

Fault-Tolerant Quantum Simulation of Generalized Hubbard Models

Andreas Juul Bay-Smidt^{1,2,*} Frederik Ravn Klausen^{1,3} Christoph Sünderhauf⁴ Róbert Izsák⁴,
Gemma C. Solomon^{1,2,†} and Nick S. Blunt^{4,‡}

¹*NNF Quantum Computing Programme, Niels Bohr Institute, University of Copenhagen, Denmark*

²*Nano-Science Center and Department of Chemistry, University of Copenhagen, Denmark*

³*Department of Mathematics, Princeton University, USA*

⁴*Riverlane, Cambridge CB2 3BZ, United Kingdom*



(Received 24 January 2025; accepted 30 July 2025; published 12 September 2025)

Quantum simulations of strongly interacting fermionic systems, such as those described by the Hubbard model, are promising candidates for useful early fault-tolerant quantum computing applications. This paper presents tile Trotterization, a generalization of plaquette Trotterization, which uses a set of tiles to construct Trotter decompositions of arbitrary lattice Hubbard models. The tile Trotterization scheme also enables the simulation of more complex models, including the extended Hubbard model. We improve previous Hubbard-model commutator bounds, further provide tight commutator bounds for periodic extended Hubbard models, and demonstrate the use of tensor network methods for this task. We consider applications of tile Trotterization to simulate hexagonal lattice Hubbard models and compare the resource requirements of tile Trotterization for performing quantum phase estimation to a qubitization-based approach, demonstrating that tile Trotterization scales more efficiently with system size. These advancements significantly broaden the potential applications of early fault-tolerant quantum computers to models of practical interest in materials research and organic chemistry.

DOI: [10.1103/gr4t-blw5](https://doi.org/10.1103/gr4t-blw5)

I. INTRODUCTION

Simulating quantum systems of interacting electrons is an important and complex challenge in the study of molecules and materials. Quantum simulation of fermionic systems is considered one of the most promising applications of quantum computers [1–3], which has motivated resource estimates for quantum simulation of complex systems such as the FeMoco-complex and the cytochrome P450 enzyme [4–6]. Such studies have investigated the electronic structure problem in both first and second quantization, using Gaussian, plane wave, or Bloch basis sets [6–12]. Despite significant advances in recent years, general electronic structure simulations of practically relevant and classically intractable systems are estimated to require millions of physical qubits and at least billions of T gates

[5,6], which is far beyond the capabilities of today's quantum hardware.

Recently, quantum simulations of the Hubbard model [13] have received increased attention due to the model's relatively low resource requirements, making it a promising candidate for early demonstrations of practical quantum advantage [14–16]. The Hubbard model is a model of interacting electrons, which, despite its simplicity compared with the full electronic structure Hamiltonian, is able to describe important aspects of the physics of real materials, and has long been considered a potential model for high-temperature superconductivity [17–19].

Two recent papers by Kivlichan *et al.* [20] and Campbell [21] introduce efficient Trotterization schemes [22–24] to implement time evolution for the Hubbard model. Hamiltonian time evolution allows for the calculation of dynamical observables such as Green's functions, which are used to access important quantities including the many-body density of states and spectral functions. Time evolution is also an important subroutine in quantum phase estimation (QPE) [25–28], statistical phase estimation (SPE) [29–33], and in other ground-state [34,35] and Gibbs-state [36] sampling methods. QPE can also be performed using the more modern qubitization framework [37–39], which has also been developed for the Hubbard model [40]. These prior studies focus on the on-site square-lattice

*Contact author: andreas.bay-smidt@nbi.ku.dk

†Contact author: gsolomon@chem.ku.dk

‡Contact author: nick.blunt@riverlane.com

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

Hubbard model [20,21,40] and exclude a broader range of lattices and more complicated electronic interaction models. Very recently, Ref. [41] introduced an extension to the square-lattice Trotter scheme in Ref. [21], allowing for beyond-nearest-neighbor hopping terms and multiorbital interactions.

Extending these simulation methods to other lattices enables the simulation of a broader range of real-life materials with complex electronic behavior. This is especially relevant for lattices that introduce frustration, such as the kagome and triangular lattices, which are challenging to simulate classically and may provide valuable insights into exotic phases of matter [42,43]. Introducing more complicated electron interaction models allows for studying the effects of nonlocal electron-electron interactions on charge order [44,45], charge-density waves [46], pair density waves [47] and alternative types of superconductivity [47–49]. The simplest model employed to study the physics of certain kagome metals is the kagome-Hubbard model which includes nearest-neighbor electron interactions [46–48,50,51]. Conjugated hydrocarbon molecules can also be described by the Hubbard model [52,53] or more complex extensions, such as the extended Hubbard model [54–56] or the PPP model [57–60]. The electronic structure of nanographene is especially interesting because it exhibits topological frustration and strong correlations leading to unconventional magnetic properties [61,62], and can be well described by the PPP model [63].

In this study, we introduce tile Trotterization to extend the applicability of plaquette Trotterization (PLAQ) (developed by Campbell in Ref. [21]) to arbitrary lattice Hubbard models, and introduce strategies for including longer-range interaction terms, used for example in the extended Hubbard model. Tile Trotterization generalizes using plaquettes as the components spanning the square lattice to components of other shapes (called tiles), that can be used to cover any lattice with nearest-neighbor-hopping terms to construct efficient Trotter decompositions. We define a set of tiles based on complete bipartite graphs, which we show can be implemented efficiently, contrary to tiles of arbitrary shape that generally have a high implementation cost. We also provide examples demonstrating how tile Trotterization can be used to simulate various hexagonal lattice Hubbard models, including per-Trotter-step gate costs and Trotter error norms. Tiling-based approaches have also been considered in the context of quantum simulation of spin models [64], and these could potentially also benefit from more advanced tiling strategies.

We also provide tight bounds on the commutator spectral norms used to estimate Trotter error. In this, we demonstrate a natural approach to divide the commutator into operators acting on a limited number of lattice sites, the spectral norm of which can be estimated by numerical techniques such as the density-matrix renormalization

group (DMRG) algorithm, or other methods generally. Our commutator bounds are presented in three lemmas: Lemma F1 for the standard Hubbard model and Lemmas F2 and F3 for the extended Hubbard model. We apply these lemmas to evaluate tight commutator bounds for the periodic hexagonal lattice Hubbard and extended Hubbard models. In Eqs. (31) and (32), we provide expressions for the non-trivial parts of the Trotter errors for periodic hexagonal lattice Hubbard and extended Hubbard model Trotter steps.

We analyze the performance of tile Trotterization by comparing it to a qubitization-based approach. Trotterization and qubitization have different dependencies on system size and simulation accuracy, leading to interesting trade-offs when comparing their performance for specific applications [65]. We construct qubitized quantum walk operators that are optimized for the Hubbard model on the periodic hexagonal lattice, building upon previous work by Babbush *et al.* in Ref. [40]. We provide further optimizations that reduce the cost of the qubitized quantum walk operators and present a detailed analysis of gate and qubit costs. To compare the Trotterization and qubitization approaches, we consider the task of energy estimation by QPE for the periodic hexagonal lattice Hubbard model. We obtain $\mathcal{O}(N^{3/2}\epsilon^{-3/2})$ T complexity for implementing QPE using tile Trotterization, compared to $\mathcal{O}(N^2\epsilon^{-1})$ for qubitized QPE, where N is the number of lattice sites and ϵ is the target accuracy. Contrary to common belief that qubitization generally is asymptotically more efficient than Trotterization, we find that tile Trotterization scales better when ϵ is either constant or allowed to scale with the system size.

Overall, we find that tile Trotterization implementations of QPE for the Hubbard model and the extended Hubbard model can be performed with T -gate costs in the range 10^6 – 10^7 for classically nontrivial system sizes, making tile-Trotterization-based quantum algorithms promising candidates for early fault-tolerant quantum computing applications.

The paper is structured as follows: In Sec. II we define the generalized Hubbard model, including the standard Hubbard model and the extended Hubbard model used throughout this paper. Section III introduces tile Trotterization while Sec. IV presents a concrete application of tile Trotterization. In Sec. V we present our qubitization approach, and finally, Sec. VI presents a QPE resource comparison of the simulation methods and models discussed in this paper.

II. HUBBARD MODEL HAMILTONIANS

We consider generalized Hubbard models of the form

$$H = H_h + H_C, \quad (1)$$

with hopping terms given by

$$H_h = -\tau \sum_{i,j,\sigma} R_{ij} a_{i\sigma}^\dagger a_{j\sigma}, \quad (2)$$

where τ represents the hopping parameter. Here, R_{ij} is the adjacency matrix of the lattice with $R_{ij} = 1$ if i and j are neighbors and $R_{ij} = 0$ otherwise. The operators $a_{i\sigma}^\dagger$ and $a_{i\sigma}$ are fermionic creation and annihilation operators acting on a spin- σ orbital at lattice site i . The potential energy, or Coulomb (C), terms have the form

$$H_C = \sum_{i=1}^N U_i n_{i\uparrow} n_{i\downarrow} + \sum_{i \neq j} \sum_{\sigma, \sigma'} C_{i\sigma, j\sigma'} n_{i\sigma} n_{j\sigma'}, \quad (3)$$

where U_i is the on-site interaction strength on site i and $C_{i\sigma, j\sigma'}$ is the interaction strength between electrons in spin orbitals $i\sigma$ and $j\sigma'$ on different sites. The number operator, $n_{i\sigma}$, is defined as $n_{i\sigma} = a_{i\sigma}^\dagger a_{i\sigma}$. The form of H_C varies across the Hamiltonians considered in this paper.

In all systems and models discussed throughout, we represent the number of lattice sites as N and assign two spin orbitals, $\sigma \in \{\uparrow, \downarrow\}$, to each lattice site, such that the total number of spin orbitals is $2N$.

A. The Hubbard model

First, we introduce the standard Hubbard model, or simply the Hubbard model, with Coulomb interactions between electrons of opposite spin on the same lattice site. The Hubbard model is defined as

$$H_H = H_h + H_I, \quad (4)$$

with potential energy term given by

$$H_I = U \sum_{i=1}^N n_{i\uparrow} n_{i\downarrow}, \quad (5)$$

where U represents the onsite interaction strength.

There are many fermion-to-qubit mappings that could be applied to this Hamiltonian, including those which exploit locality [66,67]. In this paper we work with the standard Jordan-Wigner (JW) mapping, which has been considered in previous studies of the Hubbard model [20,21]. Here, the number of terms in the transformed Hamiltonian can be reduced by $2N$ by applying a chemical shift to the interaction term in Eq. (5)

$$H'_I = U \sum_{i=1}^N \left(n_{i\uparrow} - \frac{1}{2} \right) \left(n_{i\downarrow} - \frac{1}{2} \right) = \frac{U}{4} \sum_{i=1}^N Z_{i\uparrow} Z_{i\downarrow}, \quad (6)$$

where $Z_{i\sigma} \equiv 2n_{i\sigma} - 1$. The difference between the original and the shifted on-site interaction term is

$$H'_I - H_I = -\frac{U}{2} \sum_{i=1}^N \left(n_{i\uparrow} + n_{i\downarrow} - \frac{1}{2} \right). \quad (7)$$

In an η -electron subspace, this difference is a constant energy shift of $\Delta E_I = (U/2)(N/2 - \eta)$, which can be trivially corrected [21]. Continuing, we use the modified version Eq. (6) of the on-site interaction and refer to it as H_I .

B. The extended Hubbard model

We also consider the extended Hubbard model where Coulomb interactions between electrons on neighboring sites are introduced. We define this model as

$$H_{EH} = H_h + H_I + H_V, \quad (8)$$

with $H_C = H_I + H_V$. The Coulomb interaction between neighboring sites is defined as

$$H_V = V \sum_{\langle ij \rangle} \sum_{\sigma, \sigma'} n_{i\sigma} n_{j\sigma'}, \quad (9)$$

where V is a parameter for the nearest-neighbor Coulomb interaction. We define the nearest-neighbor summation, $\sum_{\langle ij \rangle}$, as a sum over each each lattice pair $\{i, j\}$ once, which corresponds to summing over all bonds in the lattice. The summation over the spin indices, σ and σ' , ensures that we include interactions between all spin orbitals in lattice pair $\{i, j\}$.

The number of terms in the extended Hubbard model after Jordan-Wigner transformation can be reduced by applying a chemical shift to H_V to obtain

$$\begin{aligned} H'_V &= V \sum_{\langle ij \rangle} \sum_{\sigma, \sigma'} \left(n_{i\sigma} - \frac{1}{2} \right) \left(n_{j\sigma'} - \frac{1}{2} \right) \\ &= \frac{V}{4} \sum_{\langle ij \rangle} \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'}. \end{aligned} \quad (10)$$

The difference between the original and the shifted nearest-neighbor interaction term is

$$H'_V - H_V = -\frac{V}{2} \sum_{\langle ij \rangle} \sum_{\sigma, \sigma'} \left(n_{i\sigma} + n_{j\sigma'} - \frac{1}{2} \right). \quad (11)$$

Given a k -regular interaction graph (all sites having k nearest neighbors), e.g., periodic lattice models, then $\sum_{\langle ij \rangle} \sum_{\sigma, \sigma'}$ runs over $2kN$ nearest-neighbor interactions,

and the sum over $n_{i\sigma}$ (or $n_{j\sigma'}$) can be rewritten as

$$\sum_{\langle ij \rangle} \sum_{\sigma, \sigma'} n_{i\sigma} = \frac{1}{2} \sum_{i=1}^N \sum_{j: j \sim i} \sum_{\sigma, \sigma'} n_{i\sigma} = k \sum_{i=1}^N \sum_{\sigma} n_{i\sigma}, \quad (12)$$

where $\sum_{j: j \sim i}$ is the sum over j neighbor to i and $\sum_{i=1}^N \sum_{\sigma} n_{i\sigma}$ is the total electron number operator.

In an η -electron subspace, Eq. (11) can be written in terms of the total electron number, η , and the total number of lattice sites as

$$\Delta E_V = -\frac{V}{2} \left(2k\eta - \frac{2kN}{2} \right) = Vk \left(\frac{N}{2} - \eta \right), \quad (13)$$

which is a constant energy shift. From now on we only consider the shifted version Eq. (10) of the nearest-neighbor Coulomb interaction term and refer to it as H_V .

III. TILE TROTTERIZATION

In this section, we show how to implement the time-evolution operator of generalized Hubbard models using a special variant of second-order Trotterization.

Trotter product formulas approximate the time-evolution operator by decomposing the Hamiltonian into m noncommuting terms and then applying the time evolution of these terms sequentially. Hamiltonians of the form $H = \sum_{j=1}^m H_j$ can be approximately evolved for a time t using the second-order Trotter formula

$$\left\| e^{-iHt} - \prod_{j=1}^m e^{-iH_j t/2} \prod_{j=m}^1 e^{-iH_j t/2} \right\| \leq Wt^3, \quad (14)$$

where W is the Trotter error norm given by [68]

$$W = \frac{1}{12} \sum_{b=1}^{m-1} \left\| \sum_{c>b, a>b} [[H_b, H_c], H_a] \right\| + \frac{1}{24} \sum_{b=1}^{m-1} \left\| \sum_{c>b} [[H_b, H_c], H_b] \right\|, \quad (15)$$

where $\|\cdot\|$ is the spectral norm, also known as the operator norm.

The variant of second-order Trotterization described in this paper originates from PLAQ [21], which provides an efficient implementation of the time-evolution operator of square-lattice Hubbard models by using a specific structure in both the Hubbard model and the square lattice. PLAQ uses ideas from the SO-FFFT method developed by Kivlichan *et al.* [20], which reduces the Trotter error norm by diagonalizing the hopping Hamiltonian upon each application of $e^{-iH_h t}$. This allows for a simple Trotter

decomposition into two noncommuting groups consisting of hopping terms and Coulomb terms.

A similar Hamiltonian decomposition is used in PLAQ, which also separates the Coulomb and hopping terms. However, instead of simultaneously diagonalizing the hopping terms, they are further decomposed into noncommuting hopping Hamiltonian sections that each can be more easily diagonalized, at the expense of a marginally larger Trotter error norm.

To summarize the strategies available, Fig. 1 visualizes three Trotterization approaches for the hopping Hamiltonian of a 4×4 periodic square lattice: (a) the “naive” approach, (b) PLAQ and (c) SO-FFFT. PLAQ offers a compromise, increasing parallelism and reducing non-Clifford gate counts compared to (a), while also offering a more general approach than (c) (and reduced gate counts in some scenarios). As we will show, implementing each S_1 tile in (a) has a non-Clifford cost of two arbitrary rotations, while each C_4 tile in (b) requires two arbitrary rotations and eight additional T gates; since (a) uses 32 S_1 tiles, while (b) uses only eight C_4 tiles, and given the large number of T gates required to synthesis an arbitrary rotation, the PLAQ method results in a significant reduction in non-Clifford gates.

Below, we introduce tile Trotterization, generalizing the PLAQ method by allowing quantum simulation of generalized Hubbard models on arbitrary lattices, and introducing a set of tiles to minimize the non-Clifford gate count in this general setting. Although tile Trotterization can be applied to three-dimensional lattice models, the focus in this paper is on two-dimensional models.

A. Tile Trotterization scheme

Given a generalized Hubbard model Hamiltonian of the form $H = H_h + H_C$, defined in Eq. (1), the time-evolution operator e^{-iHt} is implemented by separating the hopping Hamiltonian into S noncommuting hopping Hamiltonian sections

$$H_h = \sum_{s=1}^S H_h^s. \quad (16)$$

Each section, H_h^s , consists of N_s tile Hamiltonians in each spin sector that all commute within the same section

$$H_h^s = \sum_{n=1}^{N_s} \sum_{\sigma} H_{\sigma}^{\text{tile}, sn}, \quad (17)$$

where the hopping Hamiltonian of the n 'th tile in section s is defined as

$$H_{\sigma}^{\text{tile}, sn} = -\tau \sum_{ij} R_{ij}^{\text{tile}, sn} a_{i\sigma}^{\dagger} a_{j\sigma}, \quad (18)$$

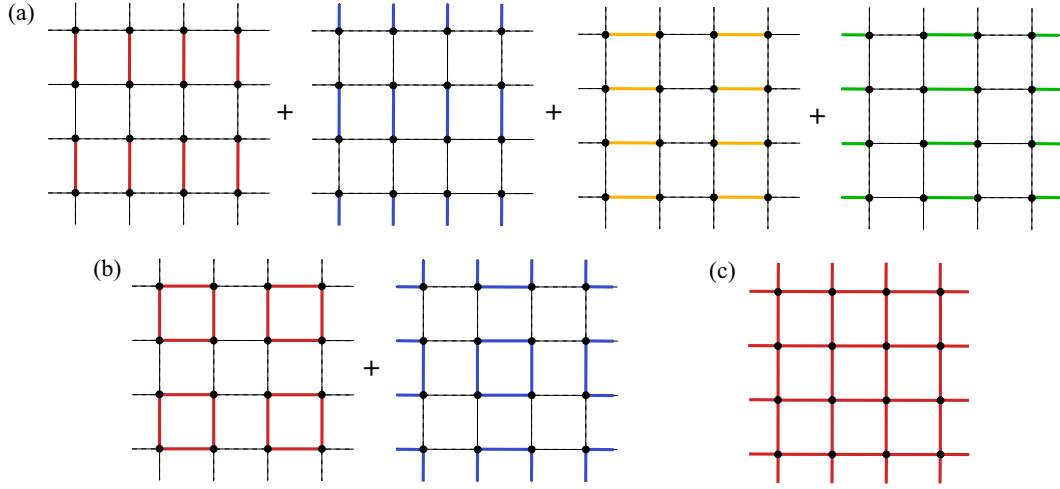


FIG. 1. Visualization of three Trotterization strategies for implementing Hamiltonian simulation for the hopping Hamiltonian, H_h , considering a 4×4 square lattice with periodic boundaries. The hopping Hamiltonian (for each spin sector) consists of terms for each “bond” or “link” on the lattice, which do not commute if two bonds share a single lattice site. The figure presents three strategies to partition terms in H_h when implementing Trotterization. In (a), H_h is partitioned into four terms, each consisting of S_1 tiles (single bonds) for which time evolution can be implemented trivially. In contrast, (c) performs $e^{-iH_h t}$ exactly by using, e.g., the fast fermionic Fourier transform (FFFT) to diagonalize H_h , as demonstrated in Ref. [20]. The approach of (b) shows the plaquette Trotterization method of Ref. [21]; here, H_h is divided into just two terms, and uses C_4 tiles, or “plaquettes,” which can be implemented with a particularly low non-Clifford cost. In the main text we discuss the trade-off between these approaches.

and $R^{\text{tile}, sn}$ is the adjacency matrix of the corresponding tile Hamiltonian. The tile Hamiltonians, Eq. (18), consist of hopping terms between neighboring lattice sites and can be represented as interaction graphs between a set of neighboring lattice points. The decomposition of the hopping Hamiltonian into sections and tiles corresponds to covering the entire lattice by tiles of different colors, where the colors are used as labels for the hopping Hamiltonian sections. These concepts are illustrated in Fig. 2, which shows examples of tiles, lattices, and a possible way to divide these lattices into sections using tiles. The tiles have to cover the entire lattice, and are distributed such that no two tiles within a section touch the same lattice site, ensuring commutativity between all tiles within a section. Note that it is possible to mix different types of tiles to cover a lattice section.

The commutativity of the tile Hamiltonians within a section means that Hamiltonian time evolution of each section can be implemented without Trotter approximation as

$$e^{-iH_h^s t} = \prod_n \prod_{\sigma} e^{-iH_{\sigma}^{\text{tile}, sn} t}, \quad (19)$$

such that the cost of $e^{-iH_h^s t}$ can be obtained by adding up the cost of implementing each tile within that section, e.g., the red hopping section of the square lattice in Fig. 2 requires eight applications of $e^{-iH_{\sigma}^{S_1} t}$ and two applications of $e^{-iH_{\sigma}^{C_4} t}$, accounting for the two spin sectors.

Time evolution of a section, $e^{-iH_h^s t}$, can be implemented exactly and efficiently if time evolution of the tile Hamiltonians can also be implemented exactly and efficiently. For general tiles, a number of rotations are needed to diagonalize the tile Hamiltonian and implement time evolution, which for fault-tolerant computation must each be implemented with a high synthesis cost. We show that the implementation cost can be significantly reduced for a certain set of complete-bipartite graph tiles, which include S_1 , S_2 , and S_4 (star graphs) and C_4 (a circle graph) that are illustrated in Fig. 2. The non-Clifford cost of implementing time evolution of these tiles is summarized in Table I. Below, we describe general features of this set of tiles and in Appendix A we provide a detailed costing of $e^{-iH_{\sigma}^{\text{tile} t}}$. Note, we also include costing of the S_3 tile in Appendix A, demonstrating concretely why certain other tiles have higher implementation cost.

More generally, we show in Appendix A6 that time evolution of any tile Hamiltonian with adjacency matrix corresponding to a complete bipartite graph of type $K_{2^a, 2^b}$, where a and b are integers, can be implemented with two arbitrary rotations and $2^{a+2} + 2^{b+2} - 8$ T gates (recovering the costs given in Table I). Notice that $K_{1, 2^k} = S_{2^k}$ and $K_{2, 2} = C_4$. When tiling a hopping Hamiltonian on a given lattice, we require that all hopping terms are covered by tiles. Therefore, a relevant measure when considering the efficiency of tiling is non-Clifford cost per bond for a given tile. A $K_{2^a, 2^b}$ graph covers 2^{a+b} bonds, resulting in an arbitrary rotation cost per bond of $2/2^{a+b}$ and a T -gate cost per bond of $(2^{a+2} + 2^{b+2} - 8)/2^{a+b}$, meaning that larger tiles

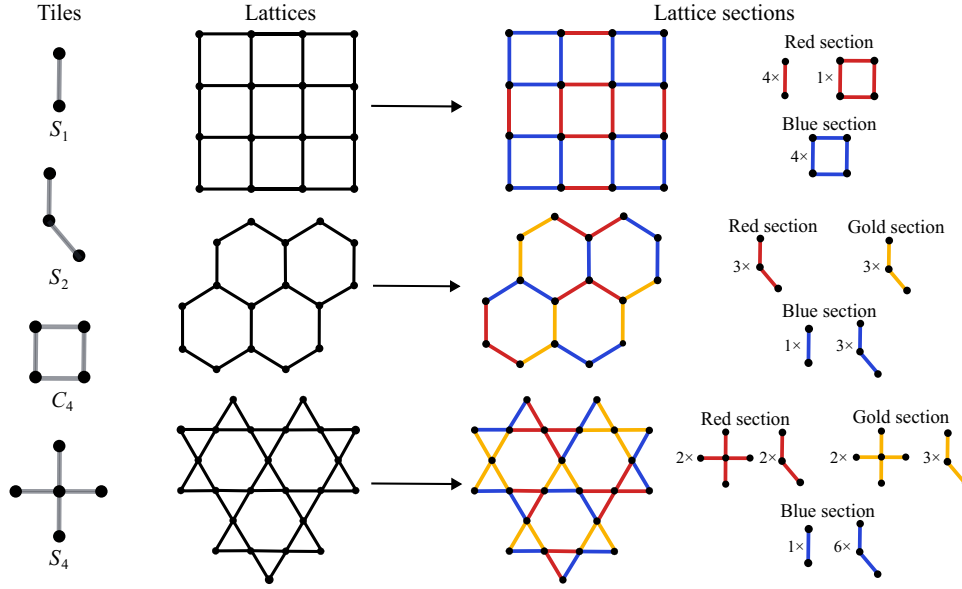


FIG. 2. Illustrations of four tiles, S_1 , S_2 , C_4 , and S_4 , and how these tiles can be used to cover three examples of lattice fragments: square, hexagonal, and kagome, in order to create lattice sections. The lattice sections are indicated by different colors: red, blue, and gold. On the right, we note how many tiles of the different types are used to cover each lattice section.

could reduce the required number of arbitrary rotations. However, these additional tiles are less relevant for two-dimensional lattices with nearest-neighbor hopping, and therefore we focus on S_1 , S_2 , S_4 , and C_4 tiles as described above.

A tile-Trotterization step of generalized Hubbard model Hamiltonians can be implemented as

$$\left\| e^{-iHt} - e^{-iH_C \frac{t}{2}} \prod_{s=1}^S e^{-iH_h^s \frac{t}{2}} \prod_{s=S}^1 e^{-iH_h^s \frac{t}{2}} e^{-iH_C \frac{t}{2}} \right\| \leq W_{\text{tile}} t^3. \quad (20)$$

A Trotter step includes applications of $e^{-iH_C t}$, and since all terms of H_C commute, the time evolution of each term can be implemented without Trotter error using two

CNOT gates and one arbitrary Z-axis rotation. The tile-Trotterization error norm, W_{tile} , can be bounded using

$$W_{\text{tile}} \leq W_{\text{SO2}} + W_h, \quad (21)$$

as shown in Appendix B, where W_{SO2} arises from the decomposition into Coulomb and hopping terms

$$\left\| e^{-i(H_h + H_C)t} - e^{-iH_C \frac{t}{2}} e^{-iH_h t} e^{-iH_C \frac{t}{2}} \right\| \leq W_{\text{SO2}} t^3, \quad (22)$$

which is evaluated using the double commutator formula given by Eq. (15). The terminology “SO2” is taken from Ref. [21], and indicates the split-operator decomposition of Coulomb and hopping terms with the ordering employed in Eq. (22). We evaluate W_{SO2} as

$$W_{\text{SO2}} = \frac{1}{12} \|[[H_C, H_h], H_h]\| + \frac{1}{24} \|[[H_C, H_h], H_C]\|. \quad (23)$$

This expression is independent of the tiling and captures the total Trotter error norm if the entire hopping Hamiltonian is simultaneously diagonalized. The additional Trotter error, W_h , depends on the number of sections and the distribution of tiles within each section

$$\left\| e^{-iH_h t} - \prod_{s=1}^S e^{-iH_h^s \frac{t}{2}} \prod_{s=S}^1 e^{-iH_h^s \frac{t}{2}} \right\| \leq W_h t^3, \quad (24)$$

and can be obtained from Eq. (15) by defining hopping Hamiltonian sections as $H_1 = H_h^1, H_2 = H_h^2, \dots, H_S = H_h^S$,

TABLE I. The non-Clifford cost of implementing $e^{-iH_{\sigma}^{\text{tile}} t}$ for tiles S_1 , S_2 , C_4 , and S_4 . The number of arbitrary rotations is equal to the number of nonzero eigenvalues of the adjacency matrix of a tile. Additional gates are required to diagonalize the tile Hamiltonian; for the four tiles considered here, this can be achieved with a non-Clifford cost of just a few T gates.

Tile	Arbitrary rotations	Additional T gates
S_1	2	0
S_2	2	4
C_4	2	8
S_4	2	12

TABLE II. Comparison of upper-bound estimates of W_{SO2} and W_h for different electron-electron interaction parameters of periodic hexagonal Hubbard and extended Hubbard models ($U/V = 2$). We take $\tau = 1$ and lattice parameters $L_x = L_y = 8$ ($N = 128$) resulting in $W_h \leq 108.6$ for each of the examples considered above (using the S_2 tiling shown in Appendix D).

Models	Model parameters		W_{SO2}	W_h/W_{SO2}
	U	V		
Hubbard	1	—	114.0	0.95
	2	—	244.8	0.44
	4	—	556.8	0.20
	8	—	1382	0.08
Extended Hubbard	1	0.5	158.9	0.68
	2	1	369.0	0.29
	4	2	942.9	0.12
	8	4	2705	0.04

where the subscripts 1, 2, and S denote the order in which the terms are applied in the Trotter decomposition. In Appendix C, we explicitly write out Eq. (15) for cases where lattices can be split into three sections and show how to compute W_h efficiently. This additional hopping Hamiltonian decomposition adds low cost in the strongly interacting regimes. We highlight this point in Table II by comparing W_{SO2} and W_h for hexagonal lattice Hubbard and extended Hubbard models for different values of U and V . The fact that $W_h \ll W_{\text{SO2}}$ in the strongly interacting regime is one of the motivations for tile Trotterization over the SO-FFFT method. However, note from Table II that in the weakly correlated regime, it can be the case that W_h/W_{SO2} is close to 1.

IV. APPLICATION OF TILE TROTTERIZATION

Here, we demonstrate how to perform costing of tile Trotterization in practice. As an example, we consider the more challenging application of the periodic hexagonal lattice extended Hubbard model in the main text. In Appendix E, we give further examples for the Hubbard model on arbitrary hexagonal lattice fragments (models without periodic boundary conditions), and the Hubbard model on a periodic hexagonal lattice. The two periodic models will then be considered in the QPE resource analysis in Sec. VIC.

A. Tile Trotterization of the periodic hexagonal lattice extended Hubbard model: gate counts

We consider the application of tile Trotterization for simulating the extended Hubbard model on the periodic hexagonal lattice. We use the periodic hexagonal lattice model described in Appendix D, which is described by the parameters L_x and L_y and contains $N = 2L_xL_y$ lattice sites.

To apply tile Trotterization, we cover the periodic hexagonal lattice with S_2 tiles to divide the lattice into

three sections: blue (b), red (r), and gold (g), as shown in Fig. 7 in Appendix D. This corresponds to decomposing the hopping Hamiltonian into three sections: $H_h = H_h^b + H_h^r + H_h^g$. The number of S_2 tiles in each section is $N_b = N_r = N_g = N/4$ for all periodic hexagonal lattice models considered in this paper, which have parameters $L_x = L_y = L$.

A single Trotter step of the extended Hubbard model, defined in Eq. (8), on the periodic hexagonal lattice is implemented as

$$e^{-iH_{EH}t} \approx e^{-i(H_I+H_V)\frac{t}{2}} e^{-iH_h^b\frac{t}{2}} e^{-iH_h^r\frac{t}{2}} e^{-iH_h^g\frac{t}{2}} \times e^{-iH_h^r\frac{t}{2}} e^{-iH_h^b\frac{t}{2}} e^{-i(H_I+H_V)\frac{t}{2}}, \quad (25)$$

with $H_{EH} = H_h + H_I + H_V$, and where we choose to implement the sections in the order $H_1 = H_h^b$, $H_2 = H_h^r$ and $H_3 = H_h^g$. Performing r repetitions of this Trotter step leads to

$$(e^{-iH_{EH}t})^r \approx e^{-i(H_I+H_V)\frac{t}{2}} \left(e^{-iH_h^b\frac{t}{2}} e^{-iH_h^r\frac{t}{2}} e^{-iH_h^g\frac{t}{2}} e^{-iH_h^r\frac{t}{2}} \times e^{-iH_h^b\frac{t}{2}} e^{-i(H_I+H_V)t} \right)^r e^{i(H_I+H_V)\frac{t}{2}}, \quad (26)$$

and therefore a single Trotter step for large r requires two applications of $e^{-iH_h^b\frac{t}{2}}$, two applications of $e^{-iH_h^r\frac{t}{2}}$, one application of $e^{-iH_h^g\frac{t}{2}}$, and one application of $e^{-i(H_I+H_V)t}$.

Under the Jordan-Wigner transformation, the system is represented by $2N$ qubits and $e^{-i(H_I+H_V)t}$ contains N terms from H_I and $6N$ terms from H_V , which can be implemented with seven layers of N arbitrary Z -axis rotations, where the angle of all gates within a given layer are the same, and 14 layers of N CNOT gates.

The cost of implementing the time evolution of the hopping terms is obtained by counting the number of applications of $e^{-iH_h^S\frac{t}{2}}$ in each section. The time evolution of the hopping Hamiltonian is decomposed into five applications of time evolution of hopping Hamiltonian sections. Each section contains $N/4$ S_2 tiles, so that we need $5 \times \frac{N}{4} \times 2$ applications of $e^{-iH_h^S\frac{t}{2}}$, accounting for the two spin sectors. The time evolution of each S_2 Hamiltonian can be implemented with two arbitrary Z -axis rotations (of the same angle) and four T gates, as shown in Appendix A. Therefore, the time evolution of the hopping term can be implemented with five layers of N arbitrary rotations of the same angle in each layer. The total non-Clifford gate cost per Trotter step is

$$N_R = 7N + 2 \times \frac{10}{4}N = 12N, \quad (27)$$

$$N_T = 4 \times \frac{10}{4}N = 10N, \quad (28)$$

where N_R is the number of arbitrary rotations and N_T is the number of T gates.

Arbitrary rotations are expensive to perform on fault-tolerant quantum computers using the surface code, most commonly being decomposed into a sequence of T and Clifford gates. Hamming weight phasing (HWP) can be used to reduce the number of arbitrary rotations by trading them for additional ancilla qubits and Toffoli gates. In particular, HWP is applicable when applying a layer of repeated rotation gates of the same angle. In this case, we can introduce ancilla qubits to calculate the Hamming weight of the logical state, and apply a smaller number of rotations to this weight [20,69]. See also Appendix E 2 for more discussion on using HWP for tile Trotterization.

The time evolution of the extended Hubbard model contains 12 layers of N arbitrary rotations, where the angle of all gates within a given layer are the same. Using HWP, we choose to implement m arbitrary rotations simultaneously using $m - 1$ ancilla qubits, with the requirement that N is an integer multiple of m , leading to a worst-case total gate cost of [21]

$$N_R = \frac{12N}{m} [\log_2(m) + 1], \quad (29)$$

$$N_T = 10N + 4 \times \frac{12N}{m} (m - 1), \quad (30)$$

where each Toffoli gate has been converted into four T gates. The total number of qubits required for this tile-Trotterization implementation with HWP is $2N + (m - 1)$.

B. Tile Trotterization: error bounds

The tile-Trotterization error norm, W_{tile} , is obtained by summing W_{SO2} and W_h , where W_{SO2} can be evaluated from the commutator bound expression in Eq. (23). In this paper we introduce new strategies for evaluating commutator bounds for the Hubbard model and the extended Hubbard model, resulting in three lemmas in Appendix F. Lemma F1 can be used to evaluate $\|[H_I, H_h], H_h\|$ for any on-site interaction Hubbard model. Our commutator bound improves upon previous state-of-the-art bounds and for the case of the periodic hexagonal lattice Hubbard model, we improve $\|[H_I, H_h], H_h\|$ by 31.5% compared to the bound in Ref. [21]. Combining Lemma 2 from Ref. [21] with our improved commutator bound, we obtain the following final expression for W_{SO2} for the periodic hexagonal lattice Hubbard model,

$$W_{\text{SO2}} \leq \frac{U^2 \tau}{24} \|R\|_1 + \frac{9.9U\tau^2}{12} N, \quad (31)$$

where $\|\cdot\|_1$ is the Schatten one norm and R is the adjacency matrix of the lattice. We compute $\|R\|_1$ numerically and use these results to bound $\|R\|_1 \leq 1.59N$ for periodic hexagonal lattices with size $4 \leq L \leq 24$.

Our second and third lemmas apply for extended Hubbard models on periodic lattices where all lattice sites have k nearest neighbors (e.g., periodic hexagonal, square, kagome, and triangular lattices). Lemmas F2 and F3 give expressions that can be used to evaluate $\|[H_C, H_h], H_h\|$ and $\|[H_C, H_h], H_C\|$, with $H_C = H_I + H_V$. We summarize the strategies used to obtain these bounds below and give expressions for obtaining W_{SO2} for the periodic hexagonal extended Hubbard model. For detailed derivations, we refer to Appendix F.

Our strategy for calculating Trotter-error bounds is to partition or divide the nested commutators into a sum of terms acting on a smaller but still nontrivial number of spin orbitals, for which we can evaluate the spectral norm either exactly or with very high precision using numerical techniques. We derive expression that allow for constructing these partitions and which clearly define the resulting subspaces. This is done by evaluating the nested commutators for a subset of terms in the Hamiltonian using commutator and anticommutator rules provided in Appendix F 1. The derived lemmas give a natural way to adjust the partitioning of commutator terms by first partitioning the lattice appropriately, for which there is significant flexibility. To obtain numerical bounds in practice, we partition the nested commutators into exactly equivalent terms (by lattice symmetry), evaluate the spectral norm of the resulting operator numerically using DMRG, and sum over partitions to obtain the total commutator bound (for details, see Appendix F).

Combining the numerical results obtained using Lemmas F2 and F3, we obtain the following final expression for W_{SO2} for the periodic hexagonal lattice extended Hubbard model with $U/V = 2$ and $L \geq 4$,

$$W_{\text{SO2}} \leq \frac{1}{12} (2.4U^2\tau + 12.5U\tau^2) N. \quad (32)$$

We must also evaluate the additional hopping Hamiltonian error norm, W_h . We calculate this as derived in Appendix C, where Eq. (C6) gives a formula for efficiently evaluating W_h for general tilings of lattices with three sections. Figure 6 (in Appendix C) plots W_h/τ^3 as a function of system size using the periodic hexagonal lattice S_2 -tiling presented in Appendix D. This yields a linear scaling of W_h with the hexagonal lattice system size. Numerically, for even L such that $4 \leq L \leq 24$, we find

$$W_h \leq 0.8532\tau^3 N. \quad (33)$$

In Table II we compare W_h and W_{SO2} for the Hubbard and extended Hubbard model, for a periodic hexagonal lattice of size $L = 8$ ($N = 128$) for different values of U and V . These results show that the Trotter decomposition of the hopping Hamiltonian contributes low additional Trotter error for systems with $U \geq 4$. We also see that W_h comprises a smaller fraction of W_{tile} for the extended Hubbard

model compared to the standard Hubbard model, due to the larger value of W_{SO_2} obtained for the former model.

V. QUBITIZATION FOR THE HEXAGONAL HUBBARD MODEL

The discussion so far has focused on quantum simulation using tile Trotterization. In order to assess this approach, we will next consider qubitization, which is a state-of-the-art approach for performing quantum simulation, therefore providing an important comparison point [38–40]. In Appendix H, we provide quantum circuits for the qubitized quantum walk operator for the periodic hexagonal Hubbard model, and include detailed resource estimates. Our construction of the qubitization circuit follows that developed in Ref. [40], but we introduce improvements that further reduce resource estimates, and perform costing of all circuit elements in detail. For brevity, in this section we briefly summarize the key costs to perform qubitized QPE using these circuits, which will be used to perform resource estimation of QPE in Sec. VI.

The quantum walk operator used in qubitization consists of SELECT, PREPARE, and reflection operations. For the Hubbard model, the asymptotically dominant cost of performing qubitized QPE comes from implementing SELECT (controlled on an ancilla). For an $L_x \times L_y$ hexagonal lattice we obtain a T -gate cost of $C_S = 40L_xL_y - 4$, or, expressed in terms of the number of lattice points, N ,

$$C_S = 20N - 4. \quad (34)$$

Note that the $\mathcal{O}(N)$ term is identical to that obtained in Ref. [40] for the square lattice, but our circuit removes an additional $\mathcal{O}(\log(N))$ contribution.

The cost of PREPARE is more involved, and we summarize the cost of each circuit element in Table V in Appendix H. Summing the contribution from each of these elements, and taking the case where $L_x = L_y = L$, the total T -gate cost of PREPARE is

$$C_P = 46\lceil \log_2 L \rceil + 4\Theta + 4\Gamma - 24\eta_L - 16, \quad (35)$$

where η_L is the largest power of 2 that is a factor of L , Θ is the number of T gates per rotation in the UNIFORM state-preparation gadgets, and Γ is the number of T gates per each other rotation in PREPARE. Therefore, the cost of PREPARE is only logarithmic in the lattice dimension, L . For our resource estimates, we choose $\Theta = 10$ and $\Gamma = 40$. The controlled reflection operator adds an additional T -gate cost of

$$C_R = 32\lceil \log_2 L \rceil + 77, \quad (36)$$

using the scheme of Ref. [70] to perform the reflection with a single ancilla qubit.

The controlled walk operator also requires additional flag and ancilla qubits. Once again taking the case of $L_x = L_y = L$, the total number of qubits to implement the circuits in Appendix H is

$$N_{\text{qubits}} = 2N + 6\lceil \log_2 L \rceil + 15. \quad (37)$$

VI. QUANTUM PHASE ESTIMATION

In this section, we provide resource estimates for QPE based on tile Trotterization and qubitization. We begin by describing the QPE schemes used for each method.

A. Trotterized QPE

We follow the quantum phase estimation costing procedure for Trotterization methods described by Kivlichan *et al.* in Ref. [20]. This phase estimation scheme relies on adaptive phase estimation techniques [71,72], which use a single control qubit. We define the maximum allowed error in the energy estimate as ϵ , and following Ref. [21], assign $x\epsilon$ of the total error to rotation synthesis error and $(1-x)\epsilon$ to Trotter and phase estimation error. This allows us to determine the total number of Trotter steps required for the phase estimation procedure as [20,21]

$$N_{\text{PE}} = 6.203 \frac{\sqrt{W}}{(1-x)^{3/2}\epsilon^{3/2}}, \quad (38)$$

where W is the Trotter error norm.

The T -gate cost per Trotter step from synthesis of rotation gates, N_{RT} , using repeat-until-success synthesis [73], can be evaluated as [21]

$$N_{\text{RT}} = N_R \left(1.15 \log_2 \left(\frac{N_R \sqrt{3W}}{x\sqrt{1-x}\epsilon^{3/2}} \right) + 9.2 \right), \quad (39)$$

where N_R is the number of arbitrary rotations per Trotter step. The total T -gate cost for performing Trotterized QPE is given by

$$N_{T,\text{Trotter}} = N_{\text{PE}}(N_{\text{RT}} + N_T), \quad (40)$$

where N_T is the number of T gates per Trotter step that do not come from the synthesis of arbitrary rotations.

To assess the T -gate complexity, note that the number of Trotter steps scales as $N_{\text{PE}} = \mathcal{O}(N^{1/2}\epsilon^{-3/2})$ for the Hubbard model and extended Hubbard model, and the T -gate cost per step *aside from rotation synthesis* is $N_T = \mathcal{O}(N)$. If HWP is applied, then $N_R = \mathcal{O}(\log(N))$ [from Eq. (29)] in which case N_{RT} does not contribute asymptotically,

resulting in a T -gate complexity of

$$\mathcal{O}(N^{3/2}\epsilon^{-3/2}). \quad (41)$$

If HWP is not applied then $N_R = \mathcal{O}(N)$ [from Eq. (27)], and the overall T complexity picks up a logarithmic factor

$$\mathcal{O}(N^{3/2}\epsilon^{-3/2}\log(N\epsilon^{-1})). \quad (42)$$

In practice, the precision of rotation synthesis can be improved with low additional cost, and the number of T gates per arbitrary rotation is often set as a constant, meaning the logarithmic factor can be disregarded. The scaling with and without HWP reported here applies for both the extended Hubbard model (Sec. IV) and the Hubbard model systems (Secs. E 1 and E 2). The Hubbard model systems, however, have smaller prefactors coming from the lower value of W_{SO2} and fewer arbitrary rotation layers. Note that the scaling of QPE with tile Trotterization and HWP matches that for the square lattice with plaquette Trotterization, see Ref. [21], Eq. (F10).

The phase estimation procedure used for tile Trotterization requires two qubits in addition to the $2N$ system qubits: one for adaptive phase estimation and one for repeat-until-success synthesis [20,21]. Through numerical tests of the overall T -gate cost, we find that optimal values of x are approximately $x = 0.03$ without applying HWP and $x = 0.01$ with HWP.

B. Qubitized QPE

For qubitized QPE, we follow the phase estimation scheme described in Ref. [6], which uses $N_{\mathcal{W}}$ repetitions of the qubitized quantum walk operator, $\mathcal{W}$, which consists of the operators controlled SELECT (S), PREPARE (P), PREPARE † ($P^\dagger$) and controlled reflection (R). The number of repetitions of the qubitized quantum walk operator is given by

$$N_{\mathcal{W}} = \left\lceil \frac{\pi\lambda}{2\epsilon} \right\rceil, \quad (43)$$

where λ is the L1 norm of the Hamiltonian and ϵ is the allowed error in phase estimation. For the periodic hexagonal lattice Hubbard model, the L1 norm is

$$\lambda = \left(3\tau + \frac{U}{4}\right)N. \quad (44)$$

The total qubitized QPE T -gate cost is

$$N_{T,N_{\mathcal{W}}} = \left\lceil \frac{\pi\lambda}{2\epsilon} \right\rceil (C_S + C_P + C_{P^\dagger} + C_R), \quad (45)$$

where C_Q represents the T -gate cost of operation Q . Note that our implementation has the same cost for P and $P^\dagger$.

The T -gate complexity of qubitized QPE on the periodic hexagonal lattice Hubbard model scales as

$$\mathcal{O}(N^2\epsilon^{-1}). \quad (46)$$

This matches the asymptotic scaling for the square lattice obtained in Ref. [40], Eq. (61). However, our analysis also considers the non-Clifford cost of all circuit elements.

Note that this phase estimation scheme uses $\alpha_{\text{PE}} = 2\lceil\log_2(N_{\mathcal{W}} + 1)\rceil - 1$ ancilla qubits for the control register [6,74], in addition to the qubits quoted in Eq. (37). Therefore, the number of qubits depend on the Hubbard model parameters, U and τ , as well as ϵ . The unary iterator over the walk operator also requires $(4N_{\mathcal{W}} - 4)$ T gates to implement, in addition to the per-walk operator T -gate counts given in Eq. (45).

C. QPE resource estimates

In our QPE resource estimates, we disregard the cost of state preparation, although this is an important and non-trivial problem [75]. Efforts have been made to efficiently prepare correlated fermionic states of Hubbard models on quantum computers [53,76]. The state-preparation problem, and its cost for Hubbard models, is further discussed in Yoshioka *et al.* [14]. We will continue by assuming that an initial state can be prepared with a sufficient overlap with the target state.

In Table IV of Appendix G we provide Trotter error norms, qubit counts, arbitrary rotation costs and T costs per Trotter step to perform the simulations considered in this section. The qubitization resource estimates are obtained from Eq. (45), with the cost of each quantum walk operator element given by Eqs. (34)–(36).

We begin our QPE resource analysis by comparing the performance of tile Trotterization (Appendix E 2) with our qubitization-based approach (Sec. V) for simulating the periodic hexagonal lattice Hubbard model. This is done to highlight the difference in scaling and performance of the two approaches for different system sizes N , and for different maximum errors, ϵ . We consider two different error regimes in our QPE costing analysis: relative and additive errors. Relative errors are often appropriate for condensed phase applications, for example, when calculating the energy per unit cell, while absolute errors are often considered in molecular quantum chemistry, or when studying defects in solids. For a more detailed discussion of this point, we refer to Ref. [20]. The QPE resource comparisons are shown in Fig. 3 using either tile Trotterization, tile Trotterization with HWP ($\alpha = N/2 - 1$ ancillas), or qubitization for hexagonal lattice Hubbard models with parameters $L_x = L_y = L$, $U = 4$ and $\tau = 1$.

Figure 3(a) presents the total T -gate count as a function of the number of lattice sites, N , and taking the relative error case with $\epsilon = 0.005N$. The T -gate cost of qubitized QPE is lower than for Trotterized QPE for

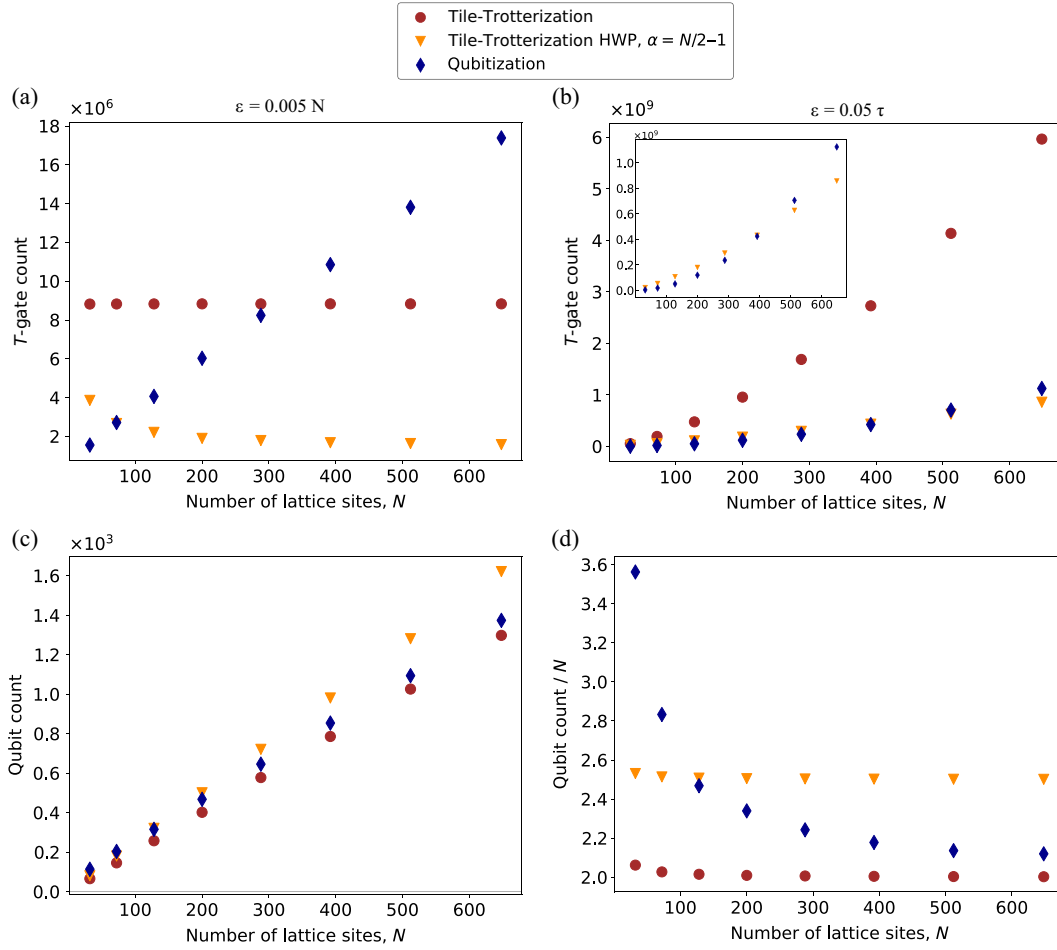


FIG. 3. Resource estimates for QPE performed on a periodic hexagonal lattice Hubbard model with $L_x = L_y = L$, using tile Trotterization (see Appendix E 2), tile Trotterization HWP (Hamming weight phasing; with $\alpha = N/2 - 1$ ancilla qubits, see Appendix E 2), and qubitization (see Sec. V). (a) Total T -gate count (including T gates from arbitrary rotation synthesis) as a function of the number of lattice sites, N , with $\epsilon = 0.005N$. (b) Total T -gate count (including T gates from arbitrary rotation synthesis) as a function of the number of lattice sites, N , with $\epsilon = 0.05\tau$. The inset shows the total T -gate count for tile Trotterization HWP and qubitization. Note that the total T -gate counts of tile-Trotterization-based QPE are upper bounds. (c) The total number of qubits used for each method. (d) The total number of qubits used for each method, divided by N .

small N (and small ϵ) but grows linearly with N in the relative error regime, eventually becoming more expensive than both Trotterization methods. The cost of tile-Trotterization-based QPE remains constant in this error regime while the tile-Trotterization HWP simulation also asymptotically approaches a constant T -gate cost. At small N , the total T -gate cost of tile-Trotterization HWP is dominated by a $\log(N)$ term from the number of arbitrary rotations per Trotter step, N_{RT} , and therefore the T -gate cost initially decreases as the logarithmic term is suppressed by the linearly growing ϵ . Eventually, the linearly growing N_T dominates the total T -gate cost per Trotter step, resulting in the total T -gate cost approaching a constant value.

Figure 3(b) shows the total T -gate count as a function of N , with an absolute error of $\epsilon = 0.05\tau$. The inset in the

top-left corner shows the T -gate cost of tile-Trotterization HWP and qubitization only, to more clearly highlight their different scaling, $\mathcal{O}(N^{3/2})$ for tile-Trotterization HWP and $\mathcal{O}(N^2)$ for qubitization. In this error regime, we once again see that the qubitization approach performs better than tile-Trotterization HWP for the smaller systems, but tile Trotterization HWP eventually reaches a lower T -gate cost due to its better system size scaling. These results only demonstrate the total T -gate cost of periodic Hubbard model simulations. We note that simulating hexagonal lattice fragments (or nanographene) with tile Trotterization (Appendix E 1) will have similar costs for models of same size.

Figure 3(c) presents the number of qubits used for the three simulation methods, and Fig. 3(d) shows the qubit count divided by N to highlight the differences in qubit

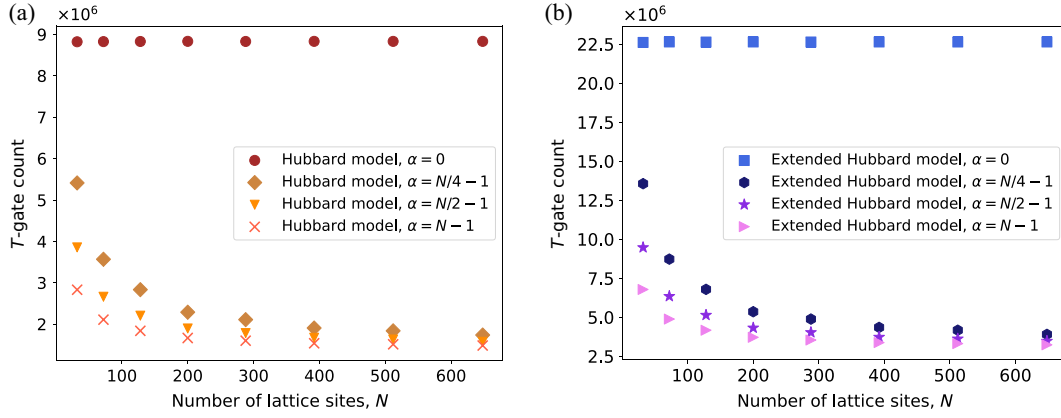


FIG. 4. Upper bounds on the total T -gate count (including T gates from arbitrary rotation synthesis) for performing QPE with tile Trotterization on periodic hexagonal lattice Hubbard models with $L_x = L_y = L$ and $\epsilon = 0.005N$, as a function of the number of lattice sites, N . We compare four different simulation approaches with $\alpha = 0$, $\alpha = N/4 - 1$, $\alpha = N/2 - 1$, and $\alpha = N - 1$ ancilla qubits for HWP for (a) the Hubbard model (Appendix E 2) and (b) the extended Hubbard model (Sec. IV).

counts. We have chosen $U = 4$, $\tau = 1$, and $\epsilon = 0.05\tau$ for the qubitization-based qubit count; the Trotter qubit counts are independent of these parameters. These figures show that the qubit counts for the three approaches are comparable and grow similarly with the system size. However, for small lattice sizes, the qubitization-based approach requires significantly more ancilla qubits relative to the $2N$ system qubits.

Figure 4 presents a comparison between Trotterized QPE resource estimates for (a) the Hubbard model and (b) the extended Hubbard model, and also the effect of varying the number of ancilla qubits, α , used for the HWP procedure. In these plots we take a relative target accuracy of $\epsilon = 0.005N$. In these QPE resource estimates, we provide four examples using $\alpha = 0$, $\alpha = N/4 - 1$, $\alpha = N/2 - 1$, and $\alpha = N - 1$ ancilla qubits for HWP. Note that the $\alpha = 0$ and $\alpha = N/2 - 1$ results for the Hubbard model correspond to those also shown in Fig. 3(a). The difference in T -gate cost for different α is largest for the small system sizes because the cost in this regime is dominated by $\log(N)$ terms with different constant factors, as shown in Eq. (E14). These differences are suppressed by ϵ growing linearly with the system size. The T -gate cost of the tile-Trotterization HWP simulations asymptotically approaches the same constant value dominated by N_T , as given by Eq. (E17).

In the relative error regime with $\epsilon = 0.005N$, our upper bounds on the T -gate cost are just below 9×10^6 for the Hubbard model and around 22.5×10^6 for the extended Hubbard model simulations using a direct tile-Trotterization approach without HWP ($\alpha = 0$).

VII. DISCUSSION AND OUTLOOK

This paper presents a Trotterization method that can be used to implement the time-evolution operator for

generalized Hubbard models on arbitrary lattices. Tile Trotterization allows for constructing Trotter decompositions of generalized Hubbard models by using a set of efficiently implementable tiles to decompose the hopping Hamiltonian, and provides a straightforward method to obtain per-Trotter-step gate counts. Tile Trotterization can also be combined with Hamming weight phasing to significantly reduce the number of arbitrary rotation gates required per Trotter step. In this paper, we focus on simulations of local two-dimensional Hubbard models, but the tile Trotterization scheme allows for arbitrary density-density interactions and can be extended to include longer-range hopping terms by constructing additional hopping Hamiltonian sections. The S_4 tiles are a natural choice for constructing sections of the next-nearest-neighbor hopping graph of the square lattice. The tile Trotterization scheme can also be applied to simulate three-dimensional lattice Hubbard models, however, specific examples and optimal tile coverings of three-dimensional lattices are left for future work.

We demonstrated applications of tile Trotterization for energy estimation by QPE for both the Hubbard model and the extended Hubbard model with nearest-neighbor Coulomb interactions, showing that the method achieves a T -gate complexity $\mathcal{O}(N^{3/2}\epsilon^{-3/2})$, where N is the number of lattice sites and ϵ is the maximum allowed error in the energy. We demonstrated the use of numerical simulations, including tensor network methods, combined with proof techniques to partition the required nested commutators into sets of smaller but nontrivial operators, allowing us to achieve tight bounds on the Trotter error. We used our commutator bounds to evaluate the Trotter error of the periodic hexagonal lattice Hubbard model and extended Hubbard model. Our commutator bound lemmas, and the numerical techniques used to evaluate them, can also be applied to other local Hubbard models and the same

complexity can be achieved, e.g., for square or kagome lattices.

We also constructed and optimized the qubitized quantum walk operator for the hexagonal lattice Hubbard model and provided detailed Toffoli gate, T -gate, and qubit counts. These were used to perform resource estimation for qubitized quantum phase estimation, demonstrating a T -gate complexity of $\mathcal{O}(N^2\epsilon^{-1})$. The Trotterization-based QPE approach studied in this paper has better scaling with respect to system size, which results in lower T -gate counts for simulations of large systems, especially when ϵ is allowed to scale extensively with N .

The results presented here demonstrate that a range of classically nontrivial model Hamiltonians can be simulated with 10^6 – 10^7 non-Clifford gates. This is several orders of magnitude less than found in many resource-estimation studies of *ab initio* chemical systems, suggesting that such model Hamiltonians are promising candidate applications for early fault-tolerant quantum computers. Going forward, it will be important to consider more detailed costing of this problem for early fault-tolerant architectures. Current resource estimates focus on counting non-Clifford gates, which have historically been expected to be most expensive to perform under traditional QEC schemes. However, recent proposals have questioned this understanding [77]. Previous resource estimates have also focused on particular architectures where the cost of Clifford gates can be ignored [78], which may not be appropriate for early FT devices. Therefore, an interesting and important task is to consider a more detailed compilation to an early FT architecture. Here, relevant questions include how routing can be efficiently performed under lattice surgery; how gates may be best parallelized; and efficient schemes for performing rotation gates with lower costs. An attempt at this was recently performed in Ref. [16], using the STAR architecture [79] and a Trotter scheme essentially using S_1 tiles.

Performing such a compilation to an early fault-tolerant architecture will introduce a number of considerations beyond those discussed in this paper. These include, for example, the cost of Clifford gates under lattice surgery and the parallelizability of a given logical circuit. We expect that the tile Trotterization scheme here is a particularly promising candidate for such early FT applications. In addition to the low gate count, circuits are relatively simple, consisting of just a small number of rotation gate layers per Trotter step (in addition to fermionic swap layers). Such considerations will also affect the efficiency of various schemes; for example, the Hamming-weight phasing method considered in this work reduces the number of rotation gates, but at the cost of additional qubits, two-qubit T gates and significantly reduced parallelizability. Indeed, applying HWP to implement a single layer of n rotation gates requires $n - 1$ Toffoli gates in adder circuits, which cannot be readily parallelized; given these costs, it

remains to be seen if applying HWP would give any advantage when implemented in a practical FT architecture. Similarly, the qubitization implementation is asymptotically dominated by unary iterators in SELECT consisting of Toffoli gates, which cannot be easily parallelized. Therefore, tile Trotterization without HWP may have greater benefits than represented in Fig. 3, for example. Separately, we also note that the tile-Trotterization scheme could be performed in combination with early fault-tolerant algorithms such as statistical phase estimation to significantly reduce circuit depths compared to those presented here [32,80]. The scheme presented in this paper also allows for significantly more complicated model Hamiltonians, such as the PPP model, which provide a link to *ab initio* quantum chemistry. Given all of these benefits, and the significant challenge of performing dynamics of model Hamiltonians in strongly correlated regimes by conventional methods, we believe that this is a promising area for practical quantum algorithms going forward.

ACKNOWLEDGMENTS

This work is supported by the Novo Nordisk Foundation, Grant No. NNF22SA0081175, NNF Quantum Computing Programme. We thank Matthew S. Teynor and Earl T. Campbell for feedback on the manuscript. We would also like to acknowledge the anonymous referees of this manuscript for their careful reviews, which led to changes that improved our manuscript.

DATA AVAILABILITY

Data files for the numerical calculation of Trotter error norms are uploaded to Zenodo [81].

APPENDIX A: TIME EVOLUTION OF TILE HAMILTONIANS

This Appendix describes how to diagonalize and perform time evolution of tile Hamiltonians for five examples of tiles: S_1 , S_2 , S_3 , S_4 , and C_4 . We also consider a more systematic analysis using the abstract set of tiles with adjacency matrices corresponding to the complete bipartite graph $K_{n,m}$. The Hamiltonian of a tile is defined as

$$H_\sigma^{\text{tile}} = -\tau \sum_{ij} R_{ij}^{\text{tile}} a_{i\sigma}^\dagger a_{j\sigma}, \quad (\text{A1})$$

where R^{tile} is the adjacency matrix describing the nearest-neighbor hopping terms contained in the tile Hamiltonian, meaning $R_{ij}^{\text{tile}} = 1$ if the tile Hamiltonian contains a hopping term between site i and j , and otherwise $R_{ij}^{\text{tile}} = 0$. The time-evolution operator can be implemented as

$$e^{-iH_\sigma^{\text{tile}}t} = e^{i\tau \sum_{ij} R_{ij}^{\text{tile}} a_{i\sigma}^\dagger a_{j\sigma}t} = e^{i\tau M_\sigma^{\text{tile}}t}, \quad (\text{A2})$$

where M_σ^{tile} is defined as

$$M_\sigma^{\text{tile}} = \sum_{ij} R_{ij}^{\text{tile}} a_{i\sigma}^\dagger a_{j\sigma}. \quad (\text{A3})$$

Since the adjacency matrix, R^{tile} , is real symmetric, it has real eigenvalues, λ_e , and eigenvectors, v_e , and it can be diagonalized as $R^{\text{tile}} = \sum_e \lambda_e v_e v_e^\dagger$. Thus,

$$M_\sigma^{\text{tile}} = \sum_{ij} \sum_e \lambda_e (v_e)_i (v_e)_j a_{i\sigma}^\dagger a_{j\sigma}, \quad (\text{A4})$$

$$= \sum_e \lambda_e \left(\sum_i (v_e)_i a_{i\sigma}^\dagger \right) \left(\sum_j (v_e)_j a_{j\sigma} \right), \quad (\text{A5})$$

$$= \sum_e \lambda_e b_{e\sigma}^\dagger b_{e\sigma}, \quad (\text{A6})$$

for $b_{e\sigma} = \sum_j (v_e)_j a_{j\sigma}$, where $(v_e)_j$ is the j 'th element of eigenvector v_e . Whenever an eigenvalue λ_e equals 0, this decomposition will reduce the number of required arbitrary rotations. Throughout this section, we only focus on the eigenvectors that have nonzero eigenvalues.

Tile Hamiltonians of any type can be diagonalized and time evolution can be performed without Trotter error. However, the difference in the cost of the diagonalization procedure between different types of tiles contribute significantly to the total cost of implementing $e^{-iH_\sigma^{\text{tile}}t}$. We show that tile Hamiltonians with hopping terms corresponding to star interaction graphs of type S_1 , S_2 , and S_4 and the circle interaction graph C_4 , where the subscript denotes the number of edges, can be efficiently diagonalized. Figure 5(a) shows five tiles: S_1 , S_2 , S_3 , C_4 , and S_4 . These tiles (except for S_3) are part of the same family of tiles: their adjacency matrices are complete bipartite graphs of type $K_{2a,2b}$, where a and b are integers (the costing of this abstract set of tiles

is presented in Sec. A 6). This structure has two major benefits for the cost of implementing time evolution: (1) they have a particularly cheap diagonalization procedure and (2) they all require just two arbitrary rotations. We also include a costing of the S_3 tile Hamiltonian (which is also a complete bipartite graph but which does not have the same eigenvector structure) to show an example where the diagonalization procedure significantly increases the total cost of $e^{-iH_\sigma^{\text{tile}}t}$. There might exist other tiles with an efficient implementation of $e^{-iH_\sigma^{\text{tile}}t}$, but the tiles given in Fig. 5(a) are sufficient for most practical purposes.

The tiles considered here, except for S_3 , can be diagonalized using the fermionic operator F_{ij} [21,82]. We let F_{ij} act on creation and annihilation operators defined in a two-spin-orbital subspace as

$$F_{ij} a_{i\sigma} F_{ij}^\dagger = \frac{1}{\sqrt{2}} (a_{i\sigma} + a_{j\sigma}), \quad (\text{A7})$$

$$F_{ij} a_{j\sigma} F_{ij}^\dagger = \frac{1}{\sqrt{2}} (a_{i\sigma} - a_{j\sigma}). \quad (\text{A8})$$

These examples are given for annihilation operators and can be trivially extended to creation operators. Under the Jordan-Wigner (JW) transformation, the qubit representation of F_{ij} acting on neighboring fermions i and j in the JW string is given by [21]

$$F_{ij} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad (\text{A9})$$

which can be implemented using just 2 T gates as shown in Fig. 5(b) (also see Fig. 8 of Ref. [20]).

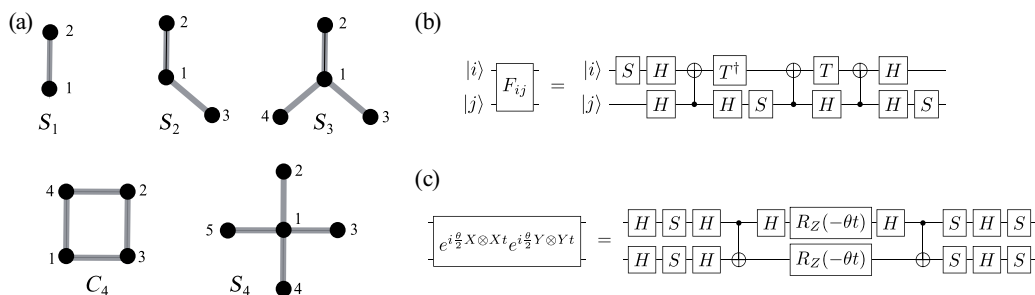


FIG. 5. (a) Illustrations of the five tiles considered in this paper: S_1 , S_2 , S_3 , C_4 , and S_4 . Each lattice site is indexed with a number that corresponds to the order of the spin orbitals in the JW string, and the edges represent nearest-neighbor hopping terms between neighboring lattice sites. The sites are ordered in this way to minimize the number of required f_{swap} operations when implementing $e^{-iH_\sigma^{\text{tile}}t}$. (b) Quantum circuit diagram for implementing fermionic operator F_{ij} on a two-qubit $\{i,j\}$ subspace in the Jordan-Wigner representation. (c) Quantum circuit diagram for implementing $e^{i\frac{\theta}{2}X \otimes X} e^{i\frac{\theta}{2}Y \otimes Y}$, up to a global phase factor of $e^{i\frac{\pi}{2}}$. Note that when controlling this operation on an ancilla, this extra controlled phase will be a Clifford gate.

The qubit implementation of fermionic operators under the JW transformation requires keeping track of the anti-symmetric fermionic properties during the computation. The qubit implementation of F_{ij} is nonlocal in the JW string ordering, such that implementing F_{ij} on nonadjacent qubits in the JW ordering would introduce additional complexity. This can be solved by using fermionic SWAP gates to update the JW ordering as needed, so that F_{ij} is only applied on adjacent qubits. The qubit representation of the fermionic SWAP gate is given by [20]

$$f_{\text{swap}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad (\text{A10})$$

which can be implemented using a SWAP gate followed by a CZ gate, having Clifford cost only.

In our costing of $e^{-iH_{\sigma}^{\text{tile}}t}$, we assume an initial JW ordering constructed to avoid using (or in the case of the S_4 tile, to minimize) fermionic SWAP gates when diagonalizing a tile Hamiltonian. The initial JW ordering within a given tile follow the labels of the sites given in Fig. 5(a). In general, the Clifford costings for implementing time evolution of the tile Hamiltonian depend on the initial JW ordering of the qubits used.

It is also important to note that different hopping sections will require different JW orderings. Therefore, in a full implementation we would require layers of f_{swap} gates in between sections, but we do not account for this Clifford only cost in this paper.

The rest of Appendix A will show how to diagonalize the tile Hamiltonians and provide quantum circuits for implementing $e^{-iH_{\sigma}^{\text{tile}}t}$ for the S_1 , S_2 , S_3 , C_4 , and S_4 tiles. We provide non-Clifford and Clifford gate counts for the implementation of each $e^{-iH_{\sigma}^{\text{tile}}t}$, and the cost of these are summarized in Table III.

TABLE III. The cost for implementing $e^{-iH_{\sigma}^{\text{tile}}t}$ for S_1 , S_2 , S_3 , C_4 and S_4 tiles, assuming an initial JW-ordering of spin orbitals given in Fig. 5(a). Here, “arbitrary rotations” refers to rotations dependent on time step, t , while “other non-Clifford rotations” are non-Clifford Z-axis rotations to diagonalize the tile Hamiltonian, not including T gates, which would therefore have a significant synthesis cost. Note that these tiles require two arbitrary rotations each while the other gate costs scale with the size of the tile. To obtain the cost, we assume an F_{ij} implementation using the quantum circuit in Fig. 5(b) and an implementation of $e^{iX \otimes X \theta} e^{iY \otimes Y \theta}$, where θ is a parameter dependent on τ and t , using the quantum circuit in Fig. 5(c). Further optimization of the Clifford gate costs could be performed, for example by merging gates from F_{ij} with gates from $e^{iX \otimes X \theta} e^{iY \otimes Y \theta}$.

Tile	Arbitrary rotations	Other non-Clifford rotations	T gates	CNOT gates	Hadamard gates	S gates	f_{swap}
S_1	2	0	0	2	8	6	0
S_2	2	0	4	8	20	12	0
C_4	2	0	8	14	32	18	0
S_4	2	0	12	20	44	24	2
S_3	2	4	4	14	32	18	0

1. S_1 tile

The S_1 tile has hopping terms between two spin orbitals $\phi_{1\sigma}$ and $\phi_{2\sigma}$, and contains hopping terms encoded by the adjacency matrix

$$R^{S_1} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad (\text{A11})$$

with eigenvalues $\lambda_+ = 1$ and $\lambda_- = -1$ and corresponding eigenvectors

$$v_+ = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{pmatrix}, \quad v_- = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{2} \end{pmatrix}. \quad (\text{A12})$$

Therefore, $M_{\sigma}^{S_1}$ can be written as

$$M_{\sigma}^{S_1} = b^{\dagger} b - c^{\dagger} c, \quad (\text{A13})$$

where b and c are given by

$$b = \frac{1}{\sqrt{2}}(a_{1\sigma} + a_{2\sigma}), \quad (\text{A14})$$

$$c = \frac{1}{\sqrt{2}}(a_{1\sigma} - a_{2\sigma}). \quad (\text{A15})$$

We can construct b and c using $V = F_{12}$ as

$$b = Va_{1\sigma}V^{\dagger}, \quad (\text{A16})$$

$$c = Va_{2\sigma}V^{\dagger}, \quad (\text{A17})$$

and implement $e^{-iH_{\sigma}^{S_1}t}$ on $|\phi_{1\sigma}\phi_{2\sigma}\rangle$, as

$$e^{-iH_{\sigma}^{S_1}t} = Ve^{i\tau(a_{1\sigma}^{\dagger}a_{1\sigma} - a_{2\sigma}^{\dagger}a_{2\sigma})t}V^{\dagger}. \quad (\text{A18})$$

Under the Jordan-Wigner transformation, we can write

$$e^{i\tau(a_{1\sigma}^{\dagger}a_{1\sigma} - a_{2\sigma}^{\dagger}a_{2\sigma})t} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{i\tau t} & 0 & 0 \\ 0 & 0 & e^{-i\tau t} & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (\text{A19})$$

which means Eq. (A18) can be further compiled to

$$e^{-iH_{\sigma}^{S_1}t} = F_{12}e^{i\tau(a_{1\sigma}^\dagger a_{1\sigma} - a_{2\sigma}^\dagger a_{2\sigma})t}F_{12}^\dagger$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\tau t) & i \sin(\tau t) & 0 \\ 0 & i \sin(\tau t) & \cos(\tau t) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = e^{i\frac{\tau}{2}X \otimes X t} e^{i\frac{\tau}{2}Y \otimes Y t}. \quad (\text{A20})$$

This operation can be implemented using the quantum circuit given in Fig. 5(c), using $\frac{\theta}{2} = \tau$ [see Eq. (18) in Ref. [83]]. This results in a non-Clifford cost of two arbitrary rotations of the same angle and additional Clifford cost of two CNOT gates, six S gates, and eight Hadamard gates for implementing $e^{-iH_{\sigma}^{S_1}t}$.

2. S_2 tile

The S_2 tile has hopping terms between three spin orbitals $\phi_{1\sigma}$, $\phi_{2\sigma}$, and $\phi_{3\sigma}$, described by the adjacency matrix

$$R^{S_2} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad (\text{A21})$$

with eigenvalues $\lambda_0 = 0$, $\lambda_+ = \sqrt{2}$, and $\lambda_- = -\sqrt{2}$ and corresponding eigenvectors

$$v_+ = \begin{pmatrix} 1/\sqrt{2} \\ 1/2 \\ 1/2 \end{pmatrix}, \quad v_- = \begin{pmatrix} 1/\sqrt{2} \\ -1/2 \\ -1/2 \end{pmatrix}. \quad (\text{A22})$$

We can write $M_{\sigma}^{S_2}$ as

$$M_{\sigma}^{S_2} = \sqrt{2}b^\dagger b - \sqrt{2}c^\dagger c, \quad (\text{A23})$$

where b and c are given by

$$b = \frac{1}{\sqrt{2}}a_{1\sigma} + \frac{1}{2}(a_{2\sigma} + a_{3\sigma}), \quad (\text{A24})$$

$$c = \frac{1}{\sqrt{2}}a_{1\sigma} - \frac{1}{2}(a_{2\sigma} + a_{3\sigma}). \quad (\text{A25})$$

Employing the unitary transformation $V = F_{23}F_{12}$, we can write

$$b = Va_{1\sigma}V^\dagger, \quad (\text{A26})$$

$$c = Va_{2\sigma}V^\dagger, \quad (\text{A27})$$

and so can implement the time evolution of the S_2 tile on $|\phi_{1\sigma}\phi_{2\sigma}\phi_{3\sigma}\rangle$ as

$$e^{-iH_{\sigma}^{S_2}t} = Ve^{i\tau\sqrt{2}(a_{1\sigma}^\dagger a_{1\sigma} - a_{2\sigma}^\dagger a_{2\sigma})t}V^\dagger. \quad (\text{A28})$$

This can be further compiled using the same strategy as shown in Eq. (A20), leading to the final expression

$$e^{-iH_{\sigma}^{S_2}t} = F_{23}e^{i\frac{\tau}{\sqrt{2}}X_1 \otimes X_2 t} e^{i\frac{\tau}{\sqrt{2}}Y_1 \otimes Y_2 t} F_{23}^\dagger. \quad (\text{A29})$$

The total gate cost for implementing the time evolution of the S_2 tile Hamiltonian is therefore two arbitrary rotations, four T gates, eight CNOT gates, 12 S gates, and 20 Hadamard gates.

3. C_4 tile

The plaquette (C_4) tile, which was first introduced in Ref. [21], has hopping terms between four spin orbitals, $\phi_{1\sigma}$, $\phi_{2\sigma}$, $\phi_{3\sigma}$, and $\phi_{4\sigma}$. Using the ordering in Fig. 5(a), the adjacency matrix is given by

$$R^{C_4} = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}, \quad (\text{A30})$$

for the Jordan-Wigner string ordering $|\phi_{1\sigma}\phi_{2\sigma}\phi_{3\sigma}\phi_{4\sigma}\rangle$. This adjacency matrix has two zero eigenvalues, $\lambda_+ = 2$ and $\lambda_- = -2$ and corresponding eigenvectors

$$v_+ = \frac{1}{2} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \quad v_- = \frac{1}{2} \begin{pmatrix} 1 \\ 1 \\ -1 \\ -1 \end{pmatrix}. \quad (\text{A31})$$

Then, $M_{\sigma}^{C_4}$ can be written as

$$M_{\sigma}^{C_4} = 2b^\dagger b - 2c^\dagger c, \quad (\text{A32})$$

with b and c given by

$$b = \frac{1}{2}(a_{1\sigma} + a_{2\sigma} + a_{3\sigma} + a_{4\sigma}), \quad (\text{A33})$$

$$c = \frac{1}{2}(a_{1\sigma} + a_{2\sigma} - a_{3\sigma} - a_{4\sigma}). \quad (\text{A34})$$

Employing the unitary transformation $V = F_{34}F_{21}F_{23}$, we can write

$$b = Va_{2\sigma}V^\dagger, \quad (\text{A35})$$

$$c = Va_{3\sigma}V^\dagger, \quad (\text{A36})$$

and implement the time evolution as

$$e^{-iH_{\sigma}^{C_4}t} = Ve^{i\tau 2(a_{2\sigma}^\dagger a_{2\sigma} - a_{3\sigma}^\dagger a_{3\sigma})t}V^\dagger. \quad (\text{A37})$$

This can be further compiled using the same strategy as shown in Eq. (A20). Then, the time evolution of the C_4 tile

Hamiltonian can be implemented as

$$e^{-iH_{\sigma}^{C_4}t} = F_{34}F_{21}e^{i\tau X_2 \otimes X_3 t}e^{i\tau Y_2 \otimes Y_3 t}F_{21}^{\dagger}F_{34}^{\dagger}, \quad (\text{A38})$$

which requires two arbitrary rotations and eight T gates. The total gate cost of $e^{-iH_{\sigma}^{C_4}t}$ is summarized in Table III. Note that the initial Jordan-Wigner ordering of the sites used here saves four f_{swap} operations in the implementation of the C_4 tile time evolution compared to the implementations in Refs. [21,41] [see Eq. (B21) in Ref. [41]].

4. S_4 tile

The S_4 tile has hopping terms between 5 spin orbitals, $\phi_{1\sigma}$, $\phi_{2\sigma}$, $\phi_{3\sigma}$, $\phi_{4\sigma}$, and $\phi_{5\sigma}$, described by the adjacency matrix

$$R^{S_4} = \begin{pmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (\text{A39})$$

This matrix has three zero eigenvalues, $\lambda_+ = 2$ and $\lambda_- = -2$ and corresponding eigenvectors

$$v_+ = \begin{pmatrix} 1/\sqrt{2} \\ 1/(2\sqrt{2}) \\ 1/(2\sqrt{2}) \\ 1/(2\sqrt{2}) \\ 1/(2\sqrt{2}) \end{pmatrix}, \quad v_- = \begin{pmatrix} 1/\sqrt{2} \\ -1/(2\sqrt{2}) \\ -1/(2\sqrt{2}) \\ -1/(2\sqrt{2}) \\ -1/(2\sqrt{2}) \end{pmatrix}. \quad (\text{A40})$$

This means $M_{\sigma}^{S_4}$ can be written as

$$M_{\sigma}^{S_4} = 2b^{\dagger}b - 2c^{\dagger}c, \quad (\text{A41})$$

where b and c are given by

$$b = \frac{1}{\sqrt{2}} \left(a_{1\sigma} + \frac{1}{2}(a_{2\sigma} + a_{3\sigma} + a_{4\sigma} + a_{5\sigma}) \right), \quad (\text{A42})$$

$$c = \frac{1}{\sqrt{2}} \left(a_{1\sigma} - \frac{1}{2}(a_{2\sigma} + a_{3\sigma} + a_{4\sigma} + a_{5\sigma}) \right). \quad (\text{A43})$$

Employing the unitary transformation $V = F_{45}F_{32}F_{34}f_{\text{swap}}^{23}F_{13}$, we can write

$$b = Va_{1\sigma}V^{\dagger}, \quad (\text{A44})$$

$$c = Va_{3\sigma}V^{\dagger}, \quad (\text{A45})$$

where f_{swap}^{23} is the fermionic SWAP operator that swaps the electrons in orbitals 2 and 3, such that the orbitals $\phi_{1\sigma}$ and $\phi_{3\sigma}$ adjacent in the Jordan-Wigner ordering after the

SWAP performed by $V^{\dagger}$. The following f_{swap}^{23} in V returns the orbitals to the initial JW ordering.

The time evolution of the S_4 tile Hamiltonian can therefore be implemented on $|\phi_{1\sigma}\phi_{2\sigma}\phi_{3\sigma}\phi_{4\sigma}\phi_{5\sigma}\rangle$ as

$$e^{-iH_{\sigma}^{S_4}t} = Ve^{i\tau 2(a_{1\sigma}^{\dagger}a_{1\sigma} - a_{3\sigma}^{\dagger}a_{3\sigma})t}V^{\dagger}. \quad (\text{A46})$$

This can be further compiled using the same strategy as shown in Eq. (A20) (remember that ϕ_1 and ϕ_3 are adjacent after the SWAP), resulting in the following expression for the time evolution of the S_4 tile Hamiltonian

$$e^{-iH_{\sigma}^{S_4}t} = F_{45}F_{32}F_{34}f_{\text{swap}}^{23}e^{i\tau X_1 \otimes X_3 t} \times e^{i\tau Y_1 \otimes Y_3 t}f_{\text{swap}}^{23}F_{34}^{\dagger}F_{32}^{\dagger}F_{45}^{\dagger}, \quad (\text{A47})$$

which requires two arbitrary rotations and 12 T gates. The total gate cost is summarized in Table III.

5. S_3 tile

Next, we demonstrate a method to implement the time evolution of the S_3 tile and highlight why this is more costly than for the other tiles described above. The S_3 tile has hopping terms between 4 spin orbitals, $\phi_{1\sigma}$, $\phi_{2\sigma}$, $\phi_{3\sigma}$, and $\phi_{4\sigma}$, described by the adjacency matrix

$$R^{S_3} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}. \quad (\text{A48})$$

This matrix has two nonzero eigenvalues, $\lambda_+ = \sqrt{3}$ and $\lambda_- = -\sqrt{3}$ and corresponding eigenvectors

$$v_+ = \begin{pmatrix} 1/\sqrt{2} \\ 1/\sqrt{6} \\ 1/\sqrt{6} \\ 1/\sqrt{6} \end{pmatrix}, \quad v_- = \begin{pmatrix} 1/\sqrt{2} \\ -1/\sqrt{6} \\ -1/\sqrt{6} \\ -1/\sqrt{6} \end{pmatrix}. \quad (\text{A49})$$

Therefore, $M_{\sigma}^{S_3}$ can be written as

$$M_{\sigma}^{S_3} = \sqrt{3}b^{\dagger}b - \sqrt{3}c^{\dagger}c, \quad (\text{A50})$$

where b and c are given by

$$b = \frac{1}{\sqrt{2}}a_{1\sigma} + \frac{1}{\sqrt{6}}(a_{2\sigma} + a_{3\sigma} + a_{4\sigma}), \quad (\text{A51})$$

$$c = \frac{1}{\sqrt{2}}a_{1\sigma} - \frac{1}{\sqrt{6}}(a_{2\sigma} + a_{3\sigma} + a_{4\sigma}). \quad (\text{A52})$$

These cannot be constructed through sequential applications of F_{ij} because this operator only creates superpositions with a prefactor of $1/\sqrt{2}$. This can be solved by

defining a new operator, G_{ij} , that acts on annihilation operators a_i and a_j as

$$G_{ij} a_i G_{ij}^\dagger = \sqrt{\frac{1}{3}} a_i + \sqrt{\frac{2}{3}} a_j, \quad (\text{A53})$$

$$G_{ij} a_j G_{ij}^\dagger = \sqrt{\frac{2}{3}} a_i - \sqrt{\frac{1}{3}} a_j, \quad (\text{A54})$$

and has matrix representation

$$G_{ij} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \sqrt{\frac{1}{3}} & \sqrt{\frac{2}{3}} & 0 \\ 0 & \sqrt{\frac{2}{3}} & -\sqrt{\frac{1}{3}} & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \quad (\text{A55})$$

This operator can be implemented using the quantum circuit shown in Fig. 8 of Ref. [20] when replacing T and $T^\dagger$ with $R_z(\arccos \frac{1}{\sqrt{3}})$ and $R_z(-\arccos \frac{1}{\sqrt{3}})$. In the standard fault-tolerant compilation scheme, these rotations must be synthesized in terms of Clifford + T (approximated up to a required precision), and therefore G_{ij} becomes significantly more costly than F_{ij} .

We define $V = F_{34} G_{23} F_{12}$, which allows us to write

$$b = V a_{1\sigma} V^\dagger, \quad (\text{A56})$$

$$c = V a_{2\sigma} V^\dagger. \quad (\text{A57})$$

Then, the time evolution of the S_3 tile Hamiltonian can be implemented on $|\phi_{1\sigma} \phi_{2\sigma} \phi_{3\sigma} \phi_{4\sigma}\rangle$ as

$$e^{-iH_{S_3}^3 t} = V e^{i\tau\sqrt{3}(a_{1\sigma}^\dagger a_{1\sigma} - a_{2\sigma}^\dagger a_{2\sigma})t} V^\dagger, \quad (\text{A58})$$

which can be further compiled using the strategy shown in Eq. (A20), resulting in the following expression

$$e^{-iH_{S_3}^3 t} = F_{34} G_{23} e^{i\frac{\sqrt{3}\tau}{2} X_1 \otimes X_2 t} e^{i\frac{\sqrt{3}\tau}{2} Y_1 \otimes Y_2 t} G_{23}^\dagger F_{34}^\dagger. \quad (\text{A59})$$

This implementation of the time-evolution operator of the S_3 tile Hamiltonian requires two applications of G_{ij} and has total non-Clifford cost of two arbitrary rotations, four $R_z(\theta)$ rotations of $\theta = \pm \arccos \frac{1}{\sqrt{3}}$ and four T gates. Assuming that we can implement the $\theta = \pm \arccos \frac{1}{\sqrt{3}}$ up to required precision ϵ with 30 T gates, the time evolution of the S_3 tile has a total cost of two arbitrary rotations and 124 T gates. This cost is far larger per hopping term implemented than especially S_2 , C_4 , and S_4 . Note that synthesis of these rotations will also contribute additional errors that we would need to consider in the QPE costing.

6. $K_{2^a, 2^b}$ tile

Lastly, we consider an abstract set of tiles, which encompasses all of the above examples, but also bipartite graphs in general. The additional tiles in this set are likely not of practical use (although could be valuable for implementing beyond-nearest-neighbor hopping), but allow for a more comprehensive analysis.

The complete bipartite graph $K_{n,m}$ has vertices $v_1 \dots v_{n+m}$ and edges between v_i and v_j if and only if $i \leq n$ and $j \geq n+1$ or $j \leq n$ and $i \geq n+1$. Notice that $K_{1,2^k} = S_{2^k}$, $K_{2,2} = C_4$, and that the S_3 tile is given by $K_{1,3}$, so this generalization covers all the cases discussed above. The adjacency matrix $R^{K_{n,m}}$ has rank 2 since it is off diagonal with 1 in the off-diagonal $n \times m$ blocks. It has eigenvalues $\pm \sqrt{nm}, 0$ with multiplicities 1, 1 and $n+m-2$, and normalized eigenvectors corresponding to the nonzero eigenvalues

$$v_+ = \left(\frac{1}{\sqrt{2n}} \mathbf{1}_n, \frac{1}{\sqrt{2m}} \mathbf{1}_m \right), \quad v_- = \left(\frac{1}{\sqrt{2n}} \mathbf{1}_n, -\frac{1}{\sqrt{2m}} \mathbf{1}_m \right), \quad (\text{A60})$$

where $\mathbf{1}_k$ denotes the vector of 1s with length k .

Then, $M_\sigma^{K_{n,m}}$ can be written as

$$M_\sigma^{K_{n,m}} = \sqrt{nm} b^\dagger b - \sqrt{nm} c^\dagger c, \quad (\text{A61})$$

with b and c given by

$$b = \frac{1}{\sqrt{2n}} \sum_{i=1}^n a_{i\sigma} + \frac{1}{\sqrt{2m}} \sum_{i=n+1}^{n+m} a_{i\sigma}, \quad (\text{A62})$$

$$c = \frac{1}{\sqrt{2n}} \sum_{i=1}^n a_{i\sigma} - \frac{1}{\sqrt{2m}} \sum_{i=n+1}^{n+m} a_{i\sigma}. \quad (\text{A63})$$

At this point it is advantageous to consider the case where $n = 2^a$ and $m = 2^b$ for non-negative integers a, b . In this case, we can define V to be

$$\prod_{k=0}^{a-1} \prod_{i=1}^{2^{a-k-1}} F_{2^k i, 2^k(i+1)} \prod_{l=0}^{b-1} \prod_{j=1}^{2^{b-l-1}} F_{2^l j + n, 2^l(j+1) + n} F_{1, 2^a + 1},$$

where whenever $a = 0$ and $b = 0$ we define the empty product to be 1. In this definition of V , we ignore f_{swap} operators that depend on the initial JW ordering of the orbitals. Then,

$$b = V a_{1\sigma} V^\dagger, \quad (\text{A64})$$

$$c = V a_{2^a+1, \sigma} V^\dagger, \quad (\text{A65})$$

and we can implement the time evolution as

$$e^{-iH_\sigma^{K_{n,m}} t} = V e^{i\tau\lambda(a_{1\sigma}^\dagger a_{1\sigma} - a_{2^a+1, \sigma}^\dagger a_{2^a+1, \sigma})t} V^\dagger, \quad (\text{A66})$$

where $\lambda = 2^{\frac{a+b}{2}}$. This can be further compiled using the same strategy as shown in Eq. (A20). Then, the time evolution of the $K_{2^a, 2^b}$ tile Hamiltonian can be implemented as

$$e^{-iH_{\sigma}^{K_{n,m}}t} = \tilde{V} e^{i\frac{\lambda}{2}\tau X_1 \otimes X_{2^a+1}t} e^{i\tau Y_1 \otimes Y_{2^a+1}t\frac{\lambda}{2}} \tilde{V}^\dagger, \quad (\text{A67})$$

with $\tilde{V} = VF_{1,2^a+1}^\dagger$, which requires two arbitrary rotations and $2^{a+2} + 2^{b+2} - 8$ T gates (since the total number of F_{ij} operators in $\tilde{V}$ is $2((2^a - 1) + (2^b - 1))$ when canceling $F_{1,2^a+1}$ and $F_{1,2^a+1}^\dagger$). Inserting appropriate values of a and b one recovers the costing above. The total number of bonds covered by the tile is 2^{a+b} . Therefore, the cost of implementing time evolution of tiles with adjacency matrices corresponding to bipartite graphs of type $K_{2^a, 2^b}$ is $2/2^{a+b}$ arbitrary rotations per bond and $(2^{a+2} + 2^{b+2} - 8)/2^{a+b}$ T gates per bond. The T -gate cost of a single arbitrary rotation is often in the range 30–50 T gates, thus, using larger tiles decreases the overall total T -gate cost required for implementing time evolution of the hopping Hamiltonian.

This section shows that tiles with adjacency matrices of type $K_{2^a, 2^b}$ have a simple and, in the case of a and/or $b > 0$, cheap implementation of time evolution of tile Hamiltonians per hopping term in a given tile. For the two-dimensional lattices considered here, we primarily find the S_2 ($K_{1,2}$), S_4 ($K_{1,4}$), and C_4 ($K_{2,2}$) tiles practically useful. However, in some cases larger tiles may

be applied, e.g., to tile three-dimensional lattices or to construct beyond nearest-neighbor hopping sections. The application of larger tiles will result in a significant saving in the number of T gates per bond.

APPENDIX B: TILE-TROTTERIZATION DECOMPOSITION

Here, we show that the Trotter error norm of tile Trotterization can be obtained as $W_{\text{tile}} \leq W_{\text{SO2}} + W_h$, given that we can compute W_{SO2} and W_h that come from the following Trotter partitionings:

$$\|e^{-i(H_h+H_C)t} - e^{-iH_C t/2} e^{-iH_h t} e^{-iH_C t/2}\| \leq W_{\text{SO2}} t^3, \quad (\text{B1})$$

$$\left\| e^{-iH_h t} - \prod_{s=1}^S e^{-iH_h^s t/2} \prod_{s=S}^1 e^{-iH_h^s t/2} \right\| \leq W_h t^3. \quad (\text{B2})$$

The tile-Trotterization error norm, W_{tile} , can be calculated as

$$\begin{aligned} & \left\| e^{-iHt} - e^{-iH_C \frac{t}{2}} \prod_{s=1}^S e^{-iH_h^s \frac{t}{2}} \prod_{s=S}^1 e^{-iH_h^s \frac{t}{2}} e^{-iH_C \frac{t}{2}} \right\| \\ & \leq W_{\text{tile}} t^3 \leq (W_{\text{SO2}} + W_h) t^3, \end{aligned} \quad (\text{B3})$$

which is shown by the following derivation:

$$\begin{aligned} & \left\| e^{-i(H_h+H_C)t} - e^{-iH_C t/2} \prod_{s=1}^S e^{-iH_h^s t/2} \prod_{s=S}^1 e^{-iH_h^s t/2} e^{-iH_C t/2} \right\| \\ & = \left\| e^{-i(H_h+H_C)t} - e^{-iH_C t/2} e^{-iH_h t} e^{-iH_C t/2} + e^{-iH_C t/2} e^{-iH_h t} e^{-iH_C t/2} - e^{-iH_C t/2} \prod_{s=1}^S e^{-iH_h^s t/2} \prod_{s=S}^1 e^{-iH_h^s t/2} e^{-iH_C t/2} \right\| \\ & \leq \left\| e^{-i(H_h+H_C)t} - e^{-iH_C t/2} e^{-iH_h t} e^{-iH_C t/2} \right\| + \left\| e^{-iH_C t/2} e^{-iH_h t} e^{-iH_C t/2} - e^{-iH_C t/2} \prod_{s=1}^S e^{-iH_h^s t/2} \prod_{s=S}^1 e^{-iH_h^s t/2} e^{-iH_C t/2} \right\| \\ & \leq W_{\text{SO2}} t^3 + \left\| e^{-iH_C t/2} \left(e^{-iH_h t} - \prod_{s=1}^S e^{-iH_h^s t/2} \prod_{s=S}^1 e^{-iH_h^s t/2} \right) e^{-iH_C t/2} \right\|, \\ & \leq W_{\text{SO2}} t^3 + \|e^{-iH_C t/2}\| \left\| \left(e^{-iH_h t} - \prod_{s=1}^S e^{-iH_h^s t/2} \prod_{s=S}^1 e^{-iH_h^s t/2} \right) \right\| \|e^{-iH_C t/2}\|, \\ & \leq (W_{\text{SO2}} + W_h) t^3. \end{aligned} \quad (\text{B4})$$

Therefore, the tile-Trotterization error norm can be bounded by a sum of simpler error norms, W_{SO2} and W_h .

APPENDIX C: THE HOPPING HAMILTONIAN TROTTER ERROR NORM W_h

Tile Trotterization splits the hopping Hamiltonian into S sections and uses the following Trotter decomposition to implement the hopping Hamiltonian,

$$\left\| e^{-iH_h t} - \prod_{s=1}^S e^{-iH_h^s t/2} \prod_{s=S}^1 e^{-iH_h^s t/2} \right\| \leq W_h t^3. \quad (\text{C1})$$

In the case where it is possible to divide a lattice into three sections of colors blue, red, and gold, we have that

$S = 3$ and that the hopping Hamiltonian sections can be written as $H_1 = H_h^b$, $H_2 = H_h^r$ and $H_3 = H_h^g$, where the subscripts 1, 2, and 3 imply the order in which the terms are implemented within the Trotter step.

The Trotter error norm, W_h , can be obtained from Eq. (15), which can be written out using a package provided by Schubert *et al.* [84], which automatically writes out Eq. (15) for an arbitrary number of noncommuting Hamiltonian terms. We evaluate Eq. (15) for $H_1 = H_h^b$, $H_2 = H_h^r$, and $H_3 = H_h^g$ as

$$\begin{aligned} W_h = & \frac{1}{12} (\|[[H_h^b, H_h^r], H_h^r]\| + \|[[H_h^b, H_h^r], H_h^g]\| + \|[[H_h^b, H_h^g], H_h^r]\| + \|[[H_h^b, H_h^g], H_h^g]\| + \|[[H_h^r, H_h^g], H_h^g]\|) \\ & + \frac{1}{24} (\|[[H_h^b, H_h^r], H_h^b]\| + \|[[H_h^b, H_h^g], H_h^b]\| + \|[[H_h^r, H_h^g], H_h^r]\|). \end{aligned} \quad (\text{C2})$$

This expression can be evaluated from the adjacency matrices of the different sections. The Hamiltonian sections can be written as

$$H_h^b = -\tau \sum_{ij\sigma} R_{ij}^b a_{i\sigma}^\dagger a_{j\sigma}, \quad (\text{C3})$$

$$H_h^r = -\tau \sum_{ij\sigma} R_{ij}^r a_{i\sigma}^\dagger a_{j\sigma}, \quad (\text{C4})$$

$$H_h^g = -\tau \sum_{ij\sigma} R_{ij}^g a_{i\sigma}^\dagger a_{j\sigma}, \quad (\text{C5})$$

where R^r , R^b , and R^g are the adjacency matrices of the red, blue, and gold lattice sections, respectively. Because the Hamiltonian section operators are free fermionic, and using the argument of Ref. [21] Appendix A, this expression for W_h can be evaluated from the adjacency matrices as

$$\begin{aligned} W_h = & \frac{\tau^3}{12} (\|[[R^b, R^r], R^r]\|_1 + \|[[R^b, R^r], R^g]\|_1 + \|[[R^b, R^g], R^r]\|_1 + \|[[R^b, R^g], R^g]\|_1 + \|[[R^r, R^g], R^g]\|_1) \\ & + \frac{\tau^3}{24} (\|[[R^b, R^r], R^b]\|_1 + \|[[R^b, R^g], R^b]\|_1 + \|[[R^r, R^g], R^r]\|_1), \end{aligned} \quad (\text{C6})$$

where $\|\cdot\|_1$ is the Schatten one norm and τ is the hopping parameter. This expression can be used to evaluate W_h for any model where the lattice can be divided into three lattice sections.

The hopping section error norm, W_h , is evaluated numerically for the periodic hexagonal lattice using the lattice division described in Appendix D, and scales linearly in N . This is shown in Fig. 6 for a lattice with parameters $L_x = L_y = L$ in the range $4 \leq L \leq 24$ for even L and hopping parameter $\tau = 1$. This figure shows W_h/τ^3 as a function of the number of lattice sites N , ranging between

$N = 32$ and $N = 1152$, demonstrating the linear dependence of W_h on the number of lattice sites. We obtain a bound on W_h using the largest constant factor: $W_h/\tau^3 N$, of the points plotted in Fig. 6, leading to

$$W_h \leq 0.8532\tau^3 N, \quad (\text{C7})$$

for periodic hexagonal lattices with even L in the range $4 \leq L \leq 24$ using the S_2 tiling shown in Appendix D.

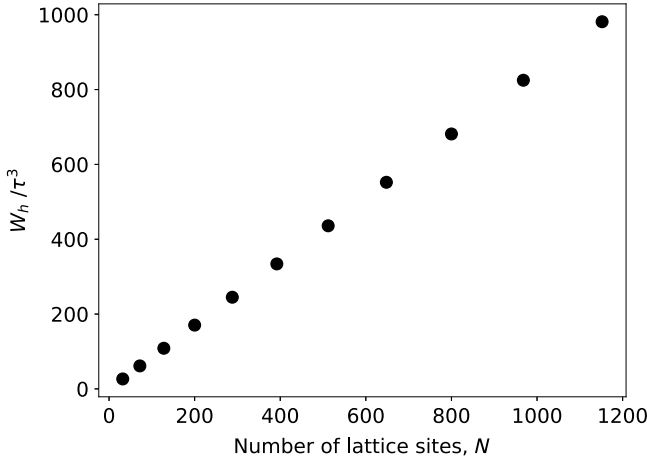


FIG. 6. Plot of W_h/τ^3 as a function of the number of lattice sites, N , for the periodic hexagonal lattice, using the division described in Appendix D.

APPENDIX D: PERIODIC HEXAGONAL LATTICE MODEL

We consider the periodic hexagonal lattice model shown in Fig. 7(a). Each lattice point pair of color white and gray are labeled by (l_x, l_y) , which is defined from the (x, y) positions of the lattice pairs as $(x, y) = (a \cdot l_x, b \cdot l_y)$, where a and b are the lattice vectors. We define the lattice size using the parameters $L_x = \max(l_x) + 1$ and $L_y = \max(l_y) + 1$, such that the number of lattice sites is $N = 2L_x L_y$.

All lattices, both periodic and nonperiodic, can be divided into sections in a number of different ways using either one type or several different types of tiles. The lattice sections influence both the Trotter error norm, W_h , the parallelizability and the per-Trotter-step gate count. The effect of W_h is however insignificant relative to the total

Trotter error norm W_{tile} . Therefore, to construct efficient hopping Hamiltonian decompositions we focus on two aspects: (1) minimizing the number of sections to improve parallelizability and (2) minimizing the number of tiles in each section to reduce gate counts (assuming we use S_1 , S_2 , C_4 , and S_4). These two points are generally not possible to optimize simultaneously. For example, the tiling of the kagome lattice shown in Fig. 2 can be achieved using more S_4 tiles than the choice demonstrated. This would reduce the overall gate cost of the hopping Hamiltonian but with an additional cost of significantly more sections and reduced parallelizability. Therefore, when choosing efficient tiling of the lattice models, we aim to keep the number of sections to a minimum and try to avoid using S_1 tiles.

We choose to divide our periodic hexagonal lattice models into three sections of colors red, blue, and gold using S_2 tiles as shown for $L_x = L_y = 4$ in Fig. 7(b). The dashed bonds with labels 1 and 1' indicate a single bond that ensures periodicity of the lattice. This lattice section division ensures that we have $N_b = N_r = N_g = N/4$ S_2 tiles in each section, which will be the case for all periodic hexagonal lattice models studied in this paper.

Figures 8(a)–8(c) show the lattice divisions of the periodic hexagonal lattice model for $L = 6, 8$, and 10 , with a lattice section division following the same structure as for the $L = 4$ case.

Note that it is not a requirement that each section contains the same number of tiles, and in fact this does not necessarily give the most optimal implementation. As shown in Eq. (E1), one Trotter step of hexagonal lattice Hubbard models with three lattice sections uses two applications of $e^{-iH_h^b t/2}$, two applications of $e^{-iH_h^r t/2}$ and one application of $e^{-iH_h^g t}$. Therefore, the gate costings of a Trotter step might be reduced by putting as many tiles as possible into

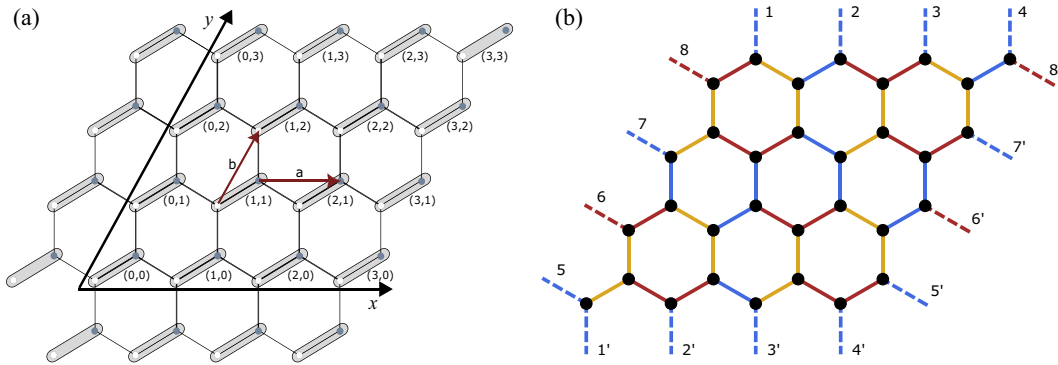


FIG. 7. (a) The periodic hexagonal lattice model where each pair of lattice points are labeled with (l_x, l_y) defined from the (x, y) positions of the lattice points as $(x, y) = (l_x \cdot a, l_y \cdot b)$, where a and b are lattice vectors. The size of the lattice is given by $L_x = \max(l_x) + 1$ and $L_y = \max(l_y) + 1$. For this lattice, we have $L_x = L_y = 4$ and $N = 2L_x L_y = 32$. Periodic boundary conditions are applied in the x and y directions, which is illustrated through the periodic image in the lattice sketch. (b) The division of a periodic hexagonal lattice with size $L_x = L_y = 4$ into three colored sections: blue, red, and gold, using S_2 tiles. The dashed bonds with attached number indicate the bonds that ensure periodicity. For example, the dashed bonds labeled with 1 and 1' indicate the same bond. Each section is covered by $N_b = N_r = N_g = N/4$ S_2 tiles

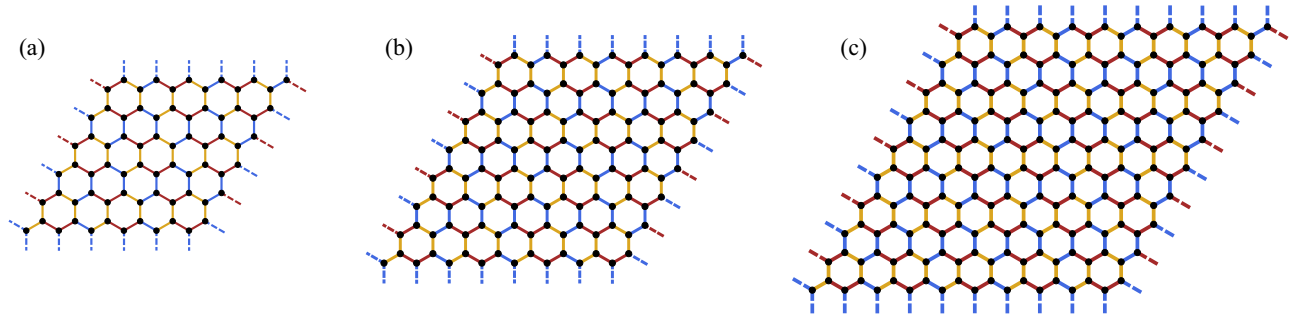


FIG. 8. Lattice sections used in this paper for periodic hexagonal lattice models with lattice size $L_x = L_y = L$, where (a) $L = 6$, (b) $L = 8$, and (c) $L = 10$, which corresponds to (a) $N = 72$, (b) $N = 128$, (c) $N = 200$. The number of tiles in each section of the periodic hexagonal lattice models is $N_r = N_b = N_g = N/4$ S_2 tiles.

the gold sections while still maintaining the commutativity properties of the gold section. This optimization has not been implemented for the models considered here, but may be considered for concrete future applications in order to take full advantage of the tile-Trotterization method.

APPENDIX E: TILE TROTTERIZATION APPLICATIONS

In the main text we considered tile Trotterization of the extended Hubbard model on a periodic hexagonal lattice. Here, we consider the application of tile Trotterization for two additional systems: the Hubbard model on a nonperiodic hexagonal lattice (hexagonal lattice fragments) and the periodic hexagonal lattice Hubbard model.

Section E 1 shows that tile Trotterization is applicable also to nonperiodic lattice models of arbitrary shapes. For this application, we provide gate counts and Trotter error norms for a specific class of hexagonal lattice fragments. Section E 2 presents a simpler version of the main text application and discusses tile Trotterization of the periodic hexagonal lattice Hubbard model. In our derivation of the gate counts, we consider this periodic model as a special case of the hexagonal lattice fragments in Sec. E 1. The results obtained in Sec. E 2 are used in the quantum phase estimation section of the main text (Sec. VI) to generate the numerical results used in Figs. 3 and 4(a).

1. Tile Trotterization of the Hubbard model on hexagonal lattice fragments

We consider all nonperiodic hexagonal lattice fragments with the requirement that all lattice sites are part of at least one full hexagon. These fragments have two distinct types of sites: center sites where a lattice site has three nearest neighbors and edge sites where a lattice site has two nearest neighbors. We denote the number of center sites by N_c and the number of edge sites by N_{ed} .

To apply tile Trotterization, we cover the fragments by S_2 tiles and divide them into three sections of colors: blue (b), red (r), and gold (g). Two examples of fragments and

the division into sections are shown in Fig. 9. We use gray as the color for the edge sites and black as the color for the center sites. Continuing, we use N_b , N_r , and N_g as the number of S_2 tiles used in sections b, r, and g.

A single Trotter step of the Hubbard model, defined in Eq. (4), on these fragments is implemented as

$$e^{-i(H_h+H_I)t} \approx e^{-iH_I \frac{t}{2}} e^{-iH_h^b \frac{t}{2}} e^{-iH_h^r \frac{t}{2}} e^{-iH_h^g \frac{t}{2}} \times e^{-iH_h^r \frac{t}{2}} e^{-iH_h^b \frac{t}{2}} e^{-iH_I \frac{t}{2}}, \quad (\text{E1})$$

where we have chosen to implement the sections in the order $H_1 = H_h^b$, $H_2 = H_h^r$, and $H_3 = H_h^g$. Performing r repetitions of this Trotter step gives

$$\left(e^{-i(H_h+H_I)t}\right)^r \approx e^{-iH_I \frac{t}{2}} \left(e^{-iH_h^b \frac{t}{2}} e^{-iH_h^r \frac{t}{2}} e^{-iH_h^g \frac{t}{2}} \times e^{-iH_h^r \frac{t}{2}} e^{-iH_h^b \frac{t}{2}} e^{-iH_I \frac{t}{2}}\right)^r e^{iH_I \frac{t}{2}}, \quad (\text{E2})$$

so that the cost of one Trotter step for large r is two applications of $e^{-iH_h^b t/2}$, two applications of $e^{-iH_h^r t/2}$, one application of $e^{-iH_h^g t}$ and one application of $e^{-iH_I t}$.

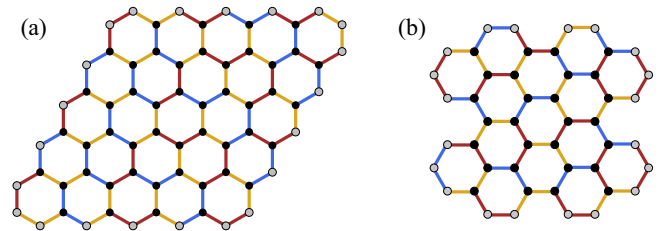


FIG. 9. (a) Example of a hexagonal lattice fragment covered by S_2 tiles in three lattice sections with $N_{ed} = 22$ edge sites (gray) and $N_c = 48$ center sites (black). (b) Another example of a hexagonal lattice fragment with a structure inspired by nanographene molecules studied in Ref. [62], which show interesting π -spin properties. This structure is also covered by S_2 tiles in three lattice sections and has $N_{ed} = 20$ edge sites and $N_c = 28$ center sites.

Using the Jordan-Wigner transformation, the system is represented by $2N$ qubits and $e^{-iH_I t}$ contains N terms, and can be implemented with one layer of N arbitrary Z -axis rotations of the same angle and two layers of N CNOT gates.

The costing of implementing the time evolution of each hopping section is performed by counting the number of applications of $e^{-iH_{\sigma^2} t}$ in each section. This Trotter implementation requires $4N_b$, $4N_r$, and $2N_g$ applications of the time-evolution operator of the S_2 tile Hamiltonian, accounting for both spin sectors. Only counting the non-Clifford gates, the time evolution of the S_2 tile Hamiltonian can be implemented using two arbitrary Z -axis rotations and four T gates as shown in Appendix A. This leads to a total non-Clifford cost per Trotter step of N_R arbitrary rotations and N_T T gates,

$$N_R = N + 8N_b + 8N_r + 4N_g, \quad (\text{E3})$$

$$N_T = 16N_b + 16N_r + 8N_g. \quad (\text{E4})$$

The tile-Trotterization error norm is evaluated as shown in Eq. (21), using W_{S_2} and W_h . First, we find an expression for W_{S_2} from Eq. (23), which shows that we only need to evaluate $\|[[H_I, H_h], H_I]\|$ and $\|[[H_I, H_h], H_h]\|$. These two commutator bounds are given by Lemmas 1 and 2 of Ref. [21]. The first commutator bound is given by

$$\|[[H_I, H_h], H_I]\| \leq U^2 \|H_h\| = U^2 \tau \|R\|_1, \quad (\text{E5})$$

where $\|\cdot\|_1$ is the Schatten one norm and R is the adjacency matrix of the fragments. The second commutator bound can be evaluated as

$$\|[[H_I, H_h], H_h]\| \leq \frac{U}{2} \sum_i (\|T_i, H_h\| + 2\|T_i\|^2), \quad (\text{E6})$$

where T_i is an operator containing all hopping terms that interact with site i . Throughout this paper, we use the notation H_k^i (instead of T_i as in Ref. [21]) to represent all hopping operators interacting with site i , where k denotes the number of nearest neighbors of site i . The norm $\|H_k^i\|$ is only dependent on the character of the specific site i . The fragments have two distinct sites: center and edge sites, therefore

$$\sum_i 2\|H_k^i\|^2 = 2\|H_3^i\|^2 N_c + 2\|H_2^i\|^2 N_{ed}. \quad (\text{E7})$$

The two norms $\|H_3^i\|$ and $\|H_2^i\|$ are evaluated as

$$\|H_3^i\| = 2\sqrt{3}\tau, \quad (\text{E8})$$

$$\|H_2^i\| = 2\sqrt{2}\tau, \quad (\text{E9})$$

using the properties of free fermionic operators shown in Appendix A in Ref. [21].

The norm of the commutator $\|[[H_k^i, H_h]]\|$ is not only dependent on the character of site i but also on neighbor sites of i and can be evaluated for each site i as shown in Appendix A in Ref. [21]. For these models, we find that $\max(\|[[H_k^i, H_h]]\|) = 2\sqrt{6}\tau^2$, using the periodic hexagonal lattice model with $L_x = L_y = L = 4$ as H_h . Note that the norm is independent of lattice size for $L \geq 4$. The maximum value of the commutator norm is found for the center sites that only has center sites as neighbors. We use this to bound $\sum_i \|[[H_k^i, H_h]]\|$ as

$$\sum_i \|[[H_k^i, H_h]]\| \leq 2\sqrt{6}\tau^2 N, \quad (\text{E10})$$

where the equal sign only holds for periodic hexagonal lattices where all sites have three nearest neighbors (with $N_c = N$ and $N_{ed} = 0$).

This leads to the final expression for the second commutator bound

$$\|[[H_I, H_h], H_h]\| \leq U\tau^2 (12N_c + 8N_{ed} + \sqrt{6}N). \quad (\text{E11})$$

We obtain W_h using Eq. (15) and the order of the hopping sections in the Trotter step: $H_1 = H_h^b$, $H_2 = H_h^r$ and $H_3 = H_h^g$. Equation (15) is written out for this splitting in Eq. (C2) in Appendix C. Equation (C6) shows a formula for efficiently calculating W_h for all tile Trotterization applications where the lattice can be divided into three sections.

2. Tile Trotterization of the Hubbard model on the periodic hexagonal lattice

We use the periodic hexagonal lattice model and the division into sections shown in Figs. 7 and 8, and described in Appendix D. This corresponds to decomposing the hopping Hamiltonian as $H_h = H_h^b + H_h^r + H_h^g$, using $N_b = N_r = N_g = N/4$ S_2 tiles in each section.

A single Trotter step of the Hubbard model on the periodic hexagonal lattice is implemented in the same way as for the arbitrary fragments described in Eqs. (E1)–(E2), so that the cost of one Trotter step (when performing r Trotter steps total) consists of two applications of $e^{-iH_h^b t/2}$, two applications of $e^{-iH_h^r t/2}$, one application of $e^{-iH_h^g t}$ and one application of $e^{-iH_I t}$. Therefore, the cost per Trotter step can again be evaluated as in Eqs. (E3)–(E4), but for the periodic case we can replace the number of tiles in each section by $N/4$, simplifying the non-Clifford cost to

$$N_R = 6N, \quad (\text{E12})$$

$$N_T = 10N. \quad (\text{E13})$$

Arbitrary rotations are expensive to perform on fault-tolerant quantum computers because such operations cannot be protected for arbitrary rotation angles. Hamming-weight phasing (HWP) [20,21,69] is a method to reduce

the number of arbitrary rotations required, which can be applied in the case where many rotations of the same angle are performed in parallel. HWP allows for trading m arbitrary rotations of the same angle with $m - 1$ clean ancilla qubits and $m - 1$ Toffoli gates. The Hubbard model time evolution contains six layers of N arbitrary rotations, where the angle of all gates within a given layer are the same. We choose to implement m of these arbitrary rotations simultaneously, where N is an integer multiple of m , which reduces the number of arbitrary rotations per layer to $N_R = \lfloor \log_2(m) + 1 \rfloor$ using $N_{\text{Tof}} = m - 1$ Toffoli gates and $\alpha = m - 1$ clean ancilla qubits. To implement the total amount of $6N$ arbitrary rotations, we need $6N/m$ layers of m arbitrary rotations. This leads to a total gate count per Trotter step of

$$N_R = \frac{6N}{m} \lfloor \log_2(m) + 1 \rfloor, \quad (\text{E14})$$

$$N_{\text{Tof}} = \frac{6N}{m} (m - 1), \quad (\text{E15})$$

$$N_T = 10N. \quad (\text{E16})$$

Each Toffoli gate can be converted into four T gates to obtain a total T -gate cost of

$$N_T = 10N + 4N_{\text{Tof}} = 10N + 4 \times \frac{6N}{m} (m - 1). \quad (\text{E17})$$

Given that we choose m as a fraction of N , the number of arbitrary rotations per Trotter step scales logarithmically in N and the number of T gates scale linearly in N . The total qubit count needed for this tile-Trotterization implementation is $2N + (m - 1)$.

Note that choosing HWP with $m < N$ reduces the parallelization of arbitrary rotations in each Trotter step because only m arbitrary rotations can be performed simultaneously. Additionally, choosing $m = N$ comes with a significant additional qubit cost so $m = N/4$ or $m = N/2$ might be more advantageous depending on the available resources.

The tile-Trotterization error norm is again evaluated from Eq. (21), and W_{SO2} can be obtained from the commutator bounds given by Lemmas 1 and 2 in Appendix C of Ref. [21]. This leads to the following bounds for the periodic hexagonal lattice:

$$\|[[H_I, H_h], H_I]\| \leq U^2 \|H_h\| = U^2 \tau \|R\|_1, \quad (\text{E18})$$

where $\|\cdot\|_1$ is the Schatten one norm and R is the adjacency matrix of the periodic hexagonal lattice. We note that $\|R\|_1$ has at worst case linear scaling in N . Using the strategies provided by Ref. [21], we can bound the second

commutator by

$$\begin{aligned} \|[[H_I, H_h], H_h]\| &\leq \frac{U}{2} \sum_i (\|H_k^i, H_h\| + 2\|H_k^i\|^2) \\ &= (12 + \sqrt{6})U\tau^2 N. \end{aligned} \quad (\text{E19})$$

Using that for all sites i in the periodic hexagonal lattice

$$\|H_3^i\| = 2\sqrt{3}\tau, \quad (\text{E20})$$

$$\|[H_3^i, H_h]\| = 2\sqrt{6}\tau^2. \quad (\text{E21})$$

We used the adjacency matrix of the periodic hexagonal lattice model with $L_x = L_y = L = 4$ to evaluate Eq. (E21). We note that $[H_k^i, H_h]$ is a local operator and its norm remains constant for $L \geq 4$.

In this paper, we introduce a new way of evaluating the commutator bound $\|[[H_I, H_h], H_h]\|$ that improves upon the commutator bound given by Lemma 2 in Ref. [21]. In Appendix F, we introduce Lemma F1 that can be applied to obtain the following bound for the periodic hexagonal Hubbard model:

$$\|[[H_I, H_h], H_h]\| \leq 9.9U\tau^2 N, \quad (\text{E22})$$

which is a 31.5% improvement compared to the bound given in Eq. (E19). For the QPE resource estimation results presented in Figs. 3 and 4(a), we use the $\|[[H_I, H_h], H_I]\|$ given by Eq. (E18) and the $\|[[H_I, H_h], H_h]\|$ bound given by Eq. (E22).

The lattice is divided into three sections and we can therefore obtain W_h using Eq. (C6) in Appendix C. The hopping Hamiltonian Trotter error norm for the periodic hexagonal lattice division used in this paper scales linearly with N , as shown in Fig. 6. In the numerical examples for the hexagonal lattice periodic Hubbard model given in this paper with $U = 4$ and $\tau = 1$, we find that W_h constitutes around 16% of the total Trotter error norm W_{tile} .

The results presented in this section are used to obtain the QPE resource estimates for the Hubbard model in Sec. VI.

APPENDIX F: COMMUTATOR BOUNDS

The tile-Trotterization error norm of generalized Hubbard models is a sum of W_{SO2} and W_h . In this Appendix, we show how to evaluate commutator bounds used to calculate W_{SO2} . The W_{SO2} Trotter error is evaluated using the expression

$$W_{\text{SO2}} = \frac{1}{24} \|[[H_C, H_h], H_C]\| + \frac{1}{12} \|[[H_C, H_h], H_h]\|. \quad (\text{F1})$$

The generalized Hubbard models used throughout this paper are defined as

$$H = H_h + H_C, \quad (\text{F2})$$

with the Coulomb term given by $H_C = H_I$ for the Hubbard model and $H_C = H_I + H_V$ for the extended Hubbard model. In the commutator bound derivations for the extended Hubbard model, we limit ourselves to lattices where all sites have the same number of nearest neighbors (k -regular graphs), which allows us to use the modified interaction term defined in Eq. (10). For the commutator bounds presented here, we use the following definitions of the terms H_h , H_I , and H_V

$$H_h = -\tau \sum_{i,j,\sigma} R_{ij} a_{i\sigma}^\dagger a_{j\sigma}, \quad (\text{F3})$$

$$H_I = \frac{U}{4} \sum_{i=1}^N Z_{i\uparrow} Z_{i\downarrow}, \quad (\text{F4})$$

$$H_V = \frac{V}{4} \sum_{\langle ij \rangle} \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'}. \quad (\text{F5})$$

For $i \neq j$, we also define the hopping operator $B_{ij\sigma}$ (hopping term from spin orbital $i\sigma$ to spin orbital $j\sigma$) as

$$B_{ij\sigma} = -\tau a_{i\sigma}^\dagger a_{j\sigma}, \quad (\text{F6})$$

and define B_{ij} as

$$B_{ij} = \sum_{\sigma \in \{\uparrow, \downarrow\}} B_{ij\sigma}. \quad (\text{F7})$$

Continuing, we write the sum over spins as $\sum_{\sigma \in \{\uparrow, \downarrow\}} = \sum_{\sigma}$. Using the B_{ij} notation, we can write H_h as

$$H_h = \sum_{\langle ij \rangle} (B_{ij} + B_{ji}), \quad (\text{F8})$$

where $\sum_{\langle ij \rangle}$ runs over $(k/2)N$ nearest-neighbor terms for all lattices where each lattice site has k nearest neighbors.

Before evaluating the commutator bound $\|[[H_I, H_h], H_h]\|$ in Sec. F3, $\|[[H_C, H_h], H_h]\|$ in Sec. F4 and $\|[[H_C, H_h], H_C]\|$ in Sec. F5, we establish a set of commutator and anticommutator rules used for the derivations (Sec. F1). In Sec. F2, we describe our procedure for evaluating the spectral norms. Finally, in Sec. F6, we evaluate the commutator bounds of interest exactly for the one-dimensional Hubbard and extended Hubbard model periodic chain to give some intuition for the tightness of the bounds.

1. Commutator and anticommutator rules

In the following derivations, we will need to consider the commutator and anticommutators between Coloumb terms (of the form $Z_{i\sigma} Z_{j\sigma'}$ for $i\sigma \neq j\sigma'$) and hopping terms (of the form $B_{ij\sigma} = -\tau a_{i\sigma}^\dagger a_{j\sigma}$ for $i \neq j$).

To aid in this task, we establish a set of commutator and anticommutator rules,

$$[Z_{m\sigma_1} Z_{n\sigma_2}, B_{ij\sigma}] = 0 \quad \text{if spin orbitals } m\sigma_1, n\sigma_2, i\sigma, \text{ and } j\sigma \text{ are all different}, \quad (\text{F9})$$

$$[Z_{i\sigma} Z_{j\sigma}, B_{ij\sigma}] = 0, \quad (\text{F10})$$

$$\{Z_{i\sigma} Z_{l\sigma'}, B_{ij\sigma}\} = 0 \quad \text{if spin orbitals } i\sigma, j\sigma, \text{ and } l\sigma' \text{ are all different}, \quad (\text{F11})$$

$$\{Z_{i\sigma} Z_{l\sigma'}, B_{j\sigma}\} = 0 \quad \text{if spin orbitals } i\sigma, j\sigma, \text{ and } l\sigma' \text{ are all different}. \quad (\text{F12})$$

In words, if the spin orbitals in the Coulomb term are both the same or both different to the spin orbitals in the hopping term, then the *commutator* is zero. In contrast, if one and only one of the spin orbitals is shared between the Coulomb and hopping term, then the *anticommutator* is zero. This gives the intuition for these results, which we now proceed to derive.

The commutator in Eq. (F9) is trivially true because all operators in the expression act on different spin orbitals.

Equations (F10) and (F11) can be shown by using the following anticommutation relation that applies for $\sigma \neq \sigma'$ or $i \neq j$,

$$\begin{aligned} \{Z_{i\sigma} Z_{j\sigma'}, a_{i\sigma}^\dagger\} &= \{(2n_{i\sigma} - \mathbb{1})(2n_{j\sigma'} - \mathbb{1}), a_{i\sigma}^\dagger\} \\ &= (2n_{j\sigma'} - \mathbb{1})\{(2n_{i\sigma} - \mathbb{1}), a_{i\sigma}^\dagger\} \\ &= 2(2n_{j\sigma'} - \mathbb{1})(\{n_{i\sigma}, a_{i\sigma}^\dagger\} - a_{i\sigma}^\dagger). \end{aligned} \quad (\text{F13})$$

This expression can be evaluated by calculating the anti-commutator

$$\begin{aligned} \{n_{i\sigma}, a_{i\sigma}^\dagger\} &= \{a_{i\sigma}^\dagger a_{i\sigma}, a_{i\sigma}^\dagger\} = a_{i\sigma}^\dagger a_{i\sigma} a_{i\sigma}^\dagger + a_{i\sigma}^\dagger a_{i\sigma}^\dagger a_{i\sigma} \\ &= a_{i\sigma}^\dagger a_{i\sigma} a_{i\sigma}^\dagger = (\mathbb{1} - a_{i\sigma} a_{i\sigma}^\dagger) a_{i\sigma}^\dagger = a_{i\sigma}^\dagger \end{aligned} \quad (\text{F14})$$

using $\{a_{i\sigma}^\dagger, a_{i\sigma}\} = \mathbb{1}$. Inserting this result back into Eq. (F13), we obtain

$$\{Z_{i\sigma} Z_{j\sigma'}, a_{i\sigma}^\dagger\} = 0, \quad (\text{F15})$$

which can also be shown for $a_{i\sigma}$ instead of $a_{i\sigma}^\dagger$ to obtain

$$\{Z_{i\sigma} Z_{j\sigma'}, a_{i\sigma}\} = 0. \quad (\text{F16})$$

Now we can prove Eq. (F10) using the identity $[A, BC] = \{A, B\}C - B\{A, C\}$ and Eqs. (F15)–(F16)

$$\begin{aligned} [Z_{i\sigma} Z_{j\sigma}, B_{ij\sigma}] &= -\tau [Z_{i\sigma} Z_{j\sigma}, a_{i\sigma}^\dagger a_{j\sigma}] \\ &= -\tau \left(\{Z_{i\sigma} Z_{j\sigma}, a_{i\sigma}^\dagger\} a_{j\sigma} - a_{i\sigma}^\dagger \{Z_{i\sigma} Z_{j\sigma}, a_{j\sigma}\} \right) = 0. \end{aligned} \quad (\text{F17})$$

Equation (F11) can be shown using Eq. (F15) and that, under the conditions given, $Z_{i\sigma} Z_{l\sigma'}$ commutes with $a_{j\sigma}$:

$$\begin{aligned} \{Z_{i\sigma} Z_{l\sigma'}, B_{ij\sigma}\} &= -\tau \{Z_{i\sigma} Z_{l\sigma'}, a_{i\sigma}^\dagger a_{j\sigma}\} \\ &= -\tau \{Z_{i\sigma} Z_{l\sigma'}, a_{i\sigma}^\dagger\} a_{j\sigma} = 0. \end{aligned} \quad (\text{F18})$$

Finally, Eq. (F11) is equivalent to Eq. (F12) by taking the Hermitian conjugate

$$\{Z_{i\sigma} Z_{l\sigma'}, B_{ij\sigma}\} = 0 \Leftrightarrow \{Z_{i\sigma} Z_{l\sigma'}, B_{ji\sigma}\} = 0. \quad (\text{F19})$$

2. Evaluating spectral norms numerically

In the following sections we will derive lemmas that can be used to bound Trotter error for Hubbard models, calculated as spectral norms of nested commutators. These lemmas take a nested commutator and partition it into a sum of operators acting on a reduced number of spin orbitals, or qubits. The triangle inequality is then applied to bound the Trotter error as a sum of spectral norms of these reduced operators. The spectral norms of the reduced operators are calculated numerically.

Our numerical calculations of the spectral norm are obtained using the density-matrix renormalization group (DMRG) algorithm, using the Block2 [85] code. For each result, we construct the desired operator O using OpenFermion [86], converting it to a qubit operator using the Jordan-Wigner transformation. We ran Block2 in Pauli mode where quantum numbers are not used. Since the operators under consideration are always Hermitian, the

spectral norm is equal to the maximum of the absolute values of the eigenvalues of O . We can calculate this numerically by finding the lowest eigenvalue of both O and $-O$, and taking the maximum of the absolute values of the two. For operators acting on a sufficiently small number of qubits, this can be achieved using exact eigenvalue solvers. In the following we consider somewhat larger partitions such that this is not possible, and instead use the DMRG method.

It is possible that DMRG may fail to converge, or else converge to a local minimum. In general, it is also necessary to ensure the maximum singular value is found over all symmetry sectors. We include all input and output files on Zenodo (see data availability statement) for Block2 simulations to demonstrate convergence, and the simulation parameters used. We additionally verified this methodology by comparing DMRG results to those from exact eigenvalue solvers for smaller partitions. In testing, we also performed DMRG on operators in both fermionic and qubit operator representations, ensuring consistent results between the two.

We next proceed to derive lemmas to bound Trotter errors in the Hubbard and extended Hubbard models.

3. Hubbard model: $\|[[H_I, H_h], H_h]\|$

In this section, we show that the commutator bound: $[[H_I, H_h], H_h]$ from Ref. [21], given in Eq. (E19), can be further tightened by partitioning the nested commutator into a sum of operators acting on small subsystems for which we can compute the spectral norm exactly. We consider the commutator bound for $[[H_I, H_h], H_h]$, which can be expressed as

$$[[H_I, H_h], H_h] = \frac{U}{4} \sum_i [[Z_{i\uparrow} Z_{i\downarrow}, H_h], H_h]. \quad (\text{F20})$$

We wish to provide a tight bound on the spectral norm of this nested commutator. Evaluating the bound exactly is computationally intractable for all but the smallest lattices, as the operator acts on all $2N$ qubits. Instead, we choose to partition the expression into smaller sets, for which the spectral norm can be evaluated exactly. The simplest approach would be taking the norm of Eq. (F20) and applying the triangle inequality,

$$\|[[H_I, H_h], H_h]\| \leq \frac{U}{4} \sum_i \|[[Z_{i\uparrow} Z_{i\downarrow}, H_h], H_h]\|. \quad (\text{F21})$$

However, for some lattices we may also be able to evaluate the spectral norm for larger sets of operators, acting on more qubits. In this case, we may include a set of neighboring on-site interactions and define a partition of this set. We write $P = \{X_1, X_2, \dots\}$ such that each X_i is a set,

which may contain more than one lattice site. The triangle inequality then yields

$$\|[[H_I, H_h], H_h]\| \leq \frac{U}{4} \sum_{X \in P} \left\| \sum_{i \in X} [[Z_{i\uparrow} Z_{i\downarrow}, H_h], H_h] \right\|. \quad (\text{F22})$$

In the derivation of the commutator bound expression $\|[[H_I, H_h], H_h]\|$ (as well as $\|[[H_C, H_h], H_h]\|$ for the extended Hubbard model), it is useful to consider the following local hopping operator around site i

$$H_k^i = \sum_{l \sim i} (B_{il} + B_{li}), \quad (\text{F23})$$

where $\sum_{l \sim i}$ is a sum over l neighbor to i . The operator H_k^i contains all hopping terms between site i and its nearest neighbors, with the subscript indicating the number of nearest neighbors k .

Lemma F1. For a Hubbard model Hamiltonian with $H_H = H_h + H_I$, defined in Eqs. (F3)–(F4),

$$\begin{aligned} & \|[[H_I, H_h], H_h]\| \\ & \leq \frac{U}{2} \sum_{X \in P} \left\| \sum_{i \in X} Z_{i\uparrow} Z_{i\downarrow} ([H_k^i, H_h] + 2(H_k^i)^2) \right\|. \end{aligned} \quad (\text{F24})$$

Proof. We evaluate the nested commutator as

$$[[H_I, H_h], H_h] = \frac{U}{4} \sum_{X \in P} \left[\left[\sum_{i \in X} Z_{i\uparrow} Z_{i\downarrow}, H_h \right], H_h \right]. \quad (\text{F25})$$

We begin by evaluating the commutator $[\sum_{i \in X} Z_{i\uparrow} Z_{i\downarrow}, H_h]$ using the commutator and anticommutator relations in Eqs. (F9)–(F12)

$$\sum_{i \in X} [Z_{i\uparrow} Z_{i\downarrow}, H_h] = \sum_{i \in X} (2Z_{i\uparrow} Z_{i\downarrow} H_k^i). \quad (\text{F26})$$

We nest the commutator with H_h and use the identity $[AB, C] = A[B, C] + [A, C]B$

$$\begin{aligned} & \sum_{i \in X} [[Z_{i\uparrow} Z_{i\downarrow}, H_h], H_h] \\ & = \sum_{i \in X} (2Z_{i\uparrow} Z_{i\downarrow} [H_k^i, H_h] + 2[Z_{i\uparrow} Z_{i\downarrow}, H_h] H_k^i). \end{aligned} \quad (\text{F27})$$

Then, we reapply Eq. (F26) and simplify the expression

$$\sum_{i \in X} [[Z_{i\uparrow} Z_{i\downarrow}, H_h], H_h] = 2 \sum_{i \in X} Z_{i\uparrow} Z_{i\downarrow} ([H_k^i, H_h] + 2(H_k^i)^2). \quad (\text{F28})$$

Next, we multiply the above expression by $U/4$, sum over the partitions, take the spectral norm and apply the triangle inequality, proving Lemma F1. ■

When making use of Lemma F1 in practice, it is useful to define the active space of the operator on the right-hand side of Eq. (F28). To do this, note that the surviving terms of $[H_k^i, H_h]$ will be localized around site i up to next-nearest neighbors.

Numerical Result F1. Periodic hexagonal lattice: We choose to partition the periodic hexagonal lattice sites into $N/4$ equivalent parts containing four lattice sites in each X with an S_3 star graph connectivity. The resulting S_3 partition operator is a 38 spin-orbital operator and we compute its norm numerically using DMRG as described in Sec. F 2, resulting in the following spectral norm of the S_3 partition operator:

$$\left\| \sum_{i \in X} Z_{i\uparrow} Z_{i\downarrow} ([H_{3,\uparrow}^i, H_h] + 2(H_{3,\uparrow}^i)^2) \right\| \leq 79.1\tau^2. \quad (\text{F29})$$

This commutator bound is identical for all S_3 partitions meaning that the entire commutator bound $\|[[H_I, H_h], H_h]\|$ can be evaluated by summing over $N/4$ S_3 partition commutator bounds, resulting in

$$\|[[H_I, H_h], H_h]\| \leq \frac{U}{2} \cdot \frac{N}{4} 79.1\tau^2 = 9.9U\tau^2 N. \quad (\text{F30})$$

Note that this result is valid for all periodic hexagonal lattices with $L_x = L_y = L \geq 4$ as defined in Appendix D. The commutator bound in Eq. (F30) improves upon the commutator bound that can be obtained using Lemma 2 from Ref. [21] by 31.5%.

4. Extended Hubbard model: $\|[[H_C, H_h], H_h]\|$

We begin by rewriting the on-site interaction term as

$$H_I = \frac{U}{4} \sum_{i=1}^N Z_{i\uparrow} Z_{i\downarrow} = \frac{U}{4k} \sum_{\langle ij \rangle} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}), \quad (\text{F31})$$

where k is the number of nearest neighbors of each lattice site and $\sum_{\langle ij \rangle}$ runs over $kN/2$ nearest-neighbor terms. This allows us to write the full Coulomb term as a sum over lattice bonds, as

$$H_C = \sum_{\langle ij \rangle} \left(\frac{U}{4k} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}) + \frac{V}{4} \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'} \right). \quad (\text{F32})$$

We define a constant, $\alpha = V/U$, and write the Coulomb term as

$$H_C = \sum_{\langle ij \rangle} \left(\frac{U}{4k} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}) + \frac{\alpha U}{4} \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'} \right)$$

$$= \frac{U}{4} \sum_{\langle ij \rangle} \left(\frac{1}{k} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}) + \alpha \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'} \right). \quad (\text{F33})$$

Once again, to provide a bound on the spectral norm of this operator, we partition the lattice into smaller sets, for which we can evaluate the spectral norm exactly. The simplest partition would be the following choice:

$$\|[[H_C, H_h], H_h]\| \leq \frac{U}{4} \sum_{\langle ij \rangle} \left\| \left[\left[\frac{1}{k} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}) + \alpha \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'}, H_h \right], H_h \right] \right\|. \quad (\text{F34})$$

To further tighten this bound, we may include multiple nearest-neighbor interaction terms inside the norm. We define the partitioning $P = \{X_1, X_2, \dots\}$ such that X_i may contain more than one nearest-neighbor pair of sites. This allows us to write a bound which will be tighter in general,

$$\|[[H_C, H_h], H_h]\| \leq \frac{U}{4} \sum_{X \in P} \left\| \sum_{(ij) \in X} \left[\left[\frac{1}{k} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}) + \alpha \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'}, H_h \right], H_h \right] \right\|. \quad (\text{F35})$$

We also define two additional local hopping operators around lattice site i , similarly to the operator defined in Eq. (F23),

$$H_{k,\sigma}^i = \sum_{l \sim i} (B_{il\sigma} + B_{li\sigma}), \quad H_{k-1,\sigma}^{ij} = \sum_{l \sim i, l \neq j} (B_{il\sigma} + B_{li\sigma}), \quad (\text{F36})$$

where $H_{k\sigma}^i$ contain all hopping terms of spin- σ between site i and its k nearest neighbors and $H_{k-1,\sigma}^{ij}$ contains hopping terms of spin- σ between site i and $k-1$ of its neighbors (all hopping terms except for the terms between neighboring sites i and j). The local hopping operators H_k^i , $H_{k,\sigma}^i$ and $H_{k-1,\sigma}^{ij}$ are used along with the rewritten Coulomb term in Eq. (F32) in the following lemma on the spectral norm of the nested commutator $[[H_C, H_h], H_h]$, for extended Hubbard models.

Lemma F2. For an extended Hubbard model Hamiltonian $H_H = H_h + H_I + H_V$, defined in Eqs. (F3)–(F5), with $\alpha = V/U$, on a lattice with N lattice sites where all sites have k nearest neighbors, then

$$\begin{aligned} \|[[H_C, H_h], H_h]\| &\leq \frac{U\alpha}{2} \sum_{X \in P} \left\| \sum_{(ij) \in X} \left(\frac{1}{k\alpha} Z_{i\uparrow} Z_{i\downarrow} ([H_k^i, H_h] + 2(H_k^i)^2) + \frac{1}{k\alpha} Z_{j\uparrow} Z_{j\downarrow} ([H_k^j, H_h] + 2(H_k^j)^2) \right. \right. \\ &\quad + Z_{i\uparrow} Z_{j\downarrow} ([H_{k,\uparrow}^i + H_{k,\downarrow}^j, H_h] + 2(H_{k,\uparrow}^i + H_{k,\downarrow}^j)^2) + Z_{i\downarrow} Z_{j\uparrow} ([H_{k,\downarrow}^i + H_{k,\uparrow}^j, H_h] + 2(H_{k,\downarrow}^i + H_{k,\uparrow}^j)^2) \\ &\quad \left. \left. + \sum_{\sigma} Z_{i\sigma} Z_{j\sigma} ([H_{k-1,\sigma}^{ij} + H_{k-1,\sigma}^{ji}, H_h] + 2(H_{k-1,\sigma}^{ij} + H_{k-1,\sigma}^{ji})^2) \right) \right\|. \quad (\text{F37}) \end{aligned}$$

Proof. We evaluate the nested commutator as

$$[[H_C, H_h], H_h] = \frac{U}{4} \sum_{X \in P} \left[\left[\sum_{(ij) \in X} \left(\frac{1}{k} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}) + \alpha \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'} \right), H_h \right], H_h \right]. \quad (\text{F38})$$

We first evaluate the commutator of the Coulomb terms and H_h using the commutator and anticommutator relations in Eqs. (F9)–(F12),

$$\left[\sum_{(i,j) \in X} \left(\frac{1}{k} (Z_{i\uparrow} Z_{i\downarrow} + Z_{j\uparrow} Z_{j\downarrow}) + \alpha \sum_{\sigma, \sigma'} Z_{i\sigma} Z_{j\sigma'} \right), H_h \right] = 2\alpha \sum_{(i,j) \in X} \left(\frac{1}{k\alpha} (Z_{i\uparrow} Z_{i\downarrow} H_k^i + Z_{j\uparrow} Z_{j\downarrow} H_k^j) + Z_{i\uparrow} Z_{j\downarrow} (H_{k,\uparrow}^i + H_{k,\downarrow}^j) \right. \\ \left. + Z_{i\downarrow} Z_{j\uparrow} (H_{k,\downarrow}^i + H_{k,\uparrow}^j) + \sum_{\sigma} Z_{i\sigma} Z_{j\sigma} (H_{k-1,\sigma}^{ij} + H_{k-1,\sigma}^{ji}) \right), \quad (\text{F39})$$

where we used the H_k^i , $H_{k,\sigma}^i$, and $H_{k-1,\sigma}^{ij}$ defined in Eqs. (F23) and (F36) to simplify the expressions. Note that for the two interaction terms between spin orbitals of the same spin, the commutation relation in Eq. (F10) shows that the hopping terms between those spin orbitals commutes with the interaction term, leading to the difference between opposite spin interaction terms and same spin interaction terms in the commutator.

We nest the above commutator with H_h and then sum over the partitions to obtain $[[H_C, H_h], H_h]$. We use the identity $[AB, C] = A[B, C] + [A, C]B$ and reapply the above equation, leading to

$$[[H_C, H_h], H_h] = \frac{U\alpha}{2} \sum_{X \in P} \sum_{(i,j) \in X} \left(\frac{1}{k\alpha} Z_{i\uparrow} Z_{i\downarrow} ([H_k^i, H_h] + 2(H_k^i)^2) + \frac{1}{k\alpha} Z_{j\uparrow} Z_{j\downarrow} ([H_k^j, H_h] + 2(H_k^j)^2) \right. \\ \left. + Z_{i\uparrow} Z_{j\downarrow} ([H_{k,\uparrow}^i + H_{k,\downarrow}^j, H_h] + 2(H_{k,\uparrow}^i + H_{k,\downarrow}^j)^2) + Z_{i\downarrow} Z_{j\uparrow} ([H_{k,\downarrow}^i + H_{k,\uparrow}^j, H_h] + 2(H_{k,\downarrow}^i + H_{k,\uparrow}^j)^2) \right. \\ \left. + \sum_{\sigma} Z_{i\sigma} Z_{j\sigma} ([H_{k-1,\sigma}^{ij} + H_{k-1,\sigma}^{ji}, H_h] + 2(H_{k-1,\sigma}^{ij} + H_{k-1,\sigma}^{ji})^2) \right). \quad (\text{F40})$$

We take the norm and apply the triangle inequality. ■

We wish to evaluate the spectral norm of the X -subspace numerically. Therefore, it is useful to define the active space of the operator on the right-hand side of Eq. (F40). The surviving terms of the commutator $[H_k^i, H_h]$ contain operators acting on lattice sites localized around site i and j up to next-nearest neighbors. This allows us to choose a partition of the entire lattice of which the spectral norm of each individual and localized part can be evaluated exactly.

Numerical Result F2. Periodic hexagonal lattice: We choose to partition the lattice into S_3 parts such that each X contains three (i, j) pairs forming an S_3 star graph. We have to sum over $N/2$ completely equivalent partitions to obtain the entire commutator bound. We evaluate the norm of the S_3 partition for parameters $V/U = \alpha = 1/2$ and $k = 3$ (each site in the periodic hexagonal lattice has three neighbors). The resulting operator acts on a 38 spin-orbital or qubit subspace and we compute its spectral norm numerically using DMRG as described in Sec. F2, resulting in the following bound:

$$\|[[H_C, H_h], H_h]\|_{\psi=2} \leq 12.5 U \tau^2 N. \quad (\text{F41})$$

Note that this result is valid for all periodic hexagonal lattices with $L_x = L_y = L \geq 4$ as defined in Appendix D.

5. Extended Hubbard model: $\|[[H_C, H_h], H_C]\|$

We next consider $[[H_C, H_h], H_C]$, which can be expressed as

$$[[H_C, H_h], H_C] = \sum_{\langle ij \rangle} \sum_{\sigma} [[H_C, B_{ij\sigma} + B_{ji\sigma}], H_C]. \quad (\text{F42})$$

Our goal is to provide a tight bound for the spectral norm of this operator. We again choose to partition the expression into smaller sets for which the spectral norm can be evaluated exactly. A simple approach would be to take

$$\left\| \sum_{\langle ij \rangle} \sum_{\sigma} [[H_C, B_{ij\sigma} + B_{ji\sigma}], H_C] \right\| \\ \leq \sum_{\langle ij \rangle} \left\| \sum_{\sigma} [[H_C, B_{ij\sigma} + B_{ji\sigma}], H_C] \right\|. \quad (\text{F43})$$

For some lattices we may also be able to evaluate the spectral norm for larger sets of operators, acting on more qubits. In this case, we may take the set of all bonds, $\langle ij \rangle = \{(i, j) \mid i \text{ and } j \text{ are neighbors}\}$, and again define a partition of this set. We label this partition as P . Then, we can write $P = \{X_1, X_2, \dots\}$, such that each X_i may contain more than one bond on the lattice, for example, we might have $X_1 = \{(1, 2), (2, 3)\}$, which contains two lattice

bonds. Note that this partitioning of the bonds on the lattice is simply a form of tiling of the lattice. Then,

$$\left\| \sum_{\langle ij \rangle} \sum_{\sigma} [[H_C, B_{ij\sigma} + B_{ji\sigma}], H_C] \right\| \leq \sum_{X \in P} \left\| \sum_{\sigma} \sum_{(ij) \in X} [[H_C, B_{ij\sigma} + B_{ji\sigma}], H_C] \right\|. \quad (\text{F44})$$

If each element $X \in P$ contains multiple bonds then Eq. (F44) will in general provide a tighter bound than Eq. (F43). We then need only ensure that the operators, O , inside the spectral norm $\|O\|$, act on sufficiently few qubits, or are sufficiently sparse, such that this norm can be calculated numerically. In general, this will depend on the lattice under consideration.

We write an expression for all the Coulomb interaction terms (from H_I and H_V) that interact with spin orbitals on lattice sites i and j

$$H_{C,ij} = \frac{U}{4} (Z_{i\uparrow}Z_{i\downarrow} + Z_{j\uparrow}Z_{j\downarrow}) + \frac{V}{4} (Z_{i\uparrow}Z_{j\uparrow} + Z_{i\uparrow}Z_{j\downarrow} + Z_{i\downarrow}Z_{j\uparrow} + Z_{i\downarrow}Z_{j\downarrow} + (Z_{i\uparrow} + Z_{i\downarrow})\Sigma^i + (Z_{j\uparrow} + Z_{j\downarrow})\Sigma^j), \quad (\text{F45})$$

where we define Σ^i as the sum over the Z operators on the nearest neighbors of i *other than* site j , and similarly we define Σ^j as the sum over Z operators on the nearest neighbors of j *other than* i . These operators are defined as

$$\Sigma^i \equiv \sum_{\sigma \in \{\uparrow, \downarrow\}} \sum_{l \sim i, l \neq j} Z_{l\sigma}, \quad \Sigma^j \equiv \sum_{\sigma \in \{\uparrow, \downarrow\}} \sum_{l \sim j, l \neq i} Z_{l\sigma}. \quad (\text{F46})$$

The operators Σ^i and Σ^j contain $2(k-1)$ operators of type $Z_{l\sigma}$. To further simplify the commutator bound expressions, we choose to give the bounds for the case where the factor between the interaction parameters U and V is constant. We define $V/U = \alpha$ and use this to rewrite $H_{C,ij}$ as

$$H_{C,ij} = \frac{U\alpha}{4} \left(\frac{1}{\alpha} Z_{i\uparrow}Z_{i\downarrow} + \frac{1}{\alpha} Z_{j\uparrow}Z_{j\downarrow} + Z_{i\uparrow}Z_{j\uparrow} + Z_{i\uparrow}Z_{j\downarrow} + Z_{i\downarrow}Z_{j\uparrow} + Z_{i\downarrow}Z_{j\downarrow} + (Z_{i\uparrow} + Z_{i\downarrow})\Sigma^i + (Z_{j\uparrow} + Z_{j\downarrow})\Sigma^j \right), \quad (\text{F47})$$

We also establish a commutator relation, which is useful for this commutator bound. Using the commutator and anticommutator relations given by Eqs. (F10)–(F12), we evaluate the commutator between $H_{C,ij}$ and the hopping

operators $B_{ij\sigma}$ and $B_{ji\sigma}$

$$[H_{C,ij}, B_{ij\sigma} + B_{ji\sigma}] = \frac{U\alpha}{2} \left(\frac{1}{\alpha} Z_{i\uparrow}Z_{i\downarrow} + \frac{1}{\alpha} Z_{j\uparrow}Z_{j\downarrow} + Z_{i\uparrow}Z_{j\downarrow} + Z_{i\downarrow}Z_{j\uparrow} + Z_{i\sigma}\Sigma^i + Z_{j\sigma}\Sigma^j \right) (B_{ij\sigma} + B_{ji\sigma}). \quad (\text{F48})$$

Finally, we define $C_{ij} \equiv (1/\alpha)Z_{i\uparrow}Z_{i\downarrow} + (1/\alpha)Z_{j\uparrow}Z_{j\downarrow} + Z_{i\uparrow}Z_{j\downarrow} + Z_{i\downarrow}Z_{j\uparrow}$, which is used to rewrite the commutator in Eq. (F48) as

$$[H_{C,ij}, B_{ij\sigma} + B_{ji\sigma}] = \frac{U\alpha}{2} (C_{ij} + Z_{i\sigma}\Sigma^i + Z_{j\sigma}\Sigma^j) (B_{ij\sigma} + B_{ji\sigma}). \quad (\text{F49})$$

Using these relations, we provide a lemma for the commutator bound $[[H_C, H_h], H_C]$ for extended Hubbard models.

Lemma F3. For an extended Hubbard model Hamiltonian $H_{EH} = H_h + H_I + H_V$, defined in Eqs. (F3)–(F5), with $\alpha = V/U$, on a lattice with N lattice sites where all sites have k nearest neighbors,

$$\begin{aligned} & \| [[H_C, H_h], H_C] \| \\ & \leq \frac{U^2\alpha^2}{4} \sum_{X \in P} \left\| \sum_{\sigma} \sum_{(ij) \in X} (C_{ij} + Z_{i\sigma}\Sigma^i + Z_{j\sigma}\Sigma^j)^2 \right. \\ & \quad \left. \times (B_{ij\sigma} + B_{ji\sigma}) \right\|, \end{aligned} \quad (\text{F50})$$

Proof. We evaluate the commutator as

$$[[H_C, H_h], H_C] = \sum_{X \in P} \left[\left[H_C, \sum_{(ij) \in X} (B_{ij\sigma} + B_{ji\sigma}) \right], H_C \right]. \quad (\text{F51})$$

We begin by evaluating the commutator: $[H_C, \sum_{(ij) \in X} (B_{ij\sigma} + B_{ji\sigma})]$. We know from the commutation relation in Eq. (F9) that $B_{ij\sigma}$ and $B_{ji\sigma}$ commute with all Coulomb terms that do not interact with sites i and j , meaning we can evaluate the commutator as $\sum_{(ij) \in X} [H_{C,ij}, B_{ij\sigma} + B_{ji\sigma}]$. Using the result of Eq. (F49), we obtain

$$\begin{aligned} & \sum_{(ij) \in X} [H_{C,ij}, B_{ij\sigma} + B_{ji\sigma}] \\ & = \sum_{(ij) \in X} \frac{U\alpha}{2} (C_{ij} + Z_{i\sigma}\Sigma^i + Z_{j\sigma}\Sigma^j) (B_{ij\sigma} + B_{ji\sigma}). \end{aligned} \quad (\text{F52})$$

The commutator in Eq. (F52) is nested with H_C . We use that the Coulomb terms commute to evaluate the nested

commutator

$$\begin{aligned}
 & [[H_C, \sum_{(i,j) \in X} (B_{ij\sigma} + B_{ji\sigma})], H_C] \\
 &= \sum_{(i,j) \in X} \frac{U\alpha}{2} (C_{ij} + Z_{i\sigma} \Sigma^i + Z_{j\sigma} \Sigma^j) [B_{ij\sigma} + B_{ji\sigma}, H_C], \\
 &= - \sum_{(i,j) \in X} \frac{U^2 \alpha^2}{4} (C_{ij} + Z_{i\sigma} \Sigma^i + Z_{j\sigma} \Sigma^j)^2 (B_{ij\sigma} + B_{ji\sigma}).
 \end{aligned} \tag{F53}$$

Then, we obtain $[[H_C, \sum_{(i,j) \in X} (B_{ij} + B_{ji})], H_C]$ by summing over σ in Eq. (F53),

$$\begin{aligned}
 & [[H_C, \sum_{(i,j) \in X} (B_{ij} + B_{ji})], H_C] \\
 &= - \frac{U^2 \alpha^2}{4} \sum_{\sigma} \sum_{(i,j) \in X} (C_{ij} + Z_{i\sigma} \Sigma^i + Z_{j\sigma} \Sigma^j)^2 (B_{ij\sigma} + B_{ji\sigma}).
 \end{aligned} \tag{F54}$$

Next, we sum over the partitions to obtain

$$\begin{aligned}
 [[H_C, H_h], H_C] &= - \frac{U^2 \alpha^2}{4} \sum_{X \in P} \sum_{\sigma} \sum_{(i,j) \in X} (C_{ij} + Z_{i\sigma} \Sigma^i \\
 &\quad + Z_{j\sigma} \Sigma^j)^2 (B_{ij\sigma} + B_{ji\sigma}),
 \end{aligned} \tag{F55}$$

and take the norm and apply the triangle inequality

$$\begin{aligned}
 \|[[H_C, H_h], H_C]\| &\leq \frac{U^2 \alpha^2}{4} \sum_{X \in P} \left\| \sum_{\sigma} \sum_{(i,j) \in X} (C_{ij} + Z_{i\sigma} \Sigma^i \right. \\
 &\quad \left. + Z_{j\sigma} \Sigma^j)^2 (B_{ij\sigma} + B_{ji\sigma}) \right\|,
 \end{aligned} \tag{F56}$$

which concludes the proof of Lemma F3. $\blacksquare$

In the above we have considered a strict partition of the set $\langle ij \rangle = \{(i,j) \mid i \text{ and } j \text{ are neighbors}\}$ into $P = \{X_1, X_2, \dots\}$. However, we could also define a set $P = \{X_1, X_2, \dots\}$ such that each bond in $\langle ij \rangle$ appears n times in total, for some integer n . In this case, we only need to divide the right-hand side of Eq. (F56) by n to account for this multiple counting. This is equivalent to simply multiplying and dividing by n on the right-hand side of Eq. (F42). In terms of terminology, this is no longer strictly a partition of the set $\langle ij \rangle$, but this does not affect its practical use (equivalently, this could be viewed a partition of the multiset where each bond on the lattice appears n times). We will use this in the numerical result below. For simplicity, we continue to use the term “partition.”

Numerical Result F3. Periodic hexagonal lattice: We choose to partition the lattice into C_6 parts such that each X contains six (i,j) pairs forming a C_6 circle graph. Each bond is contained in two C_6 partitions, meaning we have to sum over $N/2$ completely equivalent C_6 partitions and divide by two to avoid double counting the bonds when evaluating the entire commutator bound. We evaluate the norm of the C_6 partition for $V/U = \alpha = 1/2$. The resulting C_6 partition commutator is a 24-qubit operator and we compute its spectral norm numerically using DMRG as described in Sec. F2, resulting in the following bound:

$$\|[[H_C, H_h], H_C]\|_{\frac{U}{V}=2} \leq 4.80 U^2 \tau N. \tag{F57}$$

Note that this result is valid for all periodic hexagonal lattices with $L_x = L_y = L \geq 4$ as defined in Appendix D.

6. Example: One-dimensional lattices

To demonstrate the tightness of the bounds in Lemmas F1, F2, and F3 for an example, we consider the Hubbard and extended Hubbard model on periodic one-dimensional lattices. Here, it is straightforward to calculate the spectral norms exactly for a number of small lattice sizes, which can then be compared to bounds from the above lemmas.

The numerical results are plotted in Fig. 10. Exact commutator norms are calculated up to 10 lattice sites, which can be compared to the bounds calculated using Eqs. (F30), (F41), and (F57). For the Hubbard model commutator

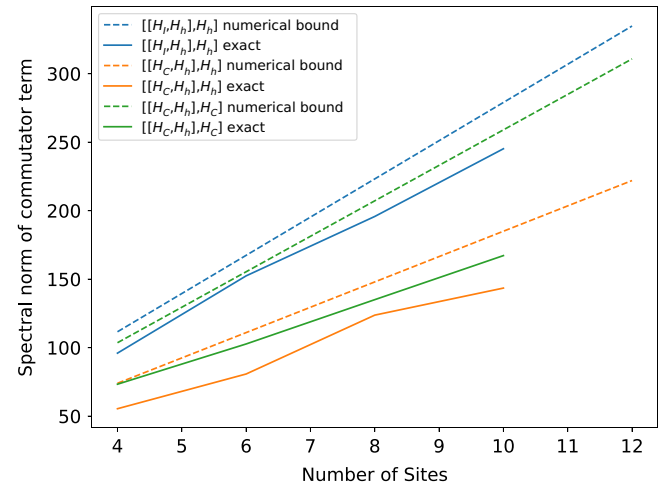


FIG. 10. Numerical results for the spectral norm of commutators considered in Appendix F, presented for a one-dimensional periodic lattice with varying number of sites. Exact norms are calculated for lattices up to 10 sites, and plotted as solid lines. These are then compared to bounds derived in Lemmas F1, F2, and F3 (plotted as dashed lines). We take a partition with two neighboring lattice sites for $\|[[H_I, H_h], H_h]\|$ and a partition containing two neighboring bonds for the extended Hubbard model bounds.

bound, we take a partition containing two neighboring lattice sites, and for the extended Hubbard model bounds we take a partition with two lattice bonds. The corresponding norms in these equations are calculated exactly by first constructing a sparse matrix representation of the operators. The results demonstrate that the bounds are always tight within a factor of at most 2, and often by a much smaller factor. These bounds could be made tighter by taking larger partitions in the above expressions, at the cost of performing spectral norm calculations for operators acting on more qubits.

APPENDIX G: TILE-TROTTERIZATION ERROR NORMS AND RESOURCE REQUIREMENTS

In Table IV we provide the tile-Trotterization error norms (W_{tile}), the total number of qubits (N_Q), and the

number of arbitrary rotations (N_R) and T gates (N_T) required per Trotter step for the hexagonal lattice Hubbard model simulations considered in Figs. 3 and 4. We use Hubbard model parameters $U = 4$, $V = 2$, and $\tau = 1$. The nearest-neighbor interaction parameter, V , is only used in the extended Hubbard model. All systems use the periodic hexagonal lattice model described in Appendix D with parameters $L_x = L_y = L$ and $4 \leq L \leq 18$, for even L . The hopping Hamiltonian Trotter-error constitutes 16% of W_{tile} for the Hubbard model and 10% of W_{tile} for the extended Hubbard model.

APPENDIX H: QUBITIZATION CIRCUITS

In this section we describe how qubitization circuits may be implemented for the Hubbard model on a periodic hexagonal lattice. Our qubitization approach builds

TABLE IV. Tile-Trotterization error norms (W_{tile}), qubit counts (N_Q), arbitrary rotation costs (N_R), and T -gate costs (N_T) for performing a single Trotter step for periodic hexagonal lattice Hubbard models with lattice parameters $L_x = L_y = L$, where $4 \leq L \leq 18$ for even L , and Hubbard model parameters $U = 4$, $V = 2$, and $\tau = 1$. The Hubbard model parameters only affects the Trotter error norms. We show data for the resources required to implement tile Trotterization for the periodic Hubbard model and the periodic extended Hubbard model without HWP ($\alpha = 0$) and with HWP using $\alpha = N/4 - 1$, $\alpha = N/2 - 1$ and $\alpha = N - 1$ ancilla qubits. The data from this table can be used to reproduce the tile-Trotterization costings shown in Figs. 3 and 4.

	$N = 2L^2$	32	72	128	200	288	392	512	648
Hubbard model	W_{tile}	166	374	665	1041	1497	2039	2663	3371
Extended Hubbard model	W_{tile}	262	592	1052	1644	2367	3222	4208	5326
Hubbard model, $\alpha = 0$	N_Q	64	144	256	400	576	784	1024	1296
	N_R	192	432	768	1200	1728	2352	3072	3888
	N_T	320	720	1280	2000	2880	3920	5120	6480
Hubbard model, $\alpha = \frac{N}{4} - 1$	N_Q	71	161	287	449	647	881	1151	1457
	N_R	96	120	144	144	168	168	192	192
	N_T	992	2352	4256	6704	9696	13 232	17 312	21 936
Hubbard model, $\alpha = \frac{N}{2} - 1$	N_Q	79	179	319	499	719	979	1279	1619
	N_R	60	72	84	84	96	96	108	108
	N_T	1040	2400	4304	6752	9744	13 280	17 360	21 984
Hubbard model, $\alpha = N - 1$	N_Q	95	215	383	599	863	1175	1535	1943
	N_R	36	42	48	48	54	54	60	60
	N_T	1064	2424	4328	6776	9768	13 304	17 384	22 008
Extended Hubbard model, $\alpha = 0$	N_Q	64	144	256	400	576	784	1024	1296
	N_R	384	864	1536	2400	3456	4704	6144	7776
	N_T	320	720	1280	2000	2880	3920	5120	6480
Extended Hubbard model, $\alpha = \frac{N}{4} - 1$	N_Q	71	161	287	449	647	881	1151	1457
	N_R	192	240	288	288	336	336	384	384
	N_T	1664	3984	7232	11 408	16 512	22 544	29 504	37 392
Extended Hubbard model, $\alpha = \frac{N}{2} - 1$	N_Q	79	179	319	499	719	979	1279	1619
	N_R	120	144	168	168	192	192	216	216
	N_T	1760	4080	7328	11 504	16 608	22 640	29 600	37 488
Extended Hubbard model, $\alpha = N - 1$	N_Q	95	215	383	599	863	1175	1535	1943
	N_R	72	84	96	96	108	108	120	120
	N_T	1808	4128	7376	11 552	16 656	22 688	29 648	37 536

upon previous work by Babbush *et al.* in Ref. [40]. In Sec. VI we compared the cost of QPE performed with both tile Trotterization and qubitization-based approaches. Here, we provide a brief introduction to the qubitization approach, and present the implementations and costing of SELECT and PREPARE operators used for our QPE resource estimates.

In Trotter-based QPE, we perform phase estimation with a Trotterized approximation of the time-evolution operator, $U = e^{-iHt}$. QPE allows us to estimate the eigenphases of this unitary, and therefore the energies of H . However, it is equally possible to perform phase estimation with other unitaries that encode the Hamiltonian. In qubitization, the unitary of interest is the walk operator, $\mathcal{W}$, which has eigenvalues

$$e^{\pm i \arccos(E_n/\lambda)}, \quad (\text{H1})$$

where E_n are the eigenvalues of H and λ is the L1 norm of H . From these, we can obtain estimates of energies, E_n .

The quantum walk operator is built from SELECT and PREPARE operators. Consider a Hamiltonian

$$H = \sum_{l=0}^{L-1} w_l P_l, \quad \lambda = \sum_{l=0}^{L-1} |w_l|, \quad (\text{H2})$$

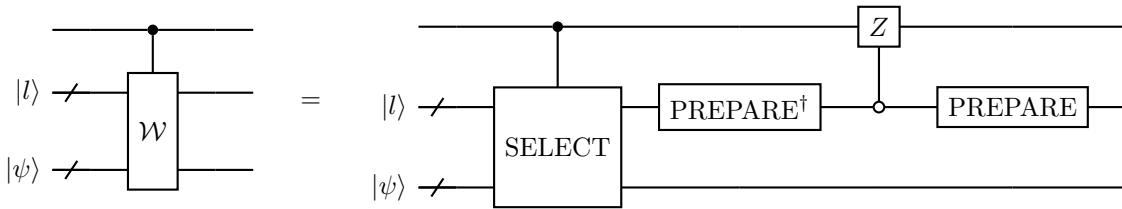
where $w_l > 0$ are coefficients and P_l are (tensor products of) Pauli operators. Then, SELECT defines a block encoding of H/λ . In particular,

$$\text{SELECT} = \sum_{l=0}^{L-1} |l\rangle\langle l| \otimes P_l, \quad (\text{H3})$$

where $|l\rangle$ are flag qubit states. Each state $|l\rangle$ flags a corresponding term in the Hamiltonian, P_l . The PREPARE operator acts on the $|0\rangle$ state of the flag qubits and prepares a state that encodes the coefficients of H . More formally, it prepares the signal state that flags the block encoding of H/λ . It can be defined by

$$\text{PREPARE}|0\rangle = \sum_{l=0}^{L-1} \sqrt{\frac{w_l}{\lambda}} |l\rangle. \quad (\text{H4})$$

The walk operator can be expressed in terms of the SELECT and PREPARE operations, and a reflection operator. In particular, it has been shown that the walk operator controlled on a single ancilla qubit can be expressed by the following operations:



Below we will describe the implementation and costing of each of these operations.

We will consider the following form of the Hubbard Hamiltonian, where the Jordan-Wigner mapping and a chemical potential shift are applied:

$$H_H^{\text{JW}} = -\frac{\tau}{2} \sum_{\langle p,q \rangle, \sigma} (X_{p\sigma} \vec{Z} X_{q\sigma} + Y_{p\sigma} \vec{Z} Y_{q\sigma}) + \frac{U}{4} \sum_{p=1}^N Z_{p\uparrow} Z_{p\downarrow}, \quad (\text{H5})$$

where X , Y , and Z are Pauli operators. We label the Hamiltonian by JW and relabel the site indices as p and q in order to distinguish from the fermionic Hamiltonians defined in II. The operators $\vec{Z}$ indicate a string of Z operators acting on all qubits between $p\sigma$ and $q\sigma$ in the JW ordering.

The gate cost of qubitization-based QPE scales with the L1 norm of the Hamiltonian. For the Hubbard model Hamiltonian, the L1 norm, λ , is given by

$$\lambda = 2\tau \times \text{no. of bonds} + \frac{U}{4} \times \text{no. of sites}. \quad (\text{H6})$$

The number of sites is N and for the periodic hexagonal lattice the number of bonds is $3N/2$, and therefore,

$$\lambda = \left(3\tau + \frac{U}{4}\right) N. \quad (\text{H7})$$

1. SELECT

We first consider the implementation of SELECT for the Hubbard model on a periodic hexagonal lattice. We index terms in the Hamiltonian using the flag

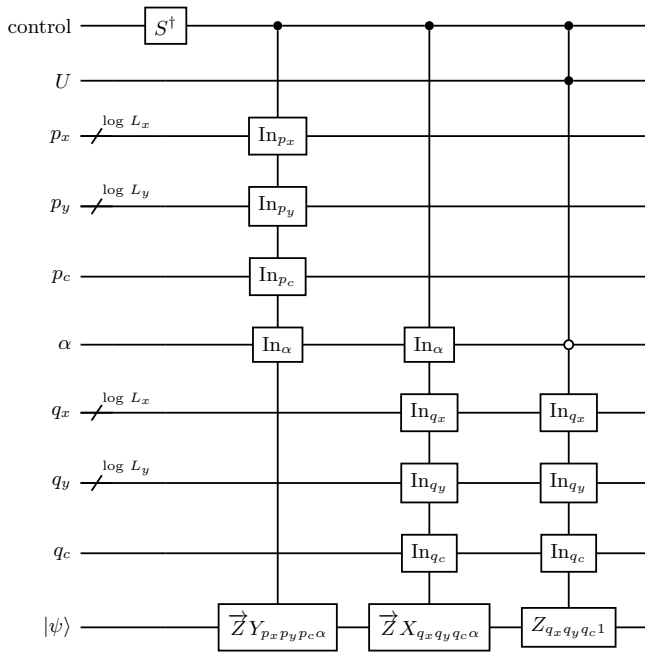


FIG. 11. Controlled SELECT circuit for the Hubbard model on an $L_x \times L_y$ periodic hexagonal lattice.

registers $|U\rangle |p_x\rangle |p_y\rangle |p_c\rangle |\alpha\rangle |q_x\rangle |q_y\rangle |q_c\rangle$. We then define the action of SELECT as

$\text{SELECT}|U, p, \alpha, q\rangle |\psi\rangle$

$$= |U, p, \alpha, q\rangle \begin{cases} Z_{p0} Z_{q1} |\psi\rangle & U \wedge (p = q) \wedge (\alpha = 0) \\ -X_{p\alpha} \vec{Z} X_{q\alpha} |\psi\rangle & \neg U \wedge (p < q) \\ -Y_{q\alpha} \vec{Z} Y_{p\alpha} |\psi\rangle & \neg U \wedge (p > q) \\ \text{UNDEFINED} & \text{otherwise.} \end{cases}$$

The labels p and q are lattice site indices that also include the “color” label, c , coming from the white and gray labels of the lattice points in each (l_x, l_y) pair, as shown in Fig. 7(a). We define the ordering $p = p_x + p_y L_y + p_c L_x L_y$ and $q = q_x + q_y L_y + q_c L_x L_y$. Note that all $c = 0$ terms come before all $c = 1$ terms within this ordering. Also note that for use in qubitized QPE, the action of the UNDEFINED block must be such that the total action of SELECT is Hermitian.

The quantum circuit to achieve this definition of SELECT, controlled on an ancilla qubit, is presented in Fig. 11, and consists of three unary iterators. The Toffoli

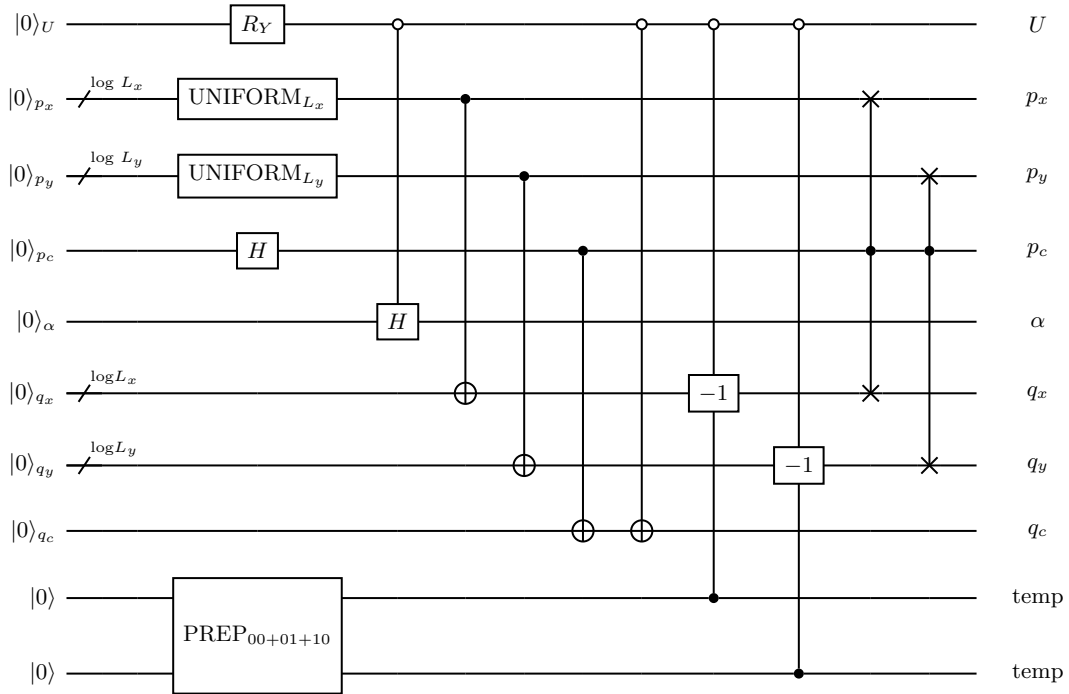


FIG. 12. PREPARE circuit for the Hubbard model on an $L_x \times L_y$ hexagonal lattice. Note that the controlled “ -1 ” operations on the $|q_x\rangle$ and $|q_y\rangle$ registers can be performed mod $2^{\lceil \log_2 L_x \rceil}$ and mod $2^{\lceil \log_2 L_y \rceil}$, respectively. They do not need to be performed mod L_x and mod L_y , as one might expect. This is discussed further in the text of Appendix H.

cost of the first two unary iterators are $4L_xL_y - 1$ each, while the Toffoli cost of the final iterator, which has two additional controls, is $(2L_xL_y - 1) + 2 = 2L_xL_y + 1$, and so the total Toffoli cost combined is $10L_xL_y - 1$. This leads

to a T -gate cost of $C_S = 40L_xL_y - 4$. Expressing this in terms of the number of lattice points, N , we obtain $C_S = 20N - 4$. This is asymptotically the most expensive subroutine in the walk operator.

2. PREPARE

The action of PREPARE is defined to act on the flag qubits prepared in state $|0\rangle$ as

$$\begin{aligned} \text{PREPARE } |0\rangle = & \sqrt{\frac{U}{4\lambda}} \sum_{p_x=0}^{L_x-1} \sum_{p_y=0}^{L_y-1} \sum_{p_c=0}^1 |1\rangle_U |p_x, p_y, p_c, 0\rangle |p_x, p_y, p_c\rangle \\ & + \sqrt{\frac{t}{2\lambda}} \sum_{p_x=0}^{L_x-1} \sum_{p_y=0}^{L_y-1} \sum_{\sigma=0}^1 |0\rangle_U (|p_x, p_y, 0, \sigma\rangle |p_x, p_y, 1\rangle + |p_x, p_y, 1, \sigma\rangle |p_x, p_y, 0\rangle \\ & + |p_x, p_y, 0, \sigma\rangle |p_x - 1, p_y, 1\rangle + |p_x - 1, p_y, 1, \sigma\rangle |p_x, p_y, 0\rangle \\ & + |p_x, p_y, 0, \sigma\rangle |p_x, p_y - 1, 1\rangle + |p_x, p_y - 1, 1, \sigma\rangle |p_x, p_y, 0\rangle). \end{aligned} \quad (\text{H8})$$

TABLE V. The cost of various circuit elements to perform the controlled quantized walk operator for the Hubbard model on the $L_x \times L_y$ periodic hexagonal Hubbard model. The T -gate counts include the cost of converting Toffoli gates into T gates. The “ancilla qubits” column is the total number of ancilla qubits for each circuit element, without reusing qubits, and not including flag qubits. The value η_{L_x} (η_{L_y}) is the largest power of 2 that is a factor of L_x (L_y), Θ is the number of T gates per rotation in the UNIFORM state-preparation circuits, and Γ is the number of T gates used to implement each of the other rotation gates in PREPARE. The “concat. success qubits” element refers to concatenating the success qubits from the two UNIFORM operations, which SELECT must be controlled on. The “reflection” step uses the result from Ref. [70] to perform a Z gate with n controls in $2n - 3$ Toffolis, with a single ancilla. In practice, ancilla qubits can mostly be shared between subroutines. The only ancillas that are not reused are the rotation qubit and success flag qubit from each UNIFORM state preparation, the concatenated success qubit and a qubit prepared in the $|T\rangle$ state in the controlled Hadamard operation. Otherwise, subroutines in PREPARE can use the $\lceil \log_2 L_x \rceil + \lceil \log_2 L_y \rceil + 3$ ancillas required by SELECT. Therefore, the total number of ancilla qubits required is $\lceil \log_2 L_x \rceil + \lceil \log_2 L_y \rceil + 9$, in addition to $2\lceil \log_2 L_x \rceil + 2\lceil \log_2 L_y \rceil + 6$ flag qubits. Compared to the Hubbard model qubitization circuits of Ref. [40], our implementation removes a term in SELECT that is logarithmic in the lattice size, and also a flag qubit.

Circuit element	Toffoli count	T count (incl. Toffolis)	Ancilla qubits
Controlled SELECT	$10L_xL_y - 1$	$40L_xL_y - 4$	$\lceil \log_2 L_x \rceil + \lceil \log_2 L_y \rceil + 3$
PREPARE:			
UNIFORM $\frac{1}{\sqrt{L_x}} \sum_{l=0}^{L_x-1} l\rangle$	$3\lceil \log_2 L_x \rceil - 3\eta_{L_x} - 3$	$4 \times \text{Toffoli count} + 2\Theta$	$\lceil \log_2 L_x \rceil - \eta_{L_x} + 2$
UNIFORM $\frac{1}{\sqrt{L_y}} \sum_{l=0}^{L_y-1} l\rangle$	$3\lceil \log_2 L_y \rceil - 3\eta_{L_y} - 3$	$4 \times \text{Toffoli count} + 2\Theta$	$\lceil \log_2 L_y \rceil - \eta_{L_y} + 2$
concat. success qubits	1	4	1
controlled Hadamard	1	4	2
controlled -1 on q_x register	$\lceil \log_2 L_x \rceil$	$4\lceil \log_2 L_x \rceil$	$\lceil \log_2 L_x \rceil$
controlled -1 on q_y register	$\lceil \log_2 L_y \rceil$	$4\lceil \log_2 L_y \rceil$	$\lceil \log_2 L_y \rceil$
Two controlled swaps	$\lceil \log_2 L_x \rceil + \lceil \log_2 L_y \rceil$	$7\lceil \log_2 L_x \rceil + 7\lceil \log_2 L_y \rceil$	0
$R_Y(\theta)$	—	Γ	1
PREP ₀₀₊₀₁₊₁₀	—	3Γ	1
Reflection	$2(2\lceil \log_2 L_x \rceil + 2\lceil \log_2 L_y \rceil + 10) - 3$	$4 \times \text{Toffoli count}$	1

The quantum circuit to achieve this definition of PREPARE is shown in Fig. 12. This consists of a number of circuit elements that we will briefly explain. The final Toffoli and T -gate costs and ancilla costs for each circuit element are summarized in Table V.

The UNIFORM operations perform uniform state preparation, and can be implemented using amplitude amplification by the approach described in Appendix A of Ref. [6]. However, we make a slight modification to the approach described there. In particular, Ref. [6] implements the ancilla rotation by using a phase-gradient ancilla register, which requires b_n ancilla qubits and $b_n - 3$ Toffoli gates. However, as noted in Appendix A of Ref. [87], when the rotation angle β is known classically, as is the case here, this can also be implemented efficiently using rotation synthesis. Here, we use this alternative approach, requiring just a single ancilla for the rotation synthesis. For this gadget, the rotation synthesis can be lower precision than elsewhere; we define Θ to be the number of T gates used for each rotation in each UNIFORM state preparation, which is separate from the number of T gates used in synthesis of other rotations in PREPARE, which we denote Γ . As described in Ref. [6], the Toffoli cost consists of two inequality tests and a reflection. Of the ancillas required, the rotation and success qubits are flag qubits and cannot be reused by other routines. We also note that the walk operator's reflection is controlled on the rotation and success qubits, which are included in the final Toffoli count.

The circuit to perform a controlled Hadamard gate is shown in Fig. 17 of the same paper by Lee *et al.* [6], which requires two ancilla qubits and one Toffoli gate. One of the ancilla qubits is prepared in a $|T\rangle$ state, which we choose to preserve for future applications of the walk operator; therefore, this ancilla qubit cannot be reused, which is accounted for in the total qubit count.

It is also necessary to prepare the state

$$|\Psi_{00+01+10}\rangle = \frac{1}{\sqrt{3}}(|00\rangle + |01\rangle + |10\rangle), \quad (\text{H9})$$

which we achieve using a circuit of the form in Fig. 13, using three rotation gates. We denote the number of T gates per rotation in this subroutine as Γ . For our costing in Sec. VI, we set $\Gamma = 40$.

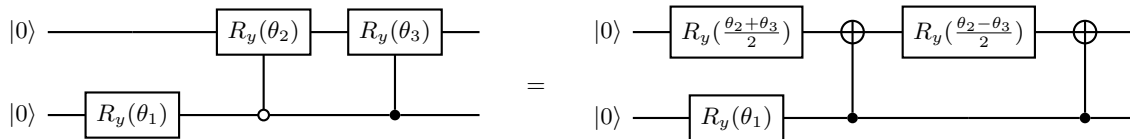


FIG. 13. Demonstration of how $\text{PREP}_{00+01+10}$ can be implemented. This requires three rotation gates, which can be implemented to precision ϵ with an expected T -gate count of $1.15 \log(1/\epsilon) + 9.2$ each. In the text, we denote the number of T gates used to implement each of these rotations by Γ , which we set to 40. The required rotation angles are $\theta_1 = 2 \tan^{-1}(1/\sqrt{2})$, $\theta_2 = \pi/4$, and $\theta_3 = 0$, with $R_y(\theta) = e^{-i\theta Y/2}$.

We next consider how to perform the controlled “ -1 ” operation. One might expect that we actually have to perform “ $-1 \bmod L_x$ ” or “ $-1 \bmod L_y$ ” operations to properly enforce periodic boundary conditions, similarly to the PREPARE circuit of Ref. [40]. However, we can avoid this complication. To see this, consider the case of an $L \times L$ periodic lattice, and let us consider the “ -1 ” operation on the $|q_x\rangle$ register. This register will consist of $n = \lceil \log_2 L \rceil$ qubits. Before this operation, the register will hold a uniform superposition of states $|l\rangle$, for $0 \leq l \leq L-1$. For $l \geq 1$ the “ -1 ” operation will act in the expected manner. For $l = 0$ it will instead give the state $|2^n - 1\rangle$. By periodic boundary conditions, we need lattice sites with $x = 0$ to be connected to sites with $x = L-1$, and so this may seem incorrect. However, provided that SELECT is implemented appropriately, this ultimately leads to the correct circuit. In particular, the SELECT circuit consists of unary iterators, which are designed to act on $|l\rangle$ for $l \leq L$. The action of a controlled unary iterator on $|l\rangle|\Psi\rangle$ is

$$|c\rangle|l\rangle|\Psi\rangle \rightarrow |c\rangle|l\rangle(P_l)^c|\Psi\rangle, \quad (\text{H10})$$

where $|c\rangle$ is the control qubit. In Ref. [40], the authors introduce optimizations to the basic SELECT circuit. This is achieved by removing “runs” of the OFF controls on the right side of the circuit. These result in the well-established “sawtooth” circuits, which can be implemented in $L-1$ Toffoli gates. However, removing these OFF controls also allows us to simplify our circuits as described above. In particular, in this case it can be seen that

$$|c\rangle|2^n - 1\rangle|\Psi\rangle \rightarrow |c\rangle|2^n - 1\rangle(P_{L-1})^c|\Psi\rangle. \quad (\text{H11})$$

In words, if the value $l = 2^n - 1$ is provided to the unary iterator, it is guaranteed to select the $l = L-1$ term, which is the desired behavior to enforce periodic boundary conditions. This is a convenient benefit of the optimized iterators, which avoids us needing to perform modular “ $-1 \bmod L$ ” addition, and instead allows us to simply perform “ $-1 \bmod 2^n$ ” addition.

The adder circuit can be further simplified, due to the fact that it always subtracts 1. For an n -bit binary number, subtracting 1 is equivalent to adding $1_1 1_2 \dots 1_n$. Therefore each bit of the value added is 1, and we can prepare and

unprepare a single ancilla qubit by using two Toffoli gates to incorporate the two controls for the adders. Then we use the circuit from Ref. [87] (Fig. 18) to perform addition with respect to this ancilla, the Toffoli cost of which is $\lceil \log_2 L \rceil - 2$. Therefore, the total Toffoli costs of the two controlled adders are $\lceil \log_2 L_x \rceil$ and $\lceil \log_2 L_y \rceil$.

- [1] R. P. Feynman, Simulating physics with computers, *Int. J. Theor. Phys.* **21**, 467 (1982).
- [2] S. Lloyd, Universal quantum simulators, *Science* **273**, 1073 (1996).
- [3] I. Kassal, S. P. Jordan, P. J. Love, M. Mohseni, and A. Aspuru-Guzik, Polynomial-time quantum algorithm for the simulation of chemical dynamics, *Proc. Natl. Acad. Sci. U.S.A.* **105**, 18681 (2008).
- [4] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Elucidating reaction mechanisms on quantum computers, *Proc. Natl. Acad. Sci. U.S.A.* **114**, 7555 (2017).
- [5] J. J. Goings, A. White, J. Lee, C. S. Tautermann, M. Degroote, C. Gidney, T. Shiozaki, R. Babbush, and N. C. Rubin, Reliably assessing the electronic structure of cytochrome P450 on today's classical computers and tomorrow's quantum computers, *Proc. Natl. Acad. Sci.* **119**, e2203533119 (2022).
- [6] J. Lee, D. W. Berry, C. Gidney, W. J. Huggins, J. R. McClean, N. Wiebe, and R. Babbush, Even more efficient quantum computations of chemistry through tensor hypercontraction, *PRX Quantum* **2**, 030305 (2021).
- [7] D. W. Berry, C. Gidney, M. Motta, J. R. McClean, and R. Babbush, Qubitization of arbitrary basis quantum chemistry leveraging sparsity and low rank factorization, *Quantum* **3**, 208 (2019).
- [8] H. H. S. Chan, R. Meister, T. Jones, D. P. Tew, and S. C. Benjamin, Grid-based methods for chemistry simulations on a quantum computer, *Sci. Adv.* **9**, eabo7484 (2023).
- [9] I. D. Kivlichan, N. Wiebe, R. Babbush, and A. Aspuru-Guzik, Bounding the costs of quantum simulation of many-body physics in real space, *J. Phys. A: Math. Theor.* **50**, 305301 (2017).
- [10] A. V. Ivanov, C. Sünderhauf, N. Holzmann, T. Ellaby, R. N. Kerber, G. Jones, and J. Camps, Quantum computation for periodic solids in second quantization, *Phys. Rev. Res.* **5**, 013200 (2023).
- [11] R. Babbush, N. Wiebe, J. McClean, J. McClain, H. Neven, and G. K.-L. Chan, Low-depth quantum simulation of materials, *Phys. Rev. X* **8**, 011044 (2018).
- [12] N. C. Rubin, D. W. Berry, F. D. Malone, A. F. White, T. Khatyar, A. E. DePrince, S. Siculo, M. Kühn, M. Kaicher, J. Lee, and R. Babbush, Fault-tolerant quantum simulation of materials using Bloch orbitals, *PRX Quantum* **4**, 040303 (2023).
- [13] J. Hubbard, Electron correlations in narrow energy bands, *Proc. R. Soc. Lond. A: Math. Phys. Sci.* **276**, 238 (1963).
- [14] N. Yoshioka, T. Okubo, Y. Suzuki, Y. Koizumi, and W. Mizukami, Hunting for quantum-classical crossover in condensed matter problems, *npj Quantum Inf.* **10**, 45 (2024).
- [15] R. Toshio, Y. Akahoshi, J. Fujisaki, H. Oshima, S. Sato, and K. Fujii, Practical quantum advantage on partially fault-tolerant quantum computer, *Phys. Rev. X* **15**, 021057 (2025).
- [16] Y. Akahoshi, R. Toshio, J. Fujisaki, H. Oshima, S. Sato, and K. Fujii, Compilation of trotter-based time evolution for partially fault-tolerant quantum computing architecture, [arXiv:2408.14929](https://arxiv.org/abs/2408.14929) [quant-ph].
- [17] X. Dong, L. D. Re, A. Toschi, and E. Gull, Mechanism of superconductivity in the Hubbard model at intermediate interaction strength, *Proc. Natl. Acad. Sci.* **119**, e2205048119 (2022).
- [18] J. Kaczmarczyk, J. Spałek, T. Schickling, and J. Büne-mann, Superconductivity in the two-dimensional Hubbard model: Gutzwiller wave function solution, *Phys. Rev. B* **88**, 115127 (2013).
- [19] P. A. Lee, N. Nagaosa, and X.-G. Wen, Doping a Mott insulator: Physics of high-temperature superconductivity, *Rev. Mod. Phys.* **78**, 17 (2006).
- [20] I. D. Kivlichan, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, W. Sun, Z. Jiang, N. Rubin, A. Fowler, A. Aspuru-Guzik, H. Neven, and R. Babbush, Improved fault-tolerant quantum simulation of condensed-phase correlated electrons via Trotterization, *Quantum* **4**, 296 (2020).
- [21] E. T. Campbell, Early fault-tolerant simulations of the Hubbard model, *Quantum Sci. Technol.* **7**, 015007 (2021).
- [22] H. F. Trotter, On the product of semi-groups of operators, *Proc. Am. Math. Soc.* **10**, 545 (1959).
- [23] M. Suzuki, General theory of fractal path integrals with applications to many-body theories and statistical physics, *J. Math. Phys.* **32**, 400 (1991).
- [24] J. B. James D. Whitfield and A. Aspuru-Guzik, Simulation of electronic structure Hamiltonians using quantum computers, *Mol. Phys.* **109**, 735 (2011).
- [25] A. Y. Kitaev, Quantum measurements and the Abelian stabilizer problem, [arXiv:quant-ph/9511026](https://arxiv.org/abs/quant-ph/9511026) [quant-ph].
- [26] D. S. Abrams and S. Lloyd, Quantum algorithm providing exponential speed increase for finding eigenvalues and eigenvectors, *Phys. Rev. Lett.* **83**, 5162 (1999).
- [27] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Quantum algorithms revisited, *Proc. R. Soc. Lond. Ser. A: Math. Phys. Eng. Sci.* **454**, 339 (1998).
- [28] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, and M. Head-Gordon, Simulated quantum computation of molecular energies, *Science* **309**, 1704 (2005).
- [29] R. D. Somma, Quantum eigenvalue estimation via time series analysis, *New J. Phys.* **21**, 123025 (2019).
- [30] L. Lin and Y. Tong, Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers, *PRX Quantum* **3**, 010318 (2022).
- [31] K. Wan, M. Berta, and E. T. Campbell, Randomized quantum algorithm for statistical phase estimation, *Phys. Rev. Lett.* **129**, 030503 (2022).
- [32] G. Wang, D. S. França, R. Zhang, S. Zhu, and P. D. Johnson, Quantum algorithm for ground state energy estimation using circuit depth with exponentially improved dependence on precision, *Quantum* **7**, 1167 (2023).
- [33] N. S. Blunt, L. Caune, R. Izsák, E. T. Campbell, and N. Holzmann, Statistical phase estimation and error mitigation on a superconducting quantum processor, *PRX Quantum* **4**, 040341 (2023).

- [34] Z. Ding, C.-F. Chen, and L. Lin, Single-ancilla ground state preparation via lindbladians, *Phys. Rev. Res.* **6**, 033147 (2024).
- [35] H.-E. Li, Y. Zhan, and L. Lin, Dissipative ground state preparation in ab initio electronic structure theory, [arXiv:2411.01470](#) [quant-ph].
- [36] C.-F. Chen, M. J. Kastoryano, F. G. S. L. Brandão, and A. Gilyén, Quantum thermal state preparation, [arXiv:2303.18224](#) [quant-ph].
- [37] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, *Quantum* **3**, 163 (2019).
- [38] D. W. Berry, M. Kieferová, A. Scherer, Y. R. Sanders, G. H. Low, N. Wiebe, C. Gidney, and R. Babbush, Improved techniques for preparing eigenstates of fermionic Hamiltonians, *npj Quantum Inf.* **4**, 22 (2018).
- [39] D. Poulin, A. Kitaev, D. S. Steiger, M. B. Hastings, and M. Troyer, Quantum algorithm for spectral measurement with a lower gate count, *Phys. Rev. Lett.* **121**, 010501 (2018).
- [40] R. Babbush, C. Gidney, D. W. Berry, N. Wiebe, J. McClean, A. Paler, A. Fowler, and H. Neven, Encoding electronic spectra in quantum circuits with linear T complexity, *Phys. Rev. X* **8**, 041015 (2018).
- [41] A. Kan and Benjamin C. B. Symons, Resource-optimized fault-tolerant simulation of the Fermi-Hubbard model and high-temperature superconductor models, *npj Quantum Inf.* **11**, 138 (2025).
- [42] Y. Yu, S. Li, S. Isakov, and E. Gull, Magnetic phases of the anisotropic triangular lattice Hubbard model, *Phys. Rev. B* **107**, 075106 (2023).
- [43] C. Wen, X. Zhu, Z. Xiao, N. Hao, R. Mondaini, H. Guo, and S. Feng, Superconducting pairing symmetry in the kagome-lattice Hubbard model, *Phys. Rev. B* **105**, 075118 (2022).
- [44] W. Kennedy, S. a. d. A. Sousa-Júnior, N. C. Costa, and R. R. dos Santos, Extended Hubbard model on a honeycomb lattice, *Phys. Rev. B* **111**, 155118 (2025).
- [45] J. Paki, H. Terletska, S. Isakov, and E. Gull, Charge order and antiferromagnetism in the extended Hubbard model, *Phys. Rev. B* **99**, 245146 (2019).
- [46] F. Ferrari, F. Becca, and R. Valentí, Charge density waves in kagome-lattice extended Hubbard models at the van Hove filling, *Phys. Rev. B* **106**, L081107 (2022).
- [47] T. Schwemmer, H. Hohmann, M. Dürrnagel, J. Potten, J. Beyer, S. Rachel, Y.-M. Wu, S. Raghu, T. Müller, W. Hanke, and R. Thomale, Sublattice modulated superconductivity in the kagome Hubbard model, *Phys. Rev. B* **110**, 024501 (2024).
- [48] J. B. Profe, L. Klebl, F. Grandi, H. Hohmann, M. Dürrnagel, T. Schwemmer, R. Thomale, and D. M. Kennes, Kagome Hubbard model from a functional renormalization group perspective, *Phys. Rev. Res.* **6**, 043078 (2024).
- [49] S.-Z. Zhou, K. Cheng, and Z.-B. Huang, Unconventional superconductivity and charge ordering in an extended ionic Hubbard model on the triangular lattice, *Res. Phys.* **61**, 107748 (2024).
- [50] M. L. Kiesel and R. Thomale, Sublattice interference in the kagome Hubbard model, *Phys. Rev. B* **86**, 121105 (2012).
- [51] M. L. Kiesel, C. Platt, and R. Thomale, Unconventional Fermi surface instabilities in the kagome Hubbard model, *Phys. Rev. Lett.* **110**, 126405 (2013).
- [52] M. Schüler, M. Rösner, T. O. Wehling, A. I. Lichtenstein, and M. I. Katsnelson, Optimal Hubbard models for materials with nonlocal Coulomb interactions: Graphene, silicene, and benzene, *Phys. Rev. Lett.* **111**, 036601 (2013).
- [53] P.-L. Dallaire-Demers, J. Romero, L. Veis, S. Sim, and A. Aspuru-Guzik, Low-depth circuit ansatz for preparing correlated fermionic states on a quantum computer, *Quantum Sci. Technol.* **4**, 045005 (2019).
- [54] T. G. Schmalz, L. Serrano-Andrés, V. Sauri, M. Merchán, and J. M. Oliva, A distance-dependent parameterization of the extended Hubbard model for conjugated and aromatic hydrocarbons derived from stretched ethene, *J. Chem. Phys.* **135**, 194103 (2011).
- [55] Y. Yoshida, N. Takemori, and W. Mizukami, Ab initio extended Hubbard model of short polyenes for efficient quantum computing, *J. Chem. Phys.* **161**, 084303 (2024).
- [56] A. L. Szabó and B. Roy, Extended Hubbard model in undoped and doped monolayer and bilayer graphene: Selection rules and organizing principle among competing orders, *Phys. Rev. B* **103**, 205135 (2021).
- [57] J. A. Pople, Electron interaction in unsaturated hydrocarbons, *Trans. Faraday Soc.* **49**, 1375 (1953).
- [58] R. Pariser and R. G. Parr, A semi-empirical theory of the electronic spectra and electronic structure of complex unsaturated molecules. I, *J. Chem. Phys.* **21**, 466 (1953).
- [59] T. G. Schmalz, From the Hubbard to the PPP model, *Croat. Chem. Acta* **86**, 419 (2013).
- [60] E. V. Boström, A. Mikkelsen, C. Verdozzi, E. Perfetto, and G. Stefanucci, Charge separation in Donor-C60 complexes with real-time Green functions: The importance of nonlocal correlations, *Nano Lett.* **18**, 785 (2018).
- [61] S. Mishra, D. Beyer, K. Eimre, S. Kezilebieke, R. Berger, O. Gröning, C. A. Pignedoli, K. Müllen, P. Liljeroth, P. Ruffieux, X. Feng, and R. Fasel, Topological frustration induces unconventional magnetism in a nanographene, *Nat. Nanotechnol.* **15**, 22 (2020).
- [62] S. Song, A. Pinar Solé, A. Matěj, G. Li, O. Stetsovych, D. Soler, H. Yang, M. Telychko, J. Li, M. Kumar, Q. Chen, S. Edalatmanesh, J. Brabec, L. Veis, J. Wu, P. Jelinek, and J. Lu, Highly entangled polyradical nanographene with coexisting strong correlation and topological frustration, *Nat. Chem.* **16**, 938 (2024).
- [63] G. Chiappe, E. Louis, E. San-Fabián, and J. A. Vergés, Can model Hamiltonians describe the electron–electron interaction in π -conjugated systems? PAH and graphene, *J. Phys.: Condens. Matter* **27**, 463001 (2015).
- [64] G. Burkard, Recipes for the digital quantum simulation of lattice spin systems, *SciPost Phys. Core* **8**, 030 (2025).
- [65] J. Aftab, D. An, and K. Trivisa, Multi-product Hamiltonian simulation with explicit commutator scaling, [arXiv:2403.08922](#) [quant-ph].
- [66] K. Setia, S. Bravyi, A. Mezzacapo, and J. D. Whitfield, Superfast encodings for fermionic quantum simulation, *Phys. Rev. Res.* **1**, 033033 (2019).
- [67] T. Guaita, On the locality of qubit encodings of local fermionic modes, *Quantum* **9**, 1644 (2025).
- [68] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, Theory of Trotter error with commutator scaling, *Phys. Rev. X* **11**, 011020 (2021).

- [69] C. Gidney, Halving the cost of quantum addition, *Quantum* **2**, 74 (2018).
- [70] T. Khattar and C. Gidney, Rise of conditionally clean ancillae for efficient quantum circuit constructions, *Quantum* **9**, 1752 (2025).
- [71] D. W. Berry, B. L. Higgins, S. D. Bartlett, M. W. Mitchell, G. J. Pryde, and H. M. Wiseman, How to perform the most accurate possible phase measurements, *Phys. Rev. A* **80**, 052114 (2009).
- [72] B. L. Higgins, D. W. Berry, S. D. Bartlett, H. M. Wiseman, and G. J. Pryde, Entanglement-free Heisenberg-limited phase estimation, *Nature* **450**, 393 (2007).
- [73] A. Bocharov, M. Roetteler, and K. M. Svore, Efficient synthesis of universal repeat-until-success quantum circuits, *Phys. Rev. Lett.* **114**, 080502 (2015).
- [74] T. N. Georges, M. Bothe, C. Sünderhauf, B. K. Berntson, R. Izsák, and A. V. Ivanov, Quantum simulations of chemistry in first quantization with any basis set, *npj Quantum Inf.* **11**, 55 (2025).
- [75] S. Lee, J. Lee, H. Zhai, Y. Tong, A. M. Dalzell, A. Kumar, P. Helms, J. Gray, Z. H. Cui, W. Liu, M. Kastoryano, R. Babbush, J. Preskill, D. R. Reichman, E. T. Campbell, E. F. Valeev, L. Lin, and G. K. L. Chan, Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry, *Nat. Commun.* **14**, 1952 (2023).
- [76] M. Motta, K. J. Sung, K. B. Whaley, M. Head-Gordon, and J. Shee, Bridging physical intuition and hardware efficiency for correlated electronic states: The local unitary cluster Jastrow ansatz for electronic structure, *Chem. Sci.* **14**, 11213 (2023).
- [77] C. Gidney, N. Shutty, and C. Jones, Magic state cultivation: Growing T states as cheap as CNOT gates, [arXiv:2409.17595](https://arxiv.org/abs/2409.17595) [quant-ph].
- [78] D. Litinski, A game of surface codes: Large-scale quantum computing with lattice surgery, *Quantum* **3**, 128 (2019).
- [79] Y. Akahoshi, K. Maruyama, H. Oshima, S. Sato, and K. Fujii, Partially fault-tolerant quantum computing architecture with error-corrected Clifford gates and space-time efficient analog rotations, *PRX Quantum* **5**, 010337 (2024).
- [80] Z. Ding and L. Lin, Simultaneous estimation of multiple eigenvalues with short-depth quantum circuit on early fault-tolerant quantum computers, *Quantum* **7**, 1136 (2023).
- [81] <https://doi.org/10.5281/zenodo.15674475>
- [82] F. Verstraete, J. I. Cirac, and J. I. Latorre, Quantum circuits for strongly correlated quantum systems, *Phys. Rev. A* **79**, 032316 (2009).
- [83] L. Bassman Oftelie, R. Van Beeumen, E. Younis, E. Smith, C. Iancu, and W. A. de Jong, Constant-depth circuits for dynamic simulations of materials on quantum computers, *Mater. Theory* **6**, 13 (2022).
- [84] A. Schubert and C. B. Mendl, Trotter error with commutator scaling for the Fermi-Hubbard model, *Phys. Rev. B* **108**, 195105 (2023).
- [85] H. Zhai, H. R. Larsson, S. Lee, Z.-H. Cui, T. Zhu, C. Sun, L. Peng, R. Peng, K. Liao, J. Tölle, J. Yang, S. Li, and G. K.-L. Chan, BLOCK2: A comprehensive open source framework to develop and apply state-of-the-art DMRG algorithms in electronic structure and beyond, *J. Chem. Phys.* **159**, 234801 (2023).
- [86] J. R. McClean *et al.*, Openfermion: The electronic structure package for quantum computers, *Quantum Sci. Technol.* **5**, 034014 (2020).
- [87] Y. R. Sanders, D. W. Berry, P. C. Costa, L. W. Tessler, N. Wiebe, C. Gidney, H. Neven, and R. Babbush, Compilation of fault-tolerant quantum heuristics for combinatorial optimization, *PRX Quantum* **1**, 020312 (2020).