre, K. W. Chan, K. Y. Tan, F. E. Hudson, K. M. Itoh, A. Morello, M. Pioro-Ladrière, A. Laucht, and A. S. Dzurak, “Silicon quantum processor unit cell operation above one Kelvin,” arXiv:1902.09126 [cond-mat, physics:quant-ph] (2019).

Andrea Morello, Jarryd J. Pla, Floris A. Zwanenburg, Kok W. Chan, Kuan Y. Tan, Hans Huebl, Mikko Möttönen, Christopher D. Nugroho, Changyi Yang, Jessica A. van Donkelaar, Andrew D. C. Alves, David N. Jamieson, Christopher C. Escott, Lloyd C. L. Hollenberg, Robert G. Clark, and Andrew S. Dzurak, “Single-shot readout of an electron spin in silicon,” Nature 467, 687–691 (2010).

M. D. Shulman, S. P. Harvey, J. M. Nichol, S. D. Bartlett, A. C. Doherty, V. Umaysk, and A. Yacoby, “Suppressing qubit dephasing using real-time Hamiltonian estimation,” Nature Communications 5, 5156 (2014).

Zhenyu Cai, Xiaosi Xu, and Simon C. Benjamin, “Mitigating Coherent Noise Using Pauli Conjugation,” arXiv:1906.06270 [quant-ph] (2019).

Jarrod R. McClean, Mollie E. Kimchi-Schwartz, Jonathan Carter, and Wibe A. de Jong, “Hybrid quantum-classical hierarchy for mitigation of decoherence on excited states,” Physical Review A 95 (2017).

Zhang Jiang, Jarrod McClean, Ryan Babbush, and Harmut Neven, “Majorana loop stabilizer codes for error correction of fermionic quantum simulations,” arXiv:1812.08190 [quant-ph] (2018).

Jarrod R. McClean, Zhang Jiang, Nicholas C. Rubin, Ryan Babbush, and Harmut Neven, “Decoding quantum errors with subspace expansions,” arXiv:1903.05786 [physics, physics:quant-ph] (2019).

Kosuke Mitarai and Keisuke Fujii, “Methodology for replacing indirect measurements with direct measurements,” arXiv:1901.00015 [quant-ph] (2018).