Solving lattice gauge theories using the quantum Krylov algorithm and qubitization

Lewis W. Anderson[1,2], Martin Kiffner[1,3], Tom O'Leary[1], Jason Crain[4,1], and Dieter Jaksch[5,1]

[1]Clarendon Laboratory, University of Oxford, Parks Road, Oxford OX1 3PU, UK
[2]IBM Research Europe, Hursley, Winchester SO21 2JN, UK
[3]PlanQC GmbH, Lichtenbergstr. 8, 85748 Garching, Germany
[4]IBM Research Europe, The Hartree Centre STFC Laboratory, Sci-Tech Daresbury, Warrington WA4 4AD, UK
[5]Institut für Quantenphysik, Universität Hamburg, 22761 Hamburg, Germany

Computing vacuum states of lattice gauge theories (LGTs) containing fermionic degrees of freedom can present significant challenges for classical computation using Monte-Carlo methods. Quantum algorithms may offer a pathway towards more scalable computation of groundstate properties of LGTs. However, a comprehensive understanding of the quantum computational resources required for such a problem is thus far lacking. In this work, we investigate using the quantum subspace expansion (QSE) algorithm to compute the groundstate of the Schwinger model, an archetypal LGT describing quantum electrodynamics in one spatial dimension. We perform numerical simulations, including the effect of measurement noise, to extrapolate the resources required for the QSE algorithm to achieve a desired accuracy for a range of system sizes. Using this, we present a full analysis of the resources required to compute LGT vacuum states using a quantum algorithm using qubitization within a fault tolerant framework. We develop of a novel method for performing qubitization of a LGT Hamiltonian based on a "linear combination of unitaries" (LCU) approach. The cost of the corresponding block encoding operation scales as $\tilde{\mathcal{O}}(N)$ with system size N. Including the corresponding prefactors, our method reduces the gate cost by multiple orders of magnitude when compared to previous LCU methods for the QSE algorithm, which scales as $\tilde{\mathcal{O}}(N^2)$ when applied to the Schwinger model. While the qubit and single circuit T-gate cost resulting from our resource analysis is appealing to early fault-tolerant implementation, we find that the number of shots required to avoid numerical instability within the QSE procedure must be significantly reduced in order to improve the feasibility of the methodology we consider and discuss how this might be achieved.

Lewis W. Anderson: lewis.anderson@ibm.com

1 Introduction

Gauge theories are quantum field theories (QFTs) in which a gauge field imposes the fundamental symmetry on the Lagrangian and mediates interactions. They play foundational roles within various fields of theoretical physics. Within particle physics, gauge theories describe fundamental particles and their interaction through the exchange of force carriers. The gauge symmetry determines the nature of the theory; with quantum electrodynamics (QED) arising from an abelian U(1) symmetry and quantum chromodynamics (QCD) from a non-abelian SU(3) symmetry [1]. Gauge field theories also play important roles in cosmology [2] and condensed matter physics [3].

Many gauge theories are too complex to study analytically and, because many phenomena of interest arise non-perturbatively, efficient numerical methods have become increasingly important for studying gauge theories. Originally formulated in seminal work by Wilson [4], numerical study of such phenomena is now routinely performed using lattice gauge theories (LGTs) in which spacetime is discretized onto a lattice of points on which the values of matter and gauge fields are defined. Wilson's path integral formulation of LGTs saw extensive computational study using Monte-Carlo methods [5]. Applying Monte-Carlo algorithms to some Hamiltonians however, leads to the so called *sign problem* or *complex action problem* [6] in which the integrand contains negative terms or the action becomes complex (and so the Boltzmann weight cannot be directly interpreted as a probability density function). In dealing with these issues, for example by using a Taylor expansion [7], computing expectation values can involve summing many highly oscillatory terms of which many cancel due to alternating signs. This behaviour leads to statistical error that grows exponentially with system size [6]. Due to the minus sign in the fermionic commutation relation the sign problem poses a significant challenge to simulations involving interacting electrons, as well as lattice QCD with non-zero Baryon density.

One of the most widely studied gauge field theories

is the Schwinger model [8] which describes charged particles and their interactions with electromagnetic fields within QED in 1+1D[1]. Being the simplest gauge theory, the Schwinger model plays an important role as a benchmark for numerical methods for solving gauge theories as well as a pedagogical tool that can be used to study a range of complex phenomena that arise in more complex quantum field theories. Such phenomena include: chiral symmetry breaking, in which an effective fermionic mass arises from pairing of fermion and anti-fermions [9]; confinement, where fermions only exist alongside their corresponding anti-particle and separating such pairs requires increasing energy as they are drawn further apart (analogous to quark confinement in 3+1D QCD) [10].

Simulating the real-time dynamics of the Schwinger model as well as the ground state in the presence of a chemical potential imbalance between different fermion flavours both give rise to the sign problem. Consequently, the lattice Schwinger model has been used to benchmark novel computational approaches which do not suffer from the sign problem[2]. A promising classical approach is to use tensor network methods [11, 12] which use a Hamiltonian, rather than path-integral, representation and thus don't suffer from the sign problem. Matrix product states (MPS), a popular tensor network ansatz, are particularly well suited to gapped locally interacting systems in one spatial dimension and have seen significant application to studying lattice gauge theories [13, 14]. MPS have been used to study mass spectra in cases with zero and various values of non-zero bare fermion mass [15], CP-violating phase transitions [16, 17], chiral condensation [18], ground and excited state energies in the thermodynamic limit [19], ground state entanglement structure [20] and real time dynamics [21–25] in the Schwinger model and related models with abelian $\mathbb{Z}_n$ or SU(2) symmetries. Despite their successes in 1+1D however, extending tensor network methods to QFTs in two or more spatial dimensions poses significant challenges. MPS methods in more than one dimension often require bond-dimensions that grow rapidly with system size and tensor networks embedded in higher dimensions, such as projected entangled pairs states (PEPS) [14], cannot be contracted efficiently.

Quantum computing may offer a pathway towards simulation of lattice gauge theories when the sign problem arises and tensor networks methods fail [26, 27]. Multiple works have investigated algorithmic approaches to lattice gauge theories [28], and the Schwinger model in particular. These quantum computing methods typically aim to solve one of two problems. The first is to simulate time evolution of the model. Notably Shaw et al. [29] provide rigorous bounds for Trotterised time evolution of the model in both NISQ and fault tolerant settings which scale as $\tilde{\mathcal{O}}(N^{3/2}T^{3/2}\sqrt{x}\Lambda)$, where N is system size, T is simulation time, x is interaction strength and Λ is the local dimension of the gauge field (see Sec. 2.2 for further details of these model parameters)[3]. Recently, Kan and Nam reported an algorithm based on a second order product-formula which recovers the scaling reported by Shaw et at al. and can be applied to higher dimensions [32]. Other authors have focused on presenting experimental implementations using quantum hardware or classical emulation methods. Notable examples of such work have demonstrated real time dynamics of the Schwinger model using Trotterisation [33, 34]; the largest simulation of which was performed using 112 qubits on an IBM superconducting device [35]; and open quantum system dynamics [36]. The second problem that has been investigated using quantum computers is finding groundstate, or vacuum states, of LGTs.

Research studying the groundstate computation for lattice gauge theories, and in particular the Schwinger model, on quantum computers has primarily focused on the use of VQE methods [33, 37–46], including as part of initial state preparation for studying dynamics [35]. The scalability of VQE is significantly impacted by barren plateaus in which gradients of the cost functions landscape become exponentially small with system size [47–54]. Moreover, the highly stochastic and time-intensive optimisation procedure of VQE means that systematic studies of the cost and scaling of the optimisation routine are challenging using both simulated and real quantum processors. Other quantum algorithms for computing ground states of lattice field theories include methods based on building the state up site-by-site [55] and using a quantum circuit for an MPS ansatz [56], which have thus far only been applied to purely Fermionic (i.e., that do no contains explicit gauge fields) Hamiltonians; as well as adiabatic methods [57], which require a iteratively performing a costly phase estimation step to ensure the state does not leave the adiabtic manifold.

In order to avoid these issues, we investigate the use of the recently proposed quantum subspace expansion (QSE) [58–62] algorithm and apply it to solving the lattice Schwinger model. In contrast to VQE, since the QSE algorithm does not involve a stochastic optimisation procedure it avoids the issue of barren plateaus (although obstacles to the scaling of the algorithm may exist) and analysis of the performance and scaling of the algorithm with numerical methods may

[1] one space and one time dimension

[2] this is not to say that such computational approaches do not lead to their own computational bottlenecks

[3] They note that this scaling with Λ is more favourable than other methods based on qubitization [30] or random compilation for Hamiltonian simulation (QDRIFT) [31].

be more fruitful. Recent work has shown that qubitization [30] can be used to implement QSE [63], as well as the related a algorithm for computing Green's functions using Lanczos recursion [64], on fault tolerant processors. Moreover, the theory underpinning QSE in such a setting is closely related that of the quantum eigenvalue transformation for unitary matrices (QETU) algorithm [65], which has been studied for U(1) gauge theories in 2+1D via both Trotterisation and qubitization [66].

1.1 Summary of results

In this article, we investigate the use of QSE for finding groundstates of a LGT. We consider all-to-all qubit connectivity and fault tolerant operation throughout[4]. We lay out a framework in which methods based on qubitization and quantum singular value transformation can be used to generate the necessary expectation values required for the lattice Schwinger model. We provide a full description and resource estimation for the qubitization procedure which is based on the linear combination of unitaries (LCU) method. As part of this, we propose a novel way of applying the LCU procedure to lattice models in which we exploit translational symmetry of the model. Compared to existing proposals for performing QSE with qubitization [63], which can be applied to the Schwinger model with a gate cost of $\tilde{\mathcal{O}}(N^2)$, the gate cost of our methods is $\tilde{\mathcal{O}}(N)$ and leads to a reduction in gate cost by multiple orders of magnitude for system sizes of hundreds of lattice sites.

To estimate the total resource cost of the QSE algorithm, we perform extensive numerical simulation of the algorithm, including the effect of measurement noise for systems of up to $N = 26$ lattice sites. We consider the effect of shot noise on expectation values and use a partitioned quantum subspace [67] to mitigate issues due to ill-conditioning within the QSE algorithm resulting from this noise. Extrapolating from the numerical results we estimate the qubit and T-gate cost of using QSE to solve the lattice Schwinger model for lattice sizes comparable to state of the art classical results. For systems of hundreds of lattice sites, we find that the number of T-gates required for the qubitization procedure are on the order of 10^6 (more precise values are given later in the paper). While such single circuit gate costs may be feasible for future fault tolerant devices, the number of shots required to control measurement noise leads to total number of circuit evaluations (and therefore runtime) beyond what we expect to be feasible. This is despite the use of partitioned quantum subspace approaches [67], which reduces the cost significantly compared to previous thresholded approaches [68]. We discuss how further algorithmic improvements may offer a pathway towards reducing the measurement cost.

Although we consider the single flavour Schwinger model, for which the groundstate problem does not lead to the sign problem, the analysis and numerics we present here can be extended to more complex models for which Monte-Carlo methods fail. The work we present here serves as a first step towards understanding and using a new quantum algorithm to study LGTs.

The remainder of this article is structured as follows. In Sec. 2 we describe the theoretical background. This includes Sec. 2.1 describing the QSE algorithm and Sec. 2.2 describing the Hamiltonian formulation of the lattice Schwinger model. In Sec. 3 we introduce the concept of quantum singular values transforms (QSVT) and qubitization. This includes Sec. 3.1 defining a block encoding and QSVT and then Sec. 3.2 in which we show how this can be used to generate a Krylov basis for QSE. In Sec. 4 we present our numerical simulations. This includes a description of how we model measurement noise in Sec. 4.1, how we use partitioned quantum subspace expansion to deal with this noise in Sec. 4.2, and a set of results in Sec. 4.3 showing the number of measurements and basis size needed to achieve a desired energy error. In Sec. 5 we calculate the resources required to run the algorithm, the mathematical details of which are deferred to Appendix C. Finally, in Sec. 6 we discuss our results.

2 Theoretical background

2.1 Quantum subspace expansion

For a Hamiltonian H, the QSE algorithm aims to calculate an approximate groundstate within a subspace spanned by a D dimensional non-orthogonal basis $\{|\psi_0\rangle, |\psi_1\rangle, \ldots, |\psi_{D-1}\rangle\}$ which can be generated by a quantum processor. Within this subspace, a trial state has the form $|\psi(\vec{c})\rangle = \sum_{i=0}^{D-1} c_i |\psi_i\rangle$. Minimising the functional $\langle\psi(\vec{c})| H |\psi(\vec{c})\rangle$ to find optimal coefficients $\vec{c}$ is equivalent to solving the generalised eigenvalue problem

$$\mathbf{H}\vec{c} = E\mathbf{S}\vec{c}, \tag{1}$$

for minimum eigenvalue E, where matrices $\mathbf{H}$ and $\mathbf{S}$ are given by

$$\mathbf{H}_{i,j} = \langle\psi_i|H|\psi_j\rangle, \quad \mathbf{S}_{i,j} = \langle\psi_i|\psi_j\rangle. \tag{2}$$

The power of the QSE algorithm relies on making a good choice of basis and being able to efficiently measure expectation values for $\mathbf{H}_{i,j}$ and $\mathbf{S}_{i,j}$. The

[4] later, we compare the required two-qubit gate depth to decoherence times for a range of quantum processors to confirm the need for fault tolerance in systems of hundreds of lattice sites.

basis $\{|\psi_k\rangle\,|k=0,1,\ldots,(D-1)\}$ is typically generated by applying (often repeatedly) operators to an input state $|\psi_0\rangle$, which is chosen to be close to the true groundstate. Many types of operators have been used to generate this basis including real and imaginary time evolution [69–74], fermionic excitation operators [75], Pauli operators [76] and using parameterised quantum circuits in variational manner [77]. Depending on the basis used, the expectation values can be calculated in different ways such as a SWAP test [78, 79] or by measuring constituent Pauli expectation values and recombining as done in VQE.

A particularly powerful basis, called a Krylov basis, is formed by repeated application of the Hamiltonian to the reference state: $\{|\psi_k\rangle = H^k\,|\psi_0\rangle\,|k = 0,1,\ldots,D-1\}$. This forms the basis for the well studied classical Lanczos algorithm for computing groundstates [80–82]. In the infinite D limit, the Krylov basis includes infinite imaginary time evolution and, assuming non-zero overlap between the initial and ground states and infinite precision arithmetic, will lead to the exact groundstate. In the case of finite sized bases for a given problem instance, the Lanczos algorithm converges exponentially with basis size, and so very good solutions can often be found using reasonable basis sizes [83, 84]. QSE using a Krylov basis is often refered to as the *quantum Krylov algorithm*. Throughout the remainder of this article we will only consider QSE using a Krylov basis and refer to it as 'QSE'. A schematic of this algorithm is shown in Fig. 1 (a).

For quantum systems in a Hilbert space, the cost of exactly evaluating and storing the Hamiltonian acting on the reference states grows exponentially with system size, making the use of a quantum processor to generate the basis within a QSE procedure very appealing. Moreover, when using a Krylov basis, $\mathbf{H}$ and $\mathbf{S}$ are Hankel matrices, specified by $\mathcal{O}(D)$ unique values given by $\mathbf{H}_{i,j} = \langle\psi_0|H^{i+j+1}|\psi_0\rangle$, $\mathbf{S}_{i,j} = \langle\psi_0|H^{i+j}|\psi_0\rangle$, rather than the $\mathcal{O}(D^2)$ unique values required in general when using operators to generate the basis that do not commute with the Hamiltonian. Of course the Hamiltonian is not a unitary operator in general, and so methods to compute expectation values must use some additional classical computation to generate the Krylov basis. Strategies to compute the non-unitary expectation values in $\mathbf{H},\mathbf{S}$ include classical post processing to combine expectation values from time evolutions [85] or methods based on qubitization [63]. We will use the latter in this work.

2.2 Hamiltonian formulation of the single flavour lattice Schwinger model

When working in a Hamiltonian formulation, the Schwinger model is often represented on a lattice by using a "staggered fermion" representation [86, 87]. As we will show in the following, the staggered fermion representation uses odd lattice sites to encode fermions and even sites to encode anti-fermions. The single flavour model is constructed as follows. We define a single component fermion field at lattice site labelled by integer n with creation and annihilation operators $\phi^\dagger(n)$ and $\phi(n)$ respectively. These obey the usual commutation relations

$$\{\phi^\dagger(m),\phi(n)\} = \delta_{m,n},\ \{\phi(m),\phi(n)\} = 0. \tag{3}$$

The gauge field is defined on links connecting sites $(n,n+1)$ with operator

$$U(n,n+1) := e^{i\theta(n)}, \tag{4}$$

which mediates interactions between neighbouring fermionic sites. Within the "compact" formalism, the onsite gauge-field becomes an angular variable on the lattice with conjugate spin variable $L(n)$ obeying

$$[\theta(m),L(n)] = i\delta_{m,n} \tag{5}$$

such that $L(n)$ has eigenvalues $L(n) = 0,\pm1,\pm2,\ldots$ and $e^{i\theta(n)}$ and $e^{-i\theta(n)}$ act as raising and lowering operators respectively. That is $e^{i\theta(n)}$ $(e^{-i\theta(n)})$ has the effect of increasing (decreasing) the value of $L(n)$ linking lattice sites $(n,n+1)$ by one, while having no effect on $L(m\neq n)$ for other lattice sites.

The dimensionless Hamiltonian for the single flavour Schwinger model is given by

$$H = H_0 + xV \tag{6}$$

with

$$H_0 = \sum_{n=1}^{N-1} L^2(n) + \mu\sum_{n=1}^{N}(-1)^n\phi^\dagger(n)\phi(n) \tag{7}$$

$$V = -i\sum_{n=1}^{N-1}\left(\phi^\dagger(n)e^{i\theta(n)}\phi(n+1) - \text{h.c}\right), \tag{8}$$

where μ and x correspond to re-scaled mass and coupling parameters [88]. As we will see shortly, the alternating sign on the mass term allows us to incorporate both matter and anti-matter particles on a bipartite lattice.

Physical states must obey Gauss's law which ensures that the electric flux into and out of a lattice site is balanced by the onsite charge. On the lattice, Gauss's law is given by [88]

$$L(n) - L(n-1) = \phi^\dagger(n)\phi(n) - \frac{1}{2}\left(1-(-1)^n\right) \quad \forall\ 0<n\leq N. \tag{9}$$

This means that excitations $\phi^\dagger(n)\phi(n) = 1$ to an even n site generates -1 unit of flux between the gauge fields into and out of the lattice site. Thus, on an even lattice site $\phi^\dagger(n)\phi(n) = 1$ is interpreted as the existence of a fermion with negative unit charge, while

$\phi^\dagger(n)\phi(n) = 0$ is an absence. Conversely, for an odd n lattice site, $\phi^\dagger(n)\phi(n) = 0$ generates $+1$ units of flux. Thus on an odd lattice site $\phi^\dagger(n)\phi(n) = 0$ can be interpreted as the existence of an anti-fermion with positive unit charge and $\phi^\dagger(n)\phi(n) = 1$ as the absence [89]. We can see from the mass term in (7) that this interpretation gives the expected mass addition μ for both fermion and anti-fermion.

It is common to impose zero field to the boundaries, namely $L(0) = L(N) = 0$. In the $x \to 0$ (no coupling) limit, the ground state can be found by inspection to be

$$L(n) = 0 \; \forall n, \quad \phi^\dagger(n)\phi(n) = \begin{cases} 0 \text{ even } n \\ 1 \text{ odd } n. \end{cases} \tag{10}$$

As one might expect for a non-interacting theory, this vacuum state corresponds to a complete absence of particles and anti-particles as well as zero gauge field.

Remark 1. *Assuming open boundaries $L(0) = L(N) = 0$, since the occupation of neighbouring gauge fields can vary by at most ± 1 due to the occupancy of a single fermionic site, the maximum value of $|L(n)|$ is $N/4$ for states obeying Gauss's law.*

This is particularly useful when studying such models with quantum computers or tensor network methods in which it is necessary to truncate the dimensionality of the gauge fields to map them onto finite dimensional systems. When using qubit or spin system with a binary encoding, the number of qubits needed to encode the gauge field subspace is $m = \lceil \log_2(N/2+1) \rceil$, which gives maximum and minimum gauge field occupancies of $\pm\Lambda$ with $\Lambda \geq N/4$. We will use such a value for m throughout this work. By using an initial state with zero field, and generating the Krylov basis with repeated action of the Hamiltonian which conserves Gauss's law, we can be sure that no states generated require field values beyond the corresponding maximum and minimum field values $\pm N/4$ and there is no loss of accuracy due to using a truncated finite dimensional gauge field space. It is likely that further truncation can be applied with minimal impact on accuracy of solution. It has been shown [90] that for time evolution and computation of spectrally isolated eigenstates it suffices to approximate the gauge field with a truncated Fock space of dimension Λ that scales polylogarithmically with inverse error. The ability to significantly truncate the dimensionality of the gauge fields has been noted in numerical tests [19, 91, 92], and is standard practice in tensor network methods for LGTs.

The fermionic operators within the Hamiltonian can be mapped to spin operators using the Jordan-Wigner transformation

$$\begin{aligned} \phi(n) &= \prod_{j<n} \left[i\sigma_3(l) \right] \sigma^-(n), \\ \phi^\dagger(n) &= \prod_{j<n} \left[-i\sigma_3(l) \right] \sigma^+(n), \end{aligned} \tag{11}$$

where $\sigma_1, \sigma_2, \sigma_3$ are the spin-half Pauli operators and $\sigma^\pm = (\sigma_1 \pm i\sigma_2)/2$. Under this transformation the Hamiltonian becomes

$$\begin{aligned} H_0 &= \sum_{n=1}^{N-1} L^2(n) + \sum_{n=1}^{N} (-1)^n \left(\frac{\mu}{2} + \frac{\mu}{2}\sigma_3(n) \right), \\ V &= \sum_{n=1}^{N-1} \left(\sigma^+(n) e^{i\theta(n)} \sigma^-(n+1) + \text{h.c} \right) \end{aligned} \tag{12}$$

and Gauss' law can be reformulated as

$$\begin{aligned} L(n) - L(n-1) = \frac{1}{2} \left(\sigma_3(n) + (-1)^n \right) \\ \forall \; 0 < n \leq N. \end{aligned} \tag{13}$$

If we assume free boundaries $L(0) = L(N) = 0$ we can eliminate the gauge field by using Gauss' law combined with the gauge transformation

$$\sigma^-(n) \to \prod_{j<n} \left[e^{-i\theta(j)} \right] \sigma^-(n), \tag{14}$$

in which case the Hamiltonian becomes

$$\begin{aligned} H_0 = &\sum_{n=1}^{N} (-1)^n \left(\frac{\mu}{2} + \frac{\mu}{2}\sigma_3(n) \right) \\ &+ \sum_{n=1}^{N-1} \left[\frac{1}{2} \sum_{m=1}^{n} \left(\sigma_3(m) + (-1)^m \right) \right]^2, \\ V = &\sum_{n=1}^{N-1} \left(\sigma^+(n)\sigma^-(n+1) + \text{h.c} \right). \end{aligned} \tag{15}$$

Within this spin-only representation, the $x \to 0$ ground state corresponds to anti-ferromagnetic state with $\sigma_3(n) = -(-1)^n \; \forall \; n$ [34].

Fig. 2 shows a schematic diagram of a state written in the original staggered fermion representation (with Gauss's law) along with the corresponding state after undergoing Jordan Wigner and gauge transformations to remove the gauge field. Similar transformations can be applied for arbitrary open boundary conditions choosing $L(0)$ and $L(N)$ to be any fixed value, this corresponds to a topological θ-term [92, 93].

3 Quantum subspace expansion via QSVT

3.1 Block encoding and QSVT

Qubitization and quantum singular value transformations (QSVT) have emerged as powerful methods for

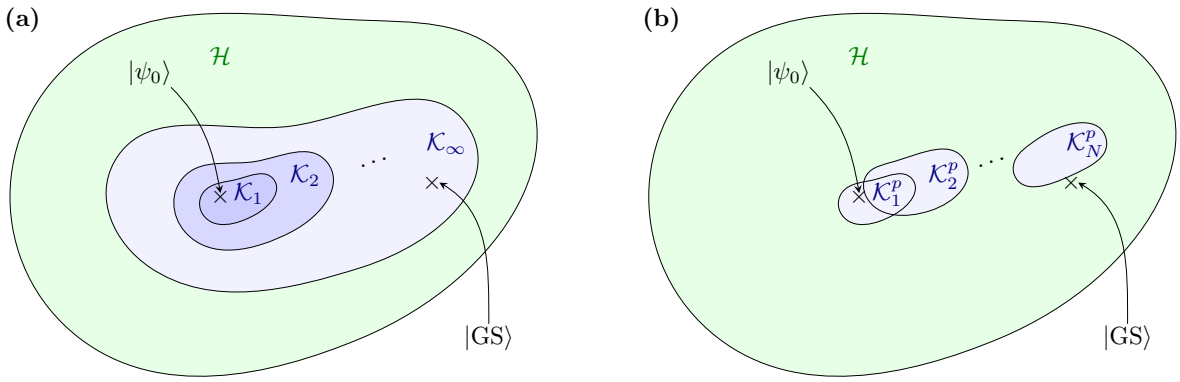


Figure 1: Schematic of the quantum subspace expansion algorithms considered in this work. **(a)** QSE algorithm using a Krylov basis. Starting from initial state $|\psi_0\rangle$ within the total Hilbert space $\mathcal{H}$, Krylov bases $\mathcal{K}_D$ with increasing basis sizes D allow for access to more of the Hilbert space. For $D = \infty$, the corresponding Krylov space $\mathcal{K}_\infty$ contains infinite imaginary time evolution and therefore the true ground state $|\mathrm{GS}\rangle$. **(b)** PQSE algorithm using a Krylov basis. In contrast to QSE, PQSE builds up a series of Krylov bases $\mathcal{K}_1^p, \mathcal{K}_2^p \ldots \mathcal{K}_n^p$ in a non-Markovian way. A selection criteria based on the variance of corresponding state (see algorithm description in Ref. [67]) at each step to create a Krylov basis $\mathcal{K}_i^p$ with automatically chosen dimension using a reference state corresponding to the optimal state in the preceding Krylov subspace $\mathcal{K}_{i-1}^p$. In practice, this improves the numerical stability of the generalised eigenvalue problem being solved and aims to find a reasonable approximation to the ground state.

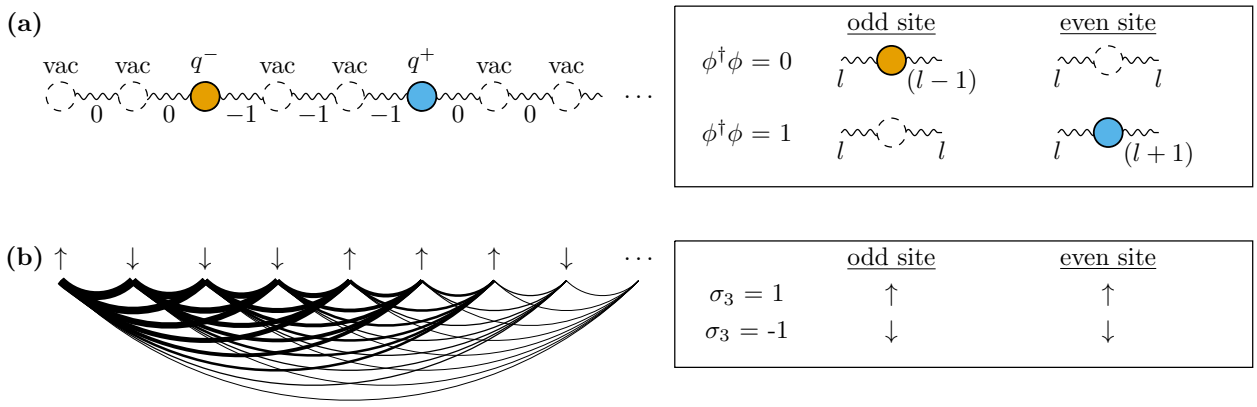


Figure 2: **(a)** State satisfying Gauss's law for Lattice Schwinger model in staggered fermion formalism. Fermionic lattice sites are indicated by (occupied or unoccupied circles) with connecting lines corresponding to gauge field. Integers below gauge field lines correspond to value of gauge field L. The box to right indicates how the value of the fermionic number operator is interpreted as existence of quark or anti-fermion, along with the gauge field flux arising from charge balance imposed by Gauss's law. **(b)** The same state after applying the Jordan-Wigner transformation and gauge-transformation to arrive at a spin model. Curves connecting spins indicate non-local interactions arising out of gauge transformation, with line thickness of each one corresponding to strength of corresponding term in Hamiltonian. The box to right indicates how existence of a quark or anti-quark corresponds to up or down spin states depending on the parity of the lattice site.

implementing arbitrary functions of non-unitary operators on quantum computers. It has been shown how many paradigmatic quantum algorithms can be implemented using such methods which in some cases offers significant improvements over resources requirements and scalings of the original proposals [94]. QSVT has become one of the most popular methods for implementing quantum phase estimation and time evolution algorithms. Although such methods are expected to require some level of fault tolerance, the latter application has already been demonstrated on a trapped-ion hardware platform [95].

In this work, we use QSVT via qubitization to generate the polynomial expectation values required to generate the Krylov basis within QSE. In the following, we describe the key idea behind qubitization and block encoding and then quantum singular value transforms.

Definition 2 (Block encoding, modified from [94])**.** *Let H be a Hamiltonian acting on Hilbert space $\mathcal{H}_s$ with spectral norm $||H|| \leq 1$. A block-encoding of H is three unitary operators $(U, G, \tilde{G})$, where U acts on $\mathcal{H}_a \otimes \mathcal{H}_s$, for some auxiliary Hilbert space $\mathcal{H}_a$, and $|G\rangle := G|0\rangle \in \mathcal{H}_a$ and $|\tilde{G}\rangle := \tilde{G}|0\rangle \in \mathcal{H}_a$, are states such that*

$$\left(\langle \tilde{G}| \otimes \mathbb{1}_s\right) U \left(|G\rangle \otimes \mathbb{1}_s\right) = H. \tag{16}$$

From this definition, we see that block encoding allows us to encode the operation of H within the subspace labelled by auxiliary states $|G\rangle$, $|\tilde{G}\rangle$ [5]. It is often the case, such as in the original proposal [30], that the auxiliary states are taken to be equal, i.e., $|G\rangle = |\tilde{G}\rangle$. This is also the case in previous works using qubitization for QSE. We will see later that we can reduce the cost of QSE via QSVT by using an asymmetric encoding $|G\rangle \neq |\tilde{G}\rangle$.

Provided we can find unitary operators U, G and $\tilde{G}$ (if not equal to G), qubitization allows us to measure operators and, as we will see later, functions of non-unitary operators using quantum circuits.

The linear combination of unitaries (LCU) method [30] is a commonly used method for block encoding local Hamiltonians that we will use in this work. LCU works as follows: We express an n qubit Hamiltonian as a linear combination of T Pauli operators:

$$H = \sum_{i=0}^{T-1} \alpha_i \hat{P}_i, \tag{17}$$

where α_i are real coefficients and $\hat{P}_i \in \{\mathbb{1}, X, Y, Z\}^{\otimes n}$ is an n Pauli operator. In most situations we can assume that $\sum_{i=0}^{T-1} |\alpha_i| = 1$, such that $||H|| < 1$, by renormalising the Hamiltonian. The block encoding unitary is

$$U = \sum_{i=1}^{T-1} |i\rangle\langle i| \otimes \hat{P}_i, \tag{18}$$

which corresponds to the application of each $\hat{P}_i$ controlled on the state of the auxiliary register being in $|i\rangle$.

As for the corresponding block encoding states, we divert from previous implementations of QSE with qubitization [63] by using an asymmetric block encoding $G \neq \tilde{G}$. For our work we will use

$$|G\rangle = \sum_{i=0}^{T-1} \sqrt{|\alpha_i|}\,|i\rangle \tag{19}$$

and

$$|\tilde{G}\rangle = \sum_{i=0}^{T-1} \mathrm{sgn}(\alpha_i)\sqrt{|\alpha_i|}\,|i\rangle\,. \tag{20}$$

It is straightforward to confirm that these definitions of U, $|G\rangle$ and $|\tilde{G}\rangle$ satisfy Definition 2 for a block encoding. We will see later that since G and $\tilde{G}$ differ only by the possible signs of α_i that they can be prepared using almost identical circuits.

The challenge in performing a block encoding in this way is to choose a labelling procedure $\{|i\rangle\}$ that makes efficient use of auxiliary qubits leading to an efficient method of generating the block encoded state. We use a labelling procedure in which $|i\rangle$ are computational basis states corresponding to a pair of n length binary numbers $(\vec{x}, \vec{z})_i$ that give a binary encoding of n qubit Pauli operator $\hat{P}_i$. If $\vec{x}$ ($\vec{z}$) contains a one in bit j, then $\hat{P}_i$ includes an X (Z) operator on qubit j. Constituent Y operators arise when both vectors contain a one in the same position as $ZX = iY$. The operation of U in (18) can then be implemented with the following three sets of gates: (1) A set of CX gates, each controlled on a qubit within the $\vec{x}$ register and targeting the corresponding qubit within the system register. (2) A set of CZ gates doing the same thing, except each controlled on one of the qubits within the $\vec{z}$ register. (3) A controlled S (i phase gate) acting between a qubit in the $\vec{x}$ register and the qubit in the same position within the $\vec{z}$ register.

To prepare $|G\rangle$ we must prepare a real superposition with amplitudes $\{\sqrt{|\alpha_i|}\}$ over computational basis states $|(\vec{x}, \vec{z})_i\rangle$. To do so, a series of multi-controlled partial CNOT and SWAP gates must be applied with control and target qubits as well as rotation angles chosen in such a way as to rotate into different states with increasing Hamming weights with the correct amplitudes. The process and order by which this is done is quite complex, a full description is given in Appendix B.2. As described by Kirby [63] this method

[5] In the case where $G = \tilde{G}$ it is this state that gives rise to the name qubitization. $\mathcal{H}_a$ acts as a qubit subspace spanned by $|G\rangle$ and its orthogonal complement. Functions of the block-encoded operator are implemented by repeated rotations around the Block sphere of this effective qubit.

requires us to apply a different multi-controlled rotation gate for each Pauli operator within the Hamiltonian. We present in Sec. 5, a method for reducing the number of these gates required by exploiting translational symmetry within the Hamiltonian such that the most costly of these gates need only be applied to single gauge link within the lattice and then effectively copied over to the remaining lattice sites. This reduces the cost from $\tilde{\mathcal{O}}(N^2)$ (if applying the method given by Kirby [63] to our Hamiltonian) to $\tilde{\mathcal{O}}(N)$. The other block encoding state $|\tilde{G}\rangle$ can be generated using the same set of rotation gates except flipping the phase of the some of the rotation angles in order to include an additional minus sign phase where needed.

In previous work [63] in which $\tilde{G} = G$, to include the negative signs on values of α_i a potentially large number of multicontrolled Z gates must be applied within U. In using our asymmetric encoding to include the required signs within $|\tilde{G}\rangle$ we remove this cost which, depending on the number of negative α_i, could otherwise dominate the cost of the algorithm.

Given access to the block encoding, functions of H can be implemented through the quantum singular value transforms (QSVT) [96]. For our purpose of computing a Krylov basis, the following theorem to generate polynomials of H will suffice.

Theorem 3 (Polynomials through QSVT, modified from [94, 96])**.** *Given Hamiltonian H with eigenvalues on the range $[-1, 1]$ and block encoding $(U, G, \tilde{G})$, we define the projectors*

$$\Pi := |G\rangle\langle G|, \quad \tilde{\Pi} := |\tilde{G}\rangle\langle\tilde{G}| \tag{21}$$

onto the subspaces spanned by $|G\rangle$ and $|\tilde{G}\rangle$ and corresponding rotations by angle $\varphi \in \mathbb{R}$,

$$\Pi_\varphi := e^{2i\varphi\Pi}, \quad \tilde{\Pi}_\varphi := e^{2i\varphi\tilde{\Pi}}. \tag{22}$$

Then, for any polynomial $f : [-1, 1] \to [-1, 1]$ of degree k,

- *if k is odd, there exists angles $\vec{\varphi} \in \mathbb{R}^k$ defining the operator*

$$U_{\vec{\varphi}} := \tilde{\Pi}_{\varphi_1} U \prod_{j=1}^{\frac{k-1}{2}} \left[\Pi_{\varphi_{2j}} U^\dagger \tilde{\Pi}_{\varphi_{2j+1}} U\right] \tag{23}$$

such that

$$\langle\tilde{G}| U_{\vec{\varphi}} |G\rangle = f(H), \tag{24}$$

- *if k is even, there exists angles $\vec{\varphi} \in \mathbb{R}^k$ defining the operator*

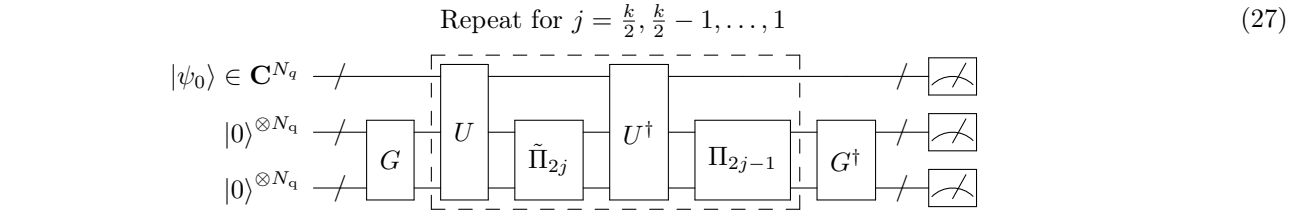
$$U_{\vec{\varphi}} := \prod_{j=1}^{\frac{k}{2}} \left[\Pi_{\varphi_{2j-1}} U^\dagger \tilde{\Pi}_{\varphi_{2j}} U\right], \tag{25}$$

such that

$$\langle G| U_{\vec{\varphi}} |G\rangle = f(H). \tag{26}$$

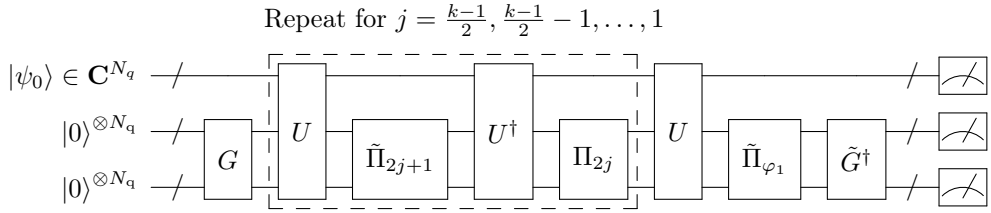
Moreover, there exists an efficient classical algorithm for computing $\vec{\varphi}$.

The required quantum circuits therefore have the form



(27)

for k even and



(28)

for k odd, where N_q are the number of qubits required to represent the system, measurements at the end of the circuit represent measuring all $3N_q$ qubits in the computational basis.

Corollary 4. *There exists $\vec{\varphi} \in \mathbb{R}^k$ such that if k is odd*

$$\left(\langle\tilde{G}| \otimes \langle\psi_0|\right) U_{\vec{\varphi}} \left(|G\rangle \otimes |\psi_0\rangle\right) = \langle\psi_0| H^k |\psi_0\rangle \tag{29}$$

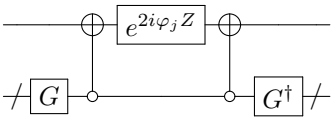
and if k is even

$$\left(\langle G| \otimes \langle\psi_0|\right) U_{\vec{\varphi}} \left(|G\rangle \otimes |\psi_0\rangle\right) = \langle\psi_0| H^k |\psi_0\rangle . \tag{30}$$

We can use this to measure the expectation values

required for **H** and **S** within a Krylov basis. The rotation gates required can be implemented as follows.

Remark 5. *The rotation about $|G\rangle$ required for the QSVT procedure in Theorem 3 by angle φ_j can be implemented with the following circuit*

$\Pi_{\varphi_j} =$ 

(31)

A similar construction can be used for $\tilde{\Pi}_{\varphi_j}$ if $\tilde{G} \neq G$.

3.2 Generating the Krylov basis using QSVT

Recalling the definition of **H** and **S** for a Krylov basis given in (2), the matrix elements for a D dimensional Krylov space $\langle\psi_0| H^k |\psi_0\rangle$ for $k = 0, 1, \ldots (2D-1)$ must be estimated on a quantum computer. Given access to G, $\tilde{G}$ and U this can be done according to Corollary 4 as follows.

1. Generate a reference state $|\psi_0\rangle$ in the system register and apply G to the auxiliary register to prepare $|G\rangle \otimes |\psi_0\rangle$.
2. For each φ_j in $\vec{\varphi}$, apply the block encoding operator U to both registers as well as the rotation operator Π_{φ_j} or $\tilde{\Pi}_{\varphi_j}$ (depending on if j is odd or even) to the auxiliary register.
3. Apply the reverse operation for initial state preparation to the system registers, as well as $G^\dagger$ or $\tilde{G}^\dagger$ depending on the parity of k to the auxiliary register.
4. Measure all system and auxiliary qubits in the computational basis and count the result if the all zero state is measured.
5. Return to step 1. and repeat until enough measurements are performed to give desired precision. The expectation value of $\langle\psi_0| H^k |\psi_0\rangle$ will be the fraction of times that the zero state is measured.

We obtain accurate gate and qubit counts for the qubitization and QSVT procedures by applying the LCU procedure along with the method for generating $|G\rangle$ described by Kirby et al. (Appendix C in Ref. [63]) to the single flavour Schwinger model to give accurate gate and qubit counts for U and $|G\rangle$ required for the qubitization and QSVT procedure. We exploit translational symmetry within our Hamiltonian to improve the gate scaling of G by a factor of system size N. By combining this method with the use of an assymetrical block encoding such that $\tilde{G} \neq G$, we are able to reduce the scaling of U by a factor of N. The cost of $\tilde{G}$ is the same as G. Since the procedures for generating $(U, G, \tilde{G})$ along with analysis resource requirements are quite involved, we defer detailed descriptions and calculation to Appendix C.

In Appendix C we show that the total number of qubits required to represent the system and implement the block encoding procedure is $\mathcal{O}(\log(N)N)$. Furthermore, the number of gate operations required scale as $\mathcal{O}(N)$ gates for U and $\mathcal{O}(\log(N)N)$ for G and $\tilde{G}$. Both of these gate costs are reduced from $\mathcal{O}(\log(N)N^2)$ by using our method to exploit translational symmetry. Exact gate costs and prefactors are given in Theorems 32 and 27 in Appendix C. The $\mathcal{O}(\log(N)N^2)$ gate cost resulting from the LCU procedure described by Kirby at al. [63], including prefactors, is shown in Theorem 34 of Appendix C.6.2. By exploiting translational symmetry, we reduce the gate cost of the algorithm by multiple orders of magnitude for relevant system sizes, as compared to the qubitization based on the work of Kirby [63]. We note that the work by Kirby et al. considers two methods of implementing the LCU procedure, we compare to the second LCU method therein which requires fewer gates at the expense of more circuit operations than the first method they describe. Further discussion of the gate costs of the QSVT procedure, as well as the total algorithmic costs, is given in Sec. 5.

4 Numerical experiments

To analyse the performance of the QSE algorithm for the Schwinger Hamiltonian, we perform classical simulations based on sparse matrix multiplication. We consider fault tolerant operation and therefore neglect the effect of device noise. To begin with, we will also neglect the effect of finite measurement noise on the expectation values of $\langle H\rangle$ before introducing it later on. For the quantum computing algorithm, we propose working with the single flavour 1+1D Schwinger Hamiltonian of (12) in which we explicitly include the gauge fields. This is in contrast to using the gauge-transformed model of (15) in which the transformed interactions contains long range terms. Although the gauge transformed representation will require fewer qubits as avoid explicitly storing the value of gauge fields, we expect the gate cost to scale more poorly. Furthermore, for more general models it is necessary to include the gauge fields as it is not possible to remove gauge fields through such a transformation for models in two or more spatial dimensions.

When performing sparse matrix multiplication with the Hamiltonian, the cost is not limited by the locality of interactions, but rather the memory usage required to store states. Thus, to calculate expectation values in $\mathbf{H}, \mathbf{S}$ for our numerical experiments, we work with the gauge transformed Hamiltonian of (15). This allows us to use dense vectors of length 2^N to represent the corresponding spin states, rather than length

$2^{(N-1)m+N}$ which would be required if we included the Hilbert space of the gauge fields. Using this representation, we are able to generate Krylov bases for up to $N = 30$ lattice sites through sparse matrix multiplication and perform direct diagonalisation of the ground state for up to $N = 26$ using double precision with up to 3TB of RAM. In all of our experiments we fix $\mu = 1.5$ and $x = 0.5$.

As reference state $|\psi_0\rangle$ we use the ground state of the corresponding $x = 0$ system given by (10). Since this state satisfies Gauss's law, and the Hamiltonian used to generate the Krylov basis commutes with each side of Gauss's law in (9), the solutions generated by the QSE algorithm will lie in the correct symmetry sector of Gauss's law. This input state is simply a computational basis state generated by applying X operators to the odd lattice sites.

We begin by simulating the algorithm with no measurement noise, running the QSE algorithm for system sizes ranging from $N = 4$ to $N = 26$. For each system size, we perform QSE with increasing basis size D until the generalised eigenvalue problem can no longer be solved due to ill-conditioning. Although we do not introduce measurement noise at this point, finite machine precision still means that the generalised eigenvalue problem becomes unstable for sufficiently large Krylov basis. As a metric for the success of the algorithm we consider the *fractional energy error* between the error in the ground state energy produced by the QSE algorithm and the interaction energy of the system (the difference between the $x = 0$ non-interacting ground state energy and the $x \neq 0$ ground state energy). This is denoted by $\Delta E/E_{\text{int}}$. As is to be expected with a Krylov basis, we find that for each value of N, the fractional energy error decreases exponentially with the basis size. For each value of N, by performing an exponential fit between $\Delta E/E_{\text{int}}$ compared to Krylov basis size we are able to estimate the basis size required to achieve a chosen energy error (assuming the the algorithm does not fail due to finite precision). For further details of the experiments and fitting, see Appendix A.1.

The result of this fitting is given in Fig. 3 in which we present an estimate for the basis size required to achieve a desired energy error of $\Delta E/E_{\text{int}} = 10^{-4}$. Considering the gradient of the curve drawn out by the data points, the rate at which the required Krylov order D increases with N, decreases as we go from $N = 4$ to $N = 26$, with the value of D becoming approximately linear with N for the later data points. Assuming the required order continues to grow linearly, or the gradient with respect to N further decreases, by fitting a straight line to the final two data points and extrapolating this to larger N, we can estimate an upper bound on the required basis for a given N. Such a line is shown in Fig. 3 and corresponds to an upper bound of $D \approx 0.057N+4.358$. This linear relationship, characterised by a shallow slope, suggests that the basis size cost does not exhibit a substantial increase as the system size grows assuming zero measurement error and infinite precision. However, to gain a full understanding of the resource cost of the algorithm we must analyse the problem in the presence of finite measurement noise. As we will see from the following sections, the behaviour and cost of the QSE algorithm in such a scenario differs significantly from the zero-noise case.

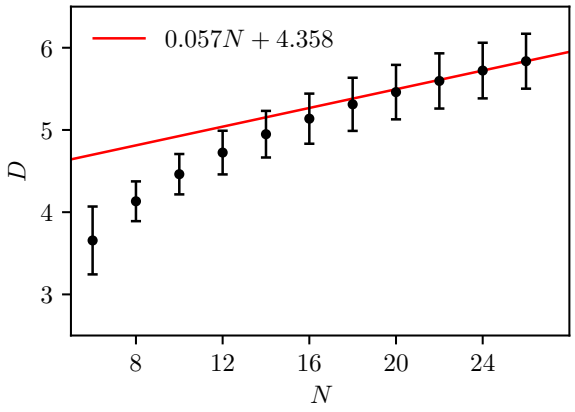


Figure 3: Estimated Krylov order D needed for QSE to achieve fractional energy error of $\Delta E/E_{\text{int}} = 10^{-4}$ in the case of no measurement noise calculated using data from Appendix A.1 for a system with $\mu = 1.5$, $x = 0.5$. Error bars correspond to standard errors on the fits to generate these points (see Appendix A.1 for details). Red line corresponds to linear fit to final two data points.

4.1 Modelling measurement noise

We now consider the effect of shot noise on the expectations values of $\mathbf{S}$ and $\mathbf{H}$ resulting from finite number of measurement. The effect of this is to perturb these matrices as $\mathbf{S} \to \mathbf{S} + \mathbf{\Delta_S}$ and $\mathbf{H} \to \mathbf{H} + \mathbf{\Delta_H}$. Like the original overlap matrices, the perturbation matrices $\mathbf{\Delta_S}, \mathbf{\Delta_H}$ are Hankel matrices[6]. Since expectation values are measured as Bernoulli trials, elements of the perturbation matrices can be modelled as random variables drawn from a normal distribution with width given by the expected variance of the expectation value.

Let $\overline{\langle\psi_0| H^k |\psi_0\rangle}$ denote the best estimate for the expectation value of $\langle\psi_0| H^k |\psi_0\rangle$ as measured according to the procedure in Sec. 3.2. If M_k measurements are used to calculate this estimate then the variance is

[6] in which each ascending skew-diagonal from left to right is constant

given by

$$\begin{aligned}\mathrm{Var}\,\overline{\langle\psi_0|\,H^k\,|\psi_0\rangle} &= \frac{\mathrm{Var}\,\langle\psi_0|\,H^k\,|\psi_0\rangle}{M_k} \\ &= \frac{\langle\psi_0|\,H^{2k}\,|\psi_0\rangle - \langle\psi_0|\,H^k\,|\psi_0\rangle^2}{M_k}.\end{aligned} \tag{32}$$

From the above, we are able to calculate the variance of each matrix element used in the generalised eigenvalue problem by calculating values of $\langle\psi_0|\,H^k\,|\psi_0\rangle$ for k twice as large as the maximum value needed for the expectation values. We choose values of M_k as follows.

We use the total calls to the block encoding operator $\Pi_\varphi U$ as a measure of the cost of performing the measurements required (counting Π_φ and $\tilde{\Pi}_\varphi$ in the same way since they have the same cost). By controlling this, we effectively control the number of gates required. Using the QSVT procedure of Theorem 3, the circuit required to generate H^k requires k applications of $\Pi_\varphi U$. Thus, given a distribution of shots $\{M_k|k = 1, \ldots, 2D+1\}$, the total number of calls to $\Pi_\varphi U$ is

$$\#\ \Pi_\varphi U \text{ calls} = \sum_{k=1}^{2D+1} kM_k. \tag{33}$$

We choose $\{M_k\}$ such that the measurement error as a fraction of expectation value is the same for all k, and as low as possible given a total number of $\Pi_\varphi U$ applications. Thus, using (32) we assign M_k as

$$M_k = \mathcal{M}\frac{\mathrm{Var}\,\langle\psi_0|\,H^k\,|\psi_0\rangle}{\langle\psi_0|\,H^k\,|\psi_0\rangle^2}, \tag{34}$$

where proportionality constant $\mathcal{M}$ is calculated to normalise the total number of block encoding calls according to (33).

To simulate the effect of the resulting measurement noise, we can then sample elements of the perturbation matrices as $\Delta_{S\,i,j} = \delta_{i+j}$, $\Delta_{H\,i,j} = \delta_{i+j+1}$, where δ_k are drawn from a normal distribution of width $\sqrt{\mathrm{Var}\,\langle\psi_0|\,H^k\,|\psi_0\rangle\,/M_k}$.

4.2 Dealing with measurement noise

For the noise levels we consider in this work, standard QSE often fails for moderately sized Krylov bases due to ill-conditioning. Therefore, for simulations with shot noise we use partitioned quantum subspace expansion (PQSE) which has been found to perform better than normal QSE and thresholded approaches in avoiding numerical instability in the presence of finite noise [67].

PQSE decomposes the full D-dimensional Krylov space into a series of smaller subspaces of dimension $\{d_i\}$ such that $\sum_i (d_i - 1) = D - 1$. Each subspace of dimensions d_i is a Krylov subspace in which the input state is the optimal state from the previous subspace of dimension d_{i-1}. Compared to the full D dimensional Krylov space, within the smaller subspaces the generalised eigenvalue problem contains smaller matrices which are less likely to be ill-conditioned. The size of each partitioned subspace is selected systematically throughout the PQSE algorithm by choosing d_i such that the solution to the corresponding generalised eigenvalue problem gives the smallest energy variance. Although the partitioned subspaces correspond to different input states, the overlap matrices $\mathbf{H}, \mathbf{S}$ required for each partition can be efficiently calculated from the same set of overlaps. As a result, PQSE requires similar overlaps as standard QSE, namely $\langle\psi_0|\,H^k\,|\psi_0\rangle$ for $k = 0, 1, \ldots, 2D$, where the single additional overlap (for $k = 2D$) is required for variance estimation. For a full description of the PQSE algorithm, we refer the reader to the original publication [67].

4.3 Results

To test the performance of the PQSE algorithm in the presence of measurement noise we simulate noise as described in Sec. 4.1 for differing numbers of calls to block encoding operators $\Pi_\varphi U$. For a given value of $\Pi_\varphi U$ calls, we split these calls among varying numbers of expectation values to apply the PQSE procedure with different total Krylov basis size D.

We considered lattice sizes varying from $N = 4$ to $N = 26$. For the extrapolation for the total resources we use data resulting from simulations using $10^2, 10^3, \ldots, 10^{15}$ calls to $\Pi_\varphi U$, for clarity, we show a representative subset of these results in Fig. 4. For each choice of $\Pi_\varphi U$ calls and N, 100 noise instances are simulated with the median and upper and lower quartiles plotted. From this we notice three things. Firstly, for a given number of calls to $\Pi_\varphi U$, as we increase the maximum Krylov order the fractional energy error starts off decreasing, until reaching a minimum at which point it begins to increase. We can interpret the initial decreasing behaviour as the error reduces due to the increased basis size - of course the error is worse than the noiseless equivalent due to error on the expectation values. Once the minimum is reached however, the matrices become sufficiently ill-conditioned such that any finite measurement noise is enough for the generalised eigenvalue problem to become unstable and so increasing order doesn't improve solution quality. In fact, when this happens increasing the maximum value of D increases the measurement error on expectation values as we have fixed the number of shots, and so error increases rapidly. Secondly, the value of D giving the lowest error increases with N. This is to be expected as noiseless results for this, as well as other Hamiltonians, typically require larger Krylov bases for larger systems to

achieve the same error. Finally, in all cases increasing the number of calls to $\Pi_\varphi U$, and therefore reducing measurement error decreases overall solution error.

As in the noiseless case, we want to predict the number of resources required, this time in terms of total calls to $\Pi_\varphi U$ rather than basis size, to achieve a desired solution accuracy for a given system size. To estimate this, we begin by taking the smallest median energy error for each number of calls to $\Pi_\varphi U$. These correspond to the lowest points on each curve in Fig. 4. This assumes that we have a reasonably good guess for the maximum order required to give the best results through PQSE. In practice, this could be inferred through some method of scaling up system size through real or simulated experiments. We see that for larger systems in Fig. 4, there are a number of choices of Maximum Krylov order that lead to near-optimal output error.

For each value of N, we find that the best median energy error achieved by PQSE decreases with a power law dependence on the number of calls to the block encoding operator $\Pi_\varphi U$. This manifests itself, rather than the log-linear relationship demonstrated in the noiseless case, as a log-log dependence between measurement error and resource cost. See Appendix A.2. Fitting to this dependence we are able to predict the number of calls to the $\Pi_\varphi U$ to achieve a desired fractional energy error. For example, the number of calls required to achieve a fractional error of 10^{-4} is shown in Fig. 5. From this we see that the number of calls to $\Pi_\varphi U$ grows approximately exponentially with N as $\#\Pi_\varphi U$ calls $\approx 10^{8.919} \cdot 1.143^N$ for the values of N considered.

Applying thresholded QSE as described by Epperly et al. [68] for the same lattice sizes gives energy errors that are approximately an order of magnitude worse than PQSE with same number of resources. See Appendix A.3 for further details.

5 Estimating resource requirements

After making an estimate of the basis size; and therefore number of calls to the qubitization procedure required, to understand the total resource cost of solving the Schwinger model, we must now compute the number of gates required to implement each part of the quantum circuit. A full description of the different parts of the circuits, along with the mathematics needed to estimate the gate cost is long-winded. Therefore, we defer a full description of the algorithmic implementation along with proofs for upper bounds of the required gate costs to Appendix C. We point the reader in particular to Appendix C.1 which outlines a sketch of the required proofs. Final results of the costing are presented in Theorems 27, 32, 35 and 38.

In Fig. 6 (a) we plot the number of gates as a function of system size N for a single step of the block encoding procedure. This is given by the cost of U plus Π_φ (which is the same as $\widetilde{\Pi}_\varphi$). Fig. 6 (b) gives the qubit cost of the algorithm which, for systems of N in the hundreds or thousands, is not far off the thousands of physical qubits targeted by, for example IBM, in the next few years. Figs. 6 (c) and (d) show a direct comparison between our improved implementations of the block encoding procedure described in the previous section and the LCU method described by Kirby et al. [63]. We see that for both T-gate count and CNOT count, our optimisations lead to significant reductions in the gate cost.

Throughout this work, we have assumed fault tolerant computation. To quickly assess whether this is necessary, or whether non error-corrected devices may be sufficient, we compare the expected runtime of the algorithmic gates to the coherence times for existing processors. Assuming that two qubit gates are run in series and give the largest contribution to runtime we find that the time needed to execute the CNOTs required for a single step of the QSVT procedure (the application of $\Pi_\varphi U$) is larger than the coherence times for systems of $N = 100$ for a range of current generation quantum processors. For systems of $N = 1000$ and $N = 10000$, the runtime of the gates is many orders of magnitude larger than the coherence times of the devices. See Appendix F for further details on what hardware specifications were used for these comparisons. Even assuming that two-qubit gates can be maximally parallelised (running $N_{\text{qubits}}/2$ two-qubit gates at the same time), the time to execute $\Pi_\varphi U$ can be of the order of $\sim$0.1% to $\sim$1000% of the coherence times. Such estimates indicate the circuit lengths required to solve the Schwinger model, at least using the implementation presented here, are beyond what is possible on currently available devices and those that we can expect in the foreseeable future. Therefore, for system sizes of hundreds of lattice sites the algorithm we present must be run in an error-corrected setting.

Having confirmed that fault-tolerance is indeed necessary for systems of hundreds of lattice sites, we now assess the feasibility of our algorithm on an error corrected device. Rather than CNOT gates dominating the execution time of the algorithm, the number of non-Clifford gates play the largest part in determining how long a computation will take to run. We estimate the total number of T-gates required to calculate the ground state energy by combining the circuit costs for a single block encoding step with the total number of calls to the block encoding estimated in extrapolating the numerical results of Section 4 as in Fig. 5. This cost is shown for a range of desired errors in Fig. 7. We see that, for systems of hundreds of lattice sites, the exponential number of measurements we require in order to control the measurement noise leads to

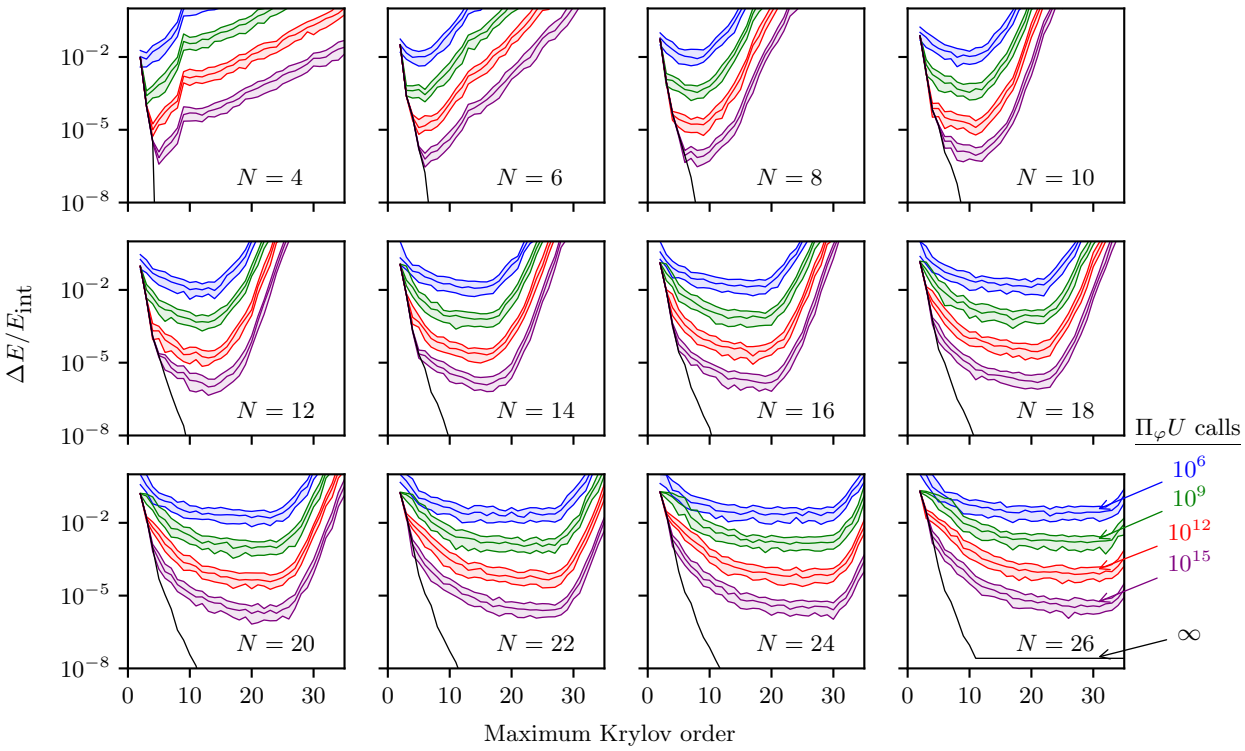


Figure 4: Fractional energy error with PQSE procedure with finite number of measurements for range of different lattice sizes N for $\mu = 1.5$, $x = 5$. Values are shown as a function of the maximum allowed Krylov basis for the PQSE procedure and as a function of number of calls to the block encoding operator $\Pi_\varphi U$ (which affect the number of measurements as defined in the main text). Lines and shaded area corresponds to the median as well as upper and lower quartiles of energy error. The black line (labelled as infinite calls to $\Pi_\varphi U$) corresponds to no measurement noise.

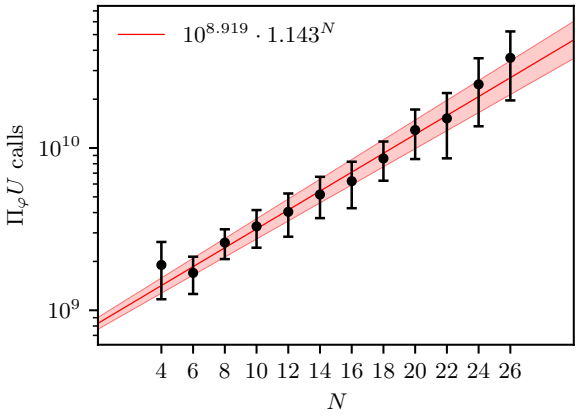


Figure 5: Estimated number of call to qubitization procedure using PQSE to achieve fractional energy error of $\Delta E/E_{\text{int}} = 10^{-4}$ for $\mu = 1.5$, $x = 0.5$, calculated using data from Fig. 4. Error bars correspond to the standard error resulting from fits used to generate the data points. Red line corresponds to exponential fit (straight line on log-linear axes) with filled area indicating uncertainty in fitting and extrapolation.

T-gate costs prohibitive within even the most optimistic predictions of fault tolerant computations [98]. However, given that we have only been able to simulate systems of up to $N = 26$ and extrapolated to systems orders of magnitude larger, it is possible that the exponential scaling we predict does not hold for larger systems. Further numerical tests, using tensor network methods to study larger system sizes and using Monte-Carlo methods to calculate the true ground state energy (which does not suffer from the sign problem for the single flavour system we consider) may shed more light on this. Whether the resource scaling would be more or less promising is unknown.

Furthermore, we repeated our analysis within a regime of strong coupling with $\mu = 1.5$ and $x = 5$ (as opposed to $x = 0.5$ considered up to this point). Details of these results, and in particular the associated plots can be found in App. D On the whole, we find a similar behaviour as in the $x = 0.5$ case. Within the noiseless regime for fixed N, $\Delta E/E_{\text{int}}$ decrees exponentially with basis size D and for fixed $\Delta E/E_{\text{int}}$, the required basis D grows approximately linearly with N. The gradient of this linear dependence is larger than in the strong coupling case, reflecting the fact that the overlap between the Néel state and the true ground state decreases more quickly with N and therefore a larger Krylov basis is needed to achieve the same error. We find a similar behaviour when considering the effect of measurement noise, giving an estimate of gate costs that grow exponentially with N and with a stronger exponential dependence that the weaker coupling regime.

6 Discussion and conclusions

In this work we presented a study of the application of the QSE algorithm to a LGT. In particular, we estimated the resources required to compute the ground state of the single flavour lattice Schwinger model for different system sizes and assess how feasible such a computation would be on early error-corrected quantum processors.

By developing a novel LCU procedure for block encoding the Schwinger model Hamiltonian and analysing the gate cost required, in combination with numerical simulation, we successfully give a upper bound estimate for the full gate cost of the algorithm.

For systems of a few hundred lattice sites, we see that the gate depth for a single circuit is beyond what we we expect to be feasible on non-error corrected devices in the near term. However, single circuit T-gate costs on the order of 10^4 to 10^6, along with hundreds of logical qubits required make such circuits appealing targets for early fault tolerant devices.

While such gate depths for individual circuits can be seen as a positive step towards useful application of QSE to LGTs, the exponential measurement cost revealed by our numerical simulations indicates that further algorithmic improvements will be required in order to use such a workflow to compute accurate ground states of LGTs - even for reasonably large error corrected devices. While the exponential measurement cost may occur for other systems and QSE methods, by performing a rigorous analysis with respect to system size scaling, we have been able to quantify and draw conclusions about the resource cost for different algorithmic approaches.

Since ground state problems for fermionic systems tend to be QMA-complete [99–102], such an exponential scaling is not particularly surprising[7]. Nor should it deter research into using QSE algorithms for LGTs but rather motivate further work aimed at reducing the algorithmic cost of such methods. Importantly, our work presents the first systematic study of the scaling of two QSE algorithms, namely thresholded and partitioned, with system size. We show that PQSE can significantly reduce the exponential scaling of the algorithm compared to the current state of the art (TQSE), serving as an important benchmark for future improvements and demonstrating the necessity to further develop noise-robust subspace methods.

A number of works have considered QSE using a basis consisting of real-time evolution [62, 69–71, 103, 104],

[7]Although the single flavour Schwinger model can be solved with Monte-Carlo methods that do not suffer from the sign problem. We anticipate quantum algorithms, like the one we present here, being applied to multi-flavour models which are known to suffer from the sign problem and therefore seem unlikely to have efficient computational solutions.

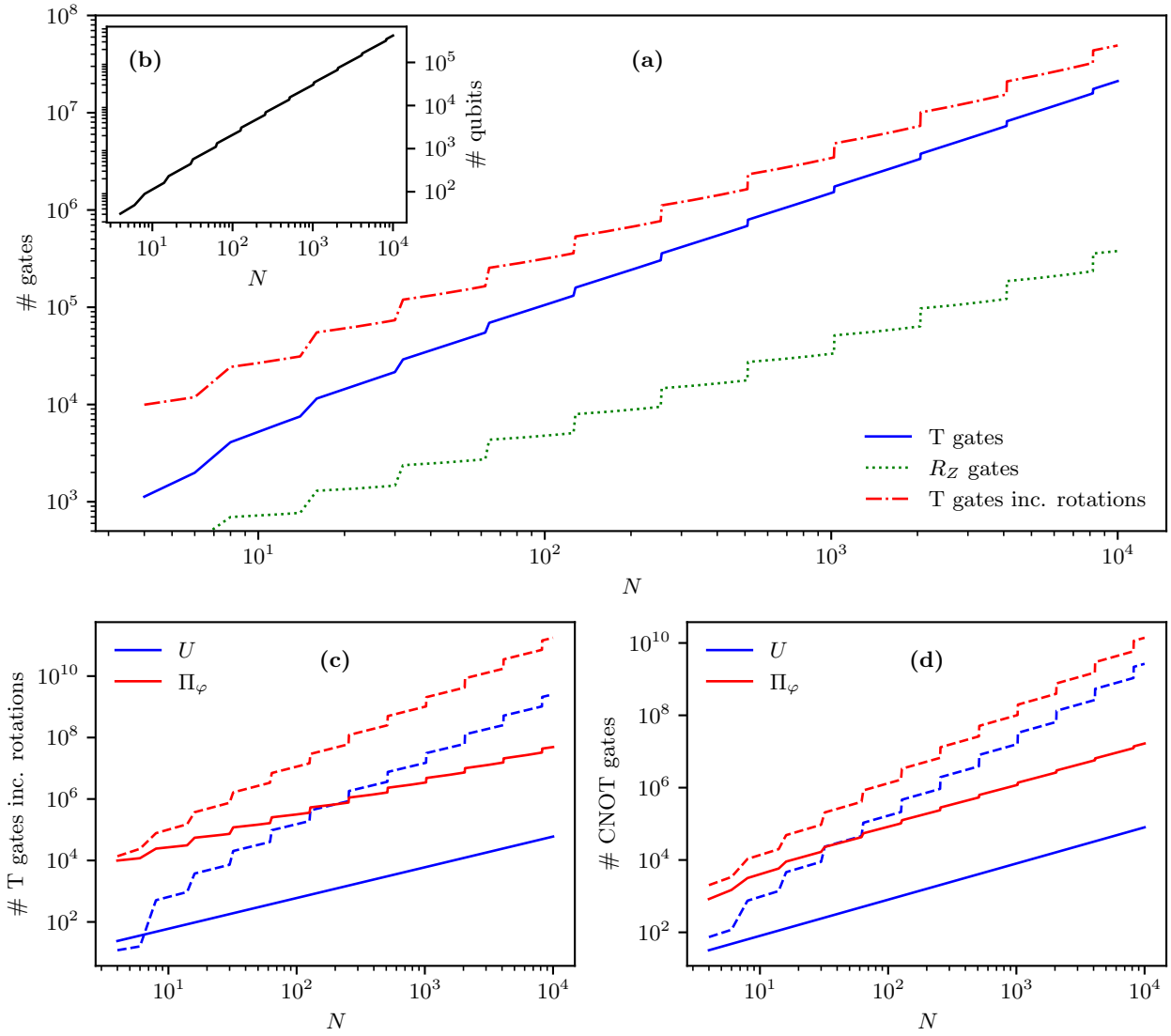


Figure 6: Upper bounds for resource costs in a single step of QSVT procedure (single application of U and Π_φ). **(a)** Total gate costs as a function of lattice size. The line labelled "T-gates" corresponds to T-gates arising from controlled operations excluding rotation gates. The line labelled "T-gates inc. rotations" includes the additional cost of converting single qubit rotation gates (line labelled "R_z gates") using Ref. [97]. Although not shown, the CNOT gate cost is indistinguishable from the T-gate line on this scale. **(b)** Total logical qubit count. **(c)** T-gate cost for individual operators U and Π_φ. Dashed lines give costs for U and Π_φ using LCU procedure as described by Kirby et al. (see Appendix B.2). Solid lines give cost using our improved methods (see Appendix C.4.6 for details). **(d)** Same as (c) except for CNOT gates.

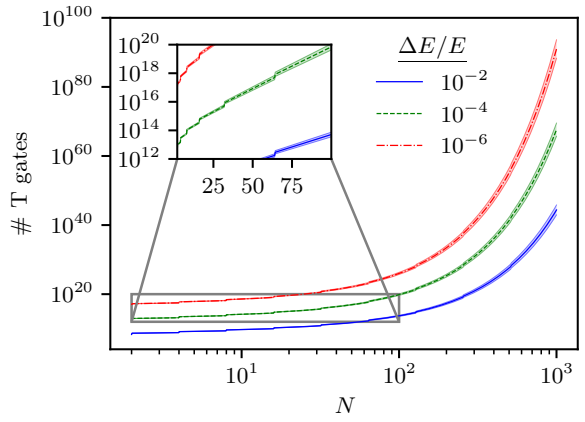


Figure 7: Total number of T-gates required for the whole PQSE procedure for the single flavour Schwinger model with $\mu = 1.5$, $x = 0.5$ as a function of lattice size N for different fractional energy errors $\Delta E/E_{\text{int}}$.

imaginary time evolution [73, 74], Chebyshev polynomials [63] and time evolved states that have been Fourier transformed with respect to a set of filter energies [71] as opposed to powers of the Hamiltonian. One particularly promising method for improving the noise resilience of QSE has been the proposal of the Gaussian power basis [105]. Generating such basis states may still be infeasible with non-fault-tolerant devices, particularly if one tries to use Trotterisation for large systems in multiple spatial dimensions. In the fault-tolerant regime however, these bases could be implemented on a quantum processor using QSVT techniques and the block encoding we present in this work. While investigating other basis methods is beyond the scope of the work, and would require more sophisticated methods for classical simulation of the overlap matrices, it would be interesting to perform the resource estimation we present here for such methods to see if this leads to more feasible gate costs. This would play an important part in developing a thorough understanding of the resources required to implement such algorithms using block-encoding or Trotterization based approaches as well as related algorithms such as QETU [66]. This may reveal regimes of both problem size and hardware specification in which the lower overhead of Trotterisation outweighs the favourable asymptotic scaling of block-encoding methods and vice-versa. As one considers scaling up system size, it seems likely that approaches based on real- or imaginary-time evolution via Trottersation give lower gate costs for smaller problem instances within the near-term non-error corrected or even early-fault tolerant era and more complex qubitization based subspace methods preferred for larger systems with the onset of large scale fault tolerant devices.

Finally, as part of our comprehensive resource analysis for ground state computation using QSE, we developed novel improvements to the LCU block encoding procedure with particular application to lattice gauge theory Hamiltonians. By exploiting symmetry within the Hamiltonian, something not present in molecular Hamiltonians for example, we can reduce the cost of the LCU procedure when applied to the Schwinger model from $\tilde{\mathcal{O}}(N^2)$ to $\tilde{\mathcal{O}}(N)$. Our procedure could be used to reduce the resources required for other quantum algorithms to study ground states or dynamics of LGTs such as phase estimation and Hamiltonian simulation, both of which have seen significant cost reductions by using qubitization for quantum chemistry in an error corrected setting [30, 106].

Code availabilty

Code for reproducing the figures presented in this paper can be found at `https://github.com/lw-anderson/quantum_krylov_for_lattice_schwinger`.

Acknowledgments

LWA thanks Nick Bultinck and Mari Carmen Bañuls for helpful discussions. LWA, MK and DJ acknowledge support from the EPSRC National Quantum Technology Hub in Networked Quantum Information Technology (EP/M013243/1) and the EPSRC Hub in Quantum Computing and Simulation (EP/T001062/1). TO acknowledges support by the EPSRC through an EPSRC iCASE studentship award in collaboration with IBM Research. DJ acknowledges support from the Hamburg Quantum Computing Initiative (HQIC) project EFRE. The project is co-financed by ERDF of the European Union and by 'Fonds of the Hamburg Ministry of Science, Research, Equalities and Districts (BWFGB)'. DJ acknowledges funding by the Cluster of Excellence 'Advanced Imaging of Matter' of the Deutsche Forschungsgemeinschaft (DFG) - EXC 2056 - project ID 390715994. The authors would like to acknowledge the use of the University of Oxford Advanced Research Computing (ARC) facility in carrying out this work [107].

References

[1] Matthew D Schwartz. *Quantum field theory and the standard model*. Cambridge University Press, 2014. DOI: 10.1017/9781139540940.

[2] David Bailin and Alexander Love. *Cosmology in gauge field theory and string theory*. Taylor & Francis, 2004. DOI: 10.1201/9780367806637.

[3] Eduardo Fradkin. *Field theories of condensed matter physics*. Cambridge University Press, 2013. DOI: 10.1017/CBO9781139015509.

[4] Kenneth G Wilson. Confinement of quarks. *Physical review D*, 10(8):2445, 1974. DOI: 10.1103/PhysRevD.10.2445.

[5] Michael Creutz, Laurence Jacobs, and Claudio Rebbi. Monte carlo computations in lattice gauge theories. *Physics Reports*, 95(4):201–282, 1983. DOI: 10.1016/0370-1573(83)90016-9.

[6] Matthias Troyer and Uwe-Jens Wiese. Computational complexity and fundamental limitations to fermionic quantum monte carlo simulations. *Physical review letters*, 94(17):170201, 2005. DOI: 10.1103/PhysRevLett.94.170201.

[7] Anders W Sandvik and Juhani Kurkijärvi. Quantum monte carlo simulation method for spin systems. *Physical Review B*, 43(7):5950, 1991. DOI: 10.1103/PhysRevB.43.5950.

[8] Julian Schwinger. Gauge invariance and mass. II. *Physical Review*, 128(5):2425, 1962. DOI: 10.1103/PhysRev.125.397.

[9] Qi-Zhou Chen and Xiang-Qian Luo. Chiral-symmetry breaking in the model with wilson fermions. *Physical Review D*, 42(4):1293, 1990. DOI: 10.1103/PhysRevD.42.1293.

[10] Kirill Melnikov and Marvin Weinstein. Lattice model: Confinement, anomalies, chiral fermions, and all that. *Physical Review D*, 62(9):094504, 2000. DOI: 10.1103/PhysRevD.62.094504.

[11] Boye Buyens, Jutho Haegeman, Frank Verstraete, and Karel Van Acoleyen. Tensor networks for gauge field theories. *arXiv preprint arXiv:1511.04288*, 2015. DOI: 10.48550/arXiv.1511.04288.

[12] Mari Carmen Banuls, Krzysztof Cichy, J. Ignacio Cirac, Karl Jansen, and Stefan Kühn. Tensor networks and their use for lattice gauge theories. *PoS*, LATTICE2018:022, 2019. DOI: 10.22323/1.334.0022.

[13] Boye Buyens, Jutho Haegeman, Karel Van Acoleyen, Henri Verschelde, and Frank Verstraete. Matrix product states for gauge field theories. *Physical review letters*, 113(9):091601, 2014. DOI: 10.1103/PhysRevLett.113.091601.

[14] Kai Zapp and Román Orús. Tensor network simulation of QED on infinite lattices: Learning from (1+1)d, and prospects for (2+1)d. *Physical Review D*, 95(11):114508, 2017. DOI: 10.1103/PhysRevD.95.114508.

[15] Mari Carmen Bañuls, K Cichy, J Ignacio Cirac, and Karl Jansen. The mass spectrum of the Schwinger model with matrix product states. *Journal of High Energy Physics*, 2013(11):1–21, 2013. DOI: 10.1007/JHEP11(2013)158.

[16] Elisa Ercolessi, Paolo Facchi, Giuseppe Magnifico, Saverio Pascazio, and Francesco V Pepe. Phase transitions in $\mathbb{Z}_n$ gauge models: Towards quantum simulations of the Schwinger-Weyl QED. *Physical Review D*, 98(7):074503, 2018. DOI: 10.1103/PhysRevD.98.074503.

[17] Lena Funcke, Karl Jansen, and Stefan Kühn. Exploring the CP-violating dashen phase in the Schwinger model with tensor networks. *Physical Review D*, 108(1):014504, 2023. DOI: 10.1103/PhysRevD.108.014504.

[18] Mari Carmen Bañuls, Krzysztof Cichy, Karl Jansen, and Hana Saito. Chiral condensate in the Schwinger model with matrix product operators. *Physical Review D*, 93(9):094512, 2016. DOI: 10.1103/PhysRevD.93.094512.

[19] Torsten V Zache, Maarten Van Damme, Jad C Halimeh, Philipp Hauke, and Debasish Banerjee. Toward the continuum limit of a (1+1)D quantum link Schwinger model. *Physical Review D*, 106(9):L091502, 2022. DOI: 10.1103/PhysRevD.106.L091502.

[20] Mari Carmen Bañuls, Krzysztof Cichy, J Ignacio Cirac, Karl Jansen, and Stefan Kühn. Efficient basis formulation for (1+1)-dimensional SU(2) lattice gauge theory: Spectral calculations with matrix product states. *Physical Review X*, 7(4):041046, 2017. DOI: 10.1103/PhysRevX.7.041046.

[21] Thomas Pichler, Marcello Dalmonte, Enrique Rico, Peter Zoller, and Simone Montangero. Real-time dynamics in U(1) lattice gauge theories with tensor networks. *Physical Review X*, 6(1):011023, 2016. DOI: 10.1103/PhysRevX.6.011023.

[22] Boye Buyens, Jutho Haegeman, Florian Hebenstreit, Frank Verstraete, and Karel Van Acoleyen. Real-time simulation of the Schwinger effect with matrix product states. *Physical Review D*, 96(11):114501, 2017. DOI: 10.1103/PhysRevD.96.114501.

[23] Giuseppe Magnifico, Marcello Dalmonte, Paolo Facchi, Saverio Pascazio, Francesco V Pepe, and Elisa Ercolessi. Real time dynamics and confinement in the $\mathbb{Z}_n$ Schwinger-Weyl lattice model for 1+1 QED. *Quantum*, 4:281, 2020. DOI: 10.22331/q-2020-06-15-281.

[24] Marco Rigobello, Simone Notarnicola, Giuseppe Magnifico, and Simone Montangero. Entanglement generation in (1+ 1)D QED scattering processes. *Physical Review D*, 104(11):114501, 2021. DOI: 10.1103/PhysRevD.104.114501.

[25] Jad C Halimeh, Maarten Van Damme, Torsten V Zache, Debasish Banerjee, and Philipp Hauke. Achieving the quantum field theory limit in far-from-equilibrium quantum link models. *Quantum*, 6:878, 2022. DOI: 10.22331/q-2022-12-19-878.

[26] Christian W Bauer, Zohreh Davoudi, A Baha Balantekin, Tanmoy Bhattacharya, Marcela Carena, Wibe A de Jong, Patrick Draper, Aida El-Khadra, Nate Gemelke, Masanori Hanada,

et al. Quantum simulation for high-energy physics. *PRX Quantum*, 4(2):027001, 2023. DOI: 10.1103/PRXQuantum.4.027001.

[27] Alberto Di Meglio, Karl Jansen, Ivano Tavernelli, Constantia Alexandrou, Srinivasan Arunachalam, Christian W Bauer, Kerstin Borras, Stefano Carrazza, Arianna Crippa, Vincent Croft, et al. Quantum computing for high-energy physics: state of the art and challenges. *PRX Quantum*, 5(3):037001, 2024. DOI: 10.1103/PRXQuantum.5.037001.

[28] Mari Carmen Banuls, Rainer Blatt, Jacopo Catani, Alessio Celi, Juan Ignacio Cirac, Marcello Dalmonte, Leonardo Fallani, Karl Jansen, Maciej Lewenstein, Simone Montangero, et al. Simulating lattice gauge theories within quantum technologies. *The European physical journal D*, 74:1–42, 2020. DOI: 10.1140/epjd/e2020-100571-8.

[29] Alexander F Shaw, Pavel Lougovski, Jesse R Stryker, and Nathan Wiebe. Quantum algorithms for simulating the lattice Schwinger model. *Quantum*, 4:306, 2020. DOI: 10.22331/q-2020-08-10-306.

[30] Guang Hao Low and Isaac L Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3: 163, 2019. DOI: 10.22331/q-2019-07-12-163.

[31] Earl Campbell. Random compiler for fast hamiltonian simulation. *Physical review letters*, 123(7):070503, 2019. DOI: 10.1103/PhysRevLett.123.070503.

[32] Angus Kan and Yunseong Nam. Simulating lattice quantum electrodynamics on a quantum computer. *Quantum Science and Technology*, 8(1):015008, 2022. DOI: 10.1088/2058-9565/aca0b8.

[33] Natalie Klco, Eugene F Dumitrescu, Alex J McCaskey, Titus D Morris, Raphael C Pooser, Mikel Sanz, Enrique Solano, Pavel Lougovski, and Martin J Savage. Quantum-classical computation of Schwinger model dynamics using quantum computers. *Physical Review A*, 98(3):032331, 2018. DOI: 10.1103/PhysRevA.98.032331.

[34] Nhung H Nguyen, Minh C Tran, Yingyue Zhu, Alaina M Green, C Huerta Alderete, Zohreh Davoudi, and Norbert M Linke. Digital quantum simulation of the Schwinger model and symmetry protection with trapped ions. *PRX Quantum*, 3(2):020324, 2022. DOI: 10.1103/PRXQuantum.3.020324.

[35] Roland C Farrell, Marc Illa, Anthony N Ciavarella, and Martin J Savage. Quantum simulations of hadron dynamics in the schwinger model using 112 qubits. *Physical Review D*, 109(11):114510, 2024. DOI: 10.1103/PhysRevD.109.114510.

[36] Wibe A de Jong, Kyle Lee, James Mulligan, Mateusz Płoskoń, Felix Ringer, and Xiaojun Yao. Quantum simulation of nonequilibrium dynamics and thermalization in the Schwinger model. *Physical Review D*, 106(5):054508, 2022. DOI: 10.1103/PhysRevD.106.054508.

[37] Christian Kokail, Christine Maier, Rick van Bijnen, Tiff Brydges, Manoj K Joshi, Petar Jurcevic, Christine A Muschik, Pietro Silvi, Rainer Blatt, Christian F Roos, et al. Self-verifying variational quantum simulation of lattice models. *Nature*, 569(7756):355–360, 2019. DOI: 10.1038/s41586-019-1177-4.

[38] A Avkhadiev, PE Shanahan, and RD Young. Accelerating lattice quantum field theory calculations via interpolator optimization using noisy intermediate-scale quantum computing. *Physical review letters*, 124(8):080501, 2020. DOI: 10.1103/PhysRevLett.124.080501.

[39] Ryan R Ferguson, Luca Dellantonio, Abdulrahim Al Balushi, Karl Jansen, Wolfgang Dür, and Christine A Muschik. Measurement-based variational quantum eigensolver. *Physical review letters*, 126(22):220501, 2021. DOI: 10.1103/PhysRevLett.126.220501.

[40] Shane Thompson and George Siopsis. Quantum computation of phase transition in the massive Schwinger model. *Quantum Science and Technology*, 7(3):035001, 2022. DOI: 10.1088/2058-9565/ac5f5a.

[41] Lena Funcke, Tobias Hartung, Karl Jansen, Stefan Kühn, Marc-Oliver Pleinert, Stephan Schuster, and Joachim von Zanthier. Exploring the phase structure of the multi-flavor Schwinger model with quantum computing. *arXiv preprint arXiv:2211.13020*, 2022. DOI: 10.48550/arXiv.2211.13020.

[42] Giulia Mazzola, Simon V Mathis, Guglielmo Mazzola, and Ivano Tavernelli. Gauge-invariant quantum circuits for U(1) and yang-mills lattice gauge theories. *Physical Review Research*, 3(4):043209, 2021. DOI: 10.1103/PhysRevResearch.3.043209.

[43] Danny Paulson, Luca Dellantonio, Jan F Haase, Alessio Celi, Angus Kan, Andrew Jena, Christian Kokail, Rick Van Bijnen, Karl Jansen, Peter Zoller, et al. Simulating 2D effects in lattice gauge theories on a quantum computer. *PRX Quantum*, 2(3):030334, 2021. DOI: 10.1103/PRXQuantum.2.030334.

[44] Luca Lumia, Pietro Torta, Glen B Mbeng, Giuseppe E Santoro, Elisa Ercolessi, Michele Burrello, and Matteo M Wauters. Two-dimensional $\mathbb{Z}_2$ lattice gauge theory on a near-term quantum simulator: Variational quantum optimization, confinement, and topological order. *PRX Quantum*, 3(2):020320, 2022. DOI: 10.1103/PRXQuantum.3.020320.

[45] Michael Meth, Jan F Haase, Jinglei Zhang,

Claire Edmunds, Lukas Postler, Alex Steiner, Andrew J Jena, Luca Dellantonio, Rainer Blatt, Peter Zoller, et al. Simulating 2D lattice gauge theories on a qudit quantum computer. *arXiv preprint arXiv:2310.12110*, 2023. DOI: 10.48550/arXiv.2310.12110.

[46] Roland C Farrell, Marc Illa, Anthony N Ciavarella, and Martin J Savage. Scalable circuits for preparing ground states on digital quantum computers: The schwinger model vacuum on 100 qubits. *PRX Quantum*, 5 (2):020315, 2024. DOI: 10.1103/PRXQuantum.5.020315.

[47] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9:4812, 2018. DOI: 10.1038/s41467-018-07090-4.

[48] Samson Wang, Enrico Fontana, Marco Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick J Coles. Noise-induced barren plateaus in variational quantum algorithms. *Nature Communications*, 12:6961, 2021. DOI: 10.1038/s41467-021-27045-6.

[49] Marco Cerezo, Akira Sone, Tyler Volkoff, Lukasz Cincio, and Patrick J Coles. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature communications*, 12:1791, 2021. DOI: 10.1038/s41467-021-21728-w.

[50] Carlos Ortiz Marrero, Mária Kieferová, and Nathan Wiebe. Entanglement-induced barren plateaus. *PRX Quantum*, 2(4):040316, 2021. DOI: 10.1103/PRXQuantum.2.040316.

[51] AV Uvarov and Jacob D Biamonte. On barren plateaus and cost function locality in variational quantum algorithms. *Journal of Physics A: Mathematical and Theoretical*, 54(24):245301, 2021. DOI: 10.1088/1751-8121/abfac7.

[52] Zoë Holmes, Kunal Sharma, Marco Cerezo, and Patrick J Coles. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum*, 3(1):010313, 2022. DOI: 10.1103/PRXQuantum.3.010313.

[53] Enrico Fontana, Dylan Herman, Shouvanik Chakrabarti, Niraj Kumar, Romina Yalovetzky, Jamie Heredge, Shree Hari Sureshbabu, and Marco Pistoia. The adjoint is all you need: Characterizing barren plateaus in quantum ansätze. *arXiv preprint arXiv:2309.07902*, 2023. DOI: 10.48550/arXiv.2309.07902.

[54] Michael Ragone, Bojko N Bakalov, Frédéric Sauvage, Alexander F Kemper, Carlos Ortiz Marrero, Martin Larocca, and M Cerezo. A unified theory of barren plateaus for deep parametrized quantum circuits. *arXiv preprint arXiv:2309.09342*, 2023. DOI: 10.48550/arXiv.2309.09342.

[55] Ali Hamed Moosavian, James R Garrison, and Stephen P Jordan. Site-by-site quantum state preparation algorithm for preparing vacua of fermionic lattice field theories. *arXiv preprint arXiv:1911.03505*, 2019. DOI: 10.48550/arXiv.1911.03505.

[56] Ali Hamed Moosavian and Stephen Jordan. Faster quantum algorithm to simulate fermionic quantum field theory. *Physical Review A*, 98(1):012332, 2018. DOI: 10.1103/PhysRevA.98.012332.

[57] Stephen P Jordan, Keith SM Lee, and John Preskill. Quantum algorithms for fermionic quantum field theories. *arXiv preprint arXiv:1404.7115*, 2014. DOI: 10.48550/arXiv.1404.7115.

[58] Jarrod R McClean, Mollie E Kimchi-Schwartz, Jonathan Carter, and Wibe A De Jong. Hybrid quantum-classical hierarchy for mitigation of decoherence and determination of excited states. *Physical Review A*, 95(4):042308, 2017. DOI: 10.1103/PhysRevA.95.042308.

[59] Jarrod R McClean, Zhang Jiang, Nicholas C Rubin, Ryan Babbush, and Hartmut Neven. Decoding quantum errors with subspace expansions. *Nature communications*, 11(1):636, 2020. DOI: 10.1038/s41467-020-14341-w.

[60] Nobuyuki Yoshioka, Hideaki Hakoshima, Yuichiro Matsuzaki, Yuuki Tokunaga, Yasunari Suzuki, and Suguru Endo. Generalized quantum subspace expansion. *Physical Review Letters*, 129(2):020502, 2022. DOI: 10.1103/PhysRevLett.129.020502.

[61] Nobuyuki Yoshioka, Takeshi Sato, Yuya O Nakagawa, Yu-ya Ohnishi, and Wataru Mizukami. Variational quantum simulation for periodic materials. *Physical Review Research*, 4(1): 013052, 2022. DOI: PhysRevResearch.4.013052.

[62] William Kirby. Analysis of quantum Krylov algorithms with errors. *Quantum*, 8:1457, 2024. ISSN 2521-327X. DOI: 10.22331/q-2024-08-29-1457.

[63] William Kirby, Mario Motta, and Antonio Mezzacapo. Exact and efficient lanczos method on a quantum computer. *Quantum*, 7:1018, 2023. DOI: 10.22331/q-2023-05-23-1018.

[64] Thomas E Baker. Lanczos recursion on a quantum computer for the Green's function and ground state. *Physical Review A*, 103(3):032404, 2021. DOI: 10.1103/PhysRevA.103.032404.

[65] Yulong Dong, Lin Lin, and Yu Tong. Ground-state preparation and energy estimation on early fault-tolerant quantum computers via quantum eigenvalue transformation of unitary matrices. *PRX quantum*, 3(4):040305, 2022. DOI: 10.1103/PRXQuantum.3.040305.

[66] Christopher F Kane, Niladri Gomes, and Michael Kreshchuk. Nearly optimal state preparation for quantum simulations of lattice gauge theories. *Physical Review A*, 110(1):012455, 2024.

[67] Tom O'Leary, Lewis W Anderson, Dieter Jaksch, and Martin Kiffner. Partitioned quantum subspace expansion. *arXiv preprint arXiv:2403.08868*, 2024. DOI: 10.48550/arXiv.2403.08868.

[68] Ethan N Epperly, Lin Lin, and Yuji Nakatsukasa. A theory of quantum subspace diagonalization. *SIAM Journal on Matrix Analysis and Applications*, 43(3):1263–1290, 2022. DOI: 10.1137/21M145954X.

[69] Robert M Parrish and Peter L McMahon. Quantum filter diagonalization: Quantum eigendecomposition without full quantum phase estimation. *arXiv preprint arXiv:1909.08925*, 2019. DOI: 10.48550/arXiv.1909.08925.

[70] Nicholas H Stair, Renke Huang, and Francesco A Evangelista. A multireference quantum krylov algorithm for strongly correlated electrons. *Journal of chemical theory and computation*, 16(4):2236–2245, 2020. DOI: 10.1021/acs.jctc.9b01125.

[71] Cristian L Cortes and Stephen K Gray. Quantum krylov subspace algorithms for ground- and excited-state energy estimation. *Physical Review A*, 105(2):022417, 2022. DOI: 10.1103/PhysRevA.105.022417.

[72] Nicholas H Stair, Cristian L Cortes, Robert M Parrish, Jeffrey Cohn, and Mario Motta. Stochastic quantum krylov protocol with double-factorized hamiltonians. *Physical Review A*, 107(3):032414, 2023. DOI: 10.1103/PhysRevA.107.032414.

[73] Mario Motta, Chong Sun, Adrian TK Tan, Matthew J O'Rourke, Erika Ye, Austin J Minnich, Fernando GSL Brandao, and Garnet Kin-Lic Chan. Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution. *Nature Physics*, 16 (2):205–210, 2020. DOI: 10.1038/s41567-019-0704-4.

[74] Kübra Yeter-Aydeniz, Raphael C Pooser, and George Siopsis. Practical quantum computation of chemical and nuclear energy levels using quantum imaginary time evolution and lanczos algorithms. *npj Quantum Information*, 6(1):63, 2020. DOI: 10.1038/s41534-020-00290-1.

[75] Tyler Takeshita, Nicholas C Rubin, Zhang Jiang, Eunseok Lee, Ryan Babbush, and Jarrod R McClean. Increasing the representation accuracy of quantum simulations of chemistry without extra quantum resources. *Physical Review X*, 10(1):011004, 2020. DOI: 10.1103/PhysRevX.10.011004.

[76] James I Colless, Vinay V Ramasesh, Dar Dahlen, Machiel S Blok, Mollie E Kimchi-Schwartz, Jarrod R McClean, Jonathan Carter, Wibe A de Jong, and Irfan Siddiqi. Computation of molecular spectra on a quantum processor with an error-resilient algorithm. *Physical Review X*, 8(1):011021, 2018. DOI: 10.1103/PhysRevX.8.011021.

[77] William J Huggins, Joonho Lee, Unpil Baek, Bryan O'Gorman, and K Birgitta Whaley. A non-orthogonal variational quantum eigensolver. *New Journal of Physics*, 22(7):073009, 2020. DOI: 10.1088/1367-2630/ab867b.

[78] Adriano Barenco, Andre Berthiaume, David Deutsch, Artur Ekert, Richard Jozsa, and Chiara Macchiavello. Stabilization of quantum computations by symmetrization. *SIAM Journal on Computing*, 26(5):1541–1557, 1997. DOI: 10.1137/S0097539796302452.

[79] Harry Buhrman, Richard Cleve, John Watrous, and Ronald De Wolf. Quantum fingerprinting. *Physical Review Letters*, 87(16):167902, 2001. DOI: 10.1103/PhysRevLett.87.167902.

[80] Cornelius Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *Journal of Research of the National Bureau of Standards*, 50, 1950. DOI: 10.6028/jres.045.026.

[81] Tae Jun Park and JC Light. Unitary quantum time evolution by iterative lanczos reduction. *The Journal of chemical physics*, 85(10):5870–5876, 1986. DOI: 10.1063/1.451548.

[82] Jane K Cullum and Ralph A Willoughby. *Lanczos algorithms for large symmetric eigenvalue computations: Vol. I: Theory*. SIAM, 2002. DOI: 10.1137/1.9780898719192.

[83] Christopher Conway Paige. *The computation of eigenvalues and eigenvectors of very large sparse matrices*. PhD thesis, University of London, 1971.

[84] Yousef Saad. On the rates of convergence of the lanczos and the block-lanczos methods. *SIAM Journal on Numerical Analysis*, 17(5):687–706, 1980. DOI: 10.1137/0717059.

[85] Kazuhiro Seki and Seiji Yunoki. Quantum power method by a superposition of time-evolved states. *PRX Quantum*, 2(1):010333, 2021. DOI: 10.1103/PRXQuantum.2.010333.

[86] Tom Banks, Leonard Susskind, and John Kogut. Strong-coupling calculations of lattice gauge theories:(1+ 1)-dimensional exercises. *Physical Review D*, 13(4):1043, 1976. DOI: 10.1103/PhysRevD.13.1043.

[87] Leonard Susskind. Lattice fermions. *Physical Review D*, 16(10):3031, 1977. DOI: 10.1103/PhysRevD.16.3031.

[88] CJ Hamer, Zheng Weihong, and J Oitmaa. Series expansions for the massive model in hamil-

tonian lattice theory. *Physical Review D*, 56(1): 55, 1997. DOI: 10.1103/PhysRevD.56.55.

[89] Philipp Hauke, David Marcos, Marcello Dalmonte, and Peter Zoller. Quantum simulation of a lattice Schwinger model in a chain of trapped ions. *Physical Review X*, 3(4):041018, 2013. DOI: 10.1103/PhysRevX.3.041018.

[90] Yu Tong, Victor V Albert, Jarrod R McClean, John Preskill, and Yuan Su. Provably accurate simulation of gauge theories and bosonic systems. *Quantum*, 6:816, 2022. DOI: 10.22331/q-2022-09-22-816.

[91] Stefan Kühn, J Ignacio Cirac, and Mari-Carmen Bañuls. Quantum simulation of the Schwinger model: A study of feasibility. *Physical Review A*, 90(4):042305, 2014. DOI: 10.1103/PhysRevA.90.042305.

[92] Boye Buyens, Simone Montangero, Jutho Haegeman, Frank Verstraete, and Karel Van Acoleyen. Finite-representation approximation of lattice gauge theories at the continuum limit with tensor networks. *Physical Review D*, 95(9):094509, 2017. DOI: 10.1103/PhysRevD.95.094509.

[93] Lena Funcke, Karl Jansen, and Stefan Kühn. Topological vacuum structure of the schwinger model with matrix product states. *Physical Review D*, 101(5):054507, 2020. DOI: 10.1103/PhysRevD.101.054507.

[94] John M Martyn, Zane M Rossi, Andrew K Tan, and Isaac L Chuang. Grand unification of quantum algorithms. *PRX Quantum*, 2(4):040203, 2021. DOI: 10.1103/PRXQuantum.2.040203.

[95] Yuta Kikuchi, Conor Mc Keever, Luuk Coopmans, Michael Lubasch, and Marcello Benedetti. Realization of quantum signal processing on a noisy quantum computer. *npj Quantum Information*, 9(1):93, 2023. DOI: 10.1038/s41534-023-00762-0.

[96] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 193–204, 2019. DOI: 10.1145/3313276.3316366.

[97] Neil J. Ross and Peter Selinger. Optimal ancilla-free Clifford+T approximation of z-rotations. *Quantum Info. Comput.*, 16(11–12): 901–953, sep 2016. ISSN 1533-7146. DOI: 10.5555/3179330.3179331.

[98] Daniel Litinski. Magic state distillation: Not as costly as you think. *Quantum*, 3:205, 2019. DOI: 10.22331/q-2019-12-02-205.

[99] Roberto Oliveira and Barbara M. Terhal. The complexity of quantum spin systems on a two-dimensional square lattice. *Quantum Info. Comput.*, 8(10):900–924, nov 2008. ISSN 1533-7146. DOI: 10.5555/2016985.2016987.

[100] Yi-Kai Liu, Matthias Christandl, and Frank Verstraete. Quantum computational complexity of the N-representability problem: QMA complete. *Physical review letters*, 98(11):110503, 2007. DOI: 10.1103/PhysRevLett.98.110503.

[101] Norbert Schuch and Frank Verstraete. Computational complexity of interacting electrons and fundamental limitations of density functional theory. *Nature physics*, 5(10):732–735, 2009. DOI: 10.1038/nphys1370.

[102] Adam D. Bookatz. QMA-complete problems. *Quantum Info. Comput.*, 14(5 & 6): 361–383, apr 2014. ISSN 1533-7146. DOI: 10.5555/2638661.2638662.

[103] Tatiana A Bespalova and Oleksandr Kyriienko. Hamiltonian operator approximation for energy measurement and ground-state preparation. *PRX Quantum*, 2(3):030318, 2021. DOI: 10.1103/PRXQuantum.2.030318.

[104] Yizhi Shen, Katherine Klymko, James Sud, David B Williams-Young, Wibe A de Jong, and Norm M Tubman. Real-time krylov theory for quantum computing algorithms. *Quantum*, 7: 1066, 2023. DOI: 10.22331/q-2023-07-25-1066.

[105] Zongkang Zhang, Anbang Wang, Xiaosi Xu, and Ying Li. Measurement-efficient quantum krylov subspace diagonalisation. *arXiv preprint arXiv:2301.13353*, 2023. DOI: 10.48550/arXiv.2301.13353.

[106] Dominic W Berry, Craig Gidney, Mario Motta, Jarrod R McClean, and Ryan Babbush. Qubitization of arbitrary basis quantum chemistry leveraging sparsity and low rank factorization. *Quantum*, 3:208, 2019. DOI: 10.22331/q-2019-12-02-208.

[107] `https://doi.org/10.5281/zenodo.22558`.

[108] Lewis W Anderson, Martin Kiffner, Panagiotis Kl Barkoutsos, Ivano Tavernelli, Jason Crain, and Dieter Jaksch. Coarse-grained intermolecular interactions on quantum processors. *Physical Review A*, 105(6):062409, 2022. DOI: 10.1103/PhysRevA.105.062409.

[109] Sam E. Ganis. Notes on the Fibonacci sequence. *The American Mathematical Monthly*, 66(2):129–130, 1959. ISSN 00029890, 19300972. DOI: 10.2307/2310016.

[110] Yong He, Ming-Xing Luo, E Zhang, Hong-Ke Wang, and Xiao-Feng Wang. Decompositions of n-qubit Toffoli gates with linear circuit complexity. *International Journal of Theoretical Physics*, 56:2350–2361, 2017. DOI: 10.1007/s10773-017-3389-4.

[111] Farrokh Vatan and Colin Williams. Optimal quantum circuits for general two-qubit gates. *Physical Review A*, 69(3):032315, 2004. DOI: 10.1103/PhysRevA.69.032315.

[112] John A Smolin and David P DiVincenzo. Five two-bit quantum gates are sufficient to implement the quantum fredkin gate. *Physical Review A*, 53(4):2855, 1996. DOI: 10.1103/PhysRevA.53.2855.

[113] Jonathan M Baker, Casey Duckering, Alexander Hoover, and Frederic T Chong. Decomposing quantum generalized toffoli with an arbitrary number of ancilla. *arXiv preprint arXiv:1904.01671*, 2019. DOI: 10.48550/arXiv.1904.01671.

[114] Ben Zindorf and Sougato Bose. Efficient implementation of multi-controlled quantum gates. *arXiv preprint arXiv:2404.02279*, 2024. DOI: 10.48550/arXiv.2404.02279.

[115] Steven A Moses, Charles H Baldwin, Michael S Allman, R Ancona, L Ascarrunz, C Barnes, J Bartolotta, B Bjork, P Blanchard, M Bohn, et al. A race-track trapped-ion quantum processor. *Physical Review X*, 13(4):041052, 2023. DOI: 10.1103/PhysRevX.13.041052.

[116] Quantinuum system model h2: Emulator data sheet, version 1.1. `www.quantinuum.com/hardware/h2`, 2023.

[117] Jwo-Sy Chen, Erik Nielsen, Matthew Ebert, Volkan Inlek, Kenneth Wright, Vandiver Chaplin, Andrii Maksymov, Eduardo Páez, Amrit Poudel, Peter Maunz, et al. Benchmarking a trapped-ion quantum computer with 29 algorithmic qubits. *arXiv preprint arXiv:2308.05071*, 2023. DOI: 10.48550/arXiv.2308.05071.

[118] Ionq forte. `www.ionq.com/quantum-systems/forte`, 2022.

[119] IBM Kyiv processor, accessed 16/09/2023 via IBM Quantum experience.

[120] Alexis Morvan, B Villalonga, X Mi, S Mandra, A Bengtsson, PV Klimov, Z Chen, S Hong, C Erickson, IK Drozdov, et al. Phase transitions in random circuit sampling. *Nature*, 634 (8033):328–333, 2024. DOI: 10.1038/s41586-024-07998-6.

A Numerical experiments and fitting

A.1 Noiseless case

These results are shown in Fig. 8(a). We see that for each value of N, the fractional energy error achieved by the algorithm decreases exponentially with Krylov basis size. As seen in Fig.8(b), for the system sizes considered, our chosen initial state has reasonably large overlap with the true ground state. This overlap decreases linearly with system size N.

To estimate the basis size required to achieve a given accuracy we perform a fit of the form $\log \Delta E/E_{\text{int}} = \chi D + \lambda$ for each value of N, where $\Delta E/E_{\text{int}}$ is the fractional energy error, and D is the Krylov basis order. χ and λ are N dependent parameters that correspond to the gradient and intercept of the lines in Fig. 8 (a). For each value of N, we fit to data-points up to $D = 10$ ($D = 4$ in the case of $N = 4$ only) with the fitting parameters χ and λ shown in Figs. 8 (c) and (d) respectively. Using these fitting parameters, we are able to estimate the Krylov basis size required to compute the ground state energy within a desired precision, there is no noise due to measurement or numerical precision (and therefore no risk of instability in the generalised eigenvalue problem).

A.2 Using partitioned QSE with measurement noise

The best median error achieved by PQSE as a function of $\Pi_\varphi U$ calls is shown in Fig. 9 (a) for each value of N. Unlike for the noiseless case in which we saw an approximately log-linear relationship between the error and basis size, for a given N we observe a strong log-log dependence between the fractional energy error and the circuit cost (number of calls to $\Pi_\varphi U$). For each value of N, we fit to this data using a function of the form $\log \Delta E/E_{\text{int}} = \chi \log(\#\Pi_\varphi U \text{ calls}) + \lambda$. Fitting parameters χ and λ are shown as a function of N in Figs. 9 (b) and (c). Using these fitting parameters allows us to estimate the number of calls required to achieve a specified accuracy for a given value of N.

A.3 Using thresholded QSE with measurement noise

Here we present the results of using thresholded QSE (TQSE) algorithm as given by Ref. [68] for the same systems as in Appendix A.2. These are shown in Fig. 10 as a direct comparison with Fig. 4 in which partitioned QSE (PQSE) [67] was used. Noise was applied in the same way as in the main text and the threshold parameter was chosen to be equal to $||\mathbf{\Delta_S}||$ which has been found, in practice to be a good choice of thresholding parameter [68]. For input states used the expectation of the Hamiltonian was approximately equal to system size N (as is the true ground state energy), therefore we rescaled expectation values used in $\mathbf{S}$ and $\mathbf{H}$ as $\langle\psi_0| H^k |\psi_0\rangle \longrightarrow \langle\psi_0| H^k |\psi_0\rangle / N^k$, which ensured elements of $\mathbf{\Delta_S}$ and $\mathbf{\Delta_H}$ were of similar absolute values (as well as the same fractions of corresponding elements in $\mathbf{S}$ and $\mathbf{H}$).

For $N \gtrsim 24$ We can see that the TQSE performed over an order of magnitude worse than PQSE for the same number of calls to U. Furthermore, fitting and extrapolation to predict the number of calls to U to achieve a given measurement error shows more severe exponential scaling. Comparing the fit of Fig. 11 to that of Fig. 5 shows that the number of calls to U required to achieve fractional measurement error of 10^{-4} is $\sim 1.300^N$ for TQSE, compared to $\sim 1.143^N$ for PQSE.

B Preparing $|G\rangle$ and U using linear combination of unitaries

In this section, we will recap the procedure that Kirby et al. described within Appendix C. of Ref. [63] to prepare U and $|G\rangle$ for a local Hamiltonian. Later we will modify this procedure to reduce the resource cost for our Hamiltonian.

B.1 Implementing U

We need to implement U given by (18), for some choice of $\{|i\rangle\}$. Let

$$|i\rangle = |\vec{x}, \vec{z}\rangle\,, \tag{35}$$

where $(\vec{x}, \vec{z})$ is the binary representation of n qubit Pauli operator $\hat{P}_i$, defined as follows. For now we will assume that all $\{\alpha_i\}$ are positive in the definition of the Hamiltonian (17) and that no $\hat{P}_i$ have a negative sign. Kirby et al. consider applying the required signs to the Pauli operator for their Hamiltonian separately, we will show how to do this for an arbitrary model later.

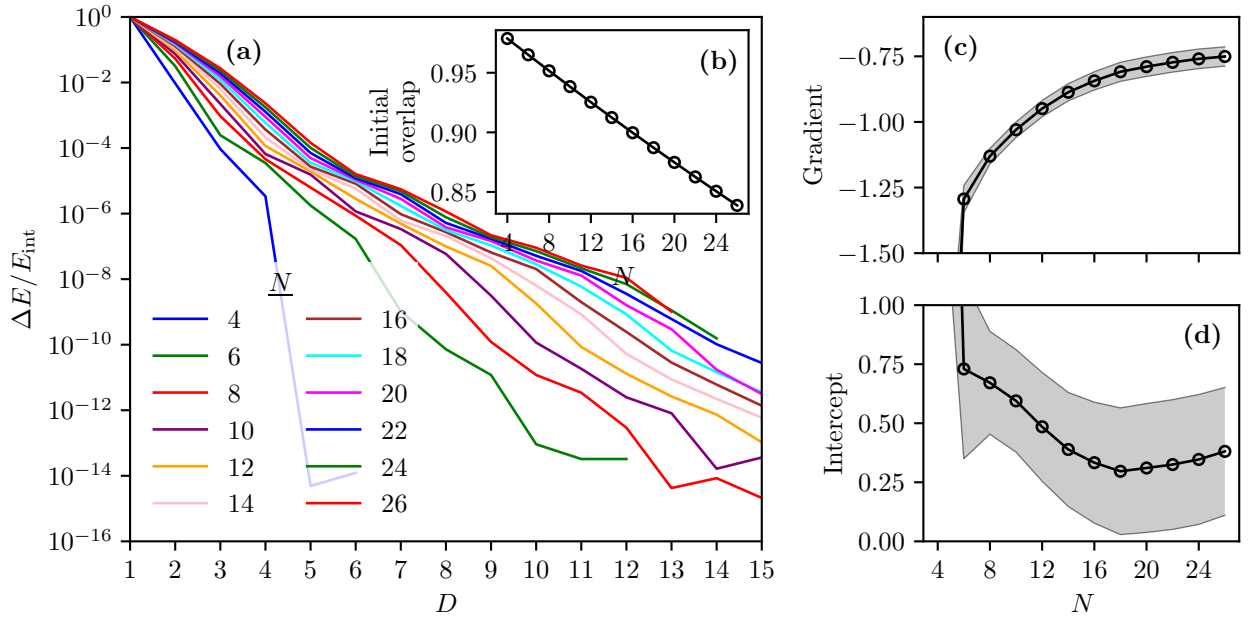


Figure 8: **(a)** Fractional energy error as a function of maximum Krylov order for different value of lattice size N for $\mu = 1.5$, $x = 5$. No measurement noise has been applied. **(b)** Overlap between initial state $|\psi_0\rangle$ and exact ground state. **(c)** & **(d)** Gradient and intercept for straight lines (on log-linear plot) fit to data in (a). Filled area indicates standard errors on estimates.

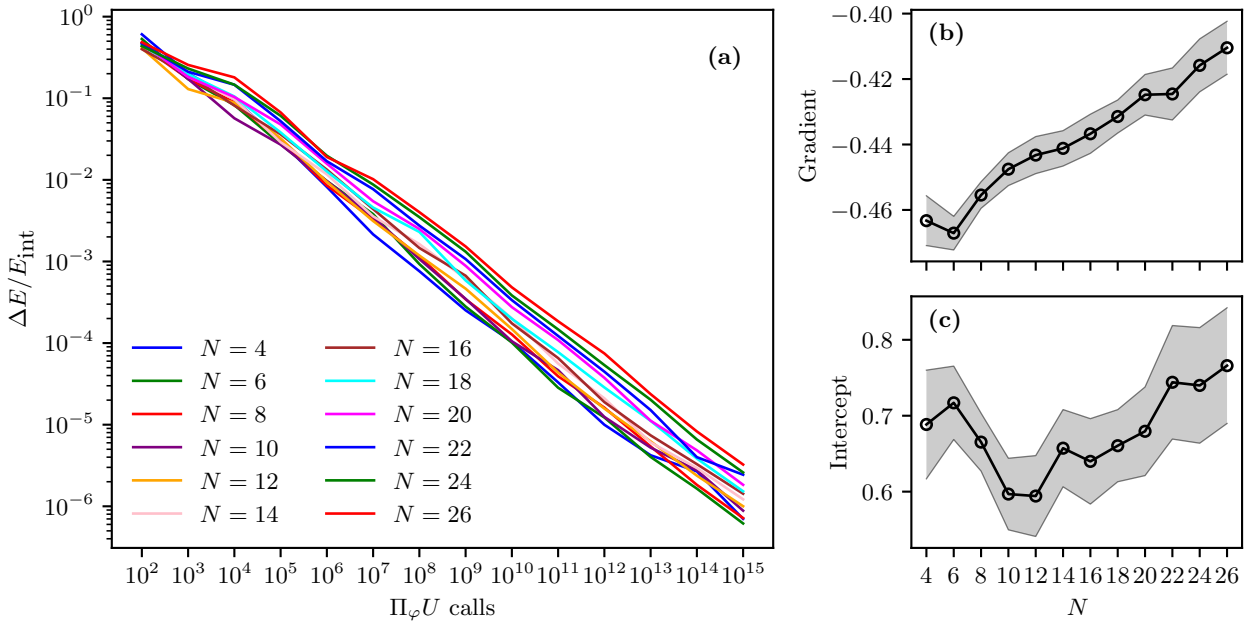


Figure 9: Median fractional energy error achieved in Fig. 4 for optimal choices of maximum Krylov order as a function of calls to $\Pi_\varphi U$. **(b)** & **(c)** Gradient and intercept for straight lines (on log-log plot) fit to data in (a). Filled area indicates standard errors on estimates.

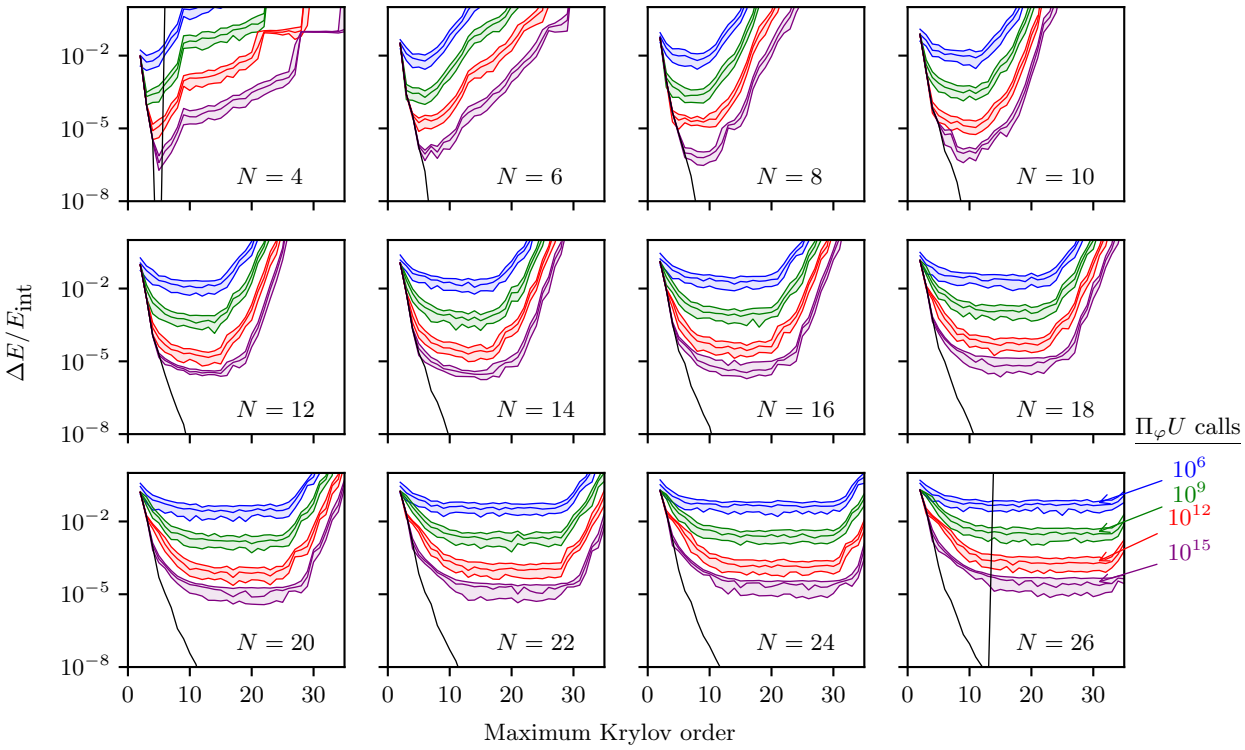


Figure 10: Fractional energy error using TQSE procedure with finite number of measurements for range of different lattice sizes N. Values are shown as a function of the Krylov basis size for the TQSE procedure and as a function of number of calls to the block encoding operator $\Pi_\varphi U$ (which affect the number of measurements as defined in the main text). Lines and shaded area corresponds to the median as well as upper and lower quartiles of energy error. The black line (labelled as infinite calls to $\Pi_\varphi U$) corresponds to no measurement noise.

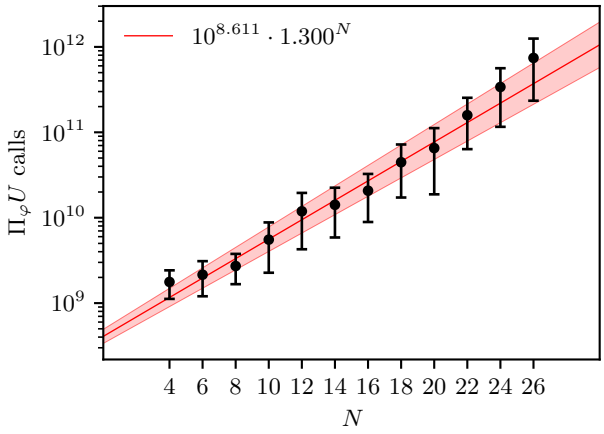


Figure 11: Estimated number of call to qubitization procedure with TQSE to achieve fractional energy error of 10^{-4} calculated using data from Fig. 10. Red line corresponds to exponential fit (straight line on log-log axes) with filled area indicating uncertainty in fitting and extrapolation.

Definition 6 (Binary representation of Pauli operator)**.** *Given a Pauli operator $\hat{P} \in \{\mathbb{1}, X, Y, Z\}^{\otimes n}$, the binary representation of $\hat{P}$ is a pair of n length binary vectors $(\vec{x}, \vec{z})$ such that a value 1 in position j of $\vec{x}$ means that $\hat{P}$ contains X acting on qubit j, and similarly for $\vec{z}$ and Z. This means $\hat{P}$ can be written as*

$$\hat{P}(\vec{x}, \vec{z}) = \bigotimes_{j=0}^{n-1} i^{x_j z_j} X^{x_j} Z^{z_j}. \tag{36}$$

Here, i is the imaginary unit which occurs when $\vec{x}$ and $\vec{z}$ contain 1 in the same position. Both X and Z acting on a qubit, along with the phase, gives Y operator since $ZX = iY$.

Using this representation, U can be implemented as

$$U = \sum_{\vec{x},\vec{z}} |\vec{x}, \vec{z}\rangle\langle\vec{x}, \vec{z}| \otimes \hat{P}(\vec{x}, \vec{z}) = \prod_{j=0}^{n-1} i^{x_j z_j} C_{x_j} X_j \cdot C_{z_j} Z_j, \tag{37}$$

which requires three layers of two qubit gates. Firstly, a layer of controlled gates that applies Z to each qubit j, controlled on the value of auxiliary (qubit z_j), a second layer doing the same for X, and then a layer that applies the two qubit controlled S gate

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & i \end{pmatrix} \tag{38}$$

between each pair of qubits (x_j, z_j).

Definition 7 (Hamming weight of a Pauli operator)**.** *Given a Pauli operator $\hat{P}$ with corresponding binary vectors $(\vec{x}, \vec{z})$ as defined in Def. 6. The Hamming weight of the operator $b(\hat{P})$ is equal to the total number of non-zero (i.e. one) elements of $\vec{x}$ and $\vec{z}$.*

The Hamming weight of each operator within the Hamiltonian will play a significant role in determining the gate cost of the qubitisation procedure.

B.2 Preparing $|G\rangle$

We must now prepare the block encoding state

$$|G\rangle = \sum_{i=0}^{T-1} \sqrt{|\alpha_i|}\, |(\vec{x}, \vec{z})_i\rangle_a\,, \tag{39}$$

which requires us to prepare a potentially arbitrary real superposition of basis states. This can be achieved using multi-controlled partial CNOT ($\mathrm{C}^n\mathrm{NOT}(\vartheta)$) gates as multi-controlled versions of the two qubit partial swap ($\mathrm{PSWAP}(\vartheta)$) gates. The action of the singly controlled partial NOT gate is

$$\mathrm{CNOT}(\vartheta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\vartheta & \sin\vartheta \\ 0 & 0 & \sin\vartheta & -\cos\vartheta \end{pmatrix}, \tag{40}$$

and the uncontrolled action of the PSWAP gate on two qubits is

$$\mathrm{PSWAP}(\vartheta) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\vartheta & -\sin\vartheta & 0 \\ 0 & \sin\vartheta & \cos\vartheta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{41}$$

for some rotation angle ϑ. We note two things, if we have a state containing basis states of Hamming weight (as defined in Def. 7) less than or equal to n, then a $\mathrm{C}^n\mathrm{NOT}(\vartheta)$ gate can rotate an n weight basis state (with 1 corresponding to every control qubit) into an $n+1$ weight basis state (with the additional Hamming weight coming from the target qubit) and will have no effect on lower weight states. Similarly, a $\mathrm{C}^n\mathrm{PSWAP}(\vartheta)$ will rotate between different states of Hamming weight $n+1$ (those with 1s in the controls and a 1 and a 0 in the two target qubits) and have no effect on lower weight states.

The following procedure can then be used to generate an arbitrary superposition of basis states.

1. Starting with zero state, apply X gate to flip a single qubit.
2. Use PSWAP(ϑ) gates, to distribute this single 1 between other qubits to generate all Hamming weight 1 states required.
3. Use a single CNOT(ϑ) gate to rotate one of the Hamming weight 1 states into a Hamming weight 2 state.
4. Use CPSWAP(ϑ) gates to distribute this Hamming weight 2 state into all Hamming weight 2 states required (Hamming weight 1 states will be unaffected).
5. Repeat steps 3 and 4, increasing the number of controls and the Hamming weight by 1 each time, until all states up to the maximum Hamming weight have been produced.

The choice of ϑ for each gate must be chosen according to the values of α_i required, noting that larger amplitudes than $\sqrt{\alpha_i}$ will be required from some rotations for basis states that will have C^{n}NOT gates applied to access the higher weight basis states.

B.3 Applying signs to $\hat{P}_i$'s and preparing $|\tilde{G}\rangle$

To prepare the state

$$|\tilde{G}\rangle = \sum_{i=0}^{T-1} \mathrm{sgn}(\alpha_i)\sqrt{|\alpha_i|}\,|(\vec{x},\vec{z})_i\rangle\,, \tag{42}$$

we can use the same procedure and rotation gates as for $|G\rangle$. The only change required is to flip the signs of some of the rotation angles for some of the partial SWAP and partial CNOT gates. By flipping the sign of the rotation angle $\vartheta \to -\vartheta$ for these gates, we induce a -1 phase shift in the resulting basis state compared to the one used in its generation. This approach ensures the incorporation of the requisite negative signs. It is crucial to monitor the sign evolution of successive bitstrings throughout this non-Markovian process. The sign of a basis state being rotated out may itself be positive or negative, contingent upon the preceding rotations and potential bitflips utilised in its generation.

C Gate costs for block encoding Schwinger model

C.1 Sketch of proofs for gate costs

We present some mathematical lemmas that will be useful for our costing in Appendix C.2. To compute the resource costs that we desire, we begin by calculating the Pauli decomposition for each term within the Schwinger model Hamiltonian. This is done in Appendix C.3. Since the gauge fields are bosonic fields these Pauli decompositions take similar forms to the operators studied in other works involving digital simulation of bosonic systems [108]. For each term in the Pauli decomposition we compute the Hamming weight of the bit string used to encode the operator within the LCU qubitization procedure.

C.1.1 Calculating the gate cost of U

Calculating an upper bound for the cost of U consists of two parts. The first set of two qubit gates to apply the controlled X, controlled Z and controlled S (i phase) gates required to apply Pauli operators $\hat{P}_i$. For the CX and CZ gates, there is one of these for each qubit within the corresponding $\vec{x}$ and $\vec{z}$ auxiliary register. The controlled S gates are each controlled on qubit within the $\vec{x}$ register, targeting the corresponding qubit within the z register. Thus the total cost of this step is just $\sim 3mN$ two qubit gates. A full proof of this cost is given in Theorem 32

We also give, in Theorem 34, the cost of U if, as described by Kirby et al. [63], we apply the phases of α_i as part of it, rather than by using an asymmetric block encoding and applying phases within $\tilde{G}$. Applying phases within U gives a gate cost that is $\mathcal{O}(N^2\log(N))$ so we achieve a significant saving by applying phases within $\tilde{G}$.

C.1.2 Calculating the gate cost of G and $\tilde{G}$

As described by Kirby et al. [63], $|G\rangle$ can be generated by applying partial rotations between different computational basis states of the $|G\rangle$ register where the different states rotated into encode the binary encoding of each Pauli decomposition with the Hamiltonian. This cost is calculated in Appendix C.4 by computing sums for the number of terms with a given Hamming weight, using the previously computed Pauli decompositions, and therefore the number of multicontrolled SWAP and partial CNOT gates required. Having computed a

decomposition of the multi-qubit gates into T-gates, CNOTs and R_z gates (Lemma 26), we can then calculate the total number of non-Clifford (T and rotation gates) along with two qubit CNOT gates.

We are able to reduce the cost of G compared to the method described by Kirby et al. [63] by exploiting the symmetry within the Hamiltonian. Noticing that gauge field and interaction terms within the Hamiltonian are the same for each gauge-link site, rather than rotating between the binary states required for all lattice sites, we perform the costly rotations into basis states $|\vec{x}, \vec{z}\rangle$ for just a single link site and then perform a less costly partial swap of all terms into the other sites. This method is described in Appendix C.4.6 and the total gate cost for G, using this method is presented in Theorem 27.

To convert the single qubit R_z gate counts into the more relevant T-gates, we make use of the approximate decomposition of R_z given by Ross et al. [97]. This decomposition requires a number of T-gates per rotation gate that scales logarithmically with the required rotation precision (see Proposition 29). Assuming that required rotation error scales inversely with the average values of the amplitudes of each Pauli decomposition (and therefore the number of terms in the Hamiltonian) allows us to estimate the number of T-gates required for the rotations within G. For further details see Appendix C.5.

If we use m qubits for each gauge link field (such that they have maximum local occupancy of $\Lambda = 2^{m-1}$, the number of gates required scales as $\mathcal{O}(m2^m + Nm)$. Recalling from Remark 1 that the maximum gauge field value is equal to the number of lattice sites, the maximum value of Λ required is N and this gate cost is $\mathcal{O}(\log(N)N)$. Without using our method to reduce the cost through by performing this partial swap, the gate cost of G would be $\mathcal{O}(\log(N)N^2)$.

We note in Corollary 37 that $\tilde{G}$ can be implemented with the same rotation gates as G except flipping the sign of some of the angles. Thus, the cost for $\tilde{G}$ is the same as the cost for G.

C.1.3 Total resource cost of algorithm

Now that we have calculated the gate costs for G and $\tilde{G}$, it is easy to calculate the gate costs of corresponding rotations Π_φ and $\tilde{\Pi}_\varphi$. This is given in Theorem 35 and is twice the cost of G with an additional number $\mathcal{O}(mN)$ number of CNOT and T-gates. Combining these with the cost of U allows us to calculate the gate cost to perform a single measurement of $\langle\psi_0| H^k |\psi_0\rangle$. The cost of this, as described in Theorem 38, is $\mathcal{O}(k \log(N)N)$.

C.2 Some useful mathematics

Before we evaluate the quantum resource required for the block encoding, we prove a few useful lemmas.

Lemma 8. *Given positive integer b,*

$$\sum_{w_y=0}^{\lfloor \frac{b}{2} \rfloor} \binom{b - w_y}{w_y} = F(b+1), \tag{43}$$

where $F(n)$ is the nth Fibonacci number.

Proof. See Ref. [109]. □

Lemma 9. *Defining $F(n)$ to be the nth Fibonacci number, then for any positive integer m,*

$$\sum_{b=2}^{m+2} F(b+1) = \frac{1}{\sqrt{5}} \left(\phi^{m+5} - (-\phi)^{-m-5}\right) - 3. \tag{44}$$

where ϕ is the golden ratio.

Proof. From the definition of the Fibonacci numbers

$$\begin{aligned} F(m+3) &= F(m+5) - F(m+4) \\ F(m+2) &= F(m+4) - F(m+3) \\ &\vdots \\ F(4) &= F(6) - F(5) \\ F(3) &= F(5) - F(4). \end{aligned} \tag{45}$$

Summing the L.H.S. and R.H.S. of the above yields

$$\sum_{b=2}^{m+2} F(b+1) = F(m+5) - F(4). \tag{46}$$

Using Binet's formula and $F(4) = 3$ yields the result. □

Lemma 10. *For any integer positive integer m,*

$$\sum_{b=2}^{m+2} bF(b+1) = \frac{m+2}{\sqrt{5}}\left(\phi^{m+5} - (-\phi)^{-m-5}\right) - \frac{1}{\sqrt{5}}\left(\phi^{m+6} - (-\phi)^{-m-6}\right) + 2. \tag{47}$$

Proof. From the definition of the Fibonacci numbers

$$\begin{aligned}
(m+2)F(m+3) =& (m+2)F(m+5) - (m+2)F(m+4)\\
(m+1)F(m+2) =& (m+1)F(m+4) - (m+1)F(m+3)\\
&\vdots\\
3F(4) =& 3F(6) - 3F(5)\\
2F(3) =& 2F(5) - 2F(4).
\end{aligned} \tag{48}$$

Summing the L.H.S. and R.H.S. of the above equations yields

$$\begin{aligned}
\sum_{b=2}^{m+2} bF(b+1) &= (m+2)F(m+5) - \sum_{b=4}^{m+3} F(b+1) - 2F(4)\\
&= (m+2)F(m+5) - F(m+6) + F(3),
\end{aligned} \tag{49}$$

where we have rewritten the sum in the second line using Lemma 9. Using Binet's formula and $F(3) = 2$ yields the result. □

Corollary 11.

$$\begin{aligned}
\sum_{b=2}^{m+2} F(b+1)(\alpha b + \beta) =& \frac{\alpha}{\sqrt{5}}\left(\phi^{m+5} - (-\phi)^{-m-5}\right) m - \frac{\alpha}{\sqrt{5}}\left(\phi^{m+6} - (-\phi)^{-m-6}\right)\\
&+ \frac{(2\alpha+\beta)}{\sqrt{5}}\left(\phi^{m+5} - (-\phi)^{-m-5}\right) + (2\alpha - 3\beta).
\end{aligned} \tag{50}$$

Proof. This follows from Lemmas 9 and 10. □

Proposition 12.

$$\sum_{b=0}^{m} \binom{m}{b}(\alpha b + \beta) = \alpha m 2^{m-1} + \beta 2^m. \tag{51}$$

Proof. This follows from standard results for the summation of binomial coefficients. □

C.3 Pauli decomposition of the Hamiltonian

In order to calculate the resources needed for the qubitization procedure, we begin by calculating the Pauli operators needed to construct the Hamiltonian.

Definition 13 (Weight of a Pauli operator)**.** *Given an operator $\hat{P}$ which can be written as single tensor product of the form $\{\mathbb{1}, X, Y, Z\}^{\otimes m}$, we define the weight $W(\hat{P})$ of $\hat{P}$ as the total number of non-identity operators within $\hat{P}$.*

Definition 14 (X,Y and Z weight of a Pauli operator)**.** *For an operator $\hat{P}$ as above, we define the X weight of an operator as the number of X operators in its Pauli decomposition. Similarly for the Y weight and Z weight. If $W(\hat{P})$ is the total Pauli weight of an operator, then $W_x(\hat{P}) + W_y(\hat{P}) + W_z(\hat{P}) = W(\hat{P})$, where $W_{x,y,z}(\hat{P})$ are the X,Y and Z weights respectively.*

Remark 15. *Given an operator $\hat{P}$ with X,Y,Z Pauli weights $W_x(\hat{P}), W_y(\hat{P}), W_z(\hat{P})$. The Hamming weight of the binary representation vector $(\vec{x}, \vec{y})$ of $\hat{P}$ as defined in Def. 7 is equal to $b(\hat{P}) = W_x(\hat{P}) + 2W_y(\hat{P}) + W_z(\hat{P})$.*

To analyse the resource requirements of the qubitization procedure, we will need to decompose each term in the Hamiltonian into a basis of Pauli operators and calculate the number of constituent operators of a given weight.

C.3.1 Gauge field terms

Lemma 16. *Consider the truncated gauge-field operator $L^2 = \sum_{l=0}^{\Lambda-1} (l - \Lambda/2)^2 \, |\underline{l}\rangle\langle\underline{l}|$ where Λ is a power of 2. If we write the operator in the Pauli basis $\{\mathbb{1}, X, Y, Z\}^{\otimes m}$, where $m = \log_2(\Lambda)$, then for every integer $0 \leq w < m$, the number of multi-qubit Pauli operators of weight w is less than or equal to the binomial coefficient $\binom{m}{w}$.*

Proof. We note the single qubit Pauli operators

$$\begin{aligned} |0\rangle\langle 0| &= \frac{1}{2}(\mathbb{1} + Z), \qquad & |1\rangle\langle 1| &= \frac{1}{2}(\mathbb{1} - Z), \\ |1\rangle\langle 0| &= \frac{1}{2}(X - iY), \qquad & |0\rangle\langle 1| &= \frac{1}{2}(X + iY). \end{aligned} \tag{52}$$

Using (52), we can rewrite this operator as

$$\sum_{l=0}^{\Lambda-1} (l - \Lambda/2)^2 \, |\underline{l}\rangle\langle\underline{l}| = \sum_{l=0}^{\Lambda-1} (l - \Lambda/2)^2 \bigotimes_{j=0}^{m-1} (\mathbb{1} + (-1)^{\mathrm{bin}(l)_j} Z), \tag{53}$$

where $\mathrm{bin}(l)_j \in \{0, 1\}$ denotes the jth bit in the binary representation of integer l. For a given l, expand the number the m qubit tensor product yields all m qubit operators in $\{\mathbb{1}, Z\}^{\otimes m}$ with coefficients ± 1, the number of unique terms containing a total of W Pauli Z operators is given by the binomial coefficient $\binom{m}{w}$. Every value of l yields operators in $\{\mathbb{1}, Z\}^{\otimes m}$, just with a different set of prefactors. Thus, the total number of unique Pauli operators of W will be at most $\binom{m}{w}$. It may be the case that, depending on the values of Λ and therefore $(l - \Lambda/2)$, some of these term cancel, however the binomial coefficient provides and upper limit on the number of terms. □

Remark 17. *We note that the only Pauli operators within the above are Z operators, there are no X or Y operators.*

C.3.2 Interaction terms

Lemma 18. *Consider the truncated raising operator $e^{-i\hat{\theta}} = \sum_{l=0}^{\Lambda} |\underline{l+1}\rangle\langle\underline{l}|$ where Λ is a power of 2 and cyclic numbering is used such that $|\underline{\Lambda}\rangle = |\underline{0}\rangle$. If we write the raising operator in the Pauli basis $\{\mathbb{1}, X, Y, Z\}^{\otimes m}$, where $m = \log_2(\Lambda)$, then for every integer $0 \leq w \leq m$, the number of multi-qubit Pauli operators of weight w present is equal to 2^w.*

Proof. As noted in Theorem 1 of Ref. [108], the Pauli decomposition of a bosonic ladder operator can be written as

$$|\underline{l+1}\rangle\langle\underline{l}| = \bigotimes_{j=k(l)+1}^{m} \left(\mathbb{1} + (-1)^{\mathrm{bin}(l)_j} Z\right) \otimes (X - iY) \otimes (X + iY)^{\otimes k(l)}, \tag{54}$$

where $\mathrm{bin}(l)_j \in \{0, 1\}$ denotes the jth bit in the binary representation of integer l and

$$k(l) := \max\{j \mid \mathrm{bin}(l)_j = 0\} \tag{55}$$

is the position of the right-most zero within that representation. We say $k(l) = m$, if l contains no zeros (i.e., $l = \Lambda - 1$). Therefore, we can write the full raising operator as

$$\sum_{l=0}^{\Lambda-1} |\underline{l+1}\rangle\langle\underline{l}| = \sum_{l=0}^{\Lambda-1} \bigotimes_{j=k(l)+1}^{m} \left(\mathbb{1} + (-1)^{\mathrm{bin}(l)_j} Z\right) \otimes (X + iY) \otimes (X - iY)^{\otimes k(l)}, \tag{56}$$

which, by rewriting the summations, can be re-expressed as

$$\sum_{\{l|k(l)=k'\}} |\underline{l+1}\rangle\langle\underline{l}| = \sum_{k'=0}^{m} \left[\sum_{\{l|k(n)=k'\}} \bigotimes_{j=k'+1}^{m} \left(\mathbb{1} + (-1)^{\mathrm{bin}(l)_j} Z \right) \right] \otimes (X+iY) \otimes (X-iY)^{\otimes k'}. \tag{57}$$

If we consider first the term within the square braces, we note that

$$\sum_{\{l|k(l)=k'\}} \bigotimes_{j=k'+1}^{m} \left(\mathbb{1} + (-1)^{\mathrm{bin}(l)_j} Z \right) = (2\mathbb{1})^{\otimes(m-k'-1)}. \tag{58}$$

To see this is the case, we use induction. Firstly, for $m = k'+1$, we note that there are only four terms (acting on a single qubit) within the summation, these correspond to $n = \Lambda$ and $n = \Lambda+1 = 0$. The summation is thus equal to $(\mathbb{1}+Z)+(\mathbb{1}-Z) = 2\mathbb{1}$. When $m = k'+2$, there are two terms within the summation, each acting on two qubits. These are equal to

$$\begin{aligned} &(\mathbb{1}+Z)\otimes(\mathbb{1}+Z) + (\mathbb{1}+Z)\otimes(\mathbb{1}-Z) + (\mathbb{1}-Z)\otimes(\mathbb{1}+Z) + (\mathbb{1}-Z)\otimes(\mathbb{1}-Z) \\ &= \big[(\mathbb{1}+Z)+(\mathbb{1}-Z)\big] \otimes \big[(\mathbb{1}+Z)+(\mathbb{1}-Z)\big] \\ &= 2\mathbb{1}\otimes 2\mathbb{1}. \end{aligned} \tag{59}$$

From this, we can see how (58) holds by induction. Therefore,

$$\sum_{\{l|k(l)=k'\}} |\underline{l+1}\rangle\langle\underline{l}| = \sum_{k'=0}^{m} (2\mathbb{1})^{\otimes(m-k'-1)} \otimes (X+iY) \otimes (X-iY)^{\otimes k'}. \tag{60}$$

Looking at the $(X+iY)\otimes(X-iY)^{\otimes k'}$ term, expanding these brackets will yield $2^{k'+1}$ terms in $\{X,Y\}^{\otimes k'+1}$, which all have weight $k'+1$. There will be $(m+1)$ unique values of $k' = 0, 1, \ldots, m$, each of which contribute $2^{k'+1}$ terms of weight $k'+1$, hence the result. □

Remark 19. *From the above proof, we can also see that if we consider all multi-qubit Pauli operators of a given w within the above expansion, the number of operators that contain a total of w_y Y operators (or equivalently X operators) is $\binom{w}{w_y}$. There are no Z operators remaining in the expansion.*

Corollary 20. *Consider a single interaction term (for a certain n) given by (13) using the truncated raising and lowering operators $e^{-i\hat{\theta}} = \sum_{l=0}^{\Lambda} |\underline{l+1}\rangle\langle\underline{l}|$ and $e^{i\hat{\theta}} = \sum_{l=0}^{\Lambda} |\underline{l}\rangle\langle\underline{l+1}|$. Decomposing the interaction term into Pauli operators onto a qubit register of size $(m+2)$, such that $m = \log_2(\Lambda)$ qubits are used for the gauge field space, and the remaining two qubits for the two spin degrees of freedom, then for every integer w satisfying $2 \leq w \leq m+2$, there will be 2^{w-1} Pauli operators of weight w. These operators will always contain an even number w_y of Pauli Y operators, for a given w the number of operators with w_y Pauli Y operators will be $\binom{w}{w_y}$.*

Proof. Decomposing the spin raising and lower operators within a single interaction term, we get

$$\begin{aligned} \sigma^+ e^{i\theta}\sigma^- + \text{h.c.} =& \frac{1}{4}\big[(X+iY)e^{i\theta}(X-iY) + \text{h.c.}\big] \\ =& \frac{1}{4}\big[Xe^{i\theta}X + iXe^{i\theta}Y - iYe^{i\theta}X + Ye^{i\theta}Y + \text{h.c.}\big]. \end{aligned} \tag{61}$$

We can see from (60), that half of the terms within the Pauli expansion of $e^{i\theta}$ are Hermitian and half are anti-Hermitian. Therefore, exactly half the terms within the expansion $Xe^{i\theta}X + iXe^{i\theta}Y - iYe^{i\theta}X - Ye^{i\theta}Y$ will be Hermitian and anti-Hermitian. Thus, in adding the Hermitian conjugate to get the full interaction term in (61), half of the terms (the anti-hermitian ones) will cancel, and the other half (the Hermitian ones) will be repeated. We note that anti-hermitian terms, and therefore those that cancel, always contain an odd number of Pauli Y operators, since in the definition of the raising operators these contain the imaginary prefactors.

Counting up the number of terms in (61), multiplying by the weights of the operator in the Pauli decomposition of $e^{i\theta}$ and taking into account the cancellation of half the terms gives the required result. □

C.3.3 Spin terms

Remark 21. *It is trivial to see that each the Pauli decomposition for an onsite spin term $(-1)^n\left[\mu + \mu\sigma^3(n)\right]/2$ consists of a single Z operator only.*

C.4 Generating the block encoding state $|G\rangle$

The circuit cost of generating $|G\rangle$ using the method described in Appendix B is entirely determined by the number of operators and the weight of each operator within the Pauli decomposition of the Hamiltonian. Here we will estimate the total number of gates needed to generate G for the 1+1D Schwinger Hamiltonian. We will start by calculating the number of vectors of each Hamming weight required to encode the Pauli decomposition of the Hamiltonian. We will consider gauge field terms, spin terms and interaction terms separately.

We will separate $|G\rangle$ into parts corresponding to each term in the Hamiltonian:

$$|G\rangle = \sum_{n=1}^{N-1} |G^{\text{field}}(n)\rangle + \sum_{n=1}^{N} |G^{\text{spin}}(n)\rangle + \sum_{n=1}^{N-1} |G^{\text{int}}(n)\rangle\,, \tag{62}$$

where $|G^{\text{field}}(n)\rangle$ is an unnormalised state containing the basis states corresponding to the binary representation of the Pauli decomposition, with appropriate amplitudes, for the Hamiltonian term for the energy of gauge field n. The definitions of $|G^{\text{int}}(n)\rangle$ and $|G^{\text{spin}}(n)\rangle$ follow similarly.

C.4.1 Binary representation of gauge field terms $|G^{\text{field}}(n)\rangle$

Consider a single term corresponding to the gauge field energy for a given site n in (12), for a given integer $0 \leq w_z \leq m$, there will be $\binom{m}{w_z}$ Pauli operators of Pauli weight $W(\hat{P}) = W_z(\hat{P}) = w_z$. This follows from Lemma 16 and Remark 17. Thus, for all $0 < w_z \leq m$, the gauge field term for a given lattice site n in the Hamiltonian will contribute a total of $\binom{m}{w_z}$ Pauli operators of Pauli weight w_z, which from Remark 15, will require binary representation vector of Hamming weight $b = w_z$. Therefore, we write the number of terms corresponding to Hamming weight b due to the fields as

$$N_b^{\text{field}} = \begin{cases} \binom{m}{b} & \text{for } 0 \leq b \leq m \\ 0 & \text{otherwise.} \end{cases} \tag{63}$$

C.4.2 Binary representation of spin terms $|G^{\text{spin}}(n)\rangle$

Next, consider a single term corresponding to the onsite energy of a single spin n. The Pauli decomposition of this operator will involve a single qubit Z operator with Pauli weight $W(\hat{P}) = W_z(\hat{P}) = 1$. Thus the N onsite spin energy terms will lead to N operators with binary representation vector of hamming weight one. Therefore, we write the number of terms corresponding to Hamming weight b due to the spins as

$$N_b^{\text{spin}} = \begin{cases} 1 & \text{for } b = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{64}$$

C.4.3 Binary representation of interaction terms $|G^{\text{int}}(n)\rangle$

Finally, consider a single term corresponding to an interaction term for a given n. We define $m' := m+2$. From Corollary 20, for a given $2 \leq w \leq m'$, there will be $\binom{w}{w_y}$ bitstrings of Pauli weight $w_y \leq w$ if w_y is even and zero otherwise. The Hamming weight of the associated binary representations will be $b = w + w_y$, which can therefore can take any value between 2 and $2m'$. The number of terms with Hamming weight b will be given by the following summation

$$N_b^{\text{int}} = \sum_{\substack{2\leq w\leq m' \\ 0\leq w_y\leq m' \\ \text{s.t. } b=w+w_y \\ w_y \text{ even}}} \binom{w}{w_y} = \sum_{\substack{w_y=\max(0,b-m') \\ w_y \text{ even}}}^{\lfloor b/2\rfloor} \binom{b-w_y}{w_y}, \tag{65}$$

where we arrive at the final line by substituting $w = b - w_y$ and simplifying the ranges on the summations appropriately. We can find an upper bound for this as follows:

$$N_b^{\text{int}} = \sum_{\substack{w_y=\max(0,b-m') \\ w_y \text{ even}}}^{\lfloor b/2\rfloor} \binom{b-w_y}{w_y} \leq \sum_{\substack{w_y=0 \\ w_y \text{ even}}}^{\lfloor b/2\rfloor} \binom{b-w_y}{w_y} \leq F(b+1), \tag{66}$$

where we have used the result for the summation from Lemma 8. Therefore,

$$N_b^{\text{int}} \leq \begin{cases} F(b+1) & 2 < b \leq 2m' \\ 0 & \text{otherwise.} \end{cases} \tag{67}$$

C.4.4 Gate cost of associated C^nPSWAPs and C^nNOTs

Now that we have the number of bitstrings for each Hamming weight, we can begin to calculate the gate cost of generating $|G\rangle = \sum_i \sqrt{\alpha_i}\,|i\rangle$. To do so, we first break down the cost of the multiqubit $C^n\text{NOT}(\vartheta)$ and $C^n\text{PSWAP}(\theta)$ in the following two remarks, before multiplying these by the total number that must be applied according to the number of binary representations required with each Hamming weight.

Remark 22 (n qubit Toffoli gate with one ancilla [110])**.** *A single* C^{n-1}NOT*, where* $n \geq 3$*, can be implemented using* $32n-96$ *T-gates,* $24n-72$ CNOT *gates and one recyclable ancilla qubit. A* $C^{n-1}Z$ *gate can be implemented with the same cost plus two additional Hadamard gates.*

Remark 23 (n qubit Toffoli gate with mulitple ancillae [110])**.** *A single* C^{n-1}NOT*, where* $n \geq 3$*, can be implemented using* $4n-8$ *T-gates,* $4n-7$ CNOT *gates and* $n-1$ *ancilla qubits. A* $C^{n-1}Z$ *gate can be implemented with the same cost plus two additional Hadamard gates.*

Remark 24 (Creating multi-controlled PSWAP and partial CNOT gates)**.** *Following the logic of Appendix C3 of Ref. [63],* $C^{n-1}\text{PSWAP}(\vartheta)$ *and* $C^n\text{NOT}(\vartheta)$ *gates can be implemented using the following gate sequences:*

$$C^{n-1}_{\{j\}}\text{PSWAP}_{k,l}(\vartheta) = e^{\frac{i\vartheta\hat{\mu}_{k,l}}{2}} \cdot C^{n-1}_{\{j\}} Z_k \cdot e^{\frac{-i\vartheta\hat{\mu}_{k,l}}{2}} \cdot C^{n-1}_{\{j\}} Z_k, \tag{68}$$

where $C^{n-1}_{\{j\}}$ *denotes controlled operator on a set of* $n-1$ *qubits* $\{j\}$ *and the operator* $\hat{\mu}_{k,l}$*, which generates the two qubit* $\text{PSWAP}_{k,l}(\vartheta)$ *gate, is defined by*

$$\hat{\mu}_{k,l} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & -i & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{69}$$

and

$$C^{n-1}_{\{j\}}\text{CNOT}_{k,l}(\vartheta) = e^{\frac{i\vartheta\hat{\nu}_{k,l}}{2}} \cdot C^{n-1}_{\{j\}} Z_k \cdot e^{\frac{-i\vartheta\hat{\nu}_{k,l}}{2}} \cdot C^{n-1}_{\{j\}} Z_k \cdot C^{n-1}_{\{j\}} S^\dagger_k, \tag{70}$$

where $\nu_{k,l}$*, which generates a two qubit* $\text{CNOT}(\vartheta)_{k,l}$ *gate, is defined by*

$$\hat{\nu}_{k,l} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}. \tag{71}$$

As discussed by Kirby et al. [63], the application of the multi-controlled $S^\dagger$ *gate can be included in the decomposition of the multi-controlled* Z *gate so that it contributes none of it's two-qubit or non-Clifford gates.*

Remark 25. *The two qubit gates* $\text{CNOT}(\vartheta)$ *and* $\text{PSWAP}(\vartheta)$ *can be constructed with two* CNOT*s, six single qubit rotations (i.e.* R_z *gates) and single qubit Clifford gates.*

This follows from the construction by Vatan and Williams ([111], Fig. 2) of a general SO(4) gate and counting the number of non-Clifford and CNOT gates required.

Lemma 26 (Implementing partial swap and partial CNOT gates.)**.** *A* $C^{n-1}\text{PSWAP}(\vartheta)$ *gate and a* $C^n\text{NOT}(\vartheta)$ *can be implemented using* $8n-16$ *T-gates,* $8n-14$ CNOT *gates,* 12 R_Z *gates and single qubit Clifford gates, provided we have access to* $n-1$ *ancilla qubits.*

Proof. This follows from Remarks 23, 24 and 25. □

C.4.5 Counting the number of C^nPSWAPs, C^nNOTs and $C^n Z$s

Following the procedure as laid out in Sec. B.2, we must apply a $C^n\text{NOT}(\vartheta)$ once for each value of possible Hamming weight (each time we encounter Step 3) up to the maximum possible values $b_{\max} = 2m+4$. We note that a single $C^n\text{PSWAP}(\vartheta)$ gate can only rotate between states that have same Hamming weight and differ in the position of only a single one bit. Thus, if we just apply a single $C^n\text{NOT}(\vartheta)$ each time we apply Step 3, say to access a Hamming weight $n+1$ state from a Hamming weight n state, then it is likely that more than one of the 1s within this new state do not match up with the position of one of the 1s in one of the weight $n+1$ states we need to rotate into with the $C^n\text{PSWAP}(\vartheta)$ gate. This would require us to apply more

than one $\mathrm{C}^n\mathrm{PSWAP}(\vartheta)$ per unique value of $(\vec{x}, \vec{z})$ gate of Hamming weight $n+1$. However, if we examine the form of the Pauli decompositions for the spin (which is trivial), field (53) and interaction terms (60) for single lattice site n, we can exploit the structure of the binary representation strings to ensure we only need to apply at most one $\mathrm{C}^n\mathrm{PSWAP}(\vartheta)$ for each unique value of $(\vec{x}, \vec{z})$, at the expense of a small number of $\mathrm{C}^n\mathrm{NOT}(\vartheta)$ gates.

Additionally, we will reduce the overall cost of the algorithm by only using the procedure of [63] (described in Section B.2 using $\mathrm{C}^n\mathrm{NOT}(\vartheta)$ and $\mathrm{C}^n\mathrm{PSWAP}(\vartheta)$) gates to generate the terms of $|G\rangle$ corresponding to gauge field and interaction energy of a single gauge link (and all fermion only terms). We will then generate the terms related to the other $N-2$ gauge sites using a procedure described in Section C.4.6.

For the gauge field energy of a single site, the binary representation vectors (considering only the m bits corresponding to the gauge field qubits of each auxiliary register) take all possible values of the form $\vec{x} = 0^m, \vec{z} \in \{0,1\}^m$, where 0^m corresponds to a zero bit repeated m times. Thus, for each unique Hamming weight $0 < b \leq m$, there is at least one binary representation vector present that differs from another of weight $b-1$ by just a single bit. Thus, for the gauge field terms, one must apply $\mathrm{C}^{b-1}\mathrm{NOT}(\vartheta)$ gate for each value of $1 \leq b \leq m$, where we have multiplied by the total number of gauge field registers, to generate the appropriate Hamming weights. This is then combined with $\left[\binom{m}{b} - 1\right]$ $\mathrm{C}^{b-2}\mathrm{PSWAP}(\vartheta)$ gates for each $1 \leq b \leq m$ to access all bitstrings of that Hamming weight. Here, the binomial coefficient arises from the number of weight b strings given by (63).

For the interaction terms of a single gauge link site, we can see from corollary 20 that the binary representation of the interaction terms have the following form

$$\vec{x} = 1^w \cdot 0^{m'-w}, \quad \vec{z} \in \pi(1^{w_y} \cdot 0^{w-w_y}) \cdot 0^{m'-w}, \tag{72}$$

for $2 \leq w_y \leq w \leq m$, with w_y even, where for notational simplicity the first two bits correspond to the two spin sites and the remaining m bits correspond to the gauge field. Here $\pi(1^{w_y} \cdot 0^{w-w_y})$ denotes the set of all permutations of a bitstring containing w_y ones and $w - w_y$ zeros. From this representation it is clear that for every weight $b > 1$ bitstring there is at least one a) bitstring of weight $b-1$ that is the same except it contains one fewer one in the $\vec{x}$ register or b) bitstring of weight $b-2$ that is the same except it contains two fewer ones in the $\vec{z}$ register. Thus, every value of b can be accessed by applying a $\mathrm{C}^{b-1}\mathrm{NOT}(\vartheta)$ to one of the $b-1$ states, or a $\mathrm{C}^{b-2}\mathrm{NOT}(\vartheta)$ to a weight $b-2$ state followed by a $\mathrm{C}^{b-1}\mathrm{NOT}(\vartheta)$. Therefore, all interaction bitstrings for each Hamming weight $2 \leq b \leq 2m'$ can be generated by applying $N-1$ total $\mathrm{C}^{b-1}\mathrm{NOT}(\vartheta)$ gates and less than or equal to $F(b+1)-1$ total $\mathrm{C}^{b-2}\mathrm{PSWAP}(\vartheta)$ gates. The latter upper bound comes from the remaining number of terms of weight b from (67).

Finally, all the onsite spin terms are of weight one and can thus be generated from an initial weight one state by $N-1$ total $\mathrm{PSWAP}(\vartheta)$ gates. These weight one states can be used as the source for the $b=1$ states for the interaction and gauge field terms.

The total number of $\mathrm{C}^{b-1}\mathrm{NOT}(\vartheta)$, $\mathrm{C}^{b-2}\mathrm{PSWAP}(\vartheta)$ for the three parts of the Hamiltonian are summarised in Table 1.

C.4.6 Reducing gate cost using translational symmetry

Naively, we could generate $|G\rangle$ by applying the multi-controlled gates within Section C.4.5 a total of $(N-1)$ times for the gauge field and interaction terms (once for each gauge link in the model) and N times for the spin terms. This would lead to a very large number of gates controlled on lots of qubits. However, by noticing that the gauge field and interaction terms have the same form for all $n = 1, 2, \ldots, N$, we can significantly reduce the cost in preparing $|G\rangle$ by partially swapping these terms for the $n=1$ site into all other lattice sites.

Consider following the procedure in the previous section to generate terms corresponding to just the gauge field and interaction terms for the first site. At this point the state will be proportional to $|G^{\text{field}}(1)\rangle + |G^{\text{int}}(1)\rangle$. Written in terms of the computation basis, each term will be of the form

$$\underbrace{|\alpha,\beta\rangle}_{x_0,z_0 \text{ site } n=1} \otimes \overbrace{|\psi\rangle}^{\text{field } n=1} \otimes \underbrace{|\alpha',\beta'\rangle}_{x_0,z_0 \text{ site } n=2} \otimes \overbrace{|0^m\rangle}^{\text{field } n=2} \otimes \underbrace{|0,0\rangle}_{x_0,z_0 \text{ site } n=3} \otimes \ldots, \tag{73}$$

Hamiltonian term	Hamming weights	# $\mathrm{C}^{b-1}\mathrm{NOT}(\vartheta)$	# $\mathrm{C}^{b-2}\mathrm{PSWAP}(\vartheta)$
Gauge fields	$0 < b \leq m$	1	$\binom{m}{b} - 1$
Interactions	$2 \leq b \leq m+2$	1	$F(b+1) - 1$
Spins	$b = 2$	0	$(N-1)$

Table 1: Number of multi-controlled gates for each Hamming weight b needed to generate state $(N-1)\,|G^{\text{field}}(1)\rangle + (N-1)\,|G^{\text{int}}(1)\rangle + \sum_{i=1}^{N} |G^{\text{spin}}(n)\rangle$ for each term in the Hamiltonian. See Section C.4.5 for details.

Hamiltonian term	# $\mathrm{CNOT}(\vartheta)$	# X	# CSWAP	# $\mathrm{C}^m\mathrm{NOT}$
Gauge fields + Interactions	$N-2$	$m(N-2)$	$(m+4)(N-2)$	$N-2$

Table 2: Gate counts to partially swap gauge field and interaction terms for a gauge link site into the remaining $N-2$ sites implementing the transformation $(N-1)\,|G^{\text{field}}(1)\rangle + (N-1)\,|G^{\text{int}}(1)\rangle + \sum_{i=1}^{N} |G^{\text{spin}}(n)\rangle \longrightarrow |G\rangle$. See Section C.4.6 for details.

where $\alpha, \beta, \alpha'\beta' \in \{0,1\}$ and $|\psi\rangle$ represents one of the m binary states corresponding to the gauge field register. If we include an ancilla qubit to which we apply a rotation gate we have

$$\underbrace{\big(\cos(\vartheta)\,|0\rangle + \sin(\vartheta)\,|1\rangle\big)}_{\text{ancilla}} \otimes |\alpha,\beta\rangle \otimes |\psi\rangle \otimes |\alpha',\beta'\rangle \otimes |0^m\rangle \otimes |0,0\rangle \otimes \dots. \tag{74}$$

Next, we apply m CSWAP (Fredkin) gates, controlled on the ancilla qubit, to swap each qubit within the m qubit $n=1$ field register, with the corresponding qubit in the $n=2$ field qubit. This yields

$$\begin{aligned}&\cos(\vartheta)\big(\,|0\rangle \otimes |\alpha,\beta\rangle \otimes |\psi\rangle \otimes |\alpha',\beta'\rangle \otimes |0^m\rangle \otimes |0,0\rangle \otimes \dots\big)\\ &+ \sin(\vartheta)\big(\,|1\rangle \otimes |\alpha,\beta\rangle \otimes |0^m\rangle \otimes |\alpha',\beta'\rangle \otimes |\psi\rangle \otimes |0,0\rangle \otimes \dots\big).\end{aligned} \tag{75}$$

We have now swapped the first two field registers, conditioned on the ancilla qubit. We remove the ancilla qubit, by applying an X gate to each qubit in the $n=2$ field register, followed by an $\mathrm{C}^m\mathrm{NOT}$ gate conditioned on the $n=2$ field register and targeting the ancilla qubit, followed by an X gate to each qubit in the in the $n=2$ field register again, and finally an X gate to the ancilla qubit. This yields

$$|0\rangle \otimes \big(\cos(\vartheta)\,|\alpha,\beta\rangle \otimes |\psi\rangle \otimes |\alpha',\beta'\rangle \otimes |0^m\rangle \otimes |0,0\rangle \otimes \dots + \sin(\vartheta)\,|\alpha,\beta\rangle \otimes |0^m\rangle \otimes |\alpha',\beta'\rangle \otimes |\psi\rangle \otimes |0,0\rangle \otimes \dots\big) \tag{76}$$

where we now have an arbitrary superposition (determined by ϑ) over states with the gauge field terms swapped, with the auxiliary qubit unentangled and returned to its initial state. For initial states corresponding to the terms corresponding to the $n=1$ gauge field energy (53), $\alpha = \beta = \alpha' = \beta' = 0$ (as there are no spin terms) and so we have effectively partially swapped the required binary encoding of the gauge field terms from $n=1$ to $n=2$. We can ensure the interaction terms (in which some of $\alpha, \beta, \alpha'\beta'$ are non zero) are correctly swapped as well by doing the following. Apply two Fredkin gates, controlled on the least significant x qubit in the $n=2$ field register, swapping the registers corresponding to the $n=2$ spin register with the $n=3$ spin register. From (57), we see that interaction terms will always contain a one within this control qubit, thus for $|\psi\rangle$ corresponding to an interaction term, we get

$$\cos(\vartheta)\,|\alpha,\beta\rangle \otimes |\psi\rangle \otimes |\alpha',\beta'\rangle \otimes |0^m\rangle \otimes |0,0\rangle \otimes \dots + \sin(\vartheta)\,|\alpha,\beta\rangle \otimes |0^m\rangle \otimes |0,0\rangle \otimes |\psi\rangle \otimes |\alpha',\beta'\rangle \otimes \dots, \tag{77}$$

where we omit the ancilla qubit which is no longer used. Finally, applying another two Fredkin gates, controlled in the same way to swap the $n=1$ and $n=2$ spin registers, we get:

$$\cos(\vartheta)\,|\alpha,\beta\rangle \otimes |\psi\rangle \otimes |\alpha',\beta'\rangle \otimes |0^m\rangle \otimes |0,0\rangle \otimes \dots + \sin(\vartheta)\,|0,0\rangle \otimes |0^m\rangle \otimes |\alpha,\beta\rangle \otimes |\psi\rangle \otimes |\alpha',\beta'\rangle \otimes \dots. \tag{78}$$

This is the form of the output for both interaction and gauge energy terms (for the latter, since all the spin bits are zero, these Fredkin gates have no effect). Thus, we have taken the bitstrings corresponding to the interaction and gauge field terms for $n=1$ and used them to generate the $n=2$ gauge field and interaction terms, avoiding the need to reapply the costly multicontrolled procedure described in Section C.4.5. i.e., following the notation in (62), we have performed

$$\frac{1}{\mathcal{N}}\left(|G^{\text{field}}(1)\rangle + |G^{\text{int}}(1)\rangle\right) \longrightarrow \frac{1}{\mathcal{N}}\left(\cos\vartheta\,|G^{\text{field}}(1)\rangle + |G^{\text{int}}(1)\rangle\right) + \frac{1}{\mathcal{N}}\left(\sin(\vartheta)\,|G^{\text{field}}(2)\rangle + |G^{\text{int}}(2)\rangle\right), \tag{79}$$

where $\mathcal{N}$ is the norm of unnormalised state $|G^{\text{field}}(1)\rangle + |G^{\text{int}}(1)\rangle$. We can keep repeating this, noticing that the spin only terms are unaffected, and choosing ϑ each time such that we end up with the transformation

$$\begin{aligned}(N-1)\left(|G^{\text{field}}(1)\rangle + |G^{\text{field}}(1)\rangle\right) + \sum_{n=1}^{N}|G^{\text{spin}}(n)\rangle \\ \longrightarrow \sum_{n=1}^{N-1}|G^{\text{field}}(n)\rangle + \sum_{n=1}^{N-1}|G^{\text{int}}(n)\rangle + \sum_{n=1}^{N}|G^{\text{spin}}(n)\rangle = |G\rangle\,.\end{aligned} \tag{80}$$

The total gate cost for this is $(N-2)$ $R_x(\vartheta)$ gates, $2m(N-2)$ X gates, $(m+4)(N-2)$ CSWAP gates and $(N-2)$ C^{m}NOT gates along with an ancilla qubit that can be reused.

C.4.7 Total gate cost for $|G\rangle$

Theorem 27 (Improved gate cost for G)**.** *For the single flavour 1+1D Schwinger model with N lattice sites and gauge fields represented by $m \geq 2$ qubits, the block encoding state $|G\rangle$ can be prepared using the following number of gates.*

$$\begin{aligned}\#T\text{ gates} &\leq 56 - 16m + 8m\frac{\phi^{m+5} - (-\phi)^{-m-5}}{\sqrt{5}} - 8\frac{\phi^{m+6} - (-\phi)^{-m-6}}{\sqrt{5}} \\ &\quad - 16\cdot 2^m + 4N + 4m2^m + 8Nm \qquad (81)\\ \#CNOT\text{ gates} &\leq 54 - 22m - 2\frac{\phi^{m+5} - (-\phi)^{-m-5}}{\sqrt{5}} - 8\frac{\phi^{m+6} - (-\phi)^{-m-6}}{\sqrt{5}} + 8m\frac{\phi^{m+5} - (-\phi)^{-m-5}}{\sqrt{5}} \\ &\quad - 14\cdot 2^m + 25N + 4m2^m + 11Nm \qquad (82)\\ \#R_z\text{ gates} &\leq -48 + 12\frac{\phi^{m+5} - (-\phi)^{-m-5}}{\sqrt{5}} + 12\cdot 2^m + 6N \qquad (83)\end{aligned}$$

where ϕ is the golden ratio. In addition, we require an unspecified number of single qubit Clifford gates as well as $2N + 2m(N-1)$ qubits to store $|G\rangle$ and m ancilla qubits for the computation.

Proof. The full procedure to generate $|G\rangle$ consists of two parts. First, generate the state

$$(N-1)\left(|G^{\text{field}}(1)\rangle + |G^{\text{field}}(1)\rangle\right) + \sum_{n=1}^{N}|G^{\text{spin}}(n)\rangle\,, \tag{84}$$

by applying the gates listed in Table 1 as described in Section C.4.5 once for the gauge field and interaction terms, and $N-1$ times for the spin terms.

The total number of T-gate, CNOT gates and R_z gates for gauge field and interaction terms for this procedure can be written as

$$\sum_{b=1}^{m}\binom{m}{b}(\alpha b + \beta) + \sum_{b=2}^{m+2}F(b+1)(\alpha b + \beta), \tag{85}$$

where for $\alpha = 8, \beta = -16$ for T-gates, $\alpha = 8, \beta = -14$ for CNOT gates and $\alpha = 0, \beta = 12$ for R_z gates according to the decomposition of C^{b-1}NOT(ϑ) and C^{b-2}PSWAP(ϑ) gates of Proposition 26. In addition, $2(N-1)$ CNOTs and $6(N-1)$ R_z gates are required for the spin terms.

Secondly, we must apply the gates listed in Table 2 as described in Section C.4.6 to implement the transformation into $|G\rangle$.

The cost of a single CNOT(ϑ) gate will be six R_z gates and two CNOTs. A single Fredkin gates (CSWAP) can be implemented with two CNOTs along with a Toffoli (CCNOT) gate [112] for a total cost of four T gates and five CNOTs (using remark 23 for the Toffoli decomposition). Each of the C^{m}NOT gates requires $4m-4$ and T gates $4m-3$ CNOTs from Remark 23. Thus the total cost for this step is

$$8Nm + 4N - 16m - 8 \quad \text{T gates}, \tag{86}$$

$$11Nm + 25N - 22m - 4 \quad \text{CNOT gates} \tag{87}$$

$$6N - 12 \quad R_z \text{ gates}. \tag{88}$$

The required expressions result from using Corollary 11 and Proposition 12 to upper bound (84) with the appropriate values of α, β and adding the result to the relevant equation of (86), (87) and (88). □

Remark 28. *If we choose $m = \lceil \log(N) + 1 \rceil$ to correspond to the maximum gauge field occupancy of Remark 1, then the gate cost of $|G\rangle$ is $\mathcal{O}(N \log N)$ and $(2N-1)\log N - \log(N)$ qubits are required.*

C.5 T gate cost of rotation gates

For error corrected quantum computing, we typically do not have access to to an arbitrary gate set but rather a restircted subset from which arbitrary operations must be performed approximately. according to the Solovay Kitaev theorem however, the gate cost of doing so scales favourably with approximation error. In our case, we need to convert the arbitrary R_z rotations into Clifford + T-gates. To do so, we use the following Lemma.

Proposition 29 (Clifford+T approximations of z-rotations [97])**.** *There exists an efficient algorithm, which is known, for approximately decomposing an R_z gate up to rotation error ϵ which requires $3\log\epsilon^{-1} + \mathcal{O}(\log\log\epsilon^{-1})$ T gates.*

We use this to make the following approximation for the number of T gates required to implement a single R_z gate in our algorithm.

Theorem 30. *Assume the following*

1. *The average fractional error required on values of α_i is given by a constant value $\bar{\epsilon}_\alpha$.*
2. *The average value of $\sqrt{|\alpha_i|}$ is approximately equal to the reciprocal of the number of Pauli terms within the Hamiltonian.*
3. *Recalling that the values of $\sqrt{|\alpha_i|}$ are generated by successive rotations using partial swap gates, for an average approximation error ϵ per R_z gate, the average error on resulting value of given $\sqrt{|\alpha_i|}$ is less than approximately $d\epsilon$, where d is the maximum number of R_z gates affecting the value of $\sqrt{|\alpha_i|}$.*

Then, the average number of T gates required per R_z is approximately less than or equal to

$$3\log\left((N-1)2^m + N\cdot 2^{m+1} + N\right) + 3\log(12m+48) + 3\log(\bar{\epsilon}_\alpha^{-1}), \tag{89}$$

up to $\mathcal{O}(\log\log)$ terms.

Proof. Given that the average value of $\sqrt{|\alpha_i|}$ equal to the reciprocal of the number of Pauli terms within the Hamiltonian. The absolute error required is approximated by

$$\epsilon_{\text{tot}} = \bar{\epsilon}_\alpha \left[(N-1)2^m + (2^{m-1})(N-1) + N\right]^{-1}, \tag{90}$$

where we have used the total number of Pauli strings resulting from Corollary 20 and Remarks **??**, 21.

From Lemma 26, each multi-qubit rotation gate for the generation of $|G\rangle$ requires $12R_z$ gates. The maximum number of rotations affecting a single value of α_i is equal to the Hamming weight of the corresponding basis state $|(\vec{x},\vec{z})_i\rangle$. The maximum Hamming weight present in the Hamiltonian is equal to $2m+4$ from the interaction terms of (61). Thus, to achieve average absolute error on $\sqrt{\alpha_i}$ that is less than or equal to ϵ_{tot}, we require $\epsilon_{R_Z} \approx (12m-48)^{-1}\epsilon_{\text{tot}}$.

From the the approximate values of ϵ_{tot} and ϵ_{R_Z} above and using Proposition 29 for the optimal decomposition of R_Z gates we get the result. □

Corollary 31. *Assuming $\bar{\epsilon}_\alpha$ is constant, if $m = \lceil \log(N) \rceil$ then this is $\mathcal{O}(\log(N))$ and the total T-gate cost for generating $|G\rangle$, including rotations, is $\mathcal{O}(N \log(N))$. This does not affect the asymptotic scaling of this part of the algorithm.*

For gate cost calculations presented in the main text, we assume that $\bar{\epsilon}_\alpha$ terms are neglible.

C.6 Applying the block encoding unitary U

C.6.1 Total gate cost for U

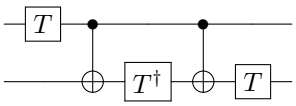
Theorem 32. *For the single flavour 1+1D Schwinger model with N lattice sites and gauge fields represented by $m \geq 2$ qubits, the block encoding state $|G\rangle$ can be prepared using the following number of gates.*

$$\#T \text{ gates} \quad = 6N + 6mN - 6m \tag{91}$$

$$\#CNOT \text{ gates} = 8N + 8mN - 8m \tag{92}$$

We require an unspecified number of single qubit Clifford gates and m ancilla qubits for the computation.

Proof. As described in Ref. [63], to implement the Pauli operators in U, we require a total of N_G, the number of qubits in $|G\rangle$, two qubit CNOT, CZ and Controlled S (see (38)) gates. The decompositions of the first two are trivial, for the latter we note that a controlled S gate performing (38) can be implemented as:



(93)

Therefore, given that $N_G = 2N + 2m(N-1)$, the number of gates required to apply the controlled Pauli operations of U is

$$3 \cdot (2N + 2m(N-1)) \quad \text{T gates}, \tag{94}$$

$$4 \cdot (2N + 2m(N-1)) \quad \text{CNOT gates}, \tag{95}$$

and zero R_z gates. □

Remark 33. *If $m = \lceil \log(N) + 1 \rceil$ as in Remark 1, then the gate cost of U is $\mathcal{O}(N \log N)$.*

C.6.2 Applying phases

If, rather than introducing the signs of α_i within $|\tilde{G}\rangle$, we follow a similar method as Appendix C of Ref. [63] and add the signs within U, we require the operation

$$U = \sum_i \text{sgn}(\alpha_i) |i\rangle\langle i| \otimes \hat{P}_i. \tag{96}$$

To understand how this would be implemented, we note that we can apply a -1 to an auxiliary state of Hamming weight n by applying a C^{n-1}Z gate to all auxiliary qubits containing 1 in the basis state of interest. This will also flip the sign of higher Hamming weight states that contain 1s in the same positions as the Hamming weight n state. Thus, the desired phases can be achieved by applying Z gates to qubits in Hamming weight one states that require a -1 phase, then applying CZ gates to Hamming weight two states that need their phases changed, flipping back the phase of any that may have been unnecessarily changed by the application of the previous single qubit phase gates, then doing the same for weight three states with CCZ gates and so on until all states have the required phase applied to them. The overall gate cost is $\mathcal{O}(\log(N)N^2)$ as given in the following theorem.

Theorem 34 (Applying phases within U)**.** *If, we add the required phases for the Block encoding of the single flavour 1+1D Schwinger model using a method analagous to that described in Appendix C of Ref. [63], the gate cost of U is*

$$\#T \text{ gates} \quad \leq (N-1)\left[32+6m-4\frac{\phi^{m+6}-(-\phi)^{-m-6}}{\sqrt{5}}+4m\frac{\phi^{m+5}-(-\phi)^{-m-5}}{\sqrt{5}}-8\cdot 2^m+2m2^m\right]+6N \tag{97}$$

$$\#CNOT \text{ gates} \leq (N-1)\left[29+8m-4\frac{\phi^{m+6}-(-\phi)^{-m-6}}{\sqrt{5}}+4m\frac{\phi^{m+5}-(-\phi)^{-m-5}}{\sqrt{5}}-7\cdot 2^m+2m2^m\right]+8N. \tag{98}$$

Proof. In addition to the controlled Pauli and phase gates of Theorem 32, we must generate the required minus signs by applying multicontrolled Z gates. In the worst case, every single bitstring of weight b requires a different phase from the total phase of the $b-1$ weight strings differing by just a single bit. Thus, each one must have a C^bZ gate applied to achieve the correct phase. Thus, we require at most $(N-1)\binom{m}{b}$ total $\mathrm{C}^{b-1}Z$ gates for each $1 \leq b \leq m$ for gauge field terms and $(N-1)F(b+1)$ total $\mathrm{C}^{b-1}Z$ for each $2 \leq b \leq m+2$ gates for interaction terms. These values correspond to the N_b^{field} and N_b^{int} in (63), (67).

As in Theorem 27, the gate cost of these terms is given by (85), this time multiplying by $(N-1)$, where for $\mathrm{C}^{b-1}Z$ gates $\alpha = 4, \beta = -8$ for T gates and $\alpha = 4, \beta = -7$ for CNOT gates. No rotation gates are needed. The total cost follows from Lemma 11 and Proposition 12 (84). □

C.7 Total gate cost for Π_φ and $\tilde{\Pi}_\varphi$

Theorem 35. *For the single flavour 1+1D Schwinger model with N lattice sites and gauge fields represented by $m \geq 2$ qubits, the rotation operator Π_φ for the QSVT procedure of Theorem 3 can be implemented with two times the cost of G from Theorem 27 plus $128mN + 128N - 128m - 192$ T gates, $96mN + 96N - 96m - 144$ CNOT gates and an R_z gate.*

Proof. As described in Remark 5, the rotation operator can be implemented with two G operators, two $\mathrm{C}^{N_G}\mathrm{NOT}$ gates and an R_z gate. From Remark 22 the cost of a $\mathrm{C}^{N_G}\mathrm{NOT}$ gate is $32N_G - 96$ T gates and $24N_G - 72$ CNOT gates along with one ancilla qubit which can be reused from the G computation. Using $N_G = 2m(N-1) + N$ and combining the costs of Gs, the $\mathrm{C}^{N_G}\mathrm{NOT}$s and rotation yields the result. □

Remark 36. *If $m = \lceil \log(N) + 1 \rceil$ as in Remark 1, then the gate cost of Π_{φ_j} is $\mathcal{O}(N \log N)$. It may be possible to reduce the multicontrolled gates by using additional ancilla [113] or by reducing the rotation to the subspace of the maximum achievable Hamming weight as in [63]. Here we sketch the steps taken to compute the quantum resource cost.*

Corollary 37. *Since $\tilde{G}$ can be generated from the same gates as G with some rotation angles differing in the phase, the cost of $\tilde{\Pi}_\varphi$ is the same as the cost of Π given in Theorem 35.*

C.8 Total gate cost of the full algorithm

Theorem 38. *A single measurement needed for the estimation of matrix element $\langle H \rangle$ can be calculated by using two applications of the gates listed in Theorem 27 and k applications of the gates in Theorems 32 and 35. Then assuming $m = \log N$, the gate cost of this is $\mathcal{O}(kN \log(N))$.*

Proof. This follows from Lemma 4 and the preceding gates costs for U, $|G\rangle$ and $|\tilde{G}\rangle$. □

D Numerical experiments in the strong coupling regime

In the majority of the main text, we studied a system with Hamiltonian parameters bare mass $\mu = 1.5$ and coupling parameter $x = 0.5$. The stronger coupling regime, in which the ground start is further from the Néel state, is more challenging for classical methods. In the following, we present additional numerical experiments, repeating the procedure described in Sec. 5 for a system with $\mu = 1.5$ and $x = 5$ using PQSE.

In Fig. 12, we present the fractional energy error for this system, analagous to Fig. 8 as a function of maximum Krylov order D assuming no measurement noise. This is analogous to Fig. 8 with $x = 5$, rather than $x = 0.5$. We see that the overlap between the initial state and the true ground state, decreasing more rapidly with N than for the more weakly coupled system, as a result, a larger Krylov basis dimension D is required to reach

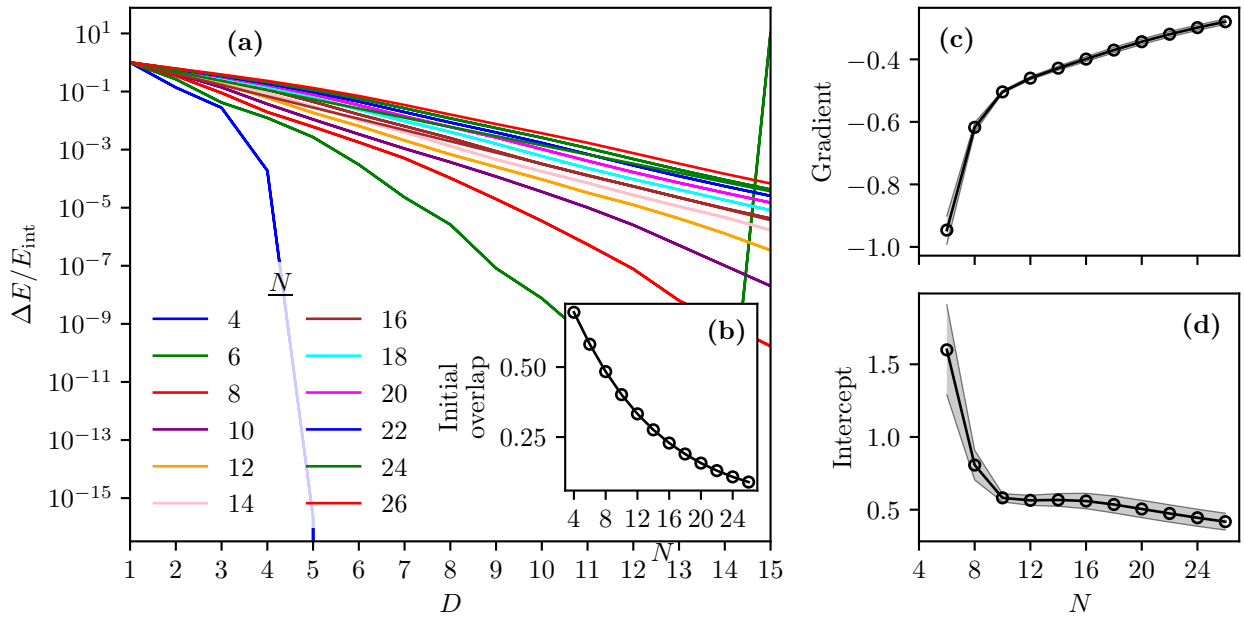


Figure 12: **(a)** Fractional energy error as a function of maximum Krylov order for different value of lattice size N for $\mu = 1.5$, $x = 5$. No measurement noise has been applied. **(b)** Overlap between initial state $|\psi_0\rangle$ and exact ground state. **(c)** & **(d)** Gradient and intercept for straight lines (on log-linear plot) fit to data in (a). Filled area indicates standard errors on estimates.

the same fractional energy error. In Fig. 13, we show the required Krylov basis size $D(N)$, to achieve a fixed fractional energy error of 10^{-4} using these results, again assuming no measurement noise. This is analogous to Fig. 3.

In Fig. 14, we present the fractional energy error for this system, analogous to Fig. 4 in the main text. In Fig. 15, we present the expected total number of calls required to the block encoding operator $\Pi_\varphi U$, analogous to Fig. 5. Finally, in Fig. 16, we present the total number of T-gates required to achieve fractional ground state energy errors of 1×10^{-2}, $1 \times 10^{-}4$ and 1×10^{-6} arising from the estimated number of block encoding calls and the estimated gate cost for the block encoding procedure, analogous to Fig. 7.

From these results, we see that the number of calls to the block encoding procedure, and therefore the number of gates required to achieve the same error for the same system size, is considerably larger than for the $\mu = 1.5$, $x = 0.5$ system.

E Comparing resource requirements for all-to-all to linear-nearest-neighbor qubit connectivity

Throughout the rest of this article, we have assumed an all-to-all qubit connectivity. In this section, we briefly discuss how a limited qubit connectivity, namely linear-nearest-neighbour (LNN) will affect the gate costs associated with the algorithm. For this we consider the following Lemma which, to the author's knowledge, is the tightest upper bound for the resource cost of multi-controlled CNOT gates for a linear nearest-neighbour topology.

Lemma 39 (n qubit Toffoli gate with linear-nearest-neighbour qubit connectivity, Theorem 8 Ref. [114])**.** *Consider a k qubit subregister with linear-nearest-neighbour qubit connectivity. A $\mathrm{C}^{n-1}\mathrm{NOT}$ gate on these qubits, where $n > 6$ and $k > n + 1$ can be implemented with $16n - 32$ T-gates, $8k + 14n - 44$ CNOT gates as well as single qubit Clifford gates.*

To the authors knowledge, this is the most optimal decomposition of a multi-qubit Toffoli gate considering a restricted qubit connectivity. We recall that the best known decomposition [110] when allowing for all-to-all connectivity and multiple ancillas was $4n - 8$ T-gates and $4n - 7$ CNOT gates (Remark 23). To leading order, the number of T-gates required for a multi-controlled Toffoli gate for LNN is four times larger than that required for all-to-all connectivity. Thus, the T-gate cost of the algorithm will be at most four times larger for a LNN

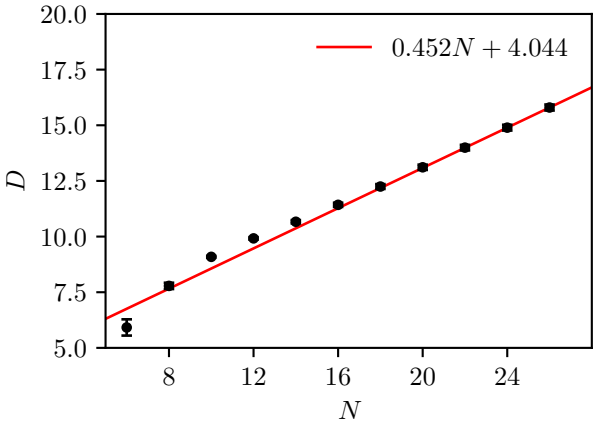


Figure 13: Estimated Krylov order D needed for QSE to achieve fractional energy error of $\Delta E/E_{\text{int}} = 10^{-4}$ in the case of no measurement noise calculated using data from Appendix A.1 for a system with $\mu = 1.5$, $x = 5$. Error bars correspond to standard errors on the fits to generate these points. Red line corresponds to linear fit to final two data points.

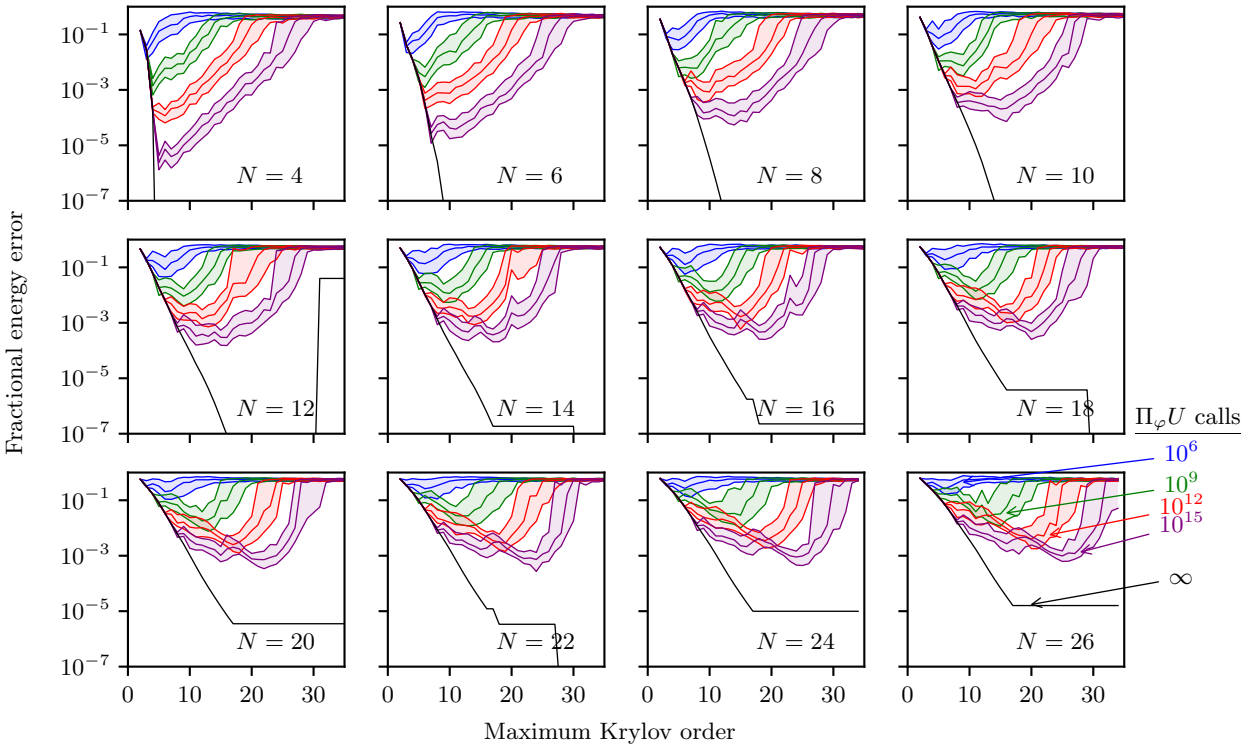


Figure 14: Fractional energy error with PQSE procedure with finite number of measurements for range of different lattice sizes N for $\mu = 1.5$, $x = 5$. Values are shown as a function of the maximum allowed Krylov basis for the PQSE procedure and as a function of number of calls to the block encoding operator $\Pi_\varphi U$ (which affect the number of measurements as defined in the main text). Lines and shaded area corresponds to the median as well as upper and lower quartiles of energy error. The black line (labelled as infinite calls to $\Pi_\varphi U$) corresponds to no measurement noise.

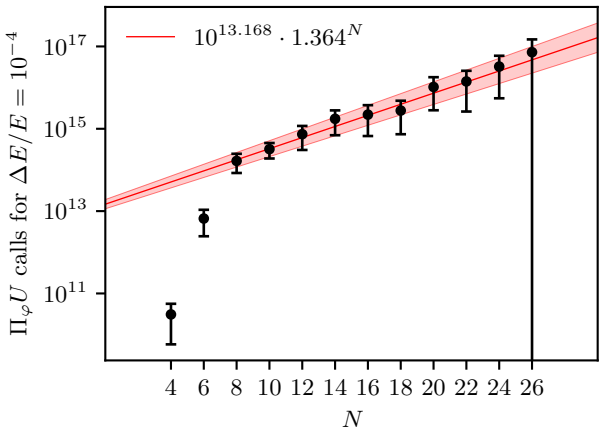


Figure 15: Estimated number of call to qubitization procedure using PQSE to achieve fractional energy error of $\Delta E/E_{\text{int}} = 10^{-4}$ for $\mu = 1.5$, $x = 5$. Error bars correspond to the standard error resulting from fits used to generate the data points. Red line corresponds to exponential fit (straight line on log-linear axes) with filled area indicating uncertainty in fitting and extrapolation.

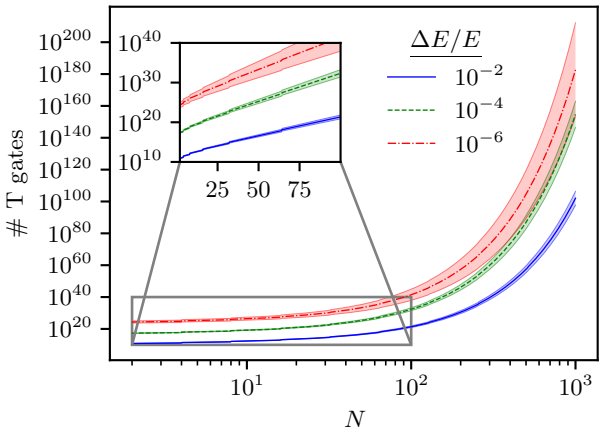


Figure 16: Total number of T-gates required for the whole PQSE procedure for the single flavour Schwinger model with $\mu = 1.5$, $x = 5$ as a function of lattice size N for different fractional energy errors $\Delta E/E_{\text{int}}$.

topology. The exact value may be considerably less as many T-gates within the computation are for other non-Clifford gates (for example the single qubit rotation gates) and will depend on what ordering is chosen fot the physical qubits (which will affect which gates need to be implemented according to the decompositions in Refs. [110] or [114]. Since the CNOT cost in Lemma E depends on the number of qubits the multi-qubit gate acts over, computing how a LNN qubit topology affects the resource cost will be more complex. However, since the CNOT cost for all-to-all qubit connectivity at system sizes of interest is already well beyond what is achievable non fault-tolerant devices, we do not consider how this cost will increase for a ressticted qubit topology.

F Hardware information for current generation quantum processors

Below we give details for current generation quantum processors used for analysis in Sec. 4. Gate times and coherence times for four processors, where they could be found or calculated, are given in Table 3.

For data on the Quantinuum H2-1 processor, two-qubit gate time was estimated in the following way. We take benchmark data for shot times and number of operations for a range of circuits from Table I of Ref. [115]. We take the shot time for the circuit, subtract the quoted initial state preparation time (32μs) and divide this by the number of two-qubit gates in the circuit (assuming two-qubit gates dominate the execution time over single qubit and SPAM gates). We take the mean of these values to estimate an average two qubit gate time, excluding the 'Transport 1Q RB, $l = 64$' circuit which contains no two-qubit gates. Coherence time is calculated by taking the quoted quadratic dephasing rate ($0.043 \cdot 2\pi$rad/s) from Ref. [116], which is quoted as being applied during qubit transport and idling. We inverted this and multiplied by the average fraction of time spent in transport from Table I of [115] to give an estimate of the typical dephasing time.

For data on the IonQ Forte processor, two-qubit gate times were estimated in the following way. Using runtime benchmark given in the supplemental data for Ref. [117], for each type of algorithm listed (each with varying problem sizes), the total single circuit time was divided by the number of native two-qubit gates (assuming that two-qubit gates dominated execution time). The mean two-qubit gate time was then calculated over the algorithm types. Estimated coherence times are taken from Ref. [118].

Using the total CNOT counts given in Fig. 6(d), we get a rough estimate of the time taken to apply a single block encoding step $\Pi_{\varphi_j} U$ by multiplying the CNOT count by the two-qubit gate time for each processor. The values calculated are shown in Table 4 both as an absolute time, as well as a multiple of the for T1 and T2 times. This makes three assumptions, (1) when compiled to the particular hardware gate set, the number of native two-qubit gates is roughly equal to the number of CNOTs we have counted, (2) total runtime is dominated by two-qubit gate operations, and (3) CNOT gates are implemented in serial. In reality, many of these CNOTs could be run in parallel or even removed by recompilation; so this is likely to be an overestimate. We can instead assume that all gates are maximally parallelised such that two-qubit gates are applied in layers of $N_{\text{qubits}}/2$ acting at the same time to give a lower bound on the run time. These values are shown in Table 5.

Processor	Type	# Qubits	T1	T2	two qubit gate time
IBM Eagle r3 [119]	Superconducting	127	275 μs	117 μs	636 ns
Google Sycamore [120]	Superconducting	70	22.9 μs	15.5 μs	20 ns
Quantinuum H2-1 [115]	Ion trap	32	?	1410 s	6.46 ms
IonQ Forte [117]	Ion trap	32	10–100 s	$\sim$1s	931 μs

Table 3: Hardware information for range of superconducting and ion trap quantum processors. T1 and T2 correspond to qubit relaxation and coherence times and two qubit gate time refers to native two-qubit gate execution times. The Quantinuum H2-1 and IonQ Forte processors, T1, T2 and two qubit gate times were calculated as described in main text of this Appendix. T1 time for the Quantinuum H2-1 could not be found. Values for other processors are median values taken from the corresponding references.

Processor	Block encoding runtime (s)			Runtime as fraction of T1			Runtime as fraction of T2		
	N=100	N=1,000	N=10,000	N=100	N=1,000	N=10,000	N=100	N=1,000	N=10,000
Eagle r3	6.62×10^{-3}	7.49×10^{-1}	1.06×10^{1}	2.41×10^{1}	2.72×10^{3}	3.86×10^{4}	5.66×10^{1}	6.40×10^{3}	9.08×10^{4}
Sycamore	2.08×10^{-4}	2.36×10^{-2}	3.34×10^{-1}	9.09×10^{0}	1.03×10^{3}	1.46×10^{4}	1.34×10^{1}	1.52×10^{3}	2.15×10^{4}
H2-1	6.73×10^{1}	7.61×10^{3}	1.08×10^{5}	-	-	-	4.77×10^{-2}	5.40×10^{0}	7.65×10^{1}
Forte	9.69×10^{0}	1.10×10^{3}	1.55×10^{4}	9.69×10^{0}	1.10×10^{2}	1.55×10^{3}	9.69×10^{0}	1.10×10^{3}	1.55×10^{4}

Table 4: Estimated block encoding runtime for different quantum processors assuming that two-qubit gates are executed in series for different system sizes. Runtimes are given in seconds as well as proportions of the T1 (where known) and T2 times for each processor using data from Table 3.

Processor	Block encoding runtime (s)			Runtime as fraction of T1			Runtime as fraction of T2		
	N=100	N=1,000	N=10,000	N=100	N=1,000	N=10,000	N=100	N=1,000	N=10,000
Eagle r3	6.64×10^{-6}	5.01×10^{-5}	5.33×10^{-5}	2.42×10^{-2}	1.82×10^{-1}	1.94×10^{-1}	5.68×10^{-2}	4.28×10^{-1}	4.55×10^{-1}
Sycamore	2.09×10^{-7}	1.58×10^{-6}	1.68×10^{-6}	9.12×10^{-3}	6.88×10^{2}	7.32×10^{-2}	1.35×10^{-2}	1.02×10^{-1}	1.08×10^{-1}
H2-1	6.75×10^{-2}	5.09×10^{-1}	5.41×10^{-1}	-	-	-	4.79×10^{-5}	3.61×10^{-4}	3.84×10^{-4}
Forte	9.73×10^{-3}	7.34×10^{-2}	7.80×10^{-2}	9.73×10^{-4}	7.34×10^{-3}	7.80×10^{-3}	9.73×10^{-3}	7.34×10^{-2}	7.80×10^{-2}

Table 5: Estimated block encoding run time for different quantum processors assuming that two-qubit gates are maximally parallelised. Run times are given as proportions of the T1 (where known) and T2 times for each processor using data from Table 3.