Non-unitary Trotter circuits for Imaginary Time Evolution

BCN:MP48J

Abstract

Imaginary time evolution (ITE) is a method to obtain the ground state energy of a fermionic Hamiltonian, which is of high interest to chemists to gauge the thermodynamic properties of the system. To implement a quantum ITE algorithm, we modify the existing Pauli gadget circuit framework for trotterised real time evolution (RTE), using ancillary qubits and mid-circuit measurements instead to simulate a block encoded version of the ITE operator rather than the RTE operator. This forms a probabilistic ITE (PITE) algorithm, where all mid-circuit measurements must be post selected for a successful implementation. The algorithm is tested on a 4-site Transverse Ising Model and a 2-site Fermi Hubbard Model through a classical statevector simulation of the quantum circuits, demonstrating its ability to reach the ground state. Further work is required to increase the probability of success, reduce the circuit depth, and increase the rate of execution for each shot.

1. Introduction to quantum circuits

Analogous to classical computing, where the unit of information is a binary digit (bit), the basis of computation is a quantum bit (qubit) in quantum computing. Unlike classical bits, which are either 0 or 1, each qubit is a vector in a Hilbert space $\mathcal{H}^2$,

$$|\psi\rangle = a\,|0\rangle + b\,|1\rangle \in \mathcal{H}^2, \quad a, b \in \mathbb{C}$$

expressed as a superposition of two basis states. The set $\{|0\rangle, |1\rangle\}$ forms the most common basis, referred to as the *computational basis*. Upon measurement, $|\psi\rangle$ collapses onto the state $|0\rangle$ with probability $|a|^2$, and $|1\rangle$ with probability $|b|^2$. This is reflected by the normalisation condition, $|a|^2 + |b|^2 = 1$, such that probabilities sum to one.

An n-qubit state is an element in $\mathcal{H}^{2n}$, created by taking tensor products of the Hilbert spaces of their individual qubits. For example, for a 2-qubit system, the statevector in the computational basis reads

$$|\psi\rangle = a\,|00\rangle + b\,|01\rangle + c\,|10\rangle + d\,|11\rangle \in \mathcal{H}^2 \otimes \mathcal{H}^2,$$

where the basis $|00\rangle$ is an abbreviation of $|0\rangle \otimes |0\rangle$, and so on for the other basis vectors. Borrowing from classical computation terminology, any state of 0's and 1's, such as $|01\ldots1\rangle$, is referred to as a bit string.

Operations on $|\psi\rangle$ must preserve the normalisation criterion to give meaningful probabilities. More generally, the inner product should be preserved upon being operated on for all $|\psi\rangle$, where $\langle\psi|\psi\rangle = 1 \implies \langle\psi|U^\dagger U|\psi\rangle = 1$. Such operations are *unitary*, identifying the property

$$U^\dagger U = I, \tag{1}$$

where I is the identity operator.

In classical computing, a set of logic gates is *universal* if any boolean function $f : \{0,1\}^n \to \{0,1\}^m \quad n, m \in \mathbb{N}$ can be implemented by a combination of gates in this set. One such example is the set of classical gates {AND, OR, NOT}. Likewise, a set of quantum gates are approximately universal if for any unitary operation U and given $\epsilon \in \mathbb{R}$, Equation (2) holds:

$$\|U - U_t \ldots U_2 U_1\| \leq \epsilon, \tag{2}$$

where $\|.\|$ indicates the spectral norm and $U_1, \ldots U_t$ are unitary quantum gates. In words, any unitary on any number of qubits can be approximated to an arbitrary precision ϵ by gates in this set. These are at best an approximation for a finite or countably infinite gate set, as the set of unitaries are uncountably infinite. Although unitaries exist in all n-dimensional spaces, the number of qubits must be finite in a physical quantum computer. This is especially true today, where *fault tolerant quantum*

computers remain an ideal; practically, we can only work with $\lessapprox 1000$ qubits in the noisy intermediate-scale quantum era (NISQ) [1]. Fortunately, the set of two-qubit unitaries are universal, such that arbitrarily sized quantum gates can be decomposed into combinations of one to two-qubit gates [2].

Altogether, this permits the quantum circuit model, where a combination of operations can be visualised as one or two-qubit gates. Any unitary operator on one or two qubits is automatically a valid quantum gate, so it may not be surprising to include the Pauli gates,

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{3}$$

equivalent to Pauli matrices in other quantum mechanical contexts. Functions of these Pauli gates can also be defined by taking the spectral decomposition of the operators. For instance, the qubit rotation gates are given by applying the exponential map on the Pauli gates

$$R_x(\theta) = e^{\left(\frac{-i\theta X}{2}\right)}, R_y(\theta) = e^{\left(\frac{-i\theta Y}{2}\right)}, R_z(\theta) = e^{\left(\frac{-i\theta Z}{2}\right)}, \tag{4}$$

for some rotation angle θ, where the exponential of an operator is related to its Taylor expansion.

Other gates include the Hadamard gate and the S gate

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}. \tag{5}$$

From Equation (3), we can see that $\{X, Y\}$ are not diagonal in the computational basis. By applying suitable basis transformations involving the gates in Equation (5), these can be diagonalised by

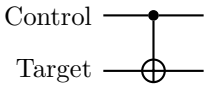
$$H^\dagger Z H = HZH = X \tag{6}$$

$$(HS)^\dagger Z H S = S^\dagger H Z H S = Y. \tag{7}$$

Finally, we introduce our fist 2-qubit gate: the controlled NOT (CNOT) gate, which requires a control and a target qubit. If the control is in the state $|0\rangle$, the target remains unchanged. If it is in $|1\rangle$, X acts on the target, flipping the qubit. Hence, the CNOT is also sometimes referred to as the CX gate. In the computational basis ($|00\rangle, |01\rangle, |10\rangle, |11\rangle$), where the control is the leftmost qubit, the matrix representation is

$$\text{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \tag{8}$$

where X is explicitly shown in the lower right block, and I on the upper right. This appears as



(9)

in a circuit diagram, where the black dot and $\oplus$ represents the control and target respectively.

To use a quantum circuit, a well defined state is given as input, typically being $|\mathbf{0}\rangle \equiv |0\rangle^{\otimes n} = |0\rangle \otimes |0\rangle \otimes \cdots \otimes |0\rangle$, ensuring a clear, predictable starting point. Qubits are measured at the end onto classical bits to extract information after being operated on by quantum gates. Mid-circuit measurements can also be made to strategically collapse $|\psi\rangle$ onto selected subspaces of $\mathcal{H}^{2^n}$, a technique used extensively in this paper.

2. Quantum Computing for Chemistry

Given a wavefunction $|\psi(t)\rangle$, where $\hat{H} = \hat{H}^\dagger$, the ground state is the eigenvector $|\phi_0\rangle$ associated with the smallest energy eigenvalue E_0. To avoid confusion, we reserve the hat notation for the Hamiltonian $\hat{H}$, instead of the Hadamard gate H. For applications in quantum chemistry, we are specifically interested in the ground state of $\hat{H}$, which details the equilibrium properties of matter at low temperatures, where quantum effects are strongest.

It has been long hypothesised that one of the main uses for quantum computers is to speed up quantum simulations and obtain the ground state [3, 4]. Summaries for two proposed methods of obtaining the ground state are given in Sections 2.2 and 2.3, but we must first convert our chemical Hamiltonian into a problem that can be implemented on a quantum computer. Concretely, we requiring a mapping of $\hat{H}$, expressed in creation and annihilation operators, onto Pauli spin operators.

2.1. Jordan Wigner Map

There are several common mappings to choose from: Jordan-Wigner (JW) [5], Parity map, and Bravyi–Kitaev [6]. Regardless of the specific mapping, the resulting $\hat{H}$ is decomposed into tensor products of Pauli operators, known as Pauli strings,

$$\hat{H} = \sum_{k=1}^{M} c_k \hat{\sigma}_{\boldsymbol{m}_k}, \tag{10}$$

where $\hat{\sigma}_{\boldsymbol{m}}$ is a Pauli string,

$$\hat{\sigma}_{\boldsymbol{m}} = \hat{\sigma}_{m_{n-1}} \otimes \ldots \otimes \hat{\sigma}_{m_0}, \quad \hat{\sigma}_m \in \{I, X, Y, Z\}.$$

We concentrate on the JW map for its simplicity, which is also used in this paper. Due to the Pauli Exclusion principle, fermions with the same quantum numbers can only occupy a maximum of one quantum state. This suggests a mapping into the computational basis $|0\rangle$ for an unoccupied state, and $|1\rangle$ for an occupied state. To preserve the antisymmetry in the wavefunction under exchange of two fermions, correct commutation relations must be obeyed, given by Equation (11),

$$\begin{aligned} c_j &\Leftrightarrow I^{\otimes j-1} \otimes \sigma^+ \otimes Z^{\otimes N-j-1} \\ c_j^\dagger &\Leftrightarrow I^{\otimes j-1} \otimes \sigma^- \otimes Z^{\otimes N-j-1}, \end{aligned} \tag{11}$$

where $\sigma^+ \equiv \frac{X+iY}{2}$ and $\sigma^- \equiv \frac{X-iY}{2}$, and c_j, $c_j^\dagger$ are creation and annihilation operators.

2.2. Quantum Phase Estimation

Quantum phase estimation (QPE) is a quantum algorithm that can estimate the phase of the eigenvalue of a unitary [7, 8]. For Hamiltonians, by setting the unitary as the real time evolution operator $U = e^{-i\hat{H}t}$, we can obtain all energy eigenvalues E_i, including E_0, as well as their respective eigenvectors.

Without going into full mathematical detail, two qubit registers are initialised, where a register is a collection of qubits. We designate the first qubit register as the 'data' qubits, which encodes $|\psi\rangle$ and is ultimately measured to produce the resulting phase information. The second register contains ancillary qubits (ancillas), which aid the calculation. The data qubits are initialised as an initial guess of the ground state $|\psi(t=0)\rangle$. The ancillas are initialised at $|\mathbf{0}\rangle$, and each applied with an H gate, placing each qubit as an equal superposition. Then a series of controlled-U^{2j+1} act on the data register, with each control being the j^{th} ancilla. After all ancillas are connected, the Inverse Quantum Fourier Transform (QFT) is applied to all ancillas, before finally measuring the ancillas onto classical bits. With probability $|c_m|^2$, this measurement simultaneously gives the measured ancillary bit string of $2^k E_m/2\pi$, which contains our energy information, and also leaves the data qubits as the corresponding $|\phi_m\rangle$. To maximise the probability of obtaining the ground state, the guess $|\psi(0)\rangle$ should maximise the overlap $\langle\psi(0)|\phi_0\rangle$, equivalent to expanding $|\psi(0)\rangle = \sum c_m |\phi_m\rangle$ and seeking the highest c_0.

QPE is an essential algorithm that is not only useful in finding ground and excited states, but also serves as the backbone of many other quantum algorithms. However, current implementations on quantum hardware are prevented as they are not fault tolerant, meaning any errors accumulated in the algorithm are not caught and corrected. In particular, applying inverse QFT requires a large amount of rotation gates, which dramatically increases readout error [9]. To address these concerns, several variants of QPE have been proposed, such as control-free QPE, aiming to replace the controlled-U operations with U itself.

2.3. Variational Quantum Eigensolver

The variational quantum eigensolver (VQE) is a hybrid quantum-classical algorithm [10], where parts of the algorithm are performed on a classical computer. This has the major advantage of being useful for NISQ devices as these algorithms use fewer qubits, offloading parts of the computation to classical computers where quantum computers do not have an advantage. VQE uses an *ansatz* wavefunction $\psi(\boldsymbol{\theta})$, parameterised by some $\boldsymbol{\theta} = (\theta_0, \theta_1, \ldots, \theta_n)$, where $\theta_i \in (-\pi, \pi]$ The algorithm relies on the Rayleigh-Ritz variational principle

$$\langle\psi(\boldsymbol{\theta})|\hat{H}|\psi(\boldsymbol{\theta})\rangle \geq E_0, \tag{12}$$

Then, the *ansatz* wavefunction is created on a quantum computer by a sequence of quantum gates U operating on an initial register $|\mathbf{0}\rangle$

$$|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})\,|\mathbf{0}\rangle\,. \tag{13}$$

By minimising the cost function $\langle\mathbf{0}|U^\dagger(\boldsymbol{\theta})\hat{H}U(\boldsymbol{\theta})|\mathbf{0}\rangle$,

$$E = \min_{\boldsymbol{\theta}} \langle\mathbf{0}|U^\dagger(\boldsymbol{\theta})\hat{H}U(\boldsymbol{\theta})|\mathbf{0}\rangle\,, \tag{14}$$

this finds the optimal $\boldsymbol{\theta}$ such that the $\langle\hat{H}\rangle$ is minimised, turning Equation (12) into an equality. The minimisation step is performed classically using any optimisation algorithm. The choice of ansatz is critical, requiring a mix between expressibility and trainability. Expressibility refers to how uniformly the ansatz explores the parameter space, but this negatively affects the trainability when minimising the cost functions, creating 'Barren Plateaus' in the cost function [11]. When $\boldsymbol{\theta}$ is computed

using gradient-based optimisers, such as Newton's method, many regions in the loss landscape exhibit large, flat regions across the parameter space. For a cost function to exhibit a barren plateau, the gradient of the objective function vanishes exponentially with the number of qubits [12]. This reduces the quantum advantage relative to classical algorithms.

3. Probabilistic Imaginary Time Evolution

The Imaginary Time Evolution operator $e^{-\hat{H}\tau}$ evolves the state $|\psi(\tau)\rangle$ through imaginary time $\tau = it$ [13]. This is a ground state projector in the infinite time limit. To see this, consider

$$|\psi(\tau)\rangle = e^{-\hat{H}\tau} |\psi(0)\rangle , \tag{15}$$

and expand out $|\psi(\tau = 0)\rangle$ in terms of energy eigenstates $|\phi_m\rangle$, which form a complete orthonormal basis for a Hermitian operator,

$$|\psi(\tau)\rangle = \sum_m c_m(0) e^{-\hat{H}\tau/\hbar} |\phi_m\rangle . \tag{16}$$

The exponential can be expanded as a Taylor series,

$$e^{-\hat{H}\tau/\hbar} = \frac{\tau \hat{I}}{\hbar} - \frac{\tau \hat{H}}{\hbar} + \dots$$

Since $|\phi_m\rangle$ are energy eigenstates, $\hat{H}^n |\phi_m\rangle = (E_m)^n |\phi_m\rangle$. Putting it back into exponential form

$$|\psi(\tau)\rangle = \sum_m c_m(0) e^{-E_m\tau/\hbar} |\phi_m\rangle , \tag{17}$$

shows that each $|\phi_m\rangle$ is exponentially decaying with a characteristic time scale $\hbar/E_m$ multiplied by some coefficient $c_m(0)$. Hence, the ground state energy decays the slowest, as E_0 is smallest by definition. For $\tau \gg 1$, the dominant term in the sum is the ground state. The aim of this paper is to efficiently simulate $e^{-\hat{H}\tau}$ using quantum circuits. We will closely follow the Pauli gadget primitive for a Trotterised real time evolution operator, but use mid circuit measurements an ancillary qubits to form a probabilistic ITE (PITE) algorithm.

3.1. Trotterisation

The Pauli strings resulting from the JW map in Section 2.1 do not commute, making their quantum circuit implementation challenging [14]. Fortunately, Trotterisation is a simple method to consider ITE on individual Pauli strings only, allowing for some error in the process. For Hamiltonians that can be expressed as a sum of local interactions, $\hat{H} = \sum_{m=1}^{M} \hat{H}_m$, where $\hat{H}_m$ act only on a small subset of particles, the first order Trotter-Suzuki formula can be used to decompose the ITE operator into products of exponentials of $\hat{H}_m$,

$$e^{-\hat{H}\tau} = \prod_{m=1}^{M} e^{-\hat{H}_m\tau} + \mathcal{O}\left((L\tau)^2\right), \tag{18}$$

where L is the system size. For systems evolving through a long τ, a substitution can be made where $r\Delta\tau = \tau$, where r is the 'trotter number' and $\Delta\tau$ a small time step to reduce the error to $\mathcal{O}\left((L\Delta\tau)^2 r\right)$, rather than $\mathcal{O}\left((Lr\Delta\tau)^2\right)$. This results in

$$e^{-\hat{H}\tau} = \left(\prod_{m=1}^{M} e^{-\hat{H}_m\Delta\tau} \right)^r + \mathcal{O}\left((L\Delta\tau)^2 r\right). \tag{19}$$

From Equation (10), we identify $\hat{H}_m = c_k \hat{\sigma}_{\boldsymbol{m}_k}$. Hence, each term inside the product in Equation (19) is an exponential of a Pauli string, its corresponding coefficient, and the time step $\Delta\tau$. We can thus focus on a systematic method to generate quantum circuits for the constituent $e^{-\hat{H}_m\tau}$ only, instead of considering the full Hamiltonian.

3.2. Block encoding scheme

Another roadblock is that $e^{-\hat{H}_m\tau}$ is not unitary, therefore not currently expressible by quantum gates. This can be mitigated by *block encoding* an operator. In the matrix representation, this corresponds to embedding $e^{-\hat{H}_m\tau}$ in a larger matrix, such that this larger matrix is unitary. Including the CNOT gate and all single qubit gates detailed in Section 1, and imposing the condition that only one ancillary qubit is used, we only need to add a non-unitary gate $\mathcal{N}_1(a)$ to make this into a universal set for non-unitary operators. $\mathcal{N}_1(a)$ has the matrix representation

$$\mathcal{N}_1(a) = \begin{pmatrix} 1 & 0 \\ 0 & a \end{pmatrix}, \tag{20}$$

which can then be embedded in a larger matrix $U_{\mathcal{N}_1(a)}$,

$$U_{\mathcal{N}_1}(a) = \begin{pmatrix} \mathcal{N}_1 & * \\ * & * \end{pmatrix}, \tag{21}$$

where $*$ are arbitrary so long as $U_{\mathcal{N}_1}(a)$ is unitary. In fact, by using the Pauli gadget primitive in Section 3.3, we only need to concern ourselves with $e^{-|c_k|\Delta\tau} e^{-c_k\Delta\tau Z}$, where $e^{-|c_k|\Delta\tau}$ is a

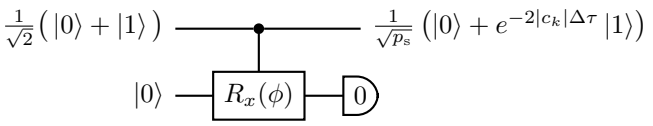


Figure 1: The circuit realisation of $U_{\mathcal{N}_1}\left(e^{-2|c_k|\Delta\tau}\right)$ for $c_k < 0$. This is equivalent to using a controlled-$R_X(\phi)$, where $\phi = 2\arccos\left(e^{-2|c_k|\Delta\tau}\right)$. The example input state is transformed exactly as Equation (20), contingent on the ancillary qubit measuring as $|0\rangle$. For $c_k < 0$, we simply surround the control with two X gates.

scaling factor [15]. Writing the operator in terms of $\mathcal{N}_1(a)$, this gives $\mathcal{N}_1\left(e^{-2|c_k|\Delta\tau}\right)$ for $c_k < 0$, and $X\mathcal{N}_1\left(e^{-2|c_k|\Delta\tau}\right)X$ for $c_k > 0$. The block encoding is mathematically equivalent to using a controlled-$R_X(\phi)$ gate, with $\phi = 2\arccos\left(e^{-2|c_k|\Delta\tau}\right)$, where the ancilla must be measured as $|0\rangle$ to be successful. On a circuit diagram, this is shown in Figure 1.

3.3. Pauli Gadget

The Pauli gadget framework is a circuit realisation for functions of Pauli strings. For our purposes, the actual construction is best illustrated with an example, where the circuit diagram is shown in Figure 2.

Consider a specific Pauli string $\hat{\sigma}_{\boldsymbol{m}_k} = IXYZ$. To write this as a quantum circuit, we first transform $\{X, Y\}$ to the diagonal computational basis by applying suitable transformation operators detailed in Section 1 at the begininng and the end of this circuit. Then, we use a 'ladder' of CNOT gates around the block encoded operator, skipping over any identity operators as they correspond to leaving its qubit untouched. Combined with implementing the controlled $R_X(\phi)$, this extends the block encoding of a single Pauli Z from Figure 1 to the entire Pauli string. This process is then repeated for each Pauli string and coefficient from the Trotter decomposition, following the same approach and adjusting the basis transformation gates accordingly, where the individual circuit blocks are joined sequentially. Finally, this entire circuit is repeated r times to simulate an imaginary time evolution up to τ.

For this ITE implementation to be successful, all mid-circuit from the ancilla measurements must yield $|0\rangle$. The success probability p_s satisfies the lower bound

$$p_s \geq \exp\{-4r\Delta\tau\lambda\}, \tag{22}$$

where $\lambda = \sum |c_k|$, the operator 1-norm.

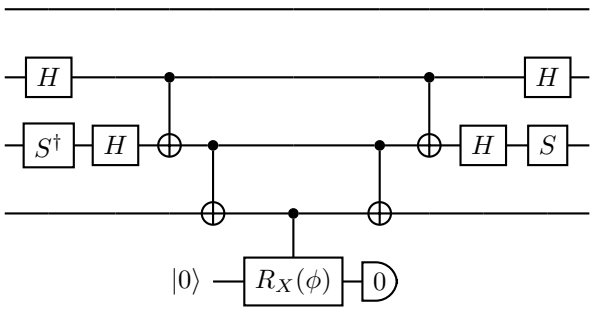


Figure 2: A circuit for simulating $e^{-|c_k|\Delta\tau}e^{-c_k\Delta\tau IXYZ}$, for $c_k < 0$, using the Pauli gadget primitive. The basis transformation gates are used to at the beginning and the end of the circuit to diagonalise X and Y, which are the second and third wires. Then, a ladder of gate joins the qubits together, ignoring the identity which leaves the qubit unchanged. Finally, a controlled $R_X(\phi)$ gate acts on an ancillary qubit initialised to $|0\rangle$, of which must measure as 0 to simulate the correct block encoding of the non-unitary operator. This is repeated for different Pauli strings until the full Hamiltonian is simulated, and repeated again to evolve the stae to τ

4. Systems of interest

The PITE will be tested on two simple systems, frequently used for testing novel quantum chemistry algorithms.

4.1. Transverse-Field Ising Model

The Transverse-Field Ising Model (TIM) has a Hamiltonian in 1-dimension given by

$$\hat{H} = -J\sum_{i=1}^{n} Z_i Z_{i+1} - h\sum_{i=1}^{n} X_i \tag{23}$$

with periodic boundary conditions such that $Z_{n+1} = Z_1$. J denotes the interaction energy between neighbouring sites, and h is a uniform, external magnetic field. It is the classical analogue of the classical Ising model, a toy model for exploring magnetic spins. By tweaking combinations of parameters such as h and temperature T, which do not appear in Equation (23) but is required when minimising the Helmholtz free energy, there exist critical parameters T_c and h_c of which phase transitions occur. Unlike the classical variant, where each spin is in one of the states from {-1, 1}, TIM allows the spins to be in a superposition, mathematically identical to a qubit. In this case, since the Hamiltonian is already given in terms of spin operators, the Pauli strings can simply be read off by expanding the sums, inserting the identity operator where necessary.

We tested a four site $n = 4$ model, with $J = 0.5$, $h = 0.1$, and a time step $\Delta\tau = 0.1$. The initial state supplied was created by applying the Hadamard gate to each qubit, $H^{\otimes n}|0\rangle^{\otimes n}$, yielding an equal superposition in all states.

4.2. Fermi Hubbard Model

The Fermi-Hubbard model (FHM) is an extension of the tight binding model, accounting for the potential energy with two electrons in the same lattice site. The Hamiltonian in second quantisation is given by

$$\hat{H} = t \sum_{\lambda=\uparrow,\downarrow} \sum_{i,j} \hat{c}^{\dagger}_{i,\lambda} \hat{c}_{j,\lambda} + U \sum_{i} \hat{c}^{\dagger}_{i,\uparrow} \hat{c}_{i,\uparrow} \hat{c}^{\dagger}_{i,\downarrow} \hat{c}_{i,\downarrow}, \quad (24)$$

where t is the hopping term, U is the on-site potential, and each second quantised operator acts on a specific site and spin. The first term of Equation (24) represents the kinetic energy of electrons hopping between neighbouring sites, from site i to j, identical to the tight binding model. This competes against the potential energy when two electrons occupy the same lattice site with the same spin. Similar to TIM, The Hubbard model also exhibits phase transitions between conducting and insulating states, depending on the fraction U/t.

Note that some definitions use a matrix t_{ij} inside the sum, allowing for different hopping terms at different neighbouring sites; here we set a constant value of $t = -0.1$ for all sites. We also set $U = 0.1$, and kept $\Delta\tau = 0.1$, identical to TIM. We chose to initialise the circuit in the singlet state $\frac{1}{\sqrt{2}}(|0110\rangle - |1001\rangle)$.

5. Results

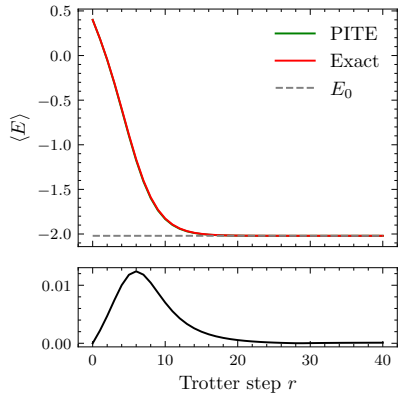
Running a quantum algorithm on a quantum computer requires *shots*, where each shot corresponds to initialising the circuit, running the qubits through the circuit, and measuring the qubits onto classical bits. In our case, this also requires post selecting all mid circuit ancillary bits to $|0\rangle$. Hence, a large number of shots are required, such that the subset of shots corresponding to a successful post selection can be used to further construct a sample mean and standard deviation for obtaining $\langle E\rangle$ at each r. To circumvent the high computational cost and also for ease of implementation, the statevector simulator was used, only possible on classical simulators of quantum computers which manipulate the statevector $|\psi\rangle$ directly, which is not observable in quantum mechanics. For each mid-circuit measurement, the post selection was artificially made successful by setting all amplitudes corresponding to a non-zero ancilla to zero, such that the probability of a failed post selection was also zero. The remaining statevector was renormalised to unity and propagated through the circuit.

Figures 3 and 4 shows the results of the TIM and FHM models respectively. Each figure contains three subfigures, showcasing $\langle E\rangle$ converging to E_0, the difference $\langle E\rangle - E_0$, and the probability of success p_s, for any given r.

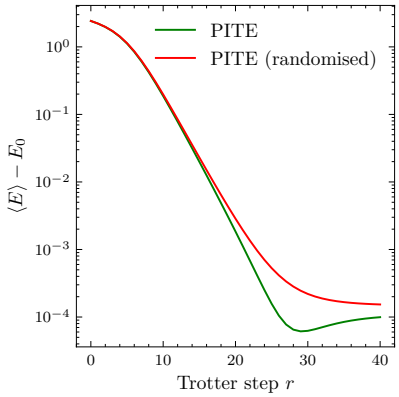
As expected, both Figures 3a and 4a show the PITE algorithm converging to the true E_0 indicated by the grey dashed line, where the rate of convergence was much faster for TIM than FHM. The red lines on both plots correspond to the exact diagonalisation. We classically apply the ITE operator by exponentiating $\hat{H}$ to obtain $e^{-\hat{H}\tau}$ as a matrix. As $e^{-\hat{H}\tau}$ is non-unitary, $e^{-\hat{H}\tau}|\psi\rangle$ is renormalised to unity before computing its expectation value. The PITE algorithm and exact diagonalisation for both subfigures are visually identical, where the red line overlaps the green line throughout the entire simulation. However, by plotting the difference in the axes below, indicated by the black line, we see that TIM has a much greater difference relative to FHM. Beyond any negligible numerical errors, this difference is mainly due to the error in Equation (19), of which the exact diagonalisation does not contain any Trotter error.

Figures 3b and 4b shows the difference between $\langle E\rangle$ and E_0. For FHM, the convergence is as expected, monotonically decreasing to 0. Curiously, this is not the case for TIM, where the green line shows a slight increase at $r = 29$. Since the JW map does not provide a unique ordering of Pauli strings, another PITE algorithm, indicated by the red line, was run with a different Pauli string ordering. This does not exhibit the same increase, suggesting that the non-commutativity affected the convergence profile. Regardless, both PITE runs reached a satisfactory precision of $\sim 10^{-4}$.

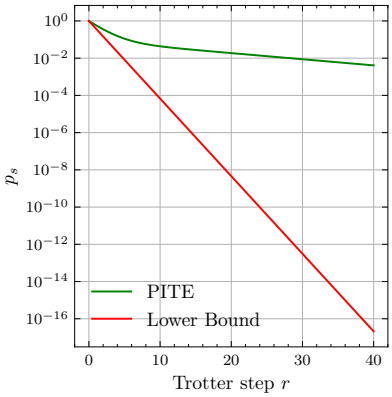
The probability of success p_s arising from the mid-circuit measurements for both models are shown in Figures 3c and 4c. The theoretical result for the lower bound from Equation (22) is shown in red. Although the worst-case p_s falls many orders of magnitude below, practically running the algorithms for the two systems yields p_s roughly between 10^{-1} and 10^{-2}, depending on the level of



(a) Energy estimate at each trotter step, where the green PITE line is not visible due to overlapping with the exact solution. The black line shows the difference between exact diagonalisation of $\hat{H}$ and PITE, mainly arising from Totterisation error.

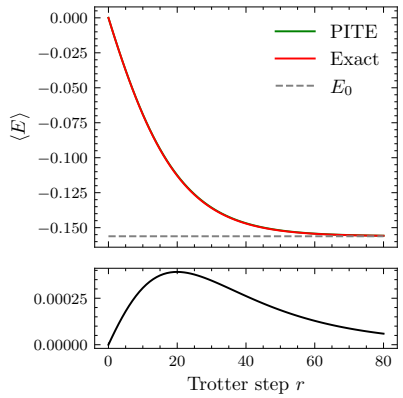


(b) Energy difference from ground state at each trotter step r. The red line is PITE with a different Pauli operator ordering, showing a different convergence trajectory to E_0.

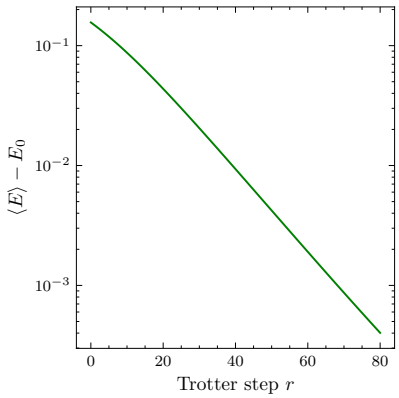


(c) Success probability p_s. Red line plots the lower bound of p_s from Equation (22).

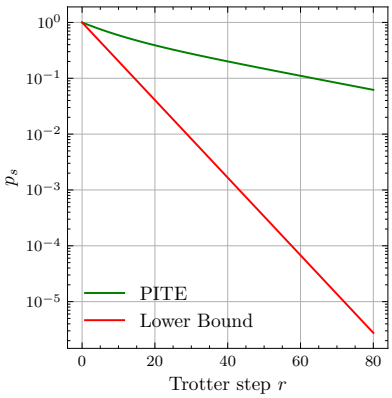
Figure 3: 4-site TIM Hamiltonian with $J = 0.5$, $h = 0.1$, $\Delta\tau = 0.1$



(a) Energy estimate, where the green PITE is not visible due to overlapping with the exact solution. The black line shows the difference between PITE and the exact solution, which is much smaller than TIM, indicating a smaller Trotter error for FHM.



(b) Energy difference from ground state at each trotter step r, monotonically decreasing as expected.



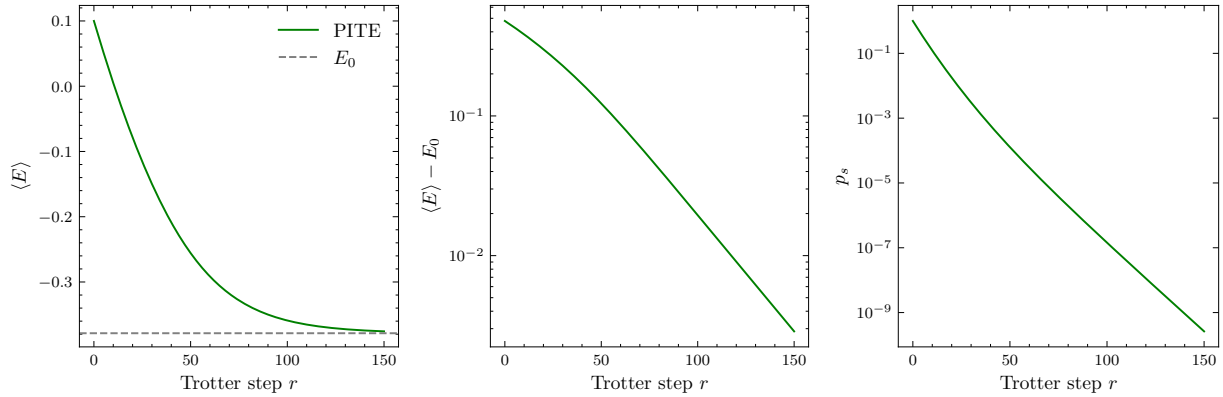
(c) Success probability p_s. Red line plots the lower bound of p_s from Equation (22).

Figure 4: 2-site FHM Hamiltonian with $U = 0.1$, $t = -0.1$, $\Delta\tau = 0.1$

precision required. This means every ten to a hundred shots would yield the true ground state, making PITE practical to run for these systems.

Although the 4-site FHM and 2-site TIM have shown promising results, the algorithm is much more difficult to extend to larger systems without sacrificing p_s. To demonstrate, Figure 5 shows a 4-site FHM, where the original FHM is extended by adding two more sites. We are still able to recover E_0 for this system, taking roughly double the trotter steps compared to the 2-site FHM to converge. However, this addition of 2 more sites drops p_s to lower than 10^{-9}. Assuming a hundred shots are needed to obtain a good estimate of $\langle E \rangle$, this increases the required number of shots to 10^{11}, rendering the problem intractable under current implementations of the algorithm.

Nevertheless, the algorithm can be improved in multiple directions, and the results presented in this paper should be considered as a proof of concept only. Fortunately, many directions for improvement for PITE are under active research, as their



(a) The PITE algorithm reaching the ground state in 150 Trotter steps.

(b) Similar to 2-site FHM, the energy difference falls monotonically.

(c) probability of success p_s. Notice how this probability drops to 10^{-9} after 150 trotter steps, making the algorithm unfeasible to run for this model.

Figure 5: 4-site FHM Hamiltonian with $U = 0.1$, $t = -0.1$, $\Delta\tau = 0.1$.

improvements are not specific to PITE itself but other classes of quantum algorithms. Firstly, a good guess for the initial state $|\psi(0)\rangle$ is required, identical to the reasoning given in Section 2.2. This ensures that a smaller τ is required to reach convergence. Hence, fewer trotter steps are necessary and the decrease in p_s is minimised. For example, rather than using an equal superposition of all basis states for TIM, a better motivated guess can be made by using a classical algorithm initially. Additionally, any techniques used to reduce circuit depth from Trotterisation is applicable here, where the depth of a circuit is the longest path of the circuit. Some approaches include reordering terms to maximise gate cancellations, and applying a higher order Trotter-Suzuki decomposition. The development of a quit-if-fail functionality on quantum hardware will also improve the time taken per shot, where the shot is discarded upon a failed post selection of an ancilla. Overall, the proposed PITE algorithm is shown to recover E_0 for all systems tested, and is easy to generalise to completely new Hamiltonians, provided that Trotterisation is possible, but the implementation on quantum hardware still await further improvements in other fields within quantum computing and quantum algorithms.

References

[1] J. Preskill, Quantum Computing in the NISQ era and beyond, Quantum 2 (2018) 79, publisher: Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften. doi:10.22331/q-2018-08-06-79.
URL https://quantum-journal.org/papers/q-2018-08-06-79/

[2] D. Deutsch, A. Barenco, A. Ekert, Universality in Quantum Computation, Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences 449 (1937) (1995) 669–677, arXiv:quant-ph/9505018. doi:10.1098/rspa.1995.0065.
URL http://arxiv.org/abs/quant-ph/9505018

[3] R. P. Feynman, Simulating physics with computers, International Journal of Theoretical Physics 21 (6) (1982) 467–488. doi:10.1007/BF02650179.
URL https://doi.org/10.1007/BF02650179

[4] J. Preskill, Quantum computing 40 years later (Jun. 2021).
URL https://arxiv.org/abs/2106.10522v2

[5] E. Wigner, On Unitary Representations of the Inhomogeneous Lorentz Group, Annals of Mathematics 40 (1) (1939) 149–204, publisher: Annals of Mathematics. doi:10.2307/1968551.
URL http://www.jstor.org/stable/1968551

[6] S. B. Bravyi, A. Y. Kitaev, Fermionic Quantum Computation, Annals of Physics 298 (1) (2002) 210–226. doi:10.1006/aphy.2002.6254.
URL https://www.sciencedirect.com/science/article/pii/S0003491602962548

[7] A. Y. Kitaev, Quantum measurements and the Abelian Stabilizer Problem, arXiv:quant-ph/9511026 (Nov. 1995). doi:10.48550/arXiv.quant-ph/9511026.
URL http://arxiv.org/abs/quant-ph/9511026

[8] T. E. O'Brien, B. Tarasinski, B. M. Terhal, Quantum phase estimation of multiple eigenvalues for small-scale (noisy) experiments, New Journal of Physics 21 (2) (2019) 023022, publisher: IOP Publishing. doi:10.1088/1367-2630/aafb8e.
URL https://dx.doi.org/10.1088/1367-2630/aafb8e

[9] H. Mohammadbagherpoor, Y.-H. Oh, A. Singh, X. Yu, A. J. Rindos, Experimental Challenges of Implementing Quantum Phase Estimation Algorithms on IBM Quantum Computer, arXiv:1903.07605 [quant-ph] (Mar. 2019). doi:10.48550/arXiv.1903.07605.
URL http://arxiv.org/abs/1903.07605

[10] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, J. L. O'Brien, A variational eigenvalue solver on a photonic quantum processor, Nature Communications 5 (1) (2014) 4213, publisher: Nature Publishing Group. doi:10.1038/ncomms5213.
URL https://www.nature.com/articles/ncomms5213

[11] E. R. Anschuetz, B. T. Kiani, Quantum variational algorithms are swamped with traps, Nature Communications 13 (1) (2022) 7760, number: 1 Publisher: Nature Publishing Group. doi:10.1038/s41467-022-35364-5.
URL https://www.nature.com/articles/s41467-022-35364-5

[12] Z. Holmes, K. Sharma, M. Cerezo, P. J. Coles, Connecting ansatz expressibility to gradient magnitudes and barren plateaus, PRX Quantum 3 (1) (2022) 010313, arXiv:2101.02138 [quant-ph, stat]. doi:10.1103/PRXQuantum.3.010313.
URL http://arxiv.org/abs/2101.02138

[13] P. Bader, S. Blanes, F. Casas, Solving the Schrödinger eigenvalue problem by the imaginary time propagation technique using splitting methods with complex coefficients, The Journal of Chemical Physics 139 (12) (2013) 124117. doi:10.1063/1.4821126.
URL https://doi.org/10.1063/1.4821126

[14] A. Tranter, P. J. Love, F. Mintert, N. Wiebe, P. V. Coveney, Ordering of Trotterization: Impact on Errors in Quantum Simulation of Electronic Structure, Entropy 21 (12) (2019) 1218, number: 12 Publisher: Multidisciplinary Digital Publishing Institute. doi:10.3390/e21121218.
URL https://www.mdpi.com/1099-4300/21/12/1218

[15] C. Leadbeater, N. Fitzpatrick, D. M. Ramo, A. J. W. Thom, Non-unitary Trotter circuits for imaginary time evolution, arXiv:2304.07917 [quant-ph] (Oct. 2023). doi:10.48550/arXiv.2304.07917.
URL http://arxiv.org/abs/2304.07917