

Optimized Quantum Simulation Algorithms for Scalar Quantum Field Theories

Andrew Hardy,^{1,*} Priyanka Mukhopadhyay,^{2,3,†} M. Sohaib Alam,^{4,5,3,‡} Robert Konik,^{6,§} Layla Hormozi,⁷ Eleanor Rieffel,^{4,¶} Stuart Hadfield,^{4,5} João Barata,⁸ Raju Venugopalan,⁸ Dmitri E. Kharzeev,^{9,8} and Nathan Wiebe^{2,10,11,**}

¹*Department of Physics, University of Toronto, Toronto, ON M5S 1A7, Canada*

²*Department of Computer Science, University of Toronto, Toronto, ON, M5S 2E4, Canada*

³*These authors made equivalent contributions to this work.*

⁴*Quantum Artificial Intelligence Laboratory (QuAIL),*

NASA Ames Research Center, Moffett Field, CA, 94035, USA

⁵*USRA Research Institute for Advanced Computer Science (RIACS), Mountain View, CA, 94043, USA*

⁶*Condensed Matter and Materials Science Division,*

Brookhaven National Laboratory, Upton, NY 11973, USA

⁷*Computational Science Initiative, Brookhaven National Laboratory, Upton, NY 11973, USA*

⁸*Physics Department, Brookhaven National Laboratory, Upton, NY 11973, USA*

⁹*Center for Nuclear Theory, Department of Physics and Astronomy,
Stony Brook University, Stony Brook, NY 11794-3800, USA*

¹⁰*Pacific Northwest National Laboratory, Richland, WA, 99354, USA*

¹¹*Canadian Institute for Advanced Studies, Toronto, ON, M5G 1M1, Canada*

We provide practical simulation methods for scalar field theories on a quantum computer that yield improved asymptotics as well as concrete gate estimates for the simulation and physical qubit estimates using the surface code. We achieve these improvements through two optimizations. First, we consider a different approach for estimating the elements of the S-matrix. This approach is appropriate in general for 1+1D and for certain low-energy elastic collisions in higher dimensions. Second, we implement our approach using a series of different fault-tolerant simulation algorithms for Hamiltonians formulated both in the field occupation basis and field amplitude basis. Our algorithms are based on either second-order Trotterization or qubitization. The cost of Trotterization in occupation basis scales as $\tilde{O}(\lambda N^7 |\Omega|^3 / (M^{5/2} \epsilon^{3/2}))$ where λ is the coupling strength, N is the occupation cutoff $|\Omega|$ is the volume of the spatial lattice, M is the mass of the particles and ϵ is the uncertainty in the energy calculation used for the S-matrix determination. Qubitization in the field basis scales as $\tilde{O}(|\Omega|^2 (k^2 \Lambda + k M^2) / \epsilon)$ where k is the cutoff in the field and Λ is a scaled coupling constant. We find in both cases that the bounds suggest physically meaningful simulations can be performed using on the order of 4×10^6 physical qubits and 10^{12} T-gates which corresponds to roughly one day on a superconducting quantum computer with surface code and a cycle time of 100 ns, placing simulation of scalar field theory within striking distance of the gate counts for the best available chemistry simulation results.

I. INTRODUCTION

Quantum simulation has been one of the great success stories of quantum computing [1–5]. It has led to the realization that non-relativistic quantum mechanics can, under most physically reasonable assumptions, be simulated in polynomial time on a quantum computer. Further, recent work in chemistry has shown that physically meaningful problems can be simulated using fewer than a million physical qubits and a few hours worth of compute time [6]. In contrast, there have been numerous classical methods developed to simulate quantum field theories (QFT), with the state of the art approaches summarized in [7]. However, even sign-problem free approaches relying on Hamiltonian representations [8, 9], require an exponential number amount of resources [10]. This opens up the possibility that quantum computing may provide the only means possible for us to numerically simulate some of the most challenging problems in quantum field theory [11–15]. Despite these advances, we do not yet know whether all physically meaningful quantum field theories can be efficiently simulated, nor are there at present detailed estimates of the memory and number of quantum operations required to perform quantum simulations of field theories.

The seminal work of Jordan, Lee and Preskill (JLP) [16, 17] provided a major step towards addressing the question of whether quantum computers can efficiently simulate quantum field theory algorithms that are capable of simulating

* andrew.hardy@mail.utoronto.ca

† priyanka.mukhopadhyay@utoronto.ca

‡ malam@usra.edu

§ rmk@bnl.gov

¶ eleanor.rieffel@nasa.gov

** nawiebe@cs.toronto.edu

scattering events for scalar ϕ^4 theory. In addition to describing aspects of the dynamics and couplings of the Higgs boson in the standard model, ϕ^4 theory is an excellent model field theory that captures non-trivial features of the dynamics of the other fundamental quantum fields in the standard model. Even in 1+1D, classical ϕ^4 theory has non-trivial bound states described by kinks and anti-kinks. The theory also contains stationary topological solitons that are periodic in time, known as “breather”-modes [18], providing insight into the nonlinear dynamics of domain walls in fields ranging from condensed matter to cosmology [18]. Further, at the very high energies now accessible at colliders, as exemplified in the parton model of Feynman and Bjorken [19, 20], scattering processes involving fermions or bosons become alike. Thus, the study of the ϕ^4 scalar field theory, at high energies, provides some insight into the dynamics of more phenomenologically rich theories, which are harder to directly simulate.

The JLP [16, 17] proposal focused on computing the elements of the Scattering Matrix, or S -matrix, that describes the entire information content of a scattering process in quantum mechanics and, more generally, in quantum field theory. Specifically, the S -matrix describes how free fields of a theory at asymptotic time $T = -\infty$ evolve through the scattering process into either the same or other asymptotic free states of the theory at time $T = \infty$. The S matrix thus allows one to predict scattering cross-sections, which represent the probability of a scattering event at a given impact parameter, which is then be integrated over a range of impact parameters. The computation of in-vacuum to the out-vacuum transitions embedded in the S matrix is, for 1 + 1 dimensions, promise-BQP complete. This means that if an arbitrary quantum computation can be thought of as such a scattering experiment, and if a classical computer could efficiently compute these matrix elements, then quantum computers would be no more powerful than probabilistic classical computers. In the language of complexity theory, BQP= BPP.

The computation of the entire S -matrix of a QFT is exponentially difficult, although specific matrix elements can be computed within additive error in polynomial time on a quantum computer. In [16] the authors provide a quantum algorithm for computing such matrix elements by directly performing a simulation and computing the overlaps between the outgoing distribution and the incoming one. They approximate the solution by preparing well separated wave packets, allowing them to evolve in time and scatter, and then measuring the resulting products of the scattering.

Since the original work of Jordan, Lee and Preskill, quantum simulation algorithms have undergone a mini revolution. The central problem in quantum simulation algorithms is to translate the unitary time dynamics of a closed quantum system into a sequence of quantum gates that, up to small approximation error, implements the unitary. At the time of the original JLP result, the dominant form of quantum simulation were product formulas [2, 3, 16] which approximate a complicated time-evolution operator as a product of simple time-evolution operators (which are often so simple that they can be synthesized from scratch into elementary gates. Since then a host of new methods have been developed for performing simulation; amongst these, qubitization [5, 21] has emerged as a leading non-Trotter based simulation approach. Neither of the two methods is believed to fully advantageous relative to the other. Qubitization provides provably optimal scaling in worst case scenarios but is inflexible and is not suited to take advantage of commutation relations between terms. Product (Trotter) formulas are the opposite in this regard: they take advantage of small commutators but fail to achieve optimal scaling in the worst case scenario.

Subsequent work has provided a number of optimized methods for simulating various aspects of ϕ^4 theory [11, 22]. There have also been considerable progress for various other field theories [23–25]. However, at present we still do not know whether there exist practical simulation algorithms for these methods in the sense that we do not yet know whether it is possible to perform such simulations using a reasonable number of gate operations and physical qubits on surface-code based quantum computers.

A. Our Contributions

We provide a cost-analysis of fault tolerant calculation of scattering matrix elements using fixed volume methods for scalar ϕ^4 theory in this paper that is appropriate in general for 1 + 1D and for certain low-energy elastic collisions in higher dimensions. We achieve this through two main contributions. First, we propose an alternative approach to JLP [16] by which we are able to readily extract values of the S -matrix through the measurement of multi-particle energies in finite volume. These techniques can also be applied in certain cases to elastic collisions in higher dimensions. Specifically, we discuss below (specifically in Claim 1) that direct calculation of the elements of the S -matrix is not often needed and the low-lying eigenvalues of the field Hamiltonian can be used to estimate, under specific circumstances, values of the S -matrix that would otherwise be impractical to compute directly as per [16, 17].

We implement our approach using a series of different fault tolerant simulation algorithms, which form the second major contribution of this paper. We consider primarily Hamiltonians formulated both in the field occupation basis and field amplitude basis. The occupation basis Hamiltonian is simulated with a Trotterization algorithm, while for the amplitude basis one we describe four simulation procedures - one with Trotterization, and the three others with modern qubitization methods. These qubitization approaches are optimized through our introduction of new

Algorithm	Qubit #	T gate #	Reference
Occ. Trotter	$O\left(N \Omega + \log\left(\frac{\lambda N}{M\epsilon_E}\right)\right)$	$O\left(\frac{\lambda N^7 \Omega ^3}{M^{5/2} \epsilon_E^{3/2}} \log\left(\lambda \frac{N \Omega }{M\epsilon_E}\right)\right)$	Theorem 12
Amp. Trotter	$O\left(\Omega \log(k) + \log_2\left(\frac{ \Omega \Lambda k (\Lambda k + M^2)}{\epsilon_E}\right)\right)$	$\tilde{O}\left(\frac{ \Omega ^{3/2} \sqrt{\Lambda^2 k^5 + \Lambda M^2 k^4} \log^4(k)}{\epsilon_E^{3/2}}\right)$	Theorem 21, Lemma 37
Amp. Qubitized I	$O\left(\Omega \log k + \log^2 k + \log\left(\frac{ \Omega [k^2 \Lambda + k M^2]}{\epsilon_E}\right)\right)$	$O\left(\frac{ \Omega ^2}{\epsilon_E} [k^2 \Lambda + k M^2] \log^2 k\right)$	Theorem 34
Amp. Qubitized IIIa	$O\left(\Omega \log k + \log\left(\frac{ \Omega [k^2 \Lambda + k M^2]}{\epsilon_E}\right)\right)$	$O\left(\frac{ \Omega ^2}{\epsilon_E} [k^2 \Lambda + k M^2] [\log^4 k]\right)$	Theorem 34
Amp. Qubitized IIIb	$O\left(\Omega \log k + \log\left(\frac{ \Omega [k^2 \Lambda + k M^2]}{\epsilon_E}\right)\right)$	$O\left(\frac{ \Omega ^2}{\epsilon_E} [k^2 \Lambda + k M^2] [\log^2 k]\right)^*$	Proposition 35

TABLE I: Qubit counts and T gate cost scalings for computing ϕ^4 scattering matrix elements as a function of the lattice size $|\Omega|$, the (rescaled) coupling constant Λ and mass M , which hide dependence on the lattice size, the amplitude cutoff k in the field amplitude basis in units of the bin size, the momentum cutoff N in the occupation basis, and the target precision in the energy estimate ϵ_E , which dictates the precision of the S-matrix through Eq. (33). The \tilde{O} notation hides further multiplicative factors at most logarithmic in all variables. (*) For Algorithm IIIb the estimate on the T-count has been obtained assuming Conjecture 28.

unitary block encodings (implemented through LCU or linear combinations of unitaries approaches [3]) of ϕ , ϕ^2 and ϕ^4 operators and in order to reduce the resource cost. Further, the implemented circuits have been optimized with recent optimization techniques [26, 27]. The Trotterization algorithm in the occupation basis performs optimally in the non-interacting limit, but is outperformed by qubitization methods (in the amplitude basis) with increasing interaction strength. However, the main appeal of the occupation basis algorithm is that the basis provides a natural extension for state preparation and measurement of direct scattering calculations in higher dimensions. In Table I we have compared the cost of implementing the various algorithms in terms of number of qubits and T-gates used. Here, we mention that we have implemented our algorithms with the Clifford+T gate set, which is the most popular, fault-tolerant, universal gate set considered for quantum simulation algorithms [15, 28–30] and affords a wide variety of exact and approximate unitary synthesis methods [31–36]. At times, it can exactly synthesize useful gates that could at best be approximately synthesized in other gate libraries [37–39]. The resource overhead in fault-tolerant implementation of the non-Clifford T gate is the highest in most error-correction schemes, including the surface code. A bound on the T-count can be used as a marker to reflect the complexity of fault-tolerant implementation of a quantum algorithm [40].

We then consider implementing the Trotter and qubitization based simulation algorithms on top of the surface code and consider the overheads of magic state distillation in the simulation algorithm. The space-time volume needed to implement a T gate is expected to dwarf the costs of all other gates by factors of 100 or greater. Our calculations for surface code costs are based on canonical approaches [40]. This facilitates the translation for others towards any preferred modern methods, (including those that may exist in future) in the ever-expanding forefront of quantum error correction [41, 42].

The organization of paper is as follows. In Section II we provide a brief review of the scalar ϕ^4 QFT. We specifically provide derivations of the Hamiltonian for ϕ^4 theory in both amplitude and occupation bases. We then discuss how phase estimation can be used to estimate the scattering amplitude in Section III. We use these techniques in Section IV to provide quantum algorithms for estimating the elements of the S-matrix (under the assumption of elastic collisions) in both the amplitude and the occupation basis. This section gives not only the asymptotic scalings required but also the constant factor analysis needed to estimate the energy within fixed error. Section V contains our analysis of the implementation of the quantum algorithms using the surface code along with the space-time needs for the simulation. We finally conclude in Section VI and discuss future applications.

II. REVIEW OF ϕ^4 SCALAR FIELD THEORY

In this section we introduce notation for the ϕ^4 scalar field theory. In order to simulate scattering processes in the theory, we need a discrete Hamiltonian representation of the continuum Lagrangian. We provide a concise definition and derivation of the Hamiltonian in both a field and occupation basis for the field theory. The regimes and limits where each basis may be ideal are discussed in more detail below.

We begin by considering the free-theory for a scalar bosonic field, denoted by $\phi(x)$. The first term in our Lagrangian

density is the free Klein-Gordon Lagrangian given by

$$\mathcal{L}_0 = \frac{1}{2} (\partial_\mu \phi)^2 - \frac{1}{2} m^2 \phi^2 = \frac{1}{2} \dot{\phi}^2 - \frac{1}{2} (\nabla \phi)^2 - \frac{1}{2} m^2 \phi^2 \quad (1)$$

where m is the bare mass of the scalar boson field. The corresponding conjugate field is given by the Euler-Lagrange equations to state

$$\pi(\mathbf{x}) \equiv \frac{\partial \mathcal{L}_0}{\partial \dot{\phi}(\mathbf{x})} \quad (2)$$

The simplest interaction to consider for this Lagrangian would be a quartic interaction term given by

$$\mathcal{L}_{int} = \lambda \phi^4 \quad (3)$$

The corresponding Hamiltonian for the theory is given as

$$H = \int d^d x \mathcal{H} = \int d^d x \left[\frac{1}{2} \pi(\mathbf{x})^2 + \frac{1}{2} (\nabla \phi(\mathbf{x}))^2 + \frac{1}{2} m^2 \phi(\mathbf{x})^2 + \lambda \phi(\mathbf{x})^4 \right] \quad (4)$$

Both formalisms considered here would also extend to interaction terms of arbitrary order ϕ^α . These higher-order terms appear in general phenomenological Landau theories of phase transitions in non-relativistic condensed matter and have rich applications in nonlinear dynamics [18]. In this work however we focus on the ϕ^4 term more in the context of the high-energy community. This interaction is the simplest renormalizable term that is also bounded from below.

We consider a theory of scalar ϕ^4 inside a d -dimensional space confined to the d -cube of side length L , considering the momentum cut-off = $\Lambda(\epsilon)$, with $^d \sqrt{|\Omega|} = P$ modes in each dimension. For simplicity, we will refer to this volume region as Ω and we assume that this region is discretized into $|\Omega| = (L/a)^d$ points with a spacing between each of the points of a .

A. Field Amplitude Basis

We begin by considering our Hilbert space for the theory to be of the form

$$\mathcal{H} = \bigotimes_{\mathbf{x} \in \Omega} \mathcal{H}_\phi, \quad (5)$$

where \mathcal{H}_ϕ is the Hilbert space that describes a field at each of the lattice sites. The discretized lattice form of the Hamiltonian is

$$H = \sum_{\mathbf{x} \in \Omega} a^d \left[\frac{1}{2} \pi(\mathbf{x})^2 + \frac{1}{2} (\nabla_a \phi)^2(\mathbf{x}) + \frac{1}{2} m^2 \phi(\mathbf{x})^2 + \frac{\lambda}{4!} \phi(\mathbf{x})^4 \right] \quad (6)$$

where Ω is the set of all (spatial) lattice sites, $\mathbf{x} = a\mathbf{n}$, where a is the lattice spacing and $\mathbf{n} \in \mathbb{Z}^{\times d}$, and d is the number of spatial dimensions.

We then consider the conjugate momentum operator

$$\pi(x) = \mathcal{F}^\dagger \phi(x) \mathcal{F}, \quad (7)$$

where \mathcal{F} is the discrete quantum Fourier transform acting on our truncated space. Starting with Eq. (6), we identify

$$\begin{aligned} (\nabla \phi(\mathbf{x}))_i &\rightarrow \Delta_i \phi(\mathbf{x}) \equiv \frac{\phi(\mathbf{x} + a\hat{x}_i) - \phi(\mathbf{x})}{a} \\ \Rightarrow \sum_{i=1}^d (\Delta_i \phi(\mathbf{x}))^2 &= \sum_{i=1}^d \frac{\phi^2(\mathbf{x} + a\hat{x}_i) + \phi^2(\mathbf{x}) - 2\phi(\mathbf{x})\phi(\mathbf{x} + a\hat{x}_i)}{a^2} \end{aligned} \quad (8)$$

If we assume periodic boundary conditions, then at each lattice site \mathbf{x} , there is one contribution of $\phi(\mathbf{x})^2$ from each of the d lattice sites at $\mathbf{x}^{(i)} = \mathbf{x} - a\hat{x}_i$ (without periodic boundary conditions, we would need to be careful about

handling the sites at the edges), and a single additional contribution coming from the $\phi(\mathbf{x})^2$ term. Therefore, the Hamiltonian becomes

$$H = \sum_{\mathbf{x} \in \Omega} \left[\frac{1}{2} a^d \pi^2(\mathbf{x}) + \frac{1}{2} (d+1 + a^2 m^2) a^{d-2} \phi^2(\mathbf{x}) + \frac{\lambda}{4!} a^d \phi^4(\mathbf{x}) - a^{d-2} \sum_{i=1}^d \phi(\mathbf{x}) \phi(\mathbf{x} + a \hat{x}_i) \right] \quad (9)$$

Let us now define

$$\begin{aligned} \Phi &:= a^{\frac{d}{2}-1} \phi, & \Pi &:= a^{\frac{d}{2}} \pi, \\ M &:= am, & \Lambda &:= a^{4-d} \lambda \end{aligned} \quad (10)$$

Then, in terms of the scaled variables, our Hamiltonian simplifies to

$$H_{amp} = \sum_{\mathbf{x} \in \Omega} \left[\frac{1}{2} \Pi^2(\mathbf{x}) + \frac{1}{2} (M^2 + d + 1) \Phi^2(\mathbf{x}) + \frac{\Lambda}{4!} \Phi^4(\mathbf{x}) - \sum_{i=1}^d \Phi(\mathbf{x}) \Phi(\mathbf{x} + a \hat{x}_i) \right] \quad (11)$$

Thus, we see that the derivative term is effectively replaced by a nearest neighbor interaction. We shall make use of this expression, which we denote by H_{amp} in both the Qubitization-based and Trotterization based implementation in the field amplitude basis (Section IV E).

The field amplitude basis is defined to be that in which the field operator is diagonal

$$\Phi |\phi\rangle = \phi |\phi\rangle \quad (12)$$

and, in the continuum, is related to the Fock basis via $|\phi\rangle = \sum_{n=0}^{\infty} |n\rangle \langle n|\phi\rangle$, where

$$\langle \phi|n\rangle = \left(\frac{M}{\pi} \right)^{1/4} \frac{1}{\sqrt{2^n n!}} e^{-\frac{M\phi^2}{2}} H_n(\sqrt{M}\phi) \quad (13)$$

where $H_n(x)$ is the (physicists') Hermite polynomial of order n . This basis has the advantage that all terms in the ϕ^4 Hamiltonian, except for the momentum term, are diagonal. As such, the basis should operate better in large λ coupling limits. The momentum operator, in turn, is related to the field operator through a Fourier transform. We discuss Trotter and qubitization approaches to simulating the ϕ^4 Hamiltonian in this basis in Section IV E.

B. Field Occupation Basis

We note that the decomposition of the Hilbert in Eq. (5) is not the only way to latticize the theory. In fact, despite its clear appeal when studying low energy properties, such a discretization might not be ideal at high energies. As we mentioned previously, an emergent particle description appears (“partons”) in high energy experiments, and therefore in this context it may be more appropriate to discretize the theory in this parton basis. Such a description starts then by dividing the Hilbert space into single particle sectors

$$\mathcal{H} = \bigotimes_l \mathcal{H}_l, \quad (14)$$

where \mathcal{H}_l describes the Hilbert of a single particle, which is span by the local vacuum and all the possible momentum (or position) modes that can be occupied. One then evolves the theory by freely propagating the particles with the quadratic part of the Hamiltonian in each particle sector, while the interactions connect different l 's. Such a procedure has been applied to several problems in high energy physics, see for example [22, 43–45].

In the same spirit, the typical approach to solving the ϕ^4 theory begins with diagonalizing the non-interacting Klein-Gordon Hamiltonian. This is done by introducing bosonic ladder operators $a_{\mathbf{p}}^\dagger, a_{\mathbf{p}}$ which correspond to the creation and annihilation of quanta inside the scalar field. They obey the canonical bosonic commutation relation

$$[a_{\mathbf{p}}, a_{\mathbf{q}}^\dagger] = (2\pi)^D \delta(p - q) \quad (15)$$

where D is the spatial dimensionality. However these operators on their own are not Hermitian. As such, we construct Hermitian operators (and their unitary exponentiation) through considering them in conjunction with their Hermitian conjugates. As such, we begin with their implementation.

To carry out our analysis in this basis, we require the mode expansion of the scalar field and its conjugate momentum

$$\begin{aligned}\phi(\mathbf{x}) &= \int \frac{d^D p}{(2\pi)^D} \frac{e^{i\mathbf{p}\cdot\mathbf{x}}}{\sqrt{2\omega_{\mathbf{p}}}} \left(a_{\mathbf{p}} + a_{-\mathbf{p}}^\dagger \right) \\ \pi(\mathbf{x}) &= \int \frac{d^D p}{(2\pi)^D} (-i) e^{i\mathbf{p}\cdot\mathbf{x}} \sqrt{\frac{\omega_{\mathbf{p}}}{2}} \left(a_{\mathbf{p}} - a_{-\mathbf{p}}^\dagger \right)\end{aligned}\quad (16)$$

where $\omega_{\mathbf{p}} = \sqrt{|\mathbf{p}|^2 + M^2}$, and D denotes the spatial dimensionality. The non-interacting part of the scalar field Hamiltonian, or the Klein-Gordon Hamiltonian H_0 , is then given by

$$\begin{aligned}H_0 &= \int d^D x \frac{1}{2} [(\pi(\mathbf{x}) \cdot \pi(\mathbf{x})) + (\nabla\phi(\mathbf{x}) \cdot \nabla\phi(\mathbf{x})) + M^2 (\phi(\mathbf{x}) \cdot \phi(\mathbf{x}))] \\ &= \int d^D x \int \frac{d^D \mathbf{p} d^D \mathbf{p}'}{(2\pi)^D (2\pi)^D} \left\{ -\frac{\sqrt{\omega_{\mathbf{p}}\omega_{\mathbf{p}'}}}{4} \left(a_{\mathbf{p}} - a_{-\mathbf{p}}^\dagger \right) \left(a_{\mathbf{p}'} - a_{-\mathbf{p}'}^\dagger \right) \right. \\ &\quad \left. + \frac{m^2 - \mathbf{p} \cdot \mathbf{p}'}{4\sqrt{\omega_{\mathbf{p}}\omega_{\mathbf{p}'}}} \left(a_{\mathbf{p}} + a_{-\mathbf{p}}^\dagger \right) \left(a_{\mathbf{p}'} + a_{-\mathbf{p}'}^\dagger \right) \right\} \\ &= \int \frac{d^D p}{(2\pi)^D} \omega_{\mathbf{p}} \left(a_{\mathbf{p}}^\dagger a_{\mathbf{p}} + \frac{1}{2} [a_{\mathbf{p}}, a_{\mathbf{p}}^\dagger] \right)\end{aligned}\quad (17)$$

The second term in the parentheses above is proportional to $\delta(0)$, and represents the zero point energy of all the harmonic oscillator modes in the free scalar field. This formally leads to an infinite contribution to the Hamiltonian, which must be removed to compute physical quantities. The resulting free Hamiltonian we consider is

$$H_0 = \int \frac{d^D p}{(2\pi)^D} \omega_{\mathbf{p}} a_{\mathbf{p}}^\dagger a_{\mathbf{p}} \quad (18)$$

This Hamiltonian is now a normal ordered product of ladder operators, where all creation operators appear on the left of all annihilation operators. This reordering is necessary to avoid spurious divergences and is denoted $H_0 = :H_0:$.

This form has two primary appeals. One is that in the limit of small λ , our approach approaches a trivial diagonal matrix. This limit up to arbitrary precision remains a challenging computational task. The primary appeal of this approach is that for explicit scattering process that require single-particle packet preparation, this protocol also has much simpler state preparation procedure. Next, the interaction part of the Hamiltonian is given by

$$\begin{aligned}H_\lambda &= \int d^D x \frac{\lambda}{4!} \phi^4 \\ &= \frac{\lambda}{4!} d^D x \int \frac{d^D \mathbf{p}_1 d^D \mathbf{p}_2 d^D \mathbf{p}_3 d^D \mathbf{p}_4}{(2\pi)^{4D} 4\sqrt{\omega_{\mathbf{p}_1}\omega_{\mathbf{p}_2}\omega_{\mathbf{p}_3}\omega_{\mathbf{p}_4}}} \left(a_{\mathbf{p}_1} e^{+i\mathbf{p}_1\cdot\mathbf{x}} + a_{\mathbf{p}_1}^\dagger e^{-i\mathbf{p}_1\cdot\mathbf{x}} \right) \left(a_{\mathbf{p}_2} e^{+i\mathbf{p}_2\cdot\mathbf{x}} + a_{\mathbf{p}_2}^\dagger e^{-i\mathbf{p}_2\cdot\mathbf{x}} \right) \\ &\quad \times \left(a_{\mathbf{p}_3} e^{+i\mathbf{p}_3\cdot\mathbf{x}} + a_{\mathbf{p}_3}^\dagger e^{-i\mathbf{p}_3\cdot\mathbf{x}} \right) \left(a_{\mathbf{p}_4} e^{+i\mathbf{p}_4\cdot\mathbf{x}} + a_{\mathbf{p}_4}^\dagger e^{-i\mathbf{p}_4\cdot\mathbf{x}} \right) \\ &= \int \frac{\lambda}{4!} d^D x \int \frac{d^D \mathbf{p}_1 d^D \mathbf{p}_2 d^D \mathbf{p}_3 d^D \mathbf{p}_4}{(2\pi)^{4D}} \frac{e^{i(\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3+\mathbf{p}_4)\cdot\mathbf{x}}}{4\sqrt{\omega_{\mathbf{p}_1}\omega_{\mathbf{p}_2}\omega_{\mathbf{p}_3}\omega_{\mathbf{p}_4}}} \\ &\quad \left(a_{\mathbf{p}_1} + a_{-\mathbf{p}_1}^\dagger \right) \left(a_{\mathbf{p}_2} + a_{-\mathbf{p}_2}^\dagger \right) \left(a_{\mathbf{p}_3} + a_{-\mathbf{p}_3}^\dagger \right) \left(a_{\mathbf{p}_4} + a_{-\mathbf{p}_4}^\dagger \right) \\ &= \frac{\lambda}{4!} \int \frac{d^D \mathbf{p}_1 d^D \mathbf{p}_2 d^D \mathbf{p}_3}{(2\pi)^{3D}} \frac{1}{4\sqrt{\omega_{\mathbf{p}_1}\omega_{\mathbf{p}_2}\omega_{\mathbf{p}_3}\omega_{-(\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3)}}} \left\{ \left(a_{\mathbf{p}_1} + a_{-\mathbf{p}_1}^\dagger \right) \left(a_{\mathbf{p}_2} + a_{-\mathbf{p}_2}^\dagger \right) \right. \\ &\quad \left. \times \left(a_{\mathbf{p}_3} + a_{-\mathbf{p}_3}^\dagger \right) \left(a_{-(\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3)} + a_{(\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3)}^\dagger \right) \right\}\end{aligned}$$

For the algorithm, we consider the normal ordered form $:H_\phi:$, discussed in B2

We consider the scalar field on a finite sized, discrete lattice, borrowing the notation from above which requires us to replace the continuous integrals with discrete Riemann sums as

$$\int_0^{2\pi/a} \frac{d^D p}{2\pi} \rightarrow \sum_{\alpha=1}^{|\Omega|} \frac{(2\pi)}{(2\pi)|\Omega|(a)^D} = \frac{1}{(a)^D} \sum_{\alpha=1}^{|\Omega|} \frac{1}{|\Omega|} \quad (19)$$

Now the Hamiltonians reads as follows

$$: H_0 := \frac{1}{|\Omega|} \sum_{\mathbf{p}} \omega_{\mathbf{p}} a_{\mathbf{p}}^\dagger a_{\mathbf{p}} \quad (20)$$

We note that $(a)^D |\Omega| = L$, but we retain this notation for if $|\Omega|$ increases with fixed L , i.e. taking the $\alpha \rightarrow 0$ continuum limit this notation makes the scaling explicit. Furthermore, we take proper discretization of our operators $a_{\mathbf{p}} \rightarrow \sqrt{a} a_{\mathbf{p}}$ so that these are dimensionless. [46]. An advantage of the momentum space-representation is that it allows for the retention of the continuum dispersion as prepared to in position space [22].

The λ term is equivalently given by

$$H_\lambda = \frac{\lambda}{4! |\Omega|^3} \sum_{\{\mathbf{p}_i\}} \frac{1}{4 \sqrt{\omega_{\mathbf{p}_1} \omega_{\mathbf{p}_2} \omega_{\mathbf{p}_3} \omega_{-(\mathbf{p}_1 + \mathbf{p}_2 + \mathbf{p}_3)}}} \left\{ \left(a_{\mathbf{p}_1} + a_{-\mathbf{p}_1}^\dagger \right) \left(a_{\mathbf{p}_2} + a_{-\mathbf{p}_2}^\dagger \right) \left(a_{-\mathbf{p}_3} + a_{\mathbf{p}_3}^\dagger \right) \left(a_{(\mathbf{p}_1 + \mathbf{p}_2 + \mathbf{p}_3)} + a_{-(\mathbf{p}_1 + \mathbf{p}_2 + \mathbf{p}_3)}^\dagger \right) \right\} \quad (21)$$

We leave the normal ordering until we consider the cases of momentum \mathbf{p}_i overlaps.

As a final note, due to the relativistic dispersion relationship the frequency $\omega_{\mathbf{p}}$ depends on the momentum as well as the mass via $\omega_{\mathbf{p}} = \sqrt{M^2 + |\mathbf{p}|^2}$. For this reason the frequencies $\omega_{\mathbf{p}}$ subtly contain the mass. If we are interested in the low momentum regime then we can take the maximum and minimum relevant frequencies to be $\omega_{min} = M$ and $\omega_{max} = \sqrt{M^2 + P_{max}^2} \sim M$ which gives us the required mass dependence of the Hamiltonian.

III. SCATTERING MATRIX ESTIMATION

The cost of any end-to-end quantum simulation depends on the observables that we wish to measure in addition to the processing required to prepare the initial state and simulate the dynamics of the quantum system. Unlike previous work, which directly examines scattering in field theories [16, 17], energy estimation is computationally simple and opens up the possibility of practical simulations of quantum dynamics.

We will now go into detail about our alternative approach to that of [16, 17], for the computation of values of the scattering matrix or S -matrix. The scattering matrix describes the relationship between the incoming and outgoing momenta for a quantum system. In quantum field theory in particular, it is useful to define the S -matrix in terms of its dynamics in an interaction frame. In particular, if we define the Hamiltonian for a field to be

$$H = H_{\text{free}} + H_{\text{int}} \quad (22)$$

where H_{free} and H_{int} are the free and interaction terms where H_{int} vanishes at infinity. In order to unambiguously define the dynamical phase of the non-interacting system as $T \rightarrow \infty$ one can go into an interaction frame of H_{free} to obtain the interaction picture Hamiltonian

$$H_I(t) = e^{iH_{\text{free}}t} H_{\text{int}} e^{-iH_{\text{free}}t}. \quad (23)$$

With this definition in mind we can express the S -matrix through its matrix elements as

$$\begin{aligned} S_{\vec{p}, \vec{k}} = \text{out} \langle p_1 p_2 \cdots | k_{\mathcal{A}} k_{\mathcal{B}} \rangle_{\text{in}} &= \lim_{T \rightarrow \infty} \underbrace{\langle p_1 p_2 \cdots |}_T \underbrace{| k_{\mathcal{A}} k_{\mathcal{B}} \rangle}_{-T} \\ &= \lim_{T \rightarrow \infty} \left\langle p_1 p_2 \cdots \left| \mathcal{T} e^{-i \int_{-T}^T dt H_I(t)} \right| k_{\mathcal{A}} k_{\mathcal{B}} \right\rangle. \end{aligned} \quad (24)$$

More specifically, by isolating the interaction terms $S = 1 + iT$, you can identify

$$\begin{aligned} &\langle p_1 \cdots p_n | iT | p_{\mathcal{A}} p_{\mathcal{B}} \rangle \\ &= \lim_{T \rightarrow \infty (1-i\epsilon)} \left(0 \left\langle p_1 \cdots p_n \left| T \left(\exp \left[-i \int_{-T}^T dt H_I(t) \right] \right) \right| p_{\mathcal{A}} p_{\mathcal{B}} \right\rangle_0 \right) \end{aligned} \quad (25)$$

The most important element of T can be given through the matrix element (\mathcal{M}), which can be thought of as the scattering amplitude and includes the important dynamics.

$$\langle p_1 p_2 \cdots | iT | k_{\mathcal{A}} k_{\mathcal{B}} \rangle = (2\pi)^4 \delta^{(4)} \left(k_{\mathcal{A}} + k_{\mathcal{B}} - \sum p_f \right) \cdot i\mathcal{M}(k_{\mathcal{A}}, k_{\mathcal{B}} \rightarrow p_f) \quad (26)$$

This \mathcal{M} will give all the necessary calculations for the cross-section if necessary. For reference: To one-loop order, we can compute \mathcal{M} in φ^4 as

$${}_0 \left\langle p_1 p_2 \left| T \left(-i \frac{\lambda}{4!} \int d^4 x \varphi_I^4(x) \right) \right| p_A p_B \right\rangle_0 = -i \lambda (2\pi)^4 \delta^{(4)}(p_A + p_B - p_1 - p_2) \quad (27)$$

which identifies $\mathcal{M} = -\lambda$.

Computing the entire S -matrix is exponentially difficult since the number of output momenta and input momenta scales exponentially with the number of particles permitted. But we can readily extract at least the elastic parts of the S -matrix through the measurement of multi-particle energies in finite volume. We will also comment on the ability to obtain inelastic information. Here we will argue that the measurement of energies in finite volume leads directly to information on the S -matrix elements. The simplest case where this occurs is for a non-interacting $1+1D$ theory and where both particles have mass m . In this case, we can go into a center of mass frame where without loss of generality both particles have momenta p and $-p$ such that $p_1 = -p_2$ and by conservation of momentum the output momenta also must be in $\{-p, p\}$. In finite volume, the momenta, $p_i, i = 1, 2$ of the particles are

$$p_i = \frac{2\pi n_i}{L}, \quad n_i \in \mathbb{Z}, \quad (28)$$

where n_i are integers indexing the two particles and L is the length of the system. The energy, E , of the two particle state is

$$E = \sum_{i=1,2} \left(m^2 + \frac{4\pi^2 n_i^2}{L^2} \right)^{1/2}, \quad (29)$$

where we have taken $c = \hbar = 1$. Now what happens when interactions are turned on? Provided the energy of the two-particle state is below threshold for particle production, the quantization condition for the two momenta is altered to

$$\begin{aligned} e^{ip_1 L} S(p_1, p_2) &= 1 = e^{2\pi n_1/L}, \\ e^{ip_2 L} S(p_2, p_1) &= 1 = e^{2\pi n_2/L}, \end{aligned} \quad (30)$$

where $S(p_1, p_2) = e^{i\delta(p_1, p_2)}$ is the two-body scattering phase, which is a phase because the scattering is elastic in $1+1$ dimensions.

By taking logarithms, the quantization conditions can be written in the form

$$\begin{aligned} \frac{2\pi n_1}{L} &= p_1 - \frac{i}{L} \log S(p_1, p_2); \\ \frac{2\pi n_2}{L} &= p_2 - \frac{i}{L} \log S(p_2, p_1). \end{aligned} \quad (31)$$

If one solves the quantization condition for p_1 and p_2 one immediately knows the energy of the two-particle state via

$$E = \sum_{i=1,2} (m^2 + p_i^2)^{1/2}, \quad (32)$$

where p_i necessarily deviate from their free values. Now if instead we start with knowledge of E via measurement, we can reverse the process and infer $S(p_1, p_2)$. And by measuring E for different L , we can determine $S(p_1, p_2)$ over a range of p_1 and p_2 because while p_i are not free, they still will behave as $1/L$ over a wide range of volumes.

In any determination of the energy, E , of a two particle state en route to the determination of an S -matrix element, there will be some uncertainty associated with its determination, δE . If we work in the center-of-mass frame and the particles have equal and opposite momentum, i.e., $p_1 = -p_2$, the associated uncertainty in the scattering phase $\delta(p, -p)$ is given by

$$\delta\delta(p, -p) = -\frac{LE}{8p} \delta E \quad (33)$$

We can see that uncertainty in the determination of the energy affects most dramatically the determination of the scattering phase at small momentum.

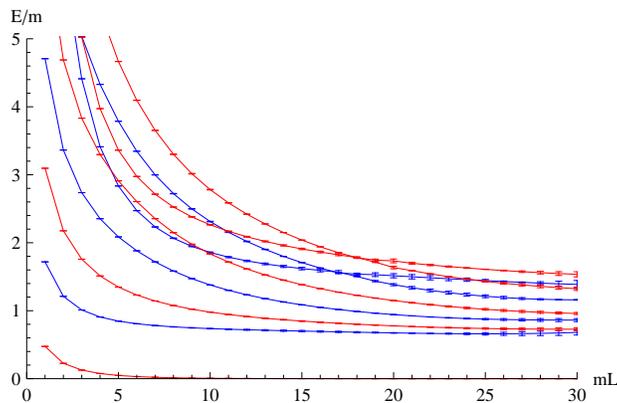


FIG. 1: The low lying spectrum of the φ^4 in its broken phase ($m^2 = -0.25$, $\lambda = 0.15$). The ground state energy has been subtracted. From Fig. 10 of [47].

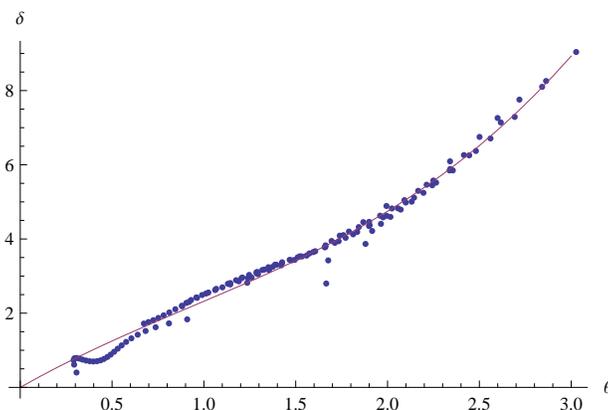


FIG. 2: The scattering phase inferred from the energies of the two kink states presented in Fig. 1. From Fig. 11 of [47].

As an illustration of this in the φ^4 theory, we will consider an example from the ordered phase of the model (i.e., $m^2 < 0$). The spectrum of the theory in this regime consists of kinks and bound states of kinks (meson-like states). As λ is increased from 0, the bound states disappear at a λ_0 from the spectrum and only kink-like states exist. At a large enough $\lambda = \lambda_c$, the model becomes disordered. In our example, we will consider the regime $\lambda_0 < \lambda < \lambda_c$.

In Fig. 1, we plot the low-lying energy levels with zero total momenta as a function of system size, L , as computed in Ref.[47] using Hamiltonian truncation methods. The ground state energy has been subtracted. The lowest lying level is the state whose energy is nearly degenerate to the ground state (we expect such a near-degeneracy in finite volume in the broken phase). Beyond this state are energies corresponding to two-kink and four-kink states. We can use the two-kink energies to back out the scattering phase as described above. This phase is plotted in Fig. 2 as a function of energy, E . Here the energy of the two-particle state is parameterized by θ via :

$$E = 2M_{kink} \cosh(\theta)$$

In this parameterization the momenta of the two kinks are $\pm M_{kink} \sinh(\theta)$.

What we have discussed so far concerns the elastic part of the $2-2$ scattering matrix. We can also access inelastic information from the measurement of energies. We can parameterize the S-matrix $S(p_1, p_2)$ solely in terms of θ (θ is related to the Mandelstam variable, s , via $s = 4m^2 \cosh^2(\theta/2)$). The unitarity condition on the S-matrix then reads

$$S(\theta + i\epsilon)S(-\theta + i\epsilon) = f(\theta), \quad (34)$$

where $f(\theta)$ is real positive on the real line. If θ_0 marks the threshold beyond which inelastic processes are possible, $f(|\theta| < \theta_0) = 1$, while $0 < f(|\theta| > \theta_0) < 1$. Using the analytic structure of the S-matrix in the complex- θ plane, we

can write $S(\theta)$ in the following form:

$$S(\theta) = \pm \prod_j \frac{\sinh(\theta) + i \sin(\beta_j)}{\sinh(\theta) - i \sin(\beta_j)} \exp \left[- \int_{-\infty}^{\infty} \frac{d\theta'}{2\pi i} \frac{\log f(\theta')}{\sinh(\theta - \theta' + i\epsilon)} \right]. \quad (35)$$

The first part of the parameterization involving the angles β_j are so-called Castillejo-Dalitz-Dyson (CDD) factors. The second part of the parameterization contains information about the inelastic part of the 2-2 scattering inasmuch as it depends on the region in θ where $f(\theta)$ differs from 1. Notice that because of analyticity, the inelastic part of the S-matrix influences the elastic region $|\theta| < \theta_0$. With sufficiently accurate measurements of the energies, one can use this parameterization to determine the number and values of the CDD angles, β_j , as well as the function $f(\theta)$. In practice, extracting the function $f(\theta)$ is difficult as it involves the inversion of an ill-posed integral equation. However the integrated quantity,

$$\int_{-\infty}^{\infty} \frac{d\theta'}{2\pi i} \frac{\log f(\theta')}{\sinh(\theta - \theta' + i\epsilon)}, \quad (36)$$

has been successfully extracted from numerical data, for example, in the case of the non-integrable Ising field theory [48].

While we have focused our discussion here on scattering in $1 + 1d$, our approach can be extended to scattering in higher dimensions, i.e., the elastic scattering phases of 2-2 particle scattering, δ_l , in different angular momentum channels can be connected to measurements of the two-particle energy in finite volume [49]. While originally formulated in [49] for two spinless identical particles, extensions have been made to identical particles with spin, asymmetric volumes, and amplitudes containing external current - see [50] for a review. In certain cases, inelastic information involving 3 particle scattering is available in higher dimensions [51–54].

Note that the last approach differs from that taken by Jordan Lee and Preskill [16, 17]. Their approach involved preparing a Gaussian wave packet in the interacting eigenbasis rather than an eigenvector of the interacting Hamiltonian via phase estimation. Their approach has the advantage that it can be applied in circumstances where the gap is large and no prior knowledge of the eigenvectors is given. Directly preparing the eigenstates as we discuss above can be more computationally efficient than the approach given by JLP, but requires efficient approximations to the eigenstates which may not be available in all settings.

To summarize, we see from this discussion that the excited state energy is a meaningful quantity for scattering theory that can be used to estimate, under specific circumstances, values of the S -matrix that would otherwise be impractical to compute. The circumstances under which the energy can be used to provide information about the scattering is given in the following claim with increasing levels of generalization:

Claim 1. *Let \tilde{S} be a finite dimensional version of the S -matrix for a Hamiltonian operator $H(t)$. The value of the excited state energies can be used to infer certain elements of \tilde{S} if one of the following occurs*

1. *We are interested in $2 \mapsto 2$ elastic scattering of particles in one spatial and one temporal dimension ($1 + 1D$) where the two-particle final state is below the threshold for particle production.*
2. *We are interested in inelastic processes in $1 + 1D$ in integrated form (as encoded in Eq.(35)).*
3. *We are interested in computing the scattering phases of elastic $2 \mapsto 2$ scattering in higher dimensions [49].*
4. *We are interested in certain relatively simple scattering processes that include particle decay in dimensions higher than $1 + 1D$. This is focused primarily on scattering involving 3-particles, an active topic of current research [51–54].*

Barring the cases mentioned in the above claim, there are other reasons why the energies of both the ground state and low lying excited states is independently interesting for the theory. It can allow us to understand phase transitions in the strongly interacting theory [55, 56]. For all the above reasons, we focus our attention on the computation of the energy of the low lying excited states which is also a well studied quantity to estimate using quantum computers [28, 30].

IV. SIMULATION ALGORITHMS

Modern simulation algorithms have converged in recent years to the point that there is no optimal single quantum simulation algorithm. Rather, different algorithms tend to have advantages and disadvantages in different regimes [15,

57, 58]. In this spirit, we provide five simulation algorithms for scalar field theory in the occupation as well as field amplitude basis. These algorithms employ Trotter decompositions or qubitization and use optimized circuits to construct the operator exponentials or oracle calls that both methods require. We further observe that the methods, as expected, have advantages and disadvantages with respect to each other. Most obviously, the Trotter algorithm for simulations in the occupation basis is by far the most efficient for weak coupling ($\lambda \ll 1$) but the qubitized field amplitude basis is likely to scale better in the strong coupling regime.

The material in this section is laid out as follows. In Section IV A we describe an algorithm for the Hamiltonian formulated in the field occupation basis, while in Section IV E we describe several algorithms for the Hamiltonian in the field amplitude basis. The techniques and concepts in these two sections can be followed independently of the other.

A. Field Occupation Basis

The interaction Hamiltonian can be decomposed into four cases based on matching momentum indices. We consider the normal ordered Hamiltonian so that the unitary phases corresponding to eigenvalues that are used to construct the S -matrix are computed with reference to the vacuum energy. First we state some essential results that are required to map the Hamiltonian from the bosonic to qubit space. Then we synthesize quantum circuits implementing the exponentiated Hamiltonian.

We discretize the Hilbert space of occupation into N distinct momentum states. We have a register of $(N + 1)V$ qubits to store information about the occupation states and we use the following one-hot unary encoding to map an occupation state to a qubit state. We index the qubits by a pair of integers, such as (p, n) , where p corresponds to a momentum mode and n to a momentum state. For each such pair (p, n) , we have a quantum state on $(N + 1)V$ qubits, in which each qubit is $|0\rangle$, except the $(p, n)^{\text{th}}$ one, which is $|1\rangle$. We denote this state by $|p, n\rangle$, which is

$$\begin{aligned} |p, n\rangle &= |0_{1,0}, \dots, 0_{p-1,N}; 0_{p,0}, \dots, 0_{p,n-1}, 1_{p,n}, 0_{p,n+1}, \dots, 0_{p,N}; 0_{p+1,0}, \dots, 0_{V,N}\rangle \\ &= \left(\bigotimes_{j=1}^{p-1} |0_{j,0}, \dots, 0_{j,N}\rangle \right) \bigotimes |0_{p,0}, \dots, 1_{p,n}, \dots, 0_{p,N}\rangle \bigotimes \left(\bigotimes_{j=p+1}^{|\Omega|} |0_{j,0}, \dots, 0_{j,N}\rangle \right). \end{aligned} \quad (37)$$

We emphasize that each Hilbert subspace $\mathcal{H}_{\mathbf{p}}$ is spanned only by vectors of Hamming weight 1 in order to ensure the unary encoding. We now consider a construction of the Hamiltonian that preserves this Hamming weight without restricting to total number conservation or conservation for each p -mode.

For convenience, we denote an operator A_{pn} acting on pn^{th} qubit by $A_{p,n}$ or $(A_n)_{\mathbf{p}}$. The qubit mapping for the creation and annihilation operators is as follows.

$$\begin{aligned} a_{\mathbf{p}}^{\dagger} &= \sum_n \sqrt{n+1} (\sigma_n^- \sigma_{n+1}^+)_{\mathbf{p}} \\ a_{\mathbf{p}} &= \sum_n \sqrt{n+1} (\sigma_n^+ \sigma_{n+1}^-)_{\mathbf{p}}, \end{aligned} \quad (38)$$

where

$$\sigma^+ = \frac{1}{2}(X - iY), \quad \sigma^- = \frac{1}{2}(X + iY). \quad (39)$$

and therefore

$$a_{\mathbf{p}}^{\dagger} |p, n\rangle = \sqrt{n+1} |p, n+1\rangle \quad \text{and} \quad a_{\mathbf{p}} |p, n\rangle = \sqrt{n} |p, n-1\rangle \quad (40)$$

Now, considering Hermitian pairing of operators, we have

$$a_{\mathbf{p}} + a_{\mathbf{p}}^{\dagger} = \frac{1}{2} \sum_n \sqrt{n+1} (X_n X_{n+1} + Y_n Y_{n+1})_{\mathbf{p}}. \quad (41)$$

In theory the number of momentum states range till infinity, but for our simulation, we truncate the Hilbert space and have N momentum states, thus n varies from 0 to N in the above summations. This truncation in the bosonic occupation is proportional to the maximum energy expected to be simulated in semi-elastic collisions $N \propto \frac{E}{\omega_{\mathbf{p}}}$. It is expected that the error in this truncation is exponentially small with respect to the cutoff [7, 59–61]. However, careful numerical analysis of these (and other finite-system size effects) will be required in implementation [62, 63].

The details of the proofs for the following lemmas have been provided in Appendix A

A similar approach was laid out in [22]. We note several distinctions between the current implementation and previous work. The previous work has two conceptual appeals. One is the use of a binary representation for the occupation basis of $a_{\mathbf{p}} + a_{\mathbf{p}}^\dagger$. Further work may examine efficient fault-tolerant arithmetic primitives required for such an encoding, which we avoid here. The second is a diagonalization of the position space field amplitude

$$\phi_{\mathbf{n}}^{(i)} \equiv \frac{a_{\mathbf{n}}^{(i)} + a_{\mathbf{n}}^{(i)\dagger}}{\sqrt{2}} \quad (42)$$

where

$$a_{\mathbf{n}} \equiv \frac{1}{\sqrt{\mathcal{V}}} \sum_{\mathbf{q}} A_{\mathbf{q}} e^{i2\pi\mathbf{n}\cdot\mathbf{q}/N_s} \quad (43)$$

and

$$A_{\mathbf{q}} \equiv \frac{1}{2} \left[\omega_{\mathbf{q}}^{-\frac{1}{2}} + \omega_{\mathbf{q}}^{\frac{1}{2}} \right] a_{\mathbf{q}} + \frac{1}{2} \left[\omega_{\mathbf{q}}^{-\frac{1}{2}} - \omega_{\mathbf{q}}^{\frac{1}{2}} \right] a_{-\mathbf{q}}^\dagger \quad (44)$$

This is implemented with a squeezing operator framework. In a fault-tolerant framework, the phase factors of the squeezing operators introduce an additional error without a clear mapping to the final error. These squeezing factors also require a large number of controlled R_Z gates. Extending such an elegant implementation into the fault-tolerant regime would be an intriguing further work.

Lemma 2. *If \hat{n}_p is the number operator on momentum mode p then for any integer $r \geq 1$ we have,*

$$(\hat{n}_p)^r = \sum_n \frac{n^r}{2} (I_n - Z_n)_p.$$

Lemma 3. *If $m \geq 1$ and $r \geq 0$ are integers then we have,*

$$(a_p^\dagger)^m (n_p)^r + (n_p)^r (a_p)^m = \frac{1}{2} \sum_n \sqrt{\frac{(n+m)!}{n!}} n^r (X_n X_{n+m} + Y_n Y_{n+m})_p. \quad (45)$$

We first consider the non-interacting term in Eq.17 and derive the circuit required to simulate the unitary exponential, with the following result.

Lemma 4 (Complexity of non-interacting Hamiltonian Simulation). *We require at most $N|\Omega|$ number of R_z gates to simulate $e^{-iH'_0 t}$, where $H'_0 =: H_0$:*

Proof. The normal ordered H_0 i.e. $:H_0:$ is,

$$:H_0 := \sum_p \omega_p a_p^\dagger a_p = \sum_p \omega_p \hat{n}_p = \sum_{n,p} \frac{n\omega_p}{2} (I_n - Z_n)_p, \quad (46)$$

using Lemma 2. The above expression, up to some global phase is equal to a sum of Z operators. And the exponential of each of them can be implemented with a R_z gate, whose angle depends on the coefficient. Thus the lemma follows. \square

a. Circuits for simulating the interacting Hamiltonian : We now consider the interacting part of the Hamiltonian i.e. H_φ , which we are re-writing as follows, for convenience. Let $S_{4\mathbf{p}} = \{\mathbf{p} = (p_1, p_2, p_3, p_4) : p_i \in \Gamma; i = 1, 2, 3, 4\}$, be a set consisting of ordered 4-tuples of the momentum mode, that respects the conservation of momentum constraint ($p_1 + p_2 = \pm(p_3 + p_4)$). We parameterize this constraint by working in the basis such that such that $\pm p_3 = p_1 + k$ and $\pm p_4 = p_2 - k$.

Now, we can divide the terms in the above sum into 4 groups, based on equality of the momentum modes in \mathbf{p} , i.e. $H_\varphi := H_{1\varphi} + H_{2\varphi} + H_{3\varphi} + H_{4\varphi}$. For each such group, we map the resulting bosonic expression into the qubit space using Equation 41, Lemma 2, Lemma 3, and obtain the Pauli expression. From this, we derive the quantum circuit for the exponentiated sum in each group. We follow the methods described in [26]. In short, we first derive an eigenbasis for a set of mutually commuting Pauli terms and calculate the number of distinct non-zero eigenvalues

(ignoring sign) for the required (weighted) sum of these Paulis. This number is equal to the number of (controlled)-rotations we require. After that, we design the explicit quantum circuit, with appropriate optimizations. Each circuit is implemented with Clifford+T+(controlled)- R_z gates.

We have delegated the detailed description of the circuit design to Appendix A and in this section we only mention the relevant results required for estimating the gate counts. As indicated, we have separated the T-gates arising from the approximately implementable (controlled)-rotation gates and the exactly implementable Toffoli or multi-controlled-X gates. So we report the number of (controlled)- R_z required and the number of T gates in Lemma 5-8 do not include the T-gates from these rotations. The total T-count estimate is obtained by multiplying the number of (controlled)- R_z gates with the T-count of individual rotation gates and adding the product to the T-count arising from the exactly implementable parts.

In the following discussions let $S_{4\vec{n}} = \{\vec{n} = (n_1, n_2, n_3, n_4) : n_i = 0, 1, \dots, N; i = 1, 2, 3, 4\}$ be an ordered 4-tuple of momentum states. We reserved \mathbf{p} for vectors of unbolded p, k, q which themselves may be D -dimensional vectors.

B. Group I: All distinct momentum modes : $\mathbf{p}_1 \neq \mathbf{p}_2, \mathbf{k} \neq \mathbf{0}$.

We take Eq.21 with Hermitian terms grouped as

$$H_\lambda = \frac{\lambda}{4!|\Omega|^3} \sum_{\{\mathbf{p}_i\}} \frac{1}{4\sqrt{\omega_{\mathbf{p}_1}\omega_{\mathbf{p}_2}\omega_{\mathbf{p}_3}\omega_{-(\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3)}}} \left\{ (a_{\mathbf{p}_1} + a_{\mathbf{p}_1}^\dagger) (a_{\mathbf{p}_2} + a_{\mathbf{p}_2}^\dagger) (a_{\mathbf{p}_3} + a_{\mathbf{p}_3}^\dagger) \left(a_{-(\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3)} + a_{-(\mathbf{p}_1+\mathbf{p}_2+\mathbf{p}_3)}^\dagger \right) \right\} \quad (47)$$

This state divides into a tensor product of terms of the following form.

$$: H_\theta := \frac{\lambda}{24|\Omega|} \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \prod_{p_i \in \mathbf{p}} \frac{1}{\sqrt{2w_{p_i}}} (a_{p_i} + a_{p_i}^\dagger) \quad (48)$$

We denote the sum of the terms having distinct momentum modes by $H_{1\varphi}$. We note that this Hamiltonian is trivially normally order because distinct momentum operators commute. After the qubit mapping using Eq. 41, we get the following.

$$\begin{aligned} H_{1\varphi} &= \frac{\lambda}{24|\Omega|^3} \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \prod_{p_i \in \mathbf{p}} \frac{1}{\sqrt{2w_{p_i}}} \left(\sum_n \frac{1}{2} \sqrt{n+1} (X_{p_i, n} X_{p_i, n+1} + Y_{p_i, n} Y_{p_i, n+1}) \right) \\ &= \frac{\lambda}{96 \cdot 16|\Omega|^3} \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \sum_{\vec{n} \in S_{4\vec{n}}} \prod_{(p_j, n_j) \in (\mathbf{p}, \vec{n})} \sqrt{\frac{n_j+1}{w_{p_j}}} (X_{p_j, n_j} X_{p_j, n_j+1} + Y_{p_j, n_j} Y_{p_j, n_j+1}) \end{aligned} \quad (49)$$

Lemma 5. *It is possible to implement $e^{-iH_{1\varphi}t}$ using at most $\frac{N^4|\Omega|^2(|\Omega|-1)}{48} cR_z, \frac{N^4|\Omega|^2(|\Omega|-1)}{4} T, \frac{11N^4|\Omega|^2(|\Omega|-1)}{16} CNOT$ and $\frac{N^4|\Omega|^2(|\Omega|-1)}{24} H$ gates per time step t .*

The proof of the above lemma, including the detail of the circuit design has been given in Appendix A.

Group II : Two distinct momentum modes : $\mathbf{p}_1 = \mathbf{p}_2, \mathbf{k} \neq \mathbf{0}$.

Let $H_{2\varphi}$ is the sum of the terms with momentum modes satisfying the given constraint. Then,

$$\begin{aligned} : H_{2\varphi} : &= \frac{\lambda}{24|\Omega|^2} \sum_{p, k} \frac{1}{4\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \left((a_p + a_p^\dagger)^2 (a_{p+k} + a_{p+k}^\dagger) (a_{p-k} + a_{p-k}^\dagger) \right) \\ &= \frac{\lambda}{96} \sum_{p, k} \frac{1}{\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \left((a_p^\dagger)^2 + (a_p)^2 + 2n_p \right) (a_{p+k} + a_{p+k}^\dagger) (a_{p-k} + a_{p-k}^\dagger), \end{aligned}$$

and after using after using Eq. 41, Lemma 2, Lemma 3 we get the following.

$$\begin{aligned}
: H_{2\varphi} : &= \frac{\lambda}{96|\Omega|^2} \sum_{p,k} \frac{1}{\omega_p \sqrt{\omega_{p+k} \omega_{p-k}}} \left(\left(\sum_n \frac{\sqrt{(n+2)(n+1)}}{2} (X_n X_{n+2} + Y_n Y_{n+2})_p \right) + \left(\sum_n n (I_n - Z_n)_p \right) \right) \\
&\quad \left(\sum_n \frac{\sqrt{n+1}}{2} (X_n X_{n+1} + Y_n Y_{n+1})_{p+k} \right) \left(\sum_n \frac{\sqrt{n+1}}{2} (X_n X_{n+1} + Y_n Y_{n+1})_{p-k} \right) \\
&= \frac{\lambda}{96|\Omega|^2} \sum_{p,k} \frac{1}{\omega_p \sqrt{\omega_{p+k} \omega_{p-k}}} \\
&\quad \times \left(\sum_{n_1, n_2, n_3} c_n^{(1)} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_p (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \right. \\
&\quad \left. + \sum_{n_1, n_2, n_3} c_n^{(2)} (I_{n_1} - Z_{n_1})_p (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \right) \quad (50)
\end{aligned}$$

where $c_n^{(1)} = \frac{\sqrt{(n+2)(n+1)(n_2+1)(n_3+1)}}{8}$, $c_n^{(2)} = \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4}$.

Lemma 6. *It is possible to implement $e^{-iH_{2\varphi}t}$ using at most $\frac{N^3|\Omega|^2}{3} cR_z$, $\frac{8N^3|\Omega|^2}{3} T$, $\frac{20N^3|\Omega|^2}{3}$ CNOT and $\frac{2N^3|\Omega|^2}{3} H$ gates per time step t .*

The proof of the above lemma, including the detail of the circuit design has been given in Appendix A.

Group III : Two distinct momentum modes : $\mathbf{p}_1 \neq \mathbf{p}_2, \mathbf{k} = \mathbf{0}$.

We use $H_{3\varphi}$ to denote the sum of the terms with momentum mode satisfying the given constraint. Then,

$$\begin{aligned}
: H_{3\varphi} : &= \frac{\lambda}{24|\Omega|^2} \sum_{p_1, p_2} \frac{1}{4\omega_{p_1} \omega_{p_2}} (a_{p_1} + a_{p_1}^\dagger)^2 (a_{p_2} + a_{p_2}^\dagger)^2 \\
&= \frac{\lambda}{96} \sum_{p_1, p_2} \frac{1}{\omega_{p_1} \omega_{p_2}} ((a_{p_1}^\dagger)^2 + (a_{p_1})^2 + 2\hat{n}_{p_1}) ((a_{p_2}^\dagger)^2 + (a_{p_2})^2 + 2\hat{n}_{p_2}). \quad (51)
\end{aligned}$$

After applying Eq. 41, Lemma 2, Lemma 3 we obtain the following.

$$\begin{aligned}
&: H_{3\varphi} : \\
&= \frac{\lambda}{96|\Omega|^2} \sum_{p_1, p_2} \frac{1}{\omega_{p_1} \omega_{p_2}} \left(\sum_n \frac{\sqrt{(n+1)(n+2)}}{2} (X_n X_{n+2} + Y_n Y_{n+2})_{p_1} + \sum_n n (I_n - Z_n)_{p_1} \right) \\
&\quad \left(\sum_n \frac{\sqrt{(n+1)(n+2)}}{2} (X_n X_{n+2} + Y_n Y_{n+2})_{p_2} + \sum_n n (I_n - Z_n)_{p_2} \right) \\
&= \frac{\lambda}{96|\Omega|^2} \sum_{p_1, p_2} \frac{1}{\omega_{p_1} \omega_{p_2}} \left(\sum_{n_1, n_2} c_n^{(3)} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1} (X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2} \right. \\
&\quad + \sum_{n_1, n_2} c_n^{(4)} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1} (I_{n_2} - Z_{n_2})_{p_2} + \sum_{n_1, n_2} c_n^{(5)} (X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2} (I_{n_1} - Z_{n_1})_{p_1} \\
&\quad \left. + \sum_{n_1, n_2} n_1 n_2 (I_n - Z_{p_1, n_1} - Z_{p_2, n_2} + Z_{p_1, n_1} Z_{p_2, n_2}) \right) \quad (52)
\end{aligned}$$

where $c_n^{(3)} = \frac{\sqrt{(n+2)(n+1)(n_2+2)(n_2+1)}}{4}$, $c_n^{(4)} = \frac{n_2 \sqrt{(n+2)(n+1)}}{2}$, $c_n^{(5)} = \frac{n_1 \sqrt{(n_2+2)(n_2+1)}}{2}$.

Lemma 7. *It is possible to implement $e^{-iH_{3\varphi}t}$ using at most $2N^2|\Omega|(|\Omega|-1) cR_z$, $8N^2|\Omega|(|\Omega|-1) T$, $16N^2|\Omega|(|\Omega|-1)$ CNOT and $3N^2|\Omega|(|\Omega|-1) H$ gates per time step t .*

The proof of the above lemma, including the detail of the circuit design has been given in Appendix A.

	#(c) \mathbf{R}_z	# \mathbf{T}	# \mathbf{CNOT}	# \mathbf{H}	
$e^{-i\mathbf{H}_0 t}$	$N \Omega $	-	-	-	Lemma 4
$e^{-i\mathbf{H}_1 \varphi t}$	$\frac{N^4 \Omega ^2(\Omega -1)}{48}$	$\frac{N^4 \Omega ^2(\Omega -1)}{4}$	$\frac{11N^4 \Omega ^2(\Omega -1)}{16}$	$\frac{N^4 \Omega ^2(\Omega -1)}{24}$	Lemma 5
$e^{-i\mathbf{H}_2 \varphi t}$	$\frac{N^3 \Omega ^2}{3}$	$\frac{8N^3 \Omega ^2}{3}$	$\frac{20N^3 \Omega ^2}{3}$	$\frac{2N^3 \Omega ^2}{3}$	Lemma 6
$e^{-i\mathbf{H}_3 \varphi t}$	$2N^2 \Omega (\Omega -1)$	$8N^2 \Omega (\Omega -1)$	$16N^2 \Omega (\Omega -1)$	$3N^2 \Omega (\Omega -1)$	Lemma 7
$e^{-i\mathbf{H}_4 \varphi t}$	$3N \Omega $	-	$4N \Omega $	$4N \Omega $	Lemma 8

TABLE II: Summary of the number of gates required to implement the exponentials of the different Hamiltonian partitions in the occupation basis.

Group IV: All equal momentum modes : $\mathbf{p}_1 = \mathbf{p}_2, \mathbf{k} = \mathbf{0}$.

The sum of the terms with all four equal momentum mode is denoted by $H_{4\varphi}$. These terms must be normal ordered to be

$$\begin{aligned}
: H_{4\varphi} : &= \frac{\lambda}{24|\Omega|} \sum_p \frac{1}{4(\omega_p)^2} \left\{ (a_p^\dagger + a_p)^4 \right\} \\
&= \frac{\lambda}{24|\Omega|} \sum_p \frac{1}{4(\omega_p)^2} \left\{ ((a_p^\dagger)^4 + (a_p)^4) + 4((a_p^\dagger)^3 a_p + a_p^\dagger (a_p)^3) + 6((a_p^\dagger)^2 (a_p)^2) \right\} \\
&= \frac{\lambda}{96|\Omega|} \sum_p \frac{1}{(\omega_p)^2} \left(((a_p^\dagger)^4 + (a_p)^4) + 4((a_p^\dagger)^2 \hat{n}_p + \hat{n}_p (a_p)^2) + 6((\hat{n}_p)^2 - \hat{n}_p) \right);
\end{aligned}$$

and after applying Eq. 41, Lemma 2, Lemma 3 we get the following.

$$\begin{aligned}
&: H_{4\varphi} : \\
&= \frac{\lambda}{96|\Omega|} \sum_{p,n} \frac{1}{(\omega_p)^2} \left(\frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2} (X_n X_{n+4} + Y_n Y_{n+4})_p \right. \\
&\quad \left. + 2n\sqrt{(n+2)(n+1)} (X_n X_{n+2} + Y_n Y_{n+2})_p + 3(n^2 - n)(I_n - Z_n)_p \right) \tag{53}
\end{aligned}$$

Lemma 8. *It is possible to implement $e^{-iH_{4\varphi} t}$ using at most $3N|\Omega|$ (c)- R_z , $4N|\Omega|$ CNOT and $4N|\Omega|$ H gates per time step t .*

The proof of the above lemma, including the details of the circuit design, has been given in Appendix A. An estimate of the total number of gates required to implement e^{-iHt} can be obtained by summing the gate costs in Lemma 4-8. This is summarized in the following theorem, as well as in Table II.

Theorem 9. *It is possible to implement e^{-iHt} with the following number of gates per time step.*

1. $\frac{N^4|\Omega|^2(|\Omega|-1)}{48} + \frac{N^3|\Omega|^2}{3} + 2N^2|\Omega|(|\Omega|-1) + 4N|\Omega|$ (controlled)- R_z gates.
2. $\frac{N^4|\Omega|^2(|\Omega|-1)}{4} + \frac{8N^3|\Omega|^2}{3} + 8N^2|\Omega|(|\Omega|-1)$ additional T gates.
3. $\frac{11N^4|\Omega|^2(|\Omega|-1)}{16} + \frac{20N^3|\Omega|^2}{3} + 16N^2|\Omega|(|\Omega|-1) + 4N|\Omega|$ CNOT gates.
4. $\frac{N^4|\Omega|^2(|\Omega|-1)}{24} + \frac{2N^3|\Omega|^2}{3} + 3N^2|\Omega|(|\Omega|-1) + 4N|\Omega|$ H gates.

Proof. The proof follows from Lemmas 5, 5,6,7, and 8, which are summarized in II. □

C. Trotter Error Analysis

In this section we discuss and bound the various errors that occur during simulation of the occupation basis Hamiltonian. For convenience, we restate the occupation basis Hamiltonian in the following equation.

$$H_{occ} = H_0 + H_\lambda \quad (54)$$

$$: H_0 : = \sum_p \omega_p a_p^\dagger a_p \quad (55)$$

$$: H_\lambda : = H_{1\varphi} + H_{2\varphi} + H_{3\varphi} + H_{4\varphi} \quad (56)$$

a. I. Trotter Error : One primary source of error is the one inherent in the Trotterization procedure, where we express the exponential of sum of operators as product of the exponentiated operators. This is true if and only if the operators are commuting. If there are non-commuting parts then the resultant error in the approximation is referred to as the *Trotter error*. Quite a few bounds on the Trotter error have been derived before [64–66], but we use the one in [67], which shows the dependence on nested commutators.

If a Hamiltonian $H = \sum_{\gamma=1}^{\Gamma} H_\gamma$ is a sum of Γ fragment Hamiltonians, then $e^{-i\tau H}$ can be approximated by product of exponentials, using the p^{th} order Trotter-Suzuki formula [68], $\mathcal{S}_p(\tau) = e^{-i\tau H} + \mathcal{A}(\tau)$, where $\|\mathcal{A}(\tau)\| \in O(\tilde{\alpha}_{comm}\tau^{p+1})$ if each H_γ are Hermitian [67]. Here $\tilde{\alpha}_{comm} = \sum_{\gamma_1, \gamma_2, \dots, \gamma_{p+1}=1}^{\Gamma} \|[H_{\gamma_{p+1}}, \dots [H_{\gamma_2}, H_{\gamma_1}]]\|$. In most applications, it is quite cumbersome, if not impossible, to derive a rigorous analytic expression of the nested commutators, in order to tightly bound higher order Trotter error. We use the following bound from [27] in this work.

$$\tilde{\alpha}_{comm} \leq 2^{p-(p'+1)} \sum_{\gamma_{i_1}, \gamma_{i_2}, \dots, \gamma_{i_{p'+1}}} \|[H_{\gamma_{p'+1}}, [\dots [H_{\gamma_3}, [H_{\gamma_2}, H_{\gamma_1}]] \dots]]\| \left(\sum_{\gamma=1}^{\Gamma} \|H_\gamma\| \right)^{p-p'} \quad [1 \leq p' \leq p] \quad (57)$$

The above formula is specifically useful in scenarios where we can compute tight bounds till level $p' < p$ of nesting. Then we combine the tighter bound till level p' and a looser sum of norm bound for the remaining levels. In this paper we consider the 2^{nd} order Trotter error in our implementations. In Appendix B we derive rigorous expressions and bounds on the first level commutators and norms. Then we use the above equation in order to bound the higher order errors.

We primarily focus on these low level Trotter errors for two reasons. The symmetric low-level Trotter errors provide tighter bounds [67]. They also appear to provide more accurate results when compared to explicit calculations [69].

Lemma 10. *Let H be the occupation basis Hamiltonian derived in Eqs. 46-53. Let $\mathcal{S}_2(\tau)$ be a 2^{nd} order Trotter-Suzuki approximation for $e^{-i\tau H}$. Then,*

$$\|e^{-iH\tau} - \mathcal{S}_p(\tau)\| \in O(\tilde{\alpha}_{comm}\tau^3), \quad (58)$$

where

$$\tilde{\alpha}_{comm} \in O\left(\frac{\lambda^2 N^6}{M^5}\right). \quad (59)$$

The details of this proof are provided in Appendix B.

D. Total T-gate cost estimate

Here we review the various contributions to the error in phase estimation. Based on previous simulation work, the most efficient distribution of error seems a highly precise phase estimation calculation to offset a single Trotter step [6]. In particular, we focus on the scheme of [70], which allows us to obtain optimal constant factors using Fourier-based phase estimation. We use an approximate quantum Fourier transform (AQFT) [71] in the preparation. The number of T-gates required to approximate an n -qubit QFT circuit up to error ϵ_{QFT} is

$$8n \log_2 \left(\frac{n}{\epsilon_{QFT}} \right) + \log_2 \left(\frac{n}{\epsilon_{QFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{n}{\epsilon_{QFT}} \right)}{\epsilon_{QFT}} \right). \quad (60)$$

Lemma 11. *There exists ϵ'_E such that for all sufficiently small $\epsilon_E \leq \epsilon'_E$ the total number of implementations of $\mathcal{S}_2(\tau)$ required to estimate an eigenstate within error ϵ_E is bounded above by*

$$2^m \leq \frac{\pi^2 \sqrt{\tilde{\alpha}_{\text{comm}}}}{\epsilon_E^{3/2}}. \quad (61)$$

Proof. The variance on the phase measured from QPE is given as

$$\epsilon_V = \langle \cos((\hat{\theta} - \tilde{\theta})) \rangle^2 - 1, \quad (62)$$

known as the Holevo variance, where $\tilde{\theta}$ is the true phase and $\hat{\theta}$ is the measured outcome. This allows simple analytic results for a compact phase that may otherwise have artificially high variance when peaked near the boundary. This can be computed using the phase estimation algorithm implemented in [70]. The estimated phase can be written as

$$\theta_{\text{est}} = \theta_{\text{true}} + \theta + \epsilon_{\text{prep}}, \quad (63)$$

where θ is a random variable with zero mean and Holevo variance ϵ_V describing the output of phase estimation and ϵ_{prep} represents the systematic errors in the phase that arise because of approximate gate synthesis, approximate QFT and approximation due to Trotterization. In the limit of small variance, with high probability we have the following.

$$\begin{aligned} \epsilon_\theta &= \sqrt{\mathbb{E}[(\theta_{\text{est}} - \theta_{\text{true}})^2]} \approx \sqrt{\epsilon_V(\theta) + (\pi\epsilon_{\text{QFT}} + \epsilon_{\text{Trotter}} + \epsilon_{\text{synth}})^2} \\ &\approx \sqrt{\left(\frac{\pi}{2^{m+1}}\right)^2 + (\pi\epsilon_{\text{QFT}} + \epsilon_{\text{synth}} + \epsilon_{\text{Trotter}})^2}, \end{aligned} \quad (64)$$

We select the following in order to get the above expression.

$$\epsilon_{\text{Trotter}} \leq \frac{\sqrt{2}\epsilon_\theta}{4}, \quad \epsilon_{\text{QFT}} \leq \frac{\sqrt{2}\epsilon_\theta}{8\pi}, \quad \epsilon_{\text{synth}} \leq \frac{\sqrt{2}\epsilon_\theta}{8}, \quad 2^m \geq \frac{\pi}{\sqrt{2}\epsilon_\theta} \quad (65)$$

We require that the error in the Trotter-Suzuki expansion to be at most $\epsilon_{\text{Trotter}} = \tilde{\alpha}_{\text{comm}}\tau^3$. Further, let

$$\epsilon_\theta = \epsilon_E \tau, \quad (66)$$

where ϵ_E is the estimate in the energy that comes from rescaling the estimate of the phase by a factor of τ . Thus it suffices to choose

$$\tau = \sqrt{\frac{\epsilon_E}{2^{3/2}\tilde{\alpha}_{\text{comm}}}} \quad (67)$$

and therefore, the error in the phase is related to the error in the energy estimate as

$$\epsilon_\theta = \sqrt{\frac{\epsilon_E^3}{2^{3/2}\tilde{\alpha}_{\text{comm}}}} \quad (68)$$

and therefore for $\pi\sqrt{\tilde{\alpha}_{\text{comm}}}/\epsilon_E^{3/2} \geq \frac{1}{1-2^{-1/8}}$ at most

$$2^m = \left\lceil \frac{\pi\sqrt{\tilde{\alpha}_{\text{comm}}}}{2^{1/8}\epsilon_E^{3/2}} \right\rceil \leq \frac{\pi\sqrt{\tilde{\alpha}_{\text{comm}}}}{\epsilon_E^{3/2}}. \quad (69)$$

Finally, as the analysis in [70] is tight in the limit as $\epsilon_E \rightarrow 0$ let us assume that ϵ_E is small enough so that

$$\left| \sqrt{\mathbb{E}[(\theta_{\text{est}} - \theta_{\text{true}})^2]} - \sqrt{\left(\frac{\pi}{2^{m+1}}\right)^2 + (\pi\epsilon_{\text{QFT}} + \epsilon_{\text{Trotter}} + \epsilon_{\text{synth}})^2} \right| \leq \epsilon_\theta. \quad (70)$$

We then can ensure a sufficient value of m by taking $\epsilon_E \rightarrow \epsilon_E/2$ and using the remaining error budget to accommodate the error due to m being finite. Using the observation that since $2^{3/2} < \pi$ it suffices to take

$$\frac{\pi\sqrt{\tilde{\alpha}_{\text{comm}}}}{(\epsilon_E/2)^{3/2}} \leq \frac{\pi^2\sqrt{\tilde{\alpha}_{\text{comm}}}}{\epsilon_E^{3/2}} = 2^m \quad (71)$$

And thus this sufficient value is an upper bound on the necessary value. \square

Theorem 12. *Given an eigenstate $|\psi\rangle$ of H such that $H|\psi\rangle = E|\psi\rangle$, the occupation basis Hamiltonian stated in (56), then it there exists a quantum algorithm that outputs with probability greater than $2/3$ a value \hat{E} such that $|\hat{E} - E| \leq \epsilon_E$, using a number of T gates that scales as*

$$\mathcal{O}\left(\frac{\lambda N^7 |\Omega|^3}{M^{5/2} \epsilon_E^{3/2}} \log\left(\frac{N|\Omega|}{M\epsilon_E}\right)\right)$$

T gates and $\mathcal{O}(N|\Omega|)$ qubits, plus an additional $\mathcal{O}\left(\log\left(\frac{\lambda N}{M\epsilon_E}\right)\right)$ ancillary qubits required for phase estimation. Here M is the particle mass for the field, the log argumenent comes from $1/\epsilon_r$ $\epsilon_r = \frac{\sqrt{2}\epsilon_E}{8N_r} \sqrt{\frac{\epsilon_E}{2^{3/2}\tilde{\alpha}_{comm}}}$, $N_r \in \mathcal{O}(N^4|\Omega|^3)$ and $\tilde{\alpha}_{comm} \in \mathcal{O}\left(\frac{\lambda^2 N^6}{M^5}\right)$.

Proof. Suppose we allocate ϵ_r as an upper bound on the permitted synthesis error per R_z gate. From Theorem 9 and Table II we find that the total number of (controlled)- R_z required for each Trotter step is at most

$$N_r := \frac{N^4|\Omega|^2(|\Omega| - 1)}{48} + \frac{N^3|\Omega|^2}{3} + 2N^2|\Omega|(|\Omega| - 1) + 4N|\Omega| \quad (72)$$

and so using the T-count estimate in [32] and assuming the T-count of controlled- R_z is at most the T-count of R_z [36], the expected number of T-gates from rotations is at most

$$N_{T/R_z} \leq N_r (3.067 \log_2(2/\epsilon_r) - 4.327).$$

Also, from Theorem 9 we require the following additional number of T-gates.

$$N_+ := \frac{N^4|\Omega|^2(|\Omega| - 1)}{4} + \frac{8N^3|\Omega|^2}{3} + 8N^2|\Omega|(|\Omega| - 1). \quad (73)$$

So, the total number of T-gates per Trotter step is

$$\mathcal{G}_T \leq N^4|\Omega|^2(|\Omega| - 1)[0.064 \log_2(2/\epsilon_r) + 0.16] + N^3|\Omega|^2[1.022 \log_2(2/\epsilon_r) + 1.224] \quad (74)$$

$$+ N^2|\Omega|(|\Omega| - 1)[6.134 \log_2(2/\epsilon_r) - 0.654] + 4N|\Omega|[3.067 \log_2(2/\epsilon_r) - 4.327] \\ \in \mathcal{O}(N^4|\Omega|^3 \log_2(2/\epsilon_r)). \quad (75)$$

Using Lemmas 10 and 11, the total number of T-gates required to achieve an eigenstate within total error ϵ_E is at most

$$\mathcal{G}_T \cdot 2^m \leq \frac{\pi^2 \sqrt{\alpha_{comm}}}{\epsilon_E^{3/2}} (N^4|\Omega|^2(|\Omega| - 1)[0.064 \log_2(2/\epsilon_r) + 0.16] + N^3|\Omega|^2[1.022 \log_2(2/\epsilon_r) + 1.224] \\ + N^2|\Omega|(|\Omega| - 1)[6.134 \log_2(2/\epsilon_r) - 0.654] + 4N|\Omega|[3.067 \log_2(2/\epsilon_r) - 4.327]) \\ \in \mathcal{O}\left(\frac{\lambda N^7 |\Omega|^3}{m^{5/2} \epsilon_E^{3/2}} \log(1/\epsilon_r)\right). \quad (76)$$

The total synthesis error due to approximation of the rotation gates is $\epsilon_{synth} = \epsilon_r \cdot N_r$, where the value of ϵ_{synth} is given in Eq. 65. Plugging in the values of the timestep τ and ϵ_θ from Eq. (67) and (66), respectively we obtain the following bound on ϵ_r in order to ensure that the final error in the estimate of the energy is at most ϵ_E .

$$\epsilon_r \leq \frac{\sqrt{2}\epsilon_E \tau}{8N_r} = \frac{\sqrt{2}\epsilon_E}{8N_r} \sqrt{\frac{\epsilon_E}{2^{3/2}\tilde{\alpha}_{comm}}} \quad (77)$$

To obtain the total number of ancillary qubits used, we get

$$m \in \mathcal{O}\left(\log\left(\frac{\alpha_{comm}^{1/2}}{\epsilon_E^{3/2}}\right)\right) \\ \in \mathcal{O}\left(\log\left(\frac{\lambda N}{M\epsilon_E}\right)\right) \quad (78)$$

where we have repeatedly used $\log(A^m/B^n) \in \mathcal{O}(\log(A/B))$ for constant $m, n > 0$. Thus the theorem is proved. \square

E. Field Amplitude Basis

In this section we describe algorithms to simulate the scalar ϕ^4 Hamiltonian H_{amp} expressed in the amplitude basis, as given in Eq. 11. We use two types of algorithms - qubitization [21, 72, 73] and Trotterization [68]. For the former we describe three approaches, which mainly differ in the LCU (linear combination of unitaries) decomposition of the operators. Similar approaches have been considered in [25]. In Section IV E 1 we discuss a decomposition of Φ using an equal weight LCU decomposition of the field operator (Algorithm I). Next, in Section IV E 2 we describe another LCU decomposition of Φ as sum of mainly Z-operators, which not only helps us in developing a qubitization based algorithm (Algorithm IIIa) in IV E 3, but it also makes the expression amenable to Trotterization (Algorithm II), as discussed in Section IV E 2. In Section IV E 4 we describe another more compact LCU decomposition of all operators using binary representation of integers. With this we describe another qubitization algorithm (Algorithm IIIb). Before describing our algorithms we mention the following results which have been used in the LCU based approaches in order to reduce the gate complexity.

a. Recursive block encoding : We use the following theorem to recursively block encode H_{amp} using a divide and conquer approach, as described in [27], where it has been shown that with such an approach it is possible to block encode with a smaller number of gates. Suppose without loss of generality, we have a Hamiltonian H_i expressed as a linear combination of unitaries (LCU) i.e. $H_i = \sum_{j=1}^{M_i} h_{ij} U_{ij}$, such that $\lambda_i = \sum_j |h_{ij}|$. In this case, we can have a $(\lambda_i, \log M_i, 0)$ -block encoding of H_i using an ancilla preparation subroutine and a unitary selection subroutine, which we denote by PREP_i and SELECT_i respectively.

$$\text{PREP}_i |0\rangle^{\log M_i} = \sum_{j=1}^{M_i} \sqrt{\frac{h_{ij}}{\lambda_i}} |j\rangle \quad (79)$$

$$\text{SELECT}_i = \sum_{j=1}^{M_i} |j\rangle \langle j| \otimes U_{ij} \quad (80)$$

It can be shown that [74]

$$\langle 0 | \text{PREP}_i^\dagger \cdot \text{SELECT}_i \cdot \text{PREP}_i | 0 \rangle = \frac{H_i}{\lambda_i}. \quad (81)$$

Suppose we have M Hamiltonians denoted by H_1, \dots, H_M , each of which has an LCU decomposition and for each one of them we define the subroutines as in Eq. 79 and 80. Now we use these subroutines to define the following,

$$\text{PREP} |0\rangle^{\log M + \sum_i \log M_i} = \left(\sum_{i=1}^M \sqrt{\frac{w_i \lambda_i}{\mathcal{A}}} |i\rangle \right) \otimes \bigotimes_{i=1}^M \text{PREP}_i \quad (82)$$

$$\text{SELECT} = \sum_{i=1}^M \left(|i\rangle \langle i| \otimes \bigotimes_{k=1}^{i-1} \mathbb{I} \otimes \text{SELECT}_i \otimes \bigotimes_{k=i+1}^M \mathbb{I} \right), \quad (83)$$

where $w_i > 0$ and $\mathcal{A} = \sum_{i=1}^M w_i \lambda_i$. We can use the above two subroutines to block encode a linear combination of Hamiltonians as follows.

Theorem 13 ([27]). *Let $H = \sum_{i=1}^M w_i H_i$ be the sum of M Hamiltonians and each of them is expressed as sum of unitaries as : $H_i = \sum_{j=1}^{M_i} h_{ij} U_{ij}$ such that $\lambda_i = \sum_j |h_{ij}|$, $w_i > 0$. Each of the summand Hamiltonian is block-encoded using the subroutines defined in Eq. 79 and 80. Then, we can have an $(\mathcal{A}, \lceil \log_2(M) \rceil, 0)$ -block encoding of H , where $\mathcal{A} = \sum_{i=1}^M w_i \lambda_i$, using the ancilla preparation subroutine (PREP) defined in Eq. 82 and the unitary selection subroutine (SELECT) defined in Eq. 83.*

1. The PREP subroutine has an implementation cost of $\mathcal{C}_{\text{PREP}} = \sum_{i=1}^M \mathcal{C}_{\text{PREP}_i} + \mathcal{C}_w$, where $\mathcal{C}_{\text{PREP}_i}$ is the number of gates to implement PREP_i and \mathcal{C}_w is the cost of preparing the state $\sum_{i=1}^M \sqrt{\frac{w_i \lambda_i}{\mathcal{A}}} |i\rangle$.
2. The SELECT subroutine can be implemented with a set of multi-controlled-X gates - $\{M_i \text{ pairs of } C^{\log_2 M_i + 1} X \text{ gates} : i = 1, \dots, M\}$, M pairs of $C^{\log M} X$ gates and $\sum_{i=1}^M M_i$ single-controlled unitaries - $\{cU_{ij} : j = 1, \dots, M_i; i = 1, \dots, M\}$.

Additionally, suppose in the above theorem, all the H_i are the same but they act on disjoint subspaces. In this case, each PREP_i is the same and so it is sufficient to keep only one copy of PREP_i in the PREP subroutine of Eq. 82. We can absorb w_i in the weights of the unitaries obtained in the LCU decomposition of H_i . Thus, in this case we have

$$\text{PREP } |0\rangle^{\log M + \log M_i} = \left(\sqrt{\frac{1}{M}} \sum_{i=1}^M |i\rangle \right) \otimes \text{PREP}_i. \quad (84)$$

We require only $\lceil \log M \rceil$ H gates to prepare the superposition in the first register by padding out the number of such subspaces to be a power of 2. We also need to make slight modifications in the SELECT procedure. This time, we keep an extra ancilla qubit, initialized to 0, in each subspace. Given a particular state of the first register, we select a subspace by flipping the qubit in the corresponding subspace. The unitaries in each subspace are now additionally controlled on this qubit (of its own subspace).

b. Group of multi-controlled-X gate synthesis : We can further optimize the number of gates by implementing the group of multi-controlled-unitaries in the SELECT subroutines, using the following theorem [27]. Here we partition the control qubits into different groups, store intermediate information in some ancillae and then implement the required logic using these intermediate results.

Theorem 14 ([27]). *Consider the unitary $U = \sum_{j=0}^{M-1} |j\rangle\langle j| \otimes U_j$ for unitary operators U_j that can be implemented controllably. We assume M is a power of 2 for simplicity. Suppose we have $\log_2 M$ qubits and M (compute-uncompute) pairs of $C^{\log_2 M} X$ gates for selecting the M basis states. Let $r_1, \dots, r_n \geq 1$ be positive fractions such that $\sum_{i=1}^n \frac{1}{r_i} = 1$ and $\frac{\log_2 M}{r_i}$ are integers. Then, U can be implemented with a circuit with*

$$\sum_{i=1}^n M^{\frac{1}{r_i}} C^{\frac{\log_2 M}{r_i}} X + MC^n X$$

(compute-uncompute) pairs of gates, M applications of controlled U_j and at most $\sum_{i=1}^n M^{\frac{1}{r_i}}$ ancillae.

Following the construction in [75, 76], the number of T-gates required to implement such multiply controlled gates is

$$\mathcal{T}_n = \sum_{i=1}^n M^{\frac{1}{r_i}} \left(\frac{4 \log_2 M}{r_i} - 4 \right) + M(4n - 4). \quad (85)$$

With the help of logical AND gadgets we do not require to use any T-gate for the uncomputation part.

c. Equal weight LCU: Note that the Hamiltonian Eq. (11) consists of 4 different families of terms, the Π^2 , $\Phi\Phi$, Φ^2 , and Φ^4 terms, each of which share the same coefficients. In principle, if we can further ensure that the LCU decompositions of each of those terms provides the same weight to every unitary in the LCU, then we can exploit this structure to drastically simplify the PREP and SELECT circuits. To achieve such an equal weight LCU, we will make use of the following lemma

Lemma 15. *Consider any arbitrary diagonal matrix*

$$A = \sum_{j=0}^{L-1} n_j |j\rangle\langle j| \quad (86)$$

where $n_j \in \mathbb{Z}^+$. Then,

$$A = \frac{1}{2} \sum_{i=0}^{2n_{max}-1} U^{(i)}, \quad \text{where} \quad (87)$$

$$U_{j,k}^{(i)} = \delta_{jk} [2\Theta(n_{max} + n_j - i - 1) - 1]$$

where the step-function is defined as

$$\Theta(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{if } x < 0 \end{cases} \quad (88)$$

Proof. We simply note that $n_j = \frac{1}{2} [(n_{max} + n_j) - (n_{max} - n_j)]$, we can describe such a matrix as a sum consisting of unitaries with entries only in ± 1 such that the (j, j) -th entry in the first $(n_{max} + n_j)$ unitaries equals $+1$, and equals -1 in the remaining $(n_{max} - n_j)$ unitaries. \square

Note that while the dimensionality of the $U_{j,k}^i$ is given by L , the number of terms in the LCU Eq. (87) is independent of L .

d. LCU decomposition of integer diagonal matrices : If a diagonal matrix consists of integers only then we can have an LCU decomposition consisting of $O(\log m_{max})$ signature matrices, where m_{max} is the maximum absolute value of any of its entries. This can be obtained by a binary decomposition of each integer, as stated below.

Lemma 16 ([27]). *Let M_I is a $N \times N$ diagonal integer matrix, which has N' positive integers whose maximum value is m'_{max} and N'' negative integers such that $m''_{max} = \max_i \{|M_I[i, i]| : M_I[i, i] < 0\}$. Then, $M = c_0 \mathbb{I} + \sum_{i=1}^{N'+N''} c_i D_i$, where D_i are signature matrices and $N' \leq \lceil \log_2(m'_{max} + 1) \rceil = \zeta'$ and $N'' \leq \lceil \log_2(m''_{max} + 1) \rceil = \zeta''$. Also, $\sum_{i=0}^{N'+N''} |c_i| \leq 2^{\zeta'} - 1$.*

Specifically, we can use the following lemma to have an LCU decomposition of $\frac{\phi}{\Delta\phi} = \text{diag}(-k, \dots, -1, 0, 1, \dots, k-1)$.

Lemma 17 (Lemma 11 in [27]). *Let $U = \mathbb{I}_{(\zeta)} \otimes \dots \otimes \mathbb{I}_{(\ell+1)} \otimes Z_{(\ell)} \otimes \mathbb{I}_{(\ell-1)} \otimes \dots \otimes \mathbb{I}_{(1)}$ is a tensor product of ζ single-qubit unitaries where Z is applied on qubit ℓ and \mathbb{I} on the rest. Then U is a diagonal matrix of the following form.*

$$\begin{aligned} U_{j,j} &= 1 & \text{if} & \quad j = 2^\ell k, 2^\ell k + 1, \dots, 2^\ell k + 2^{\ell-1} - 1 \\ &= -1 & \text{if} & \quad j = 2^\ell k + 2^{\ell-1}, 2^\ell k + 2^{\ell-1} + 1, \dots, 2^\ell k + 2^\ell - 1 \\ & & \text{where} & \quad k = 0, 1, \dots, 2^{\zeta-\ell} - 1 \end{aligned}$$

It follows that

$$\frac{\phi}{\Delta\phi} = -\frac{1}{2} \sum_{j=0}^{\log_2 k} 2^j Z_j - \frac{1}{2} \mathbb{I}. \quad (89)$$

e. Error in the scalar field : We also bound the error in the scalar field in terms of the target energy scale as below.

Theorem 18. *Let $|\phi_{max}|$ be the maximum allowed value of the scalar field. Then, it suffices to take*

$$|\phi_{max}| = \left(\frac{\epsilon E_{max}}{C(M, \Lambda, d) |\Omega|} \right)^{1/4} \quad (90)$$

where $E_{max} > 0$ is the maximum energy scale we wish to allow in our simulation of the Hamiltonian H_{amp} given in Eq. (11), $\epsilon = \Pr(H_{amp} \geq E_{max})$ is the probability that a measurement of the Hamiltonian exceeds E_{max} , $|\Omega|$ is the lattice size, and $C(M, \Lambda, d) = M^2 + 3d + \frac{\Lambda}{4!} + \frac{3}{2}$.

Proof. Let $|\phi_{max}|$ be the maximum allowed value of the scalar field in our simulation. Then, in the field amplitude basis defined by $\hat{\Phi}|\phi\rangle = \phi|\phi\rangle$, it readily follows that $\langle \phi | \hat{\Phi}^n | \phi \rangle \leq |\phi_{max}|^n$ at a single site. Similarly, since the momentum operator is related to the field operator via a Fourier transform, $\hat{\Pi} = \mathcal{F}^\dagger \hat{\Phi} \mathcal{F}$, then letting $|\tilde{\phi}\rangle = \mathcal{F}|\phi\rangle$, it also readily follows for a single lattice site that

$$\begin{aligned} \langle \phi | \hat{\Pi}^n | \phi \rangle &= \langle \tilde{\phi} | \hat{\Phi}^n | \tilde{\phi} \rangle \\ &\leq |\phi_{max}|^n \end{aligned} \quad (91)$$

Now, the Hamiltonian is given by

$$H_{amp} = \sum_{\vec{x} \in \Omega} \left[\frac{1}{2} \Pi^2(\vec{x}) + (M^2 + d + 1) \Phi^2(\vec{x}) + \frac{\Lambda}{4!} \Phi^4(\vec{x}) - 2 \sum_{i=1}^d \Phi(\vec{x}) \Phi(\vec{x} + a\hat{x}_i) \right] \quad (92)$$

It therefore follows that

$$\begin{aligned}
\langle \phi | H_{amp} | \phi \rangle &= \sum_{\vec{x} \in \Omega} \langle \phi | \left[\frac{1}{2} \Pi^2(x) + (M^2 + d + 1) \Phi^2(\vec{x}) + \frac{\Lambda}{4!} \Phi^4(\vec{x}) - 2 \sum_{i=1}^d \Phi(\vec{x}) \Phi(\vec{x} + a\hat{x}_i) \right] | \phi \rangle \\
&\leq \sum_{\vec{x} \in \Omega} \left[\left(\frac{1}{2} + (M^2 + d + 1) + 2d \right) |\phi_{max}|^2 + \frac{\Lambda}{4!} |\phi_{max}|^2 \right] \\
&\leq \sum_{\vec{x} \in \Omega} \left[\left(\frac{1}{2} + M^2 + 3d + 1 + \frac{\Lambda}{4!} \right) |\phi_{max}|^2 \right] \\
&= C(M, \Lambda, d) |\Omega| |\phi_{max}|^4
\end{aligned} \tag{93}$$

where $C(M, \Lambda, d) = M^2 + 3d + 1 + \frac{\Lambda}{4!} + \frac{1}{2}$. Let $E_{max} > 0$ be the maximum allowed energy scale in our simulation. By Markov's inequality then,

$$\begin{aligned}
\Pr(H_{amp} \geq E_{max}) &\leq \frac{\langle \phi | H_{amp} | \phi \rangle}{E_{max}} \\
&\leq \frac{C(M, \Lambda, d) |\Omega| |\phi_{max}|^4}{E_{max}}
\end{aligned} \tag{94}$$

from which the statement of the theorem readily follows. \square

1. Algorithm 1 : Equal weight LCU

Here, we describe how to construct the PREP and SELECT oracles for the Φ^4 Hamiltonian H_{amp} appearing in Eq. (11), using an equal weight LCU for the operators appearing in the Hamiltonian. By ensuring that we provide the same coefficient to each of the unitaries appearing in the LCU decompositions of Φ , Φ^2 , Φ^4 and Π^2 , we essentially ensure that we have a total of only 4 families of terms each sharing the same coefficient within that family of terms. This drastically simplifies the PREP circuit, the only non-trivial part of which is to prepare a 2-qubit state, while the rest of it is composed of transversal Hadamards. We also describe how to efficiently carry out a SELECT circuit that does not require selecting each of the unitaries within a family in succession, but can rather apply all of them simultaneously using a comparator.

First, we describe the decompositions of the operators appearing in the Hamiltonian of Eq. (11). We apply Lemma 15 to obtain these decompositions. The Φ , Φ^2 and Φ^4 terms are all built from the same basic diagonal operator $\frac{\hat{\Phi}}{\Delta\Phi} = \text{Diag}(-k + 1, \dots, 0, \dots, k)$.

a. Decompositions First, for the $\hat{\Phi}$ operator, we use $L = 2k$, $n_{max} = k$, and $n_j = j - k + 1$ in Eq. (87) to get

$$\begin{aligned}
\frac{\hat{\Phi}}{\Delta\Phi} &= \frac{1}{2} \sum_{i=0}^{2k-1} U^{(i)}, \quad \text{where} \\
U^{(i)} &= - \sum_{j=0}^{i-1} |j\rangle\langle j| + \sum_{j=i}^{2k-1} |i\rangle\langle i| = \sum_{j=0}^{2k-1} [2\Theta(j-i) - 1] |j\rangle\langle j|
\end{aligned} \tag{95}$$

On any two sites a and b , the operator $-\hat{\Phi}_a \hat{\Phi}_b$ is then simply given by

$$-\frac{\hat{\Phi}_a \hat{\Phi}_b}{(\Delta\Phi)^2} = \frac{1}{4} \sum_{i_a, j_a=0}^{2k-1} \sum_{i_b, j_b=0}^{2k-1} [2\Theta(j_a - i_a) - 1] [1 - 2\Theta(j_b - i_b)] |j_a\rangle\langle j_a| \otimes |j_b\rangle\langle j_b| \tag{96}$$

For the $\hat{\Phi}^2$ term, we use $L = 2k^2$, $n_{max} = k^2$ and $n_j = (k - j - 1)^2$ in Eq. (87) to get

$$\begin{aligned}
\left(\frac{\hat{\Phi}}{\Delta\Phi} \right)^2 &= \frac{1}{2} \sum_{i=0}^{2k^2-1} U^{(i)}, \quad \text{where} \\
U^{(i)} &= \sum_{j=0}^{2k-1} [2\Theta(k^2 + (k - j - 1)^2 - i - 1) - 1] |j\rangle\langle j|
\end{aligned} \tag{97}$$

Similarly, for the $\hat{\Phi}^4$ term, we use $L = 2k$, $n_{max} = k^4$ and $n_j = (k - j - 1)^4$ in Eq. (87) to get

$$\left(\frac{\hat{\Phi}}{\Delta\Phi}\right)^4 = \frac{1}{2} \sum_{i=0}^{2k^4-1} U^{(i)}, \quad \text{where}$$

$$U^{(i)} = \sum_{j=0}^{2k-1} [2\Theta(k^4 + (k - j - 1)^4 - i - 1) - 1] |j\rangle \langle j| \quad (98)$$

b. PREP circuit As the above decompositions illustrate, we have LCU decompositions for each of the terms in H_{amp} such that all the unitaries share the same coefficients within each of 4 different families. In the language of Theorem 13, this means that we have $M = 4$ different families. These equal weight LCU decompositions lead to a much simpler PREP circuit. In particular, we only need to prepare a non-trivial $\log_2 M = 2$ qubit register whose amplitudes encode the shared coefficients of the 4 groups of terms in the Hamiltonian (11). The PREP_{*i*} (sub-PREPARE) circuits for each of these groups, which would control which of the $U^{(i)}$'s in the LCUs above would be selectively applied, then simply become a layer of transversal Hadamard gates.

In other words, the PREP circuit then simplifies to

$$\text{PREP} = \text{PREP}_F \otimes H^{\otimes \log_2 |\Omega|} \otimes H^{\otimes \log_2 2k^4} \quad (99)$$

where

$$\text{PREP}_F |0\rangle^{\otimes 2} = \alpha_{\pi^2} |00\rangle + \alpha_{\phi^2} |01\rangle + \alpha_{\phi^4} |10\rangle + \alpha_{\phi\phi} |11\rangle \quad (100)$$

where the amplitudes α_j encode the coefficients of the 4 families of terms in the Hamiltonian (11)

$$|\alpha_{\pi^2}|^2/\mathcal{N} = \frac{1}{2}, \quad |\alpha_{\phi^2}|^2 = (M^2 + d + 1)/\mathcal{N}$$

$$|\alpha_{\phi^4}|^2 = \frac{\Lambda}{4!}/\mathcal{N}, \quad |\alpha_{\phi\phi}|^2 = 2/\mathcal{N} \quad (101)$$

where $\mathcal{N} = \sum_{j \in \{\pi^2, \phi^2, \phi^4, \phi\phi\}} |\alpha_j|^2$. Note that this is not the same as the coefficient 1-norm of the Hamiltonian, which instead is given by

$$|\alpha|_1 = |\Omega| \left[\frac{k^4 \Delta_\phi^4 \Lambda}{4!} + k^2 \Delta_\phi^2 \left(M^2 + 3d + \frac{3}{2} \right) \right] \quad (102)$$

where we take $\Delta_\phi = \sqrt{2\pi/2^{n_Q}} = \sqrt{\pi/k}$ where $n_Q = \log_2 2k$ is the number of qubits required to implement the field register at a single lattice site. This choice ensures that the free part of the Hamiltonian corresponds to the usual eigenspectrum of a harmonic oscillator.

Thus, the PREP circuit will contribute at most $O(1)$ T gates, by preparing a superposition of the four non-trivial coefficients in the ϕ^4 Hamiltonian, since the LCUs for all the operators appearing in the Hamiltonian are all equal weight. We need only prepare a non-trivial 2-qubit state, with real amplitudes. It was shown by Vatan and Williams [77] that such an $SO(4)$ operator requires at most 12 RZ gates and 2 CNOTs.

We now use the result of [28] that gives us a T gate cost of $3.067 \log_2(2/\epsilon_{synth}) - 4.327$ to compile an RZ gate to target precision ϵ_{synth} . Other such cost estimates include the result of [78], which estimates that on average, the T gate cost of a synthesized/compiled unitary V ϵ_{synth} -approximating a single RZ gate U (i.e. $|U - V| \leq \epsilon_{synth}$) is $1.15 \log_2(1/\epsilon_{synth})$, while [79] estimates a worst case upper bound of $10 + 4 \log_2(1/\epsilon_{synth})$ T gates per RZ gate. Since the errors are additive, the total T gate cost increases by the number of RZ gates times the compilation cost of a single RZ gate. Thus, the total T gate cost of the PREP circuit is given by

$$\text{Count}(T)_{\text{PREP}} = 36.804 \log_2(2/\epsilon_{synth}) - 51.924 \quad (103)$$

c. Simultaneous SELECT In principle, with the simplification of the PREP circuit described above, we could selectively apply each of the unitaries in a particular LCU by controlling on the state of some register prepared in equal superposition of all bitstrings. However, this decomposition also allows us to simultaneously apply all the SELECT_{*i*} circuits corresponding to these unitaries. In particular, by making use of the comparator CMP operation, defined as

$$\text{CMP} |i\rangle |j\rangle |0\rangle = |i\rangle |j\rangle |j < i\rangle \quad (104)$$

we can apply the following lemma

Lemma 19. For integers i and j , we have

$$\text{CMP}^\dagger (\mathbb{I} \otimes \mathbb{I} \otimes Z) \text{CMP} |i\rangle |j\rangle |0\rangle = (2\Theta(j-i) - 1) |i\rangle |j\rangle |0\rangle \quad (105)$$

where the step function $\Theta(x)$ is defined in Eq. (88).

from which we readily obtain the following corollary

Corollary 20. Let $|\psi\rangle = \sum_{j=0}^{D-1} \alpha_j |j\rangle$ denote the state of the scalar field at some lattice point, and $|\overline{\mp}\rangle = \sum_{i=0}^{D-1} \frac{1}{\sqrt{D}} |i\rangle$ an equal superposition register. Then,

$$|\overline{\mp}\rangle \otimes \frac{\hat{\Phi}}{\Phi_{max}} |\psi\rangle \otimes |0\rangle = \text{CMP}^\dagger (\mathbb{I} \otimes \mathbb{I} \otimes Z) \text{CMP} |\overline{\mp}\rangle |\psi\rangle |0\rangle \quad (106)$$

where $\Phi_{max} = k\Delta\Phi$.

In this way, we obviate the need to selectively apply each of the unitaries $U^{(i)}$ appearing in Eq. (95) individually. Instead, we can apply all of them in superposition by a single application of the comparator and its inverse. This simplifies the SELECT_i circuits necessary to implement each of the terms in the Hamiltonian. Fig 3 demonstrates this circuit equivalence, and the advantage it confers. Applying each of the equally weighted unitaries individually would require $O(|\Omega|k)$ many operations controlled on the state of $O(\log |\Omega|k)$ many qubits, leading to a T gate count of $O(|\Omega|k \log |\Omega|k)$. Instead, we have $O(|\Omega|)$ many single-qubit operations controlled on the state of $O(\log |\Omega|)$ many qubits, leading to a significantly reduced T gate count of $O(|\Omega| \log |\Omega|)$ for the SELECT operation.

Note that instead of the comparator, we are also free to apply any other operator CMP' of the form

$$\text{CMP}' |i\rangle |j\rangle |\text{scratch}\rangle |0\rangle = |\Psi_{i,j,\text{scratch}}\rangle |j < i\rangle \quad (107)$$

which may significantly reduce the gate cost. An example of such an operation would be to compute the high carry bit of $\overline{j+i} = j-i$, which is 1 iff $j < i$, but leaves the state of the $|i\rangle$, $|j\rangle$ and other ancillary qubits $|\text{scratch}\rangle$ entangled. Once the appropriate phase has been extracted from the $|j < i\rangle$ qubit, we simply run the inverse of this operation. Noting also that

$$|\mathbf{x}\rangle \langle \mathbf{x}| \otimes (U^\dagger V U) + |\mathbf{x}^\perp\rangle \langle \mathbf{x}^\perp| \otimes \mathbb{I} = (\mathbb{I} \otimes U^\dagger) (|\mathbf{x}\rangle \langle \mathbf{x}| \otimes V + |\mathbf{x}^\perp\rangle \langle \mathbf{x}^\perp| \otimes \mathbb{I}) (\mathbb{I} \otimes U) \quad (108)$$

this means that the controlled version of the entire operation $H^{\otimes m} \text{CMP}'^\dagger Z_{anc} \text{CMP}' H^{\otimes m}$ simply needs to control the Z_{anc} operation. An example circuit involving the CMP' operation is depicted in Fig. 4.

We can now split up the entire SELECT circuit as

$$\text{SELECT} = \sum_{i \in \{\pi^2, \phi^2, \phi^4, \phi\phi\}} |\alpha_i\rangle \langle \alpha_i| \otimes \sum_{a=0}^{|\Omega|-1} |a\rangle \langle a| \otimes \text{SELECT}_{i,a} \quad (109)$$

where the left-most register controls the family of terms in the Hamiltonian, the middle register controls which site we apply an operator to, and finally the SELECT_i circuits applies the relevant transformations for each of the family of terms as described below, and further detailed in Appendix C 1 b.

d. $\text{SELECT}_{\phi\phi}$ term To implement the $-\Phi\Phi$ term, we can simply make use of Corollary (20) and the fact that $XZX = -Z$ to obtain

$$|\overline{\mp}\rangle \otimes \left(-\frac{\hat{\Phi}}{\Phi_{max}} \right) |\psi\rangle \otimes |0\rangle = \text{CMP}^\dagger (\mathbb{I} \otimes \mathbb{I} \otimes XZX) \text{CMP} |\overline{\mp}\rangle |\psi\rangle |0\rangle \quad (110)$$

Then, for any two lattice sites a and b , the operator $(-\hat{\Phi}_a \hat{\Phi}_b) / (\Delta\Phi)^2$ is then implemented by the product of operations given in Eqs. (20) and (110). In D spatial dimensions, a lattice site has $2D$ neighbors. Since the operation $(\text{CMP}^\dagger Z_{anc} \text{CMP})$ squares to the identity, this means we only need to apply the operation $(\text{CMP}^\dagger X_{anc} Z_{anc} X_{anc} \text{CMP})$ on each of the neighbors of a given lattice site, without applying a similar operation on the given lattice site itself. Thus, the entire sub-select unitary for this family of terms is then given by

$$\text{SELECT}_{\phi\phi,a} = \sum_{b=0}^{2D-1} |b\rangle \langle b| \otimes (\text{CMP}^\dagger Z_{anc} \text{CMP})_a (\text{CMP}^\dagger X_{anc} Z_{anc} X_{anc} \text{CMP})_b \quad (111)$$

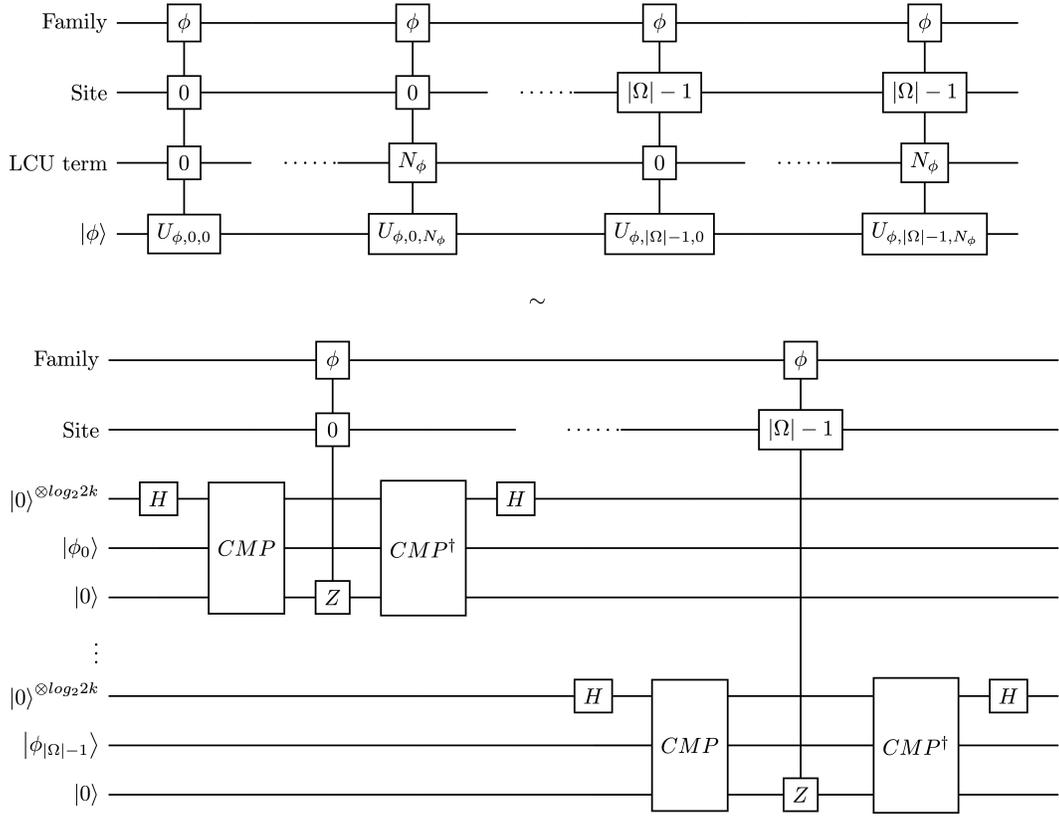


FIG. 3: Circuit equivalence demonstrating how the comparator CMP can implement the entire sum of unitaries in Eq. (95). The “family” register controls which of the 4 families of terms in the Hamiltonian we want to apply, the “site” register controls which of the lattice sites we want to apply the operator to, and the “LCU term” register controls which of the unitaries of Eq. (95) we wish to apply.

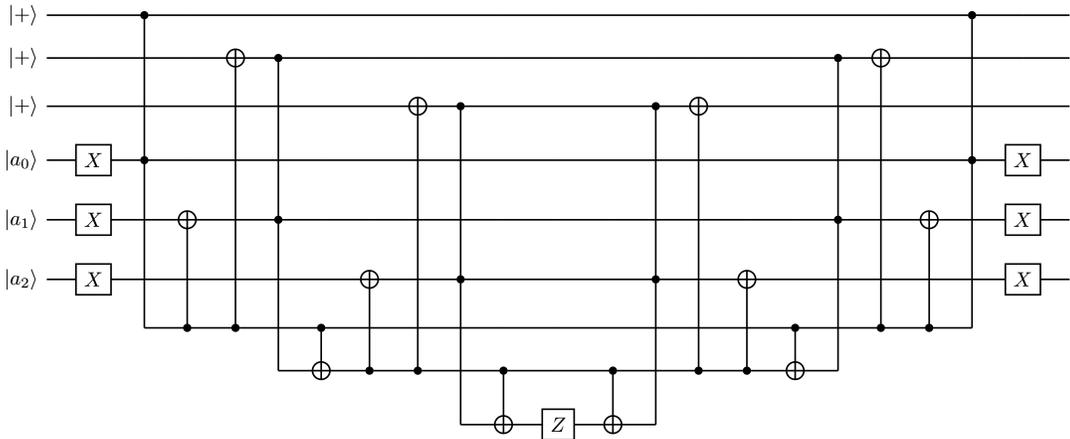


FIG. 4: An example circuit for $k = 4$ implementing the operation $\text{CMP}'^{\dagger} Z_{anc} \text{CMP}'$ using the logical AND construction of [76]. The field register $|\phi\rangle = |a_2\rangle|a_1\rangle|a_0\rangle$ is compared against an equal superposition of bitstrings using CMP' , the comparison recorded onto an ancilla using Z_{anc} , followed by the uncomputation CMP'^{\dagger} which involves measurements and post-measurement Clifford operations.

where the $|b\rangle$ register, consisting of $\log_2 2D$ qubits, controls which of the neighbors we apply the operation to. For notational simplicity, in the expression above, the subscripts a and b denote not just the respective lattice sites, but

also the associated ancilla required to implement the CMP, as well as the X_{anc} and Z_{anc} operations. As described earlier, the only operation inside the parantheses above that needs to be controlled is the Z_{anc} operation, since all other operations on either side of it are self-conjugate. Any given lattice site will receive one application of $(\text{CMP}^\dagger Z_{anc} \text{CMP})$ and $2D$ applications of $(\text{CMP}^\dagger X_{anc} Z_{anc} X_{anc} \text{CMP})$ coming from its neighbors. Between these successive applications, CMP and its inverse multiply to the identity, as do several applications of the X operator. Therefore, per lattice site, we require one application each of CMP and its conjugate, two applications of the X_{anc} operator, and $2D + 1$ applications of a controlled Z_{anc} operator, each of which is controlled on the state of $\log_2(8D|\Omega|)$ ($= 2 + \log_2|\Omega| + \log_2(2D)$) many qubits, where the factor of 2 comes from the PREP circuit, the factor of $\log_2|\Omega|$ from the lattice size, and $\log_2(2D)$ from the number of neighboring lattice sites in D spatial dimensions.

A C^n -Z gate can be implemented [76, 80] with $n - 2$ ancillas initialized in the $|0\rangle$ state, using $n - 1$ logical AND computations (and just as many uncomputations) along with 2 Hadamard gates. Each logical AND computation requires 4 T gates. Therefore, the controlled Z_{anc} operations naively require $O(D|\Omega|\log_2(D|\Omega|))$ T gates across the entire lattice, and in each of the D spatial dimensions.

A more efficient approach using unary iteration is also described in [70], which brings the cost for implementing such a SELECT operation consisting of L operations down to $4L - 4$ T gates, instead of $O(L\log_2 L)$. Briefly, this method exploits the fact that successive control patterns differ in at most a constant number of bits, so that the collective circuit can be simplified. In our case, we have $L = |\Omega|(2D + 1)$ many controlled operations to apply across the entire lattice, which incurs a T gate cost of $4L - 4 = 4|\Omega|(2D + 1) - 4$.

As described before, the CMP operations may be replaced by CMP' operations described by Eq. (107). For two n -bit numbers i and j , the operation CMP' records the Boolean value of $j < i$ onto an ancillary qubit, and requires n many logical AND computations. Each of these in turn can be implemented using 4 T gates. The CMP'^\dagger computations are simply the logical AND uncomputations that require measurements on the ancilla and classically conditioned post-measurement Clifford operations. In our case, we require the comparison of two $\log_2(2k)$ -bit numbers. Thus, the CMP' operations contribute a total T gate count of $4|\Omega|\log_2(2k)$ across the entire lattice.

The total T gate count for the $\phi\phi$ part of the Hamiltonian is therefore

$$\text{Count}(\text{T})_{\phi\phi} = 4|\Omega|(2D + \log_2 k + 2) - 4 \quad (112)$$

e. SELECT $_{\Phi^2}$ term The sub-select circuit for Φ^2 term can be constructed quite similarly. The comparator is still used to extract the relevant phase. The only difference is that now we must massage an ancillary register with some initial set of operations that we summarize as $U_{initial}^{\phi^2}$.

Specifically, given $|\psi\rangle = \sum_j c_j |j\rangle$, denoting the field register at some lattice point, and another ancillary register $|k\rangle$, we require

$$U_{initial}^{\phi^2} \left(\sum_j c_j |j\rangle \right) |k\rangle |scratch\rangle |0\rangle = |\Psi'\rangle |k^2 + (j + 1 - k)^2\rangle. \quad (113)$$

Once the value $k^2 + (j + 1 - k)^2$ has been written onto a register, we can repeat the procedure described earlier of using a comparator, extracting out the relevant phase, then undoing the comparison. In sum, for each lattice site a

$$\text{SELECT}_{\Phi^2,a} = \left(U_{initial}^{\phi^2,\dagger} \text{CMP}^\dagger Z_{anc} \text{CMP} U_{initial}^{\phi^2} \right)_a. \quad (114)$$

As shown in the Appendix , the cost of the $U_{initial}^{\phi^2}$ operation and its inverse incurs a total T gate cost of $8|\Omega| \left(4(\log_2 2k)^2 + 2\log_2 2k - 6 \right)$, where the leading term comes from multiplying two $\log_2 2k$ -bit numbers, while other terms come from other arithmetic operations, such as addition.

The CMP' operation (and its inverse), replacing the CMP operation as before, compares two $\log_2 2k^2$ -qubit numbers and costs $\log_2 2k^2$ many logical ANDs. Thus, the comparisons require $|\Omega|\log_2 2k^2$ many logical ANDs across the entire lattice, contributing to a T gate count of $4|\Omega|\log_2 2k^2$. In addition, we have $|\Omega|$ many multi-controlled Z gates to apply across the entire lattice, which incurs a cost of $4|\Omega| - 4$ T gates using the unary iteration method of [70].

In all, the total T gate count for this family of terms is given by

$$\begin{aligned} \text{Count}(\text{T})_{\phi^2} &= 8|\Omega| \left(4(\log_2 2k)^2 + 2\log_2 2k - 6 \right) + 4|\Omega| (\log_2 2k^2 + 1) - 4 \\ &= 8|\Omega| (4\log_2^2 k + 11\log_2 k + 1) - 4 \end{aligned} \quad (115)$$

f. SELECT $_{\pi^2}$ term Using the relation $\pi = \mathcal{F}^\dagger \phi \mathcal{F} \Rightarrow \pi^2 = \mathcal{F}^\dagger \phi^2 \mathcal{F}$, where \mathcal{F} denotes the discrete Fourier transform, we see that the SELECT $_{\pi^2}$ circuit is essentially the same as the SELECT $_{\phi^2}$ except that at each lattice site, we have additional (uncontrolled) $\log_2 2k$ -qubit Fourier transforms and their inverses at every lattice site. For each lattice site a , we have

$$\text{SELECT}_{\pi^2, a} = \left(\mathcal{F}^\dagger U_{initial}^{\phi^2, \dagger} \text{CMP}^\dagger Z_{anc} \text{CMP} U_{initial}^{\phi^2} \mathcal{F} \right)_a \quad (116)$$

where \mathcal{F} now represents the quantum Fourier transform (QFT) circuit. Here, we use the approximate QFT (AQFT) circuit of [71], which incurs a T gate cost of approximately

$$\text{Count}(T)_{AQFT} \approx 8n \log_2 \left(\frac{n}{\epsilon_{AQFT}} \right) + \log_2 \left(\frac{n}{\epsilon_{AQFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{n}{\epsilon_{AQFT}} \right)}{\epsilon_{AQFT}} \right) \quad (117)$$

for a target precision of ϵ_{AQFT} . One application each of the AQFT and its inverse at each lattice site incurs a total cost of $2|\Omega|$ times the expression in Eq. (117) with $n = \log_2 2k$. The total T gate cost for this family of terms is then

$$\begin{aligned} \text{Count}(T)_{\pi^2} &= \text{Count}(T)_{\phi^2} + \text{Count}(T)_{AQFT} \\ &= 8|\Omega| (4 \log_2^2 k + 11 \log_2 k + 1) - 4 + 16|\Omega| \log_2 2k \log_2 \left(\frac{\log_2 2k}{\epsilon_{AQFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{AQFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{AQFT}} \right)}{\epsilon_{AQFT}} \right) \end{aligned} \quad (118)$$

g. SELECT $_{\phi^4}$ term The sub-select circuit for the Φ^4 term can be constructed similarly to that of the Φ^2 term. As before, we apply an initial transformation to record an appropriate value to an ancillary register, to which the comparator is applied, and subsequently the appropriate phase extracted to match the entries of the signature matrices of the equal weight LCU Eq. (98).

Assuming $|\psi\rangle = \sum_j c_j |j\rangle$ denotes the field register at some lattice point, and we have another ancillary register initialized to $|k\rangle$, along with an entire $|scratch\rangle$ ancillary register to assist with various sub-routines, as well as another ancillary register initialized to $|0\rangle$, we require

$$U_{initial}^{\phi^4} \left(\sum_j c_j |j\rangle \right) |k\rangle |scratch\rangle |0\rangle = |\Psi'\rangle \left| k^4 + (j+1-k)^4 \right\rangle \quad (119)$$

Once the value $k^4 + (j+1-k)^4$ has been recorded in superposition, we can apply the usual comparator technique as described above to extract the relevant phase and produce the desired LCU. In all, we have for each lattice site a

$$\text{SELECT}_{\Phi^4, a} = \left(U_{initial}^{\phi^4, \dagger} \text{CMP}^\dagger Z_{anc} \text{CMP} U_{initial}^{\phi^4} \right)_a \quad (120)$$

As before, the $U_{initial}^{\phi^4}$ circuit requires the implementation of a few basic arithmetic sub-routines, including binary multiplication of two $O(\log k)$ -bit numbers, which contributes the dominant cost of $O(|\Omega|(\log k)^2)$ T gates. As shown in the appendix, this part of the circuit contributes a total T gate cost of $8|\Omega| (20 \log_2^2(2k) - 10 \log_2(2k) + 1)$.

The cost of the CMP' (simplifying the CMP operation), which now compares two $\log_2 2k^4$ -bit numbers, is $4|\Omega| \log_2 2k^4$. As in the case of the ϕ^2 term, we have $|\Omega|$ many multi-controlled Z gates to apply across the entire lattice, which contributes a cost of $4|\Omega| - 4$ T gates using the unary iteration method of [70].

In all, the total T gate count for this family of terms is

$$\text{Count}(T)_{\phi^4} = 8|\Omega| (\log_2^2 k - 6 \log_2 k - 7) - 4 \quad (121)$$

The explicit construction of the $U_{initial}^{\phi^4}$ circuit, as well as a detailed counting of the T gates, is provided in the Appendix.

h. Total cost for Hamiltonian simulation The constructions above create a block encoding of the Hamiltonian via $(\langle G| \otimes \mathbb{I}) U (|G\rangle \otimes \mathbb{I}) = H$ where $|G\rangle = U_{PREP} |\bar{0}\rangle$ and $U = U_{SELECT}$. Since $U_{SELECT}^2 = \mathbb{I}$, by Corollary 9 of [5], we have that the walk operator

$$W = \left(\left(2U_{PREP} |0\rangle \langle 0| U_{PREP}^\dagger - I \right) \otimes I \right) U_{SELECT} \quad (122)$$

also provides a block encoding of the Hamiltonian, i.e. $(\langle G | \otimes \mathbb{I}) W (|G\rangle \otimes \mathbb{I}) = H$ but in an $SU(2)$ invariant subspace containing $|G\rangle$, i.e. in qubitized form. It is this qubitized block encoding that we shall employ to perform phase estimation in order to subsequently estimate scattering matrix elements.

The cost of this block encoding is then given by the sum of twice the cost of the PREP oracle, the cost of the SELECT oracle, and the cost of the reflection operator $2|0\rangle\langle 0| - \mathbb{I}$. The cost of PREP circuit is given by Eq. (103), while the cost of the SELECT circuit is given by the sum of Eqs. (112), (115), (118) and (121). The cost of the reflection operator is at most logarithmic in all parameters, and is exponentially sub-dominant to the other costs, so we neglect it in our analysis. In all, we have

$$\begin{aligned} \text{Cost}(T)_W &= 72|\Omega| \log_2^2 k + 168|\Omega| \log_2 k + 8|\Omega|D - 32|\Omega| + 73.608 \log_2 (2/\epsilon_{\text{synth}}) \\ &\quad + 16|\Omega| \log_2 2k \log_2 \left(\frac{\log_2 2k}{\epsilon_{AQFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{AQFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{AQFT}} \right)}{\epsilon_{AQFT}} \right) - 119.848 \end{aligned} \quad (123)$$

In order to employ this block-encoding for Hamiltonian simulation using QSVT, we need

$$\Theta \left(|\alpha|_1 t + \frac{\log 1/\epsilon}{\log \left(e + \frac{\log(1/\epsilon)}{|\alpha|_1 t} \right)} \right) \quad (124)$$

many queries to the walk operator W , where the coefficient 1-norm is given by Eq. (102).

i. Ancilla count To implement the operations described above, we need several ancillary qubits. First, we need 2 ancillae for the 4 families of terms in the Hamiltonian, $\log_2 |\Omega|$ ancillae for the $|\Omega|$ sites on the lattice, and $\log_2 (2D)$ ancillae for the neighbors of a given lattice site. These are common across all the $\text{SELECT}_{i,a}$ operations, even though the last register is only used for the $\text{SELECT}_{\phi\phi,a}$ operation.

Second, the comparison operations can only be carried out in succession, and may thus be recycled for use at each lattice site (or lattice site-neighbor pair in the case of the $\text{SELECT}_{\phi\phi,a}$ operation). The largest comparisons we require are for the $\text{SELECT}_{\phi^4,a}$ operation. Each such comparison requires $\log_2 (2k^4)$ qubits to compare against, another $\log_2 (2k^4)$ qubits to hold temporary carry values, and 1 more qubit for the Z_{anc} or $(XZX)_{anc}$ operation. The controlled- Z operation further requires at most $\log_2 (|\Omega|D)$ many ancillary qubits.

Similarly, the ancillae involved in the various U_{initial} operations can be recycled for use across each site and across each family of terms. The dominant cost is for the ϕ^4 term, and provided in the appendix. In addition to the ancilla qubits, we also have $|\Omega| \log_2 (2k)$ many qubits to hold the values of the scalar field across the entire lattice. Adding all these counts together, we find

$$\text{Count}(\text{Qubits}) = |\Omega| \log_2 (2k) + 18 \log_2^2 k + 60 \log_2 k + \log_2 (|\Omega|D) + 29 \in O(|\Omega| \log k + \log^2 k) \quad (125)$$

2. Algorithm II : Trotterization with Z operators

In this section we describe a decomposition of the operators Φ, Φ^2, Φ^4 as a function of Z operators, thus making it amenable to Trotterization. In the next section we will use these LCU decompositions to estimate the simulation cost using QSVT or QSP. The sum of the ℓ_1 norm of the coefficients in these decompositions is referred to as the ℓ_1 norm of the decomposition or operator. This is also an upper bound on the spectral norm of the operators and is useful to bound the complexity of LCU based simulation algorithms.

a. Decomposition of Φ, Φ^2 and Φ^4 : We decompose $\frac{\Phi}{\Delta\Phi}$ as follows, as done in Eq. 89.

$$\frac{\Phi}{\Delta\Phi} = -\frac{1}{2} \sum_{j=0}^{\log_2 k} 2^j Z_j - \frac{1}{2} \mathbb{I}$$

Thus,

$$\begin{aligned} \left(\frac{\Phi}{\Delta\Phi} \right)^2 &= \left(\frac{\Phi}{\Delta\Phi} + \frac{\mathbb{I}}{2} \right)^2 - \left(\frac{\Phi}{\Delta\Phi} + \frac{\mathbb{I}}{2} \right) + \frac{\mathbb{I}}{4} \\ &= \frac{1}{6} (2^{2 \log_2 k + 1} + 1) \mathbb{I} + \sum_{j=0}^{\log_2 k - 1} \sum_{k > j}^{\log_2 k} 2^{j+k-1} Z_j Z_k + \frac{1}{2} \sum_{j=0}^{\log_2 k} 2^j Z_j \end{aligned} \quad (126)$$

and so $\left(\frac{\Phi}{\Delta\Phi}\right)^4$ can be obtained by squaring the above equation and hence can be expressed as sum of Z operators as well. The ℓ_1 norm of the decomposition of $\left(\frac{\Phi}{\Delta\Phi}\right)$, $\left(\frac{\Phi}{\Delta\Phi}\right)^2$ and $\left(\frac{\Phi}{\Delta\Phi}\right)^4$ is k , k^2 and k^4 , respectively. The detail of these calculations have been provided in Appendix C 2.

Plugging in the above LCU decomposition of the operators into Eq. 11, we obtain the following decomposition of H_{amp} . We keep in mind that we have a lattice with $|\Omega|$ vertices, where each vertex has D neighbors. Let us denote the set of edges by E_D .

$$H_{amp} = H_\pi + H_\phi \quad (127)$$

$$H_\pi = \sum_{\mathbf{x} \in \Omega} H_{\pi\mathbf{x}} := \sum_{\mathbf{x} \in \Omega} \frac{1}{2} \Pi^2(\mathbf{x}) = \sum_{\mathbf{x} \in \Omega} \mathcal{F} \left(\gamma_0 \mathbb{I} + \sum_j \gamma_j Z_j + \sum_{j,k} \gamma_{jk} Z_j Z_k \right) \mathcal{F}^\dagger \quad (128)$$

$$H_{\phi'} = \sum_{\mathbf{x} \in \Omega} \left(\mathbb{I} + \sum_j \alpha_j Z_j + \sum_{j,k} \alpha_{jk} Z_j Z_k + \sum_{j,k,l} \alpha_{jkl} Z_j Z_k Z_l + \sum_{j,k,l,m} \alpha_{jklm} Z_j Z_k Z_l Z_m \right) \quad (129)$$

$$H_{\phi''} = \sum_{\substack{\mathbf{x}, \mathbf{x}' \\ (\mathbf{x}, \mathbf{x}') \in E_D}} \sum_{j, j'} \beta_{jj'} (Z_j)_\mathbf{x} (Z_{j'})_{\mathbf{x}'} \quad (130)$$

$$H_\phi = H_{\phi'} + H_{\phi''} \quad (131)$$

In the above set of equations the coefficients $\gamma_0, \gamma_j, \gamma_{jk}, \alpha_j, \alpha_{jk}, \alpha_{jkl}, \alpha_{jklm}, \beta_{jj'}$ are determined by the LCU decomposition of the operators as well as Eq. 11. We ignore the identity term, since if $H_{amp} = H_{amp'} + \beta_0 \mathbb{I}$, where β_0 is a constant, then $e^{-iH_{amp}'\tau} \propto e^{-iH_{amp}\tau}$ upto some global phase. So the gate complexity to implement the exponentials is the same.

We have two non-commuting components, H_ϕ and H_π . We first analyze the gate complexity to implement $e^{-iH_\phi\tau}$ and $e^{-iH_\pi\tau}$, where τ is one Trotter step. As mentioned, Trotterization becomes quite straight-forward with this decomposition. We require 1 R_z gate and few CNOT gates (to compute the parity) for each exponentiated Z operator. Thus we need at most

$$|\Omega| \left(\binom{\log_2 k + 1}{1} + \binom{\log_2 k + 1}{2} + \binom{\log_2 k + 1}{3} + \binom{\log_2 k + 1}{4} \right) + E_D (\log_2 k + 1)^2 \quad (132)$$

number of R_z gates and at most

$$2|\Omega| \left(\binom{\log_2 k + 1}{2} + 2 \binom{\log_2 k + 1}{3} + 3 \binom{\log_2 k + 1}{4} \right) + 2E_D (\log_2 k + 1)^2 \quad (133)$$

number of CNOT gates. We observe that in order to implement the required exponential of the operators we require a number of CNOT that will realize the necessary parities i.e. the XOR of all combinations of 2, 3 and 4 qubits. Such a circuit is usually referred to as a 'parity network'. The R_z gates are placed at that point of the circuit where the corresponding parity is realized. The number of CNOT gates can be optimized using algorithms like [81, 82]. Additionally we require $2|\Omega|$ number of $\log_2 2k$ -qubit QFT. If we use the approximate QFT (AQFT) circuit of [71] then we require

$$2|\Omega| \left(8(\log_2 2k) \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right)}{\epsilon_{QFT}} \right) \right) \quad (134)$$

number of additional T gates.

Now we bound the error using the 2^{nd} order Trotter formula [67, 68], in a similar manner as discussed in Section

IV C. We need to calculate the second level nested commutators. Here, for convenience we fragment H_ϕ as follows.

$$H_\phi := H_{1\phi} + H_{2\phi} + H_{3\phi} \quad (135)$$

$$H_{1\phi} := \sum_{\mathbf{x} \in \Omega} H_{1\phi\mathbf{x}} = \sum_{\mathbf{x} \in \Omega} \frac{1}{2} (M^2 + d + 1) \Phi^2(\mathbf{x}) \quad (136)$$

$$H_{2\phi} := \sum_{\mathbf{x} \in \Omega} H_{2\phi\mathbf{x}} = \sum_{\mathbf{x} \in \Omega} \frac{\Lambda}{24} \Phi^4(\mathbf{x}) \quad (137)$$

$$H_{3\phi} := \sum_{\mathbf{x} \in \Omega} H_{3\phi\mathbf{x}} = \sum_{\mathbf{x} \in \Omega} \left(-2 \sum_{\substack{\mathbf{x}', \\ (\mathbf{x}, \mathbf{x}') \in E_D}} \Phi(\mathbf{x}) \Phi(\mathbf{x}') \right) := \sum_{\mathbf{x} \in \Omega} \sum_{\substack{\mathbf{x}', \\ (\mathbf{x}, \mathbf{x}') \in E_D}} H_{3\phi\mathbf{x}\mathbf{x}'} \quad (138)$$

We observe the following commutation relations between $H_{\pi\mathbf{x}}$ (Eq. 128) and the Hamiltonian fragments in the above equations.

$$\begin{aligned} [H_{1\phi\mathbf{x}}, H_{2\phi\mathbf{x}'}], [H_{1\phi\mathbf{x}}, H_{3\phi\mathbf{x}'}], [H_{3\phi\mathbf{x}}, H_{2\phi\mathbf{x}'}] &= 0 && \text{for all } \mathbf{x}, \mathbf{x}', \\ [H_{\pi\mathbf{x}}, H_{1\phi\mathbf{x}'}], [H_{\pi\mathbf{x}}, H_{2\phi\mathbf{x}'}] &\neq 0 && \text{if and only if } \mathbf{x} = \mathbf{x}', \\ \text{and } [H_{\pi\mathbf{x}}, H_{3\phi\mathbf{x}'}] &\neq 0 && \text{if and only if } (\mathbf{x}, \mathbf{x}') \in E_D. \end{aligned} \quad (139)$$

Now assuming $H'_{1\phi} = \sum_{\mathbf{x} \in \Omega} H'_{1\phi\mathbf{x}} = \sum_{\mathbf{x} \in \Omega} H_{1\phi\mathbf{x}} + H_{2\phi\mathbf{x}} = H_{1\phi} + H_{2\phi}$, we have

$$\begin{aligned} \tilde{\alpha}_{comm} &\leq (\|[[H_\pi, H'_{1\phi}], H_\pi] + [[H_\pi, H'_{1\phi}], H'_{1\phi}]\|) \\ &\quad + (\|[[H_\pi, H'_{1\phi}], H_{3\phi}] + [[H_\pi, H_{3\phi}], H_\pi] + [[H_\pi, H_{3\phi}], H'_{1\phi}] + [[H_\pi, H_{3\phi}], H_{3\phi}]\|) \\ &:= S_1 + S_2. \end{aligned} \quad (140)$$

For the first sum, we use Eq. (139) and observe that Eq. (57) can be applied along with triangle inequality in order to have the following bound.

$$\begin{aligned} S_1 &= \sum_{\mathbf{x} \in \Omega} (\|[[H_{\pi\mathbf{x}}, H'_{1\phi\mathbf{x}}], H_{\pi\mathbf{x}}]\| + \|[[H_{\pi\mathbf{x}}, H'_{1\phi\mathbf{x}}], H'_{1\phi\mathbf{x}}]\|) \\ &\leq \sum_{\mathbf{x} \in \Omega} (\|H_{\pi\mathbf{x}}\| (\|H'_{1\phi\mathbf{x}}\| + \|H_{\pi\mathbf{x}}\|)) \leq |\Omega| (\|H_{\pi\mathbf{x}}\| (\|H'_{1\phi\mathbf{x}}\| + \|H_{\pi\mathbf{x}}\|)) \\ &\leq \sum_{\mathbf{x} \in \Omega} \left(\frac{M^2 + d + 1}{4} \|\Pi^2(\mathbf{x}), \Phi^2(\mathbf{x})\| + \frac{\Lambda}{48} \|\Pi^2(\mathbf{x}), \Phi^4(\mathbf{x})\| \right) \left(\frac{1}{2} (M^2 + d + 1) \|\Phi^2\| + \frac{\Lambda}{24} \|\Phi^4\| + \frac{1}{2} \|\Pi^2\| \right) \\ &\leq |\Omega| \left(\frac{M^2 + d + 1}{2} \|\Pi^2(\mathbf{x})\| \|\Phi^2(\mathbf{x})\| + \frac{\Lambda}{24} \|\Pi^2(\mathbf{x})\| \|\Phi^4(\mathbf{x})\| \right) \left(\frac{1}{2} (M^2 + d + 1) \|\Phi^2\| + \frac{\Lambda}{24} \|\Phi^4\| + \frac{1}{2} \|\Pi^2\| \right) \end{aligned} \quad (141)$$

For the second sum S_2 we again use Eq. (139) and triangle inequality and obtain the following bound.

$$\begin{aligned} S_2 &\leq \sum_{\mathbf{x} \in \Omega} \sum_{\substack{\mathbf{x}', \\ (\mathbf{x}, \mathbf{x}') \in E_D \text{ or } (\mathbf{x}', \mathbf{x}'') \in E_D}} \sum_{\mathbf{x}'': (\mathbf{x}, \mathbf{x}'') \in E_D} (\|[[H_{\pi\mathbf{x}}, H'_{1\phi\mathbf{x}}], H_{3\phi\mathbf{x}\mathbf{x}'}]\| + \|[[H_{\pi\mathbf{x}}, H_{3\phi\mathbf{x}\mathbf{x}'}], H_{\pi\mathbf{x}''}]\| + \|[[H_{\pi\mathbf{x}}, H_{3\phi\mathbf{x}\mathbf{x}'}], H'_{1\phi\mathbf{x}''}]\|) \\ &\quad + (\|[[H_{\pi\mathbf{x}}, H_{3\phi\mathbf{x}\mathbf{x}'}], H_{3\phi\mathbf{x}\mathbf{x}''}]\| + \|[[H_{\pi\mathbf{x}}, H_{3\phi\mathbf{x}\mathbf{x}'}], H_{3\phi\mathbf{x}'\mathbf{x}''}]\|) \\ &\leq |\Omega| d^2 (\|H_{\pi\mathbf{x}}\| (\|H'_{1\phi\mathbf{x}}\| \|H_{3\phi\mathbf{x}\mathbf{x}'}\| + \|H_{\pi\mathbf{x}}\| \|H_{3\phi\mathbf{x}\mathbf{x}'}\|) (\|H_{\pi\mathbf{x}''}\| + \|H'_{1\phi\mathbf{x}''}\| + \|H_{3\phi\mathbf{x}\mathbf{x}''}\| + \|H_{3\phi\mathbf{x}'\mathbf{x}''}\|)) \end{aligned} \quad (142)$$

We already mentioned that the ℓ_1 norm for the decomposition of $(\frac{\Phi}{\Delta\Phi})$, $(\frac{\Phi}{\Delta\Phi})^2$ and $(\frac{\Phi}{\Delta\Phi})^4$ is k , k^2 and k^4 , respectively (Appendix C 2). Also $\|\Pi\| = \|\Phi\|$ since they are Fourier conjugates and so using triangle inequality we have,

$$\begin{aligned} \|H'_{1\phi\mathbf{x}}\| &\leq \frac{1}{2} (M^2 + d + 1) \|\Phi^2(\mathbf{x})\| + \frac{\Lambda}{24} \|\Phi^4(\mathbf{x})\| \leq \frac{1}{2} (M^2 + d + 1) k^2 \Delta^2 + \frac{\Lambda}{24} k^4 \Delta^4; \\ \|H_{3\phi\mathbf{x}\mathbf{x}'}\| &\leq 2 \|\Phi(\mathbf{x})\| \|\Phi(\mathbf{x}')\| \leq 2k^2 \Delta^2; \quad \|H_{\pi\mathbf{x}}\| \leq \frac{1}{2} \|\Pi^2(\mathbf{x})\| \leq \frac{1}{2} k^2 \Delta^2; \\ \|H_{\pi\mathbf{x}}, H'_{1\phi\mathbf{x}}\| &\leq 2 \|H_{\pi\mathbf{x}}\| \|H'_{1\phi\mathbf{x}}\| \leq \frac{1}{2} (M^2 + d + 1) k^4 \Delta^4 + \frac{\Lambda}{24} k^6 \Delta^6; \\ \|H_{\pi\mathbf{x}}, H_{3\phi\mathbf{x}\mathbf{x}'}\| &\leq 2 \|H_{\pi\mathbf{x}}\| \|H_{3\phi\mathbf{x}\mathbf{x}'}\| \leq 2k^4 \Delta^4. \end{aligned} \quad (143)$$

Thus, from Eq. (141) and 142 we have

$$\begin{aligned}
S_1 &\leq |\Omega| \left(\frac{1}{2}(M^2 + d + 1)k^4\Delta^4 + \frac{\Lambda}{24}k^6\Delta^6 \right) \left(\frac{1}{2}(M^2 + d + 2)k^2\Delta^2 + \frac{\Lambda}{24}k^4\Delta^4 \right) \\
&= |\Omega| \left(\frac{1}{4}(M^2 + d + 1)(M^2 + d + 2)k^6\Delta^6 + \frac{\Lambda}{48}(2M^2 + 2d + 3)k^8\Delta^8 + \frac{\Lambda^2}{576}k^{10}\Delta^{10} \right), \\
S_2 &\leq |\Omega|d^2 \left(\left(\frac{1}{2}(M^2 + d + 1)k^4\Delta^4 + \frac{\Lambda}{24}k^6\Delta^6 \right) 2k^2\Delta^2 \right. \\
&\quad \left. + 2k^4\Delta^4 \left(\frac{1}{2}k^2\Delta^2 + \frac{1}{2}(M^2 + d + 1)k^2\Delta^2 + \frac{\Lambda}{24}k^4\Delta^4 + 4k^2\Delta^2 \right) \right) \\
&= |\Omega|d^2 \left((2M^2 + 2d + 11)k^6\Delta^6 + \frac{\Lambda}{6}k^8\Delta^8 \right),
\end{aligned}$$

and from Eq. (140) we have

$$\begin{aligned}
\tilde{\alpha}_{comm} &\leq |\Omega| \left(\frac{\Lambda^2}{576}k^{10}\Delta^{10} + \frac{\Lambda}{48}(2M^2 + 8d^2 + 2d + 3)k^8\Delta^8 + \right. \\
&\quad \left. \left(\frac{1}{4}(M^2 + d + 1)(M^2 + d + 2) + d^2(2M^2 + 2d + 11) \right) k^6\Delta^6 \right) \\
&\in O \left[|\Omega| (\Lambda^2 k^{10} \Delta^{10} + \Lambda M^2 k^8 \Delta^8) \right].
\end{aligned} \tag{144}$$

for fixed spatial dimensionality d . We can prove the following bound on the T-gate complexity, using similar reasoning as in Theorem 12.

Theorem 21. *Given an eigenstate $|\psi\rangle$ of H such that $H_{amp}|\psi\rangle = E|\psi\rangle$, where H_{amp} is the amplitude basis Hamiltonian, as stated in Eq. (127), then there exists a quantum algorithm that outputs with probability greater than $2/3$ a value \hat{E} such that $|\hat{E} - E| \leq \epsilon_E$, using a number of T gates that scales as*

$$O \left(\frac{|\Omega|^{3/2} \sqrt{\Lambda^2 k^5 + \Lambda M^2 k^4} \log^4(k)}{\epsilon_E^{3/2}} \log \left(\frac{|\Omega| k \log k}{\epsilon_E} (\Lambda^2 k + \Lambda M^2) \right) \right)$$

T gates and $\mathcal{O}(|\Omega| \log_2(2k))$ qubits, plus an additional number of ancillary qubits required for phase estimation as detailed in Lemma 36.

Here the log argument is derived from $\epsilon_r = \frac{\sqrt{2}\epsilon_E}{8N_r} \sqrt{\frac{\epsilon_E}{2^{3/2}\tilde{\alpha}_{comm}}}$, $N_r \in \mathcal{O}(|\Omega| \log_2^4(2k))$ and

$\tilde{\alpha}_{comm} \in \mathcal{O}(|\Omega| (\Lambda^2 k^{10} \Delta^{10} + \Lambda M^2 k^8 \Delta^8))$, and $\Delta = \sqrt{\pi/k}$.

Proof. Suppose we allocate ϵ_r as an upper bound on the permitted synthesis error per R_z gate. From Eq. (132) we find that the total number of R_z required for each Trotter step is at most

$$N_r := |\Omega| \left(\binom{\log_2 k + 1}{1} + \binom{\log_2 k + 1}{2} + \binom{\log_2 k + 1}{3} + \binom{\log_2 k + 1}{4} \right) + E_D(\log_2 k + 1)^2$$

and so using the T-count estimate in [32] gives the expected number of T-gates from rotations to be at most

$$N_{T/R_z} \leq N_r (3.067 \log_2(1/\epsilon_r) - 4.327).$$

We also require the following number of additional T-gates due to the QFT performed. Since we are using approximate QFT, so we allocate an error of ϵ_{QFT} due to this. Thus from Eq. (134) we require the following additional number of T-gates.

$$N_+ := 2|\Omega| \left(8(\log_2 2k) \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right)}{\epsilon_{QFT}} \right) \right).$$

So, the total number of T-gates per Trotter step is

$$\mathcal{G}_T = N_{T/R_z} + N_+ \in O(|\Omega| \log^4(2k) \log(1/\epsilon_r)).$$

Using Lemmas 10 and 11 and Eq. (144), the total number of T-gates required to achieve an eigenstate within total error ϵ_E is at most

$$\mathcal{G}_T \cdot 2^m \leq \frac{\pi^2 \sqrt{\tilde{\alpha}_{\text{comm}}}}{\epsilon_E^{3/2}} \mathcal{G}_T \in \mathcal{O} \left(\frac{|\Omega|^{3/2} \sqrt{\Lambda^2 k^5 + \Lambda M^2 k^4} \log^4(2k)}{\epsilon_E^{3/2}} \log(1/\epsilon_r) \right)$$

where we have used $\Delta = \sqrt{\pi/k}$. The total synthesis error due to approximation of the rotation gates is $\epsilon_{\text{synth}} = \epsilon_r \cdot N_r + \epsilon_{\text{QFT}} \geq \epsilon_r N_r$, where the value of ϵ_{synth} is given in Eq. 65. Plugging in the values of the timestep τ and ϵ_θ from Eq. (67) and (66), respectively we obtain the following bound on ϵ_r in order to ensure that the final error in the estimate of the energy is at most ϵ_E .

$$\epsilon_r \leq \frac{\sqrt{2}\epsilon_E \tau}{8N_r} = \frac{\sqrt{2}\epsilon_E}{8N_r} \sqrt{\frac{\epsilon_E}{2^{3/2} \tilde{\alpha}_{\text{comm}}}}$$

Note that we get a slightly different pre-factor if we also account for the errors arising due to the use of the approximate QFT, as detailed in Lemma 36, but this does not change the asymptotic scaling derived here. Using the above, we obtain

$$\log \left(\frac{1}{\epsilon_r} \right) \in \mathcal{O} \left(\log \left(\frac{|\Omega| k \log k}{\epsilon_E} (\Lambda^2 k + \Lambda M^2) \right) \right), \quad (145)$$

hence proving the theorem. \square

3. Algorithm IIIa : LCU with Z operators

In this section we discuss an approach to simulate H_{amp} with qubitization, using Eq. 89 that expresses ϕ as sum of Z operators. Then we repeatedly square it to obtain an LCU decomposition for Φ^2 and Φ^4 , as done in previous subsection (Eq. 126). We group the terms as follows.

$$H_{1\mathbf{x}} = \sum_j \alpha_j Z_j + \sum_{j,k} \alpha_{jk} Z_j Z_k + \mathcal{F} \left(\sum_j \gamma_j Z_j + \sum_{j,k} \gamma_{jk} Z_j Z_k \right) \mathcal{F}^\dagger \quad (146)$$

$$H_{2\mathbf{x}} = \sum_{j,k,l} \alpha_{jkl} Z_j Z_k Z_l + \sum_{j,k,l,m} \alpha_{jklm} Z_j Z_k Z_l Z_m \quad (147)$$

$$H_{12} = \sum_{\mathbf{x} \in \Omega} H_{1\mathbf{x}} + H_{2\mathbf{x}} \quad (148)$$

$$H_{3\mathbf{x}\mathbf{x}'} = \sum_{j,j'} \beta_{jj'} (Z_j)_{\mathbf{x}} (Z_{j'})_{\mathbf{x}'} \quad (149)$$

$$H_3 = \sum_{(\mathbf{x}, \mathbf{x}') \in E_D} H_{3\mathbf{x}\mathbf{x}'} \quad (150)$$

$$H'_{\text{amp}} = H_{12} + H_3 \quad (151)$$

Let $L_1 = \binom{\log_2 k + 1}{1}$, $L_2 = \binom{\log_2 k + 1}{2}$, $L_3 = \binom{\log_2 k + 1}{3}$, $L_4 = \binom{\log_2 k + 1}{4}$ and $L_5 = (\log_2 k + 1)^2$. In the following theorem we summarize the number of gates required to block encode H'_{amp} by repeatedly applying Theorem 13.

Theorem 22. *Let H'_{amp} be the Hamiltonian defined in Eq. 151 and $\|H'_{\text{amp}}\|$ be the ℓ_1 norm of the coefficients defined in its decomposition (Eq. 146-150). Then it is possible to have a $(\|H'_{\text{amp}}\|, \cdot, 0)$ block-encoding of H'_{amp} with $\mathcal{O}(|\Omega| \log k)$ qubits, using the following number of rotation gates*

$$N_r \in \mathcal{O}(\log^4 k),$$

and the following number of additional T gates (that are not in the decomposition of rotation gates).

$$N_t \in \mathcal{O}(|\Omega| \log^4 k).$$

The proofs and more detailed explanations have been given in Appendix C 2. We give a very concise summary of the complete block-encoding procedure here. Let $\|H_{1\mathbf{x}}\| = \sum_j (|\alpha_j| + |\gamma_j|) + \sum_{j,k} (|\alpha_{jk}| + |\gamma_{jk}|)$, $\|H_{2\mathbf{x}}\| = \sum_{j,k,l} |\alpha_{jkl}| + \sum_{j,k,l,m} |\alpha_{jklm}|$ and $\|H_{3\mathbf{xx}'}\| = \sum_{j,j'} |\beta_{jj'}|$. We first design PREP (Eq. C61, C63, C69) and SELECT (Eq. C62, C64, C70) procedures to obtain $(\|H_{1\mathbf{x}}\|, \cdot, 0)$, $(\|H_{2\mathbf{x}}\|, \cdot, 0)$, $(\|H_{3\mathbf{xx}'}\|, \cdot, 0)$ block-encoding of $H_{1\mathbf{x}}$, $H_{2\mathbf{x}}$ and $H_{3\mathbf{xx}'}$, respectively.

Here we briefly describe the block encoding of $H_{1\bar{x}}$. We assume a bijective map between $j' = (j, k)$ to some integer in $[L_1 + 1, L_1 + L_2]$. The ancilla preparation sub-routine is as follows.

$$\begin{aligned} \text{PREP}_{1\bar{x}} |0\rangle^{1+\log_2(L_1+L_2)} &= \frac{1}{\mathcal{N}_{1\bar{x}}} \left(\sum_{j=1}^{L_1} \sqrt{\alpha_j} |j, 0\rangle + \sum_{j'=L_1+1}^{L_1+L_2} \sqrt{\alpha_{j'}} |j', 0\rangle + \sum_{j=1}^{L_1} \sqrt{\gamma_j} |j, 1\rangle \right. \\ &\quad \left. + \sum_{j'=L_1+1}^{L_1+L_2} \sqrt{\gamma_{j'}} |j', 0\rangle \right) \quad [\mathcal{N}_{1\bar{x}} \text{ is normalization constant.}] \end{aligned}$$

The last qubit is used to select the QFT. We require $1 + \log_2(L_1 + L_2)$ qubits. For the state preparation we require $1 + \log_2(L_1 + L_2)$ H, $4(L_1 + L_2) - 2$ rotation gates and $4(L_1 + L_2) + 3\log_2(L_1 + L_2) - 4$ CNOT.

The select sub-routine does the following.

$$\begin{aligned} \text{SELECT}_{1\bar{x}} |j, 0\rangle |\psi\rangle &\rightarrow |j, 0\rangle Z_j |\psi\rangle \\ \text{SELECT}_{1\bar{x}} |j, 1\rangle |\psi\rangle &\rightarrow |j, 1\rangle \mathcal{F} Z_j \mathcal{F}^\dagger |\psi\rangle \end{aligned}$$

If $j > L_1 + 1$ then we apply two Z gates depending on the mapping. The last qubit is used to selectively apply the pairs of $(\log_2 k + 1)$ -qubit QFT. We require $L_1 + L_2$ compute-uncompute pairs of $C^{\log_2(L_1+L_2)} X$ gates, which can be synthesized efficiently using Theorem 14 [27]. Assuming equal partitioning into 2 groups and using the constructions in [75, 76] we require at most

$$4\sqrt{L_1 + L_2} (\log_2(L_1 + L_2) - 2) + 4(L_1 + L_2)$$

T gates and

$$\sqrt{L_1 + L_2} (4\log_2(L_1 + L_2) - 6) + 5(L_1 + L_2)$$

CNOT gates. Additionally we require $(\log_2 k + 1) + \frac{\log_2 k (\log_2 k + 1)}{2} = \frac{(\log_2 k + 1)(\log_2 k + 2)}{2}$ number of CZ gates and two $(\log_2 k + 1)$ -qubit QFT. To implement the QFTs (approximately) [71] we require $O\left((\log_2 2k) \log_2\left(\frac{\log_2 2k}{\epsilon}\right)\right)$ T gates and almost an equal number of CNOT gates. With similar procedures we can obtain the block encoding of $H_{2\mathbf{x}}$ and $H_{3\mathbf{xx}'}$.

Next, using the recursive block encoding Theorem 13 we obtain a $(|\Omega|(\|H_{1\mathbf{x}}\| + \|H_{2\mathbf{x}}\|), \cdot, 0)$ and $(|E_D| \|H_{3\mathbf{xx}'}\|, \cdot, 0)$ block-encoding of H_{12} and H_3 , respectively. We can block encode $H_{1\bar{x}} + H_{2\bar{x}}$ using ancilla preparation sub-routine that has 1 H and 2 rotation gates. The unitary selection sub-routine adds an extra control to each unitary. For H_{12} we prepare an equal superposition of $\log_2 |\Omega|$ qubits, using $\log_2 |\Omega|$ H gates and use these to select an ancilla of each subspace. The rest of the operations of $H_{1\mathbf{x}}$ and $H_{2\mathbf{x}}$ in each subspace are controlled on this. For H_3 we prepare $\log_2 |E_D|$ qubits in equal superposition, using $\log_2 |E_D|$ H gates. We use these to select two sub-spaces. Specifically, each superimposed state selects an ancilla. From this ancilla we use two CNOTs to select an ancilla in each of the two corresponding sub-spaces. The rest of the operations of $H_{3\mathbf{xx}'}$ in pair of subspaces are controlled on these ancillae. Finally, we again apply Theorem 13 in order to obtain a $(\|H'_{amp}\|, \cdot, 0)$ block-encoding of H'_{amp} , where $\|H'_{amp}\| = \|H_{12}\| + \|H_3\|$ is the sum of the coefficients in the LCU decomposition of $H_{amp'}$ (Eq. 146-150). We prepare 1 qubit in equal superposition, using 1 H gate. Controlled on this, we implement the operations in H_{12} and H_3 .

More detail explanations and gate count estimates have been provided in Appendix C 2, and summarized in the statement of Theorem 22. The following lemma gives a bound on the ℓ_1 norm of $H_{amp'}$.

Lemma 23.

$$\begin{aligned} \|H'_{amp}\| &\leq |\Omega| \left(\frac{\lambda \Delta^4}{27} k^4 + k^2 \left(\left(\frac{M^2 + 7d + 1}{3} \right) \Delta^2 - 0.048611 \lambda \Delta^4 \right) + k (-3d \Delta^2 + 0.03125 \lambda \Delta^4) \right. \\ &\quad \left. + \Delta^2 \left(\frac{-M^2 + 8d - 4}{6} \right) - 0.0081019 \lambda \Delta^4 \right) \end{aligned} \quad (152)$$

The detail proof has been provided in Appendix C 2 a. From [21], we require

$$\mathcal{R} \in O \left(\|H'_{amp}\|t + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)} \right) \quad (153)$$

calls to the block encoding of $\frac{H'_{amp}}{\|H'_{amp}\|}$ in order to implement an ϵ -precise block encoding of $e^{-iH'_{amp}t}$. Thus the total gate complexity is obtained by multiplying \mathcal{R} with the number of gates obtained in Theorem 22.

4. Algorithm IIIb : LCU with binary decomposition of integers

In this section we discuss a more compact decomposition of the field operators. The central idea stems from the fact that by using binary representation we can express an integer diagonal matrix as a sum of $O(\log k)$ number of signature matrices, as stated in Lemma 16. In our case Φ^2 and Φ^4 are diagonal matrices, consisting of 2^{nd} and 4^{th} power of consecutive integers, respectively. If ζ' is the maximum number of bits required to express the highest integer, then we can express any other integer $n = (b_{\zeta'}, \dots, b_1) = \sum_{j=1}^{\zeta'} b_j 2^{j-1}$. For smaller integers we append leading zeros. The leftmost bit $b_{\zeta'}$ is referred to as the most significant bit, while the rightmost bit is referred to as the least significant bit. The j^{th} signature matrix is obtained by taking the j^{th} bit in the binary expansion of each diagonal integer, replacing 0s by 1 and 1s by -1.

The circuit complexity of the SELECT circuits can be bounded by the sum of gates, qubits, etc required to implement the signature matrices. In order to design efficient circuits for each signature matrix we exploit their structure, which is obtained from the binary decomposition of integers. We first prove the following.

Lemma 24. *Suppose n is a positive integer and $(b_m, b_{m-1}, \dots, b_1)$ is its binary expansion. Then,*

$$\begin{aligned} b_\ell &= 0 & \text{if} & \quad n = 2^\ell k, 2^\ell k + 1, \dots, 2^\ell k + 2^{\ell-1} - 1 \\ &= 1 & \text{if} & \quad n = 2^\ell k + 2^{\ell-1}, 2^\ell k + 2^{\ell-1} + 1, \dots, 2^\ell k + 2^\ell - 1 \end{aligned}$$

where k is a non-negative integer.

Proof. We have

$$\begin{aligned} n &= b_m 2^{m-1} + b_{m-1} 2^{m-2} + \dots + b_{\ell+1} 2^\ell + b_\ell 2^{\ell-1} + b_{\ell-1} 2^{\ell-2} + \dots + b_1 \\ &= 2^\ell (b_m 2^{m-1-\ell} + b_{m-1} 2^{m-2-\ell} + \dots + b_{\ell+1}) + b_\ell 2^{\ell-1} + (b_{\ell-1} 2^{\ell-2} + \dots + b_1) \\ &:= 2^\ell S_R + b_\ell 2^{\ell-1} + S_L. \end{aligned}$$

S_R is any non-negative integer which we denote k . S_L is also a positive integer whose minimum value is 0 when $(b_{\ell-1}, \dots, b_1) = (0, \dots, 0)$ and the maximum value is

$$2^{\ell-2} + 2^{\ell-3} + \dots + 2 + 1 = 2^{\ell-1} - 1,$$

when $(b_{\ell-1}, \dots, b_1) = (1, \dots, 1)$. It follows that if $b_\ell = 0$ then n can take any value between $2^\ell k$ to $2^\ell k + 2^{\ell-1} - 1$, while if $b_\ell = 1$ then the value of n ranges from $2^\ell k + 2^{\ell-1}$ to $2^\ell k + 2^{\ell-1} + 2^{\ell-1} - 1 = 2^\ell k + 2^\ell - 1$. \square

Lemma 25. (a) *Let n be an integer and (b_m, \dots, b_1) be the binary decomposition of n^2 . Then $b_1 = 0$ for even n and 1 for odd n and $b_2 = 0$. For $\ell > 2$, $b_\ell = 1$ if and only if $n = 2^{\ell-1}j + j'$, where j, j' are integers such that $1 \leq j' \leq 2^{\ell-1} - 1$.*

(b) *Let n be an integer and (b_m, \dots, b_1) be the binary decomposition of n^4 . Then $b_1 = 0$ for even n and 1 for odd n and $b_2 = b_3 = b_4 = 0$. For $\ell \neq 2, 3, 4$, $b_\ell = 1$ if and only if $n = 2^{\ell-2}j + j'$, where j, j' are integers such that $1 \leq j' \leq 2^{\ell-2} - 1$.*

Proof. (a) The inference about b_1 is easy to deduce. Next, we observe that for any integer n ,

$$(2n)^2 = 4n^2, \quad \text{and} \quad (2n+1)^2 = 4n^2 + 4n + 1 = 4n(n+1) + 1. \quad (154)$$

From Lemma 24 we know that -1 occurs whenever j^2 is of the form $4k+2$ and $4k+3$ and there does not exist any integer whose square can be expressed in this form, as evident from Equation 154.

Suppose $n = 2^{\ell-1}j + j'$. Then, squaring we have the following.

$$\begin{aligned} n^2 &= (2^{\ell-1}j + j')^2 = 2^{2\ell-2}j^2 + 2^\ell jj' + j'^2 \\ &= 2^\ell (2^{\ell-2}j^2 + jj') + j'^2 \end{aligned}$$

If $1 \leq j' \leq 2^{\ell-1} - 1$ then there exists at least one value of j' such that $j'^2 \pmod{2^\ell} \geq 2^{\ell-1}$. From Lemma 24 it follows that $b_\ell = 1$.

In the other direction, let $b_\ell = 1$. Then from Lemma 24 $n^2 = 2^\ell t + t'$, where t is a non-negative integer and $2^{\ell-1} \leq t' \leq 2^\ell - 1$. Let a_1, a_2 be positive integers such that $n = a_1 + a_2$. Then,

$$\begin{aligned} n^2 = 2^\ell t + t' &= (a_1 + a_2)^2 = a_1^2 + a_2^2 + 2a_1a_2 \\ &= 2a_1 \left(\frac{a_1}{2} + a_2 \right) + a_2^2. \end{aligned}$$

The above equality is satisfied if $2a_1 = 2^\ell a'_1$, for some integer a'_1 and $a_2^2 = t'$. This implies $a_1 = 2^{\ell-1}a'_1$ and definitely $a_2 < 2^{\ell-1}$. Thus n is of the desired form as stated in the statement of the lemma. Even if we simply equate powers of 2 i.e. $2a_1a_2 = 2^\ell t$, then we arrive at similar conclusions.

(b) The inference about b_1 is easy to deduce. Next, we observe that for any integer n ,

$$\begin{aligned} (2n)^4 &= 16n^2, \\ \text{and } (2n+1)^4 &= (4n(n+1)+1) = 8n(n+1)(2n(n+1)+1)+1 \\ &= 16 \left(\frac{n(n+1)}{2} \right) (2n(n+1)+1) + 1. \end{aligned} \tag{155}$$

From Lemma 24 and Equation 155 we observe that no integer of the form $(2n)^4$ or $(2n+1)^4$ can have 1 at bit positions 2, 3 and 4.

Suppose $n = 2^{\ell-2}j + j'$. Taking the fourth power, we have the following.

$$\begin{aligned} n^4 &= (2^{\ell-2}j + j')^4 \\ &= 2^{4\ell-8}j^4 + 4 \cdot 2^{3\ell-6}j^3j' + 6 \cdot 2^{2\ell-4}j^2j'^2 + 4 \cdot 2^{\ell-2}jj'^3 + j'^4 \\ &= 2^\ell (2^{3\ell-8}j^4 + 2^{3\ell-4}j^3j' + 3 \cdot 2^{2\ell-3}j^2j'^2 + jj'^3) + j'^4 \end{aligned}$$

If $1 \leq j' \leq 2^{\ell-1} - 1$ then there exists at least one value of j' such that $j'^4 \pmod{2^\ell} \geq 2^{\ell-1}$. From Lemma 24 it follows that $b_\ell = 1$.

In the other direction, let $b_\ell = 1$. Then from Lemma 24 $n^4 = 2^\ell t + t'$, where t is a non-negative integer and $2^{\ell-1} \leq t' \leq 2^\ell - 1$. Let a_1, a_2 be positive integers such that $n = a_1 + a_2$. Then,

$$\begin{aligned} n^4 = 2^\ell t + t' &= (a_1 + a_2)^4 = a_1^4 + 4a_1^3a_2 + 6a_1^2a_2^2 + 4a_1a_2^3 + a_2^4 \\ &= 2^2a_1 \left(\frac{a_1^3}{2} + a_1^2a_2 + 3\frac{a_1a_2^2}{2} + a_2^3 \right) + a_2^4. \end{aligned}$$

The above equality is satisfied if $2^2a_1 = 2^\ell a'_1$, for some integer a'_1 and $a_2^4 = t'$. This implies $a_1 = 2^{\ell-2}a'_1$ and definitely $a_2 < 2^{\ell-2}$. Thus n is of the desired form as stated in the statement of the lemma. Even if we simply equate sum of terms that have powers of 2 we arrive at similar conclusions. \square

We perform the following tests which verify the above result and also helps us conjecture certain circuit complexity bounds, as explained later. First, we numerically calculate the squares and fourth powers of all integers till $2^7 = 128$. We compute their binary decomposition and list those with 1 at particular bit positions, as shown in Tables IV and V, respectively. Next, we consider integers till 2^x where $x = 2, \dots, 12$. For each such 2^x we enumerate all integers $n \leq 2^x$ such that n^2 and n^4 have a binary decomposition with 1 at a particular bit position, as shown in Tables VI and VII, respectively. For these tables we have not mentioned the integers explicitly, due to lack of space, but these can be obtained from our code. The obtained data corroborates the conclusions drawn in the above lemma.

For convenience, we write

$$\Phi^x = \text{diag}((k-1)^x, (k-2)^x, \dots, 1, 0, 1, \dots, (k-1)^x, k^x). \quad [x \in \{2, 4\}] \tag{156}$$

Let $\zeta' = 1 + 2 \log k$ and $\zeta'' = 1 + 4 \log k$ be the maximum number of bits in the binary decomposition of k^2 and k^4 , respectively. Using Lemma 16,

$$\Phi^2 = c_0 \mathbb{I} + \sum_{\ell=1}^{\zeta'} c_\ell U_\ell \quad \text{and} \quad \Phi^4 = c'_0 \mathbb{I} + \sum_{\ell=1}^{\zeta''} c'_\ell U'_\ell, \tag{157}$$

where c_0, c'_0, c_ℓ and c'_ℓ are real coefficients. U_ℓ is a signature matrix obtained from the ℓ^{th} bit in the binary decomposition of $\{0, 1, \dots, k^2\}$ and then replacing the 0s with 1 and 1s with -1, as discussed earlier. Similarly, U'_ℓ is a signature matrix obtained from the ℓ^{th} bit in the binary decomposition of $\{0, 1, \dots, k^4\}$.

If we index the rows as $0, 1, 2, \dots, 2k-1$ then the integer j^x ($j \neq 0$) appears at rows $(k-1)-j$ and $(k-1)+j$ of the matrix ϕ^x (Eq.156). The binary decomposition of a row (equivalently, diagonal) index gives the state of the qubits for which a certain phase is incurred on the quantum circuit. There are $\zeta = \log(2k)$ qubits and let $j = (b_\zeta, b_{\zeta-1}, b_2, b_1)$ be the binary decomposition of the row indexed by integer j . Then if the j^{th} diagonal entry is -1, it implies that whenever the state of the qubits is $(b_\zeta, b_{\zeta-1}, \dots, b_2, b_1)$ a phase of -1 is incurred. A naive way of implementing a circuit is to apply multi-controlled-Z, where the controls correspond to the state of the qubits for which a -1 phase is applied. However, we can design much more compact and efficient circuits by exploiting patterns in the position of ± 1 in each signature matrix and this information can be obtained from Lemma 25. For example, this lemma implies that both U_1 and U'_1 consist of alternate +1 and -1 and hence can be implemented with a single Z on qubit 1. Also, U_2, U'_2, U'_3, U'_4 are all 0 matrices. Now, let us consider U_ℓ where $\ell > 2$. From Lemma 25, -1 appears at rows $(k-1) \pm j$ where $j = 2^{\ell-1}t + t'$, t, t' are integers and $t' \leq 2^{\ell-1} - 1$. If we consider the binary decomposition of any integer then multiplying by $2^{\ell-1}$ shifts the bits $\ell-1$ positions to the left (more significant positions) and appends $\ell-1$ zeros (the lesser significant positions). So binary representation of j can be obtained by shifting the bits in the binary decomposition of t by $\ell-1$ places to the left and then appending the $(\ell-1)$ bits in the binary decomposition of t' . If we consider a Boolean table of ζ bits, then the set of binary strings obtained from integers of the form $2^{\ell-1}t + t'$ induce a “don't care” logical condition on the most significant $\ell-1$ bits. We observe that adding or subtracting $k-1$ simply permutes the all possible strings on these “don't care” bits. Thus to implement U_ℓ it is sufficient to control on the values of the last (significant) $\ell-1$ qubits. Similar deductions can be drawn about U'_ℓ . Here we keep in mind that in this paper we focus on number of qubits and T-gates required, as already mentioned. In our circuits the T-gate contribution increases as we increase the number of controls on Z gate.

Fact 26. *Let us order the qubits q_ζ, \dots, q_1 corresponding to the bit positions b_ζ, \dots, b_1 in the binary decomposition of an integer k , as defined in Eq. 156. The leftmost bit is the most significant one. U_1, U'_1 can be implemented with a single Z on qubit 1. U_2, U'_2, U'_3, U'_4 are all 0 matrices. U_ℓ can be implemented with a circuit which has gates acting on qubits q_x, \dots, q_1 , where $x = \min(\zeta, \ell-1)$. U'_ℓ can be implemented with a circuit which has gates acting on qubits $q_{x'}, \dots, q_1$, where $x' = \min(\zeta, \ell-2)$.*

This also implies that the T-count of U_ℓ ($\ell \leq 2 \log k$) and U'_ℓ ($\ell \leq 4 \log k$) is $O(1)$ i.e. a constant independent of k (but dependent on ℓ). In other words the T-count of U_ℓ and U'_ℓ is at most κ_ℓ and κ'_ℓ , respectively. The T-count of $U_{1+2 \log k}$ and $U'_{1+4 \log k}$ is $O(\log k)$.

Now we want to bound κ_ℓ and κ'_ℓ and for this we identify the possibility of another set of “don't care” conditions from our data. From the statements of Lemma 24 and 25, their proof and previous explanations we can say that the number of controls in multi-controlled-Z and hence the number of T-gates is determined by the number of j' such that $1 \leq j' \leq 2^{\ell-1} - 1$ and $j'^2 \bmod 2^\ell \geq 2^{\ell-1}$. Similar conditions hold when considering the fourth power, as in part (b) of Lemma 25.

Lemma 27. (a) *Let $\ell > 2$ and $S_{1\ell} = \{j' : 1 \leq j' \leq 2^{\ell-1} - 1 \text{ and } j'^2 \bmod 2^\ell \geq 2^{\ell-1}\}$. If $j' \in S_{1\ell}$ then $2^{\ell-1} - j' \in S_{1\ell}$.*

(b) *Let $\ell > 4$ and $S_{2\ell} = \{j' : 1 \leq j' \leq 2^{\ell-2} - 1 \text{ and } j'^4 \bmod 2^\ell \geq 2^{\ell-1}\}$. If $j' \in S_{2\ell}$ then $2^{\ell-2} - j' \in S_{2\ell}$.*

Proof. (a) Suppose $j' \in S_{1\ell}$ and $x = 2^{\ell-1} - j'$. Then,

$$x^2 = 2^{2\ell-2} - 2^\ell j' + j'^2 = 2^\ell (2^{\ell-2} - j') + j'^2, \quad (158)$$

and if $j' \in S_{1\ell}$ then clearly $x \in S_{1\ell}$.

(a) Suppose $j' \in S_{2\ell}$ and $x = 2^{\ell-2} - j'$. Then,

$$\begin{aligned} x^2 &= 2^{4\ell-8} - 4 \cdot 2^{3\ell-6} j' + 6 \cdot 2^{2\ell-4} j'^2 - 4 \cdot 2^{\ell-2} j'^3 + j'^4 \\ &= 2^\ell (2^{3\ell-8} - 2^{2\ell-4} j' + 3 \cdot 2^{\ell-3} j'^2 - j'^3) + j'^4, \end{aligned} \quad (159)$$

and if $j' \in S_{2\ell}$ then clearly $x \in S_{2\ell}$. □

From the data in Tables IV, VI, V and VII we observe that for $\ell \leq x+1$, $|S_{1\ell}| \approx 2^{\ell-2}$ where the approximation ratio tends to 1 as ℓ increases. When $x+2 \leq \ell \leq 2x$, then $0.5 \cdot 2^{x-1} \leq |S_{1\ell}| \leq 0.94 \cdot 2^{x-1}$. Also, for $\ell \leq x+2$, $2^{\ell-4} \leq |S_{2\ell}| \leq 2^{\ell-3}$ and for $x+3 \leq \ell \leq 4x$, we have $0.6 \cdot 2^{x-2} \leq |S_{2\ell}| \leq 1.05 \cdot 2^{x-1}$. This also follows from Eq. 158 and 159 when we consider all possible values of j' such that the multiple of 2^ℓ is positive. We keep in mind that in Tables IV and VI, the listed or enumerated integers are a subset of $S_{1\ell}$ whenever $k = 2^x \geq 2^{\ell+1}$ because the remaining

integers are bigger than 2^x . For similar reasons, in Tables V and VII the listed integers are a subset of $S_{2\ell}$ whenever $k = 2^x \geq 2^{\ell+2}$.

Next, let us observe the following. Consider a circuit with x qubits and suppose a “don’t care” condition exists involving $y < x$ of them. This implies that for every possible combination of binary values on y qubits, there is a certain state of the remaining $x - y$ qubits when a -1 phase is implemented. The number of possible states is 2^y and to implement this logic in a quantum circuit we require a single Z gate controlled on the state of $x - y$ qubits. Many kinds of “don’t care” conditions exist, for example involving parity. Suppose for every possible binary combinations on y qubits where there are odd number of 1s, there exists a certain state of the $x - y$ qubits when -1 phase is incurred. Here too, there are exponential number of such states and to implement this circuit we require a number of CNOTs to check the parity and a Z-controlled on $x - y + 1$ qubits. In summary, as the number of “don’t care” conditions increases, so the number of states that satisfy a certain value also increases, usually exponentially. But the increase in the circuit complexity is less. This implies, if we consider Tables VI and VII then we can say that as the number of integers in each cell increase with respect to $k = 2^x$ and ℓ , the probability of the existence of more ”don’t care” conditions becomes higher and hence κ_ℓ and κ'_ℓ grow more slowly with respect to ℓ . We have already mentioned that the number of integers in each cell increase exponentially with respect to ℓ , but the cell at the intersection of $b_{1+2\log k}$ (or $b_{1+4\log k}$) and k (i.e. 2^x) has value 1. That is why T-count of $U_{1+2\log k}$ and $U'_{1+4\log k}$ is $O(\log k)$. Thus we conjecture the following.

Conjecture 28. The T-count needed to exactly implement U_ℓ and U'_ℓ is at most $O(\min\{\ell, \log k\})$.

In Appendix C 2 b we have explicitly constructed some circuits of signature matrices arising in this compact decomposition of Φ^2 and Φ^4 . Now, we discuss the cost of simulating H_{amp} . The procedure is similar to the one described in the previous section for Algorithm IIIa. In this case we partition H_{amp} as follows.

$$\begin{aligned}
H_{\phi^2\mathbf{x}} &= \frac{1}{2}\Pi^2(\mathbf{x}) + \frac{1}{2}(M^2 + d + 1)\Phi^2(\mathbf{x}) \\
H_{\phi^4\mathbf{x}} &= \Phi^4(\mathbf{x}) \\
H_{\phi\mathbf{x}\mathbf{x}'} &= \Phi(\mathbf{x})\Phi(\mathbf{x}') = H_{3\mathbf{x}\mathbf{x}'} \\
H_3 &= \sum_{\mathbf{x}, \mathbf{x}' \in E_D} H_{3\mathbf{x}\mathbf{x}'} \\
H'_{12} &= \sum_{\mathbf{x} \in \Omega} H_{\phi^2\mathbf{x}} + \frac{\Lambda}{4!}H_{\phi^4\mathbf{x}} \\
H_{amp} &= H'_{12} + H_3
\end{aligned} \tag{160}$$

We summarize the gate complexity for block-encoding H_{amp} in the following theorem.

Theorem 29. Let H_{amp} be the Hamiltonian defined in Eq. 160 and $\|H_{amp}\|$ be the ℓ_1 norm of the coefficients defined in its decomposition Eq. 160. Then it is possible to have a $(\|H_{amp}\|, \cdot, 0)$ block-encoding of H_{amp} with $O(|\Omega| \log k)$ qubits, using the following number of rotation gates

$$N'_r \in O(\log^2 k),$$

and the following number of additional T gates (that are not in the decomposition of rotation gates)

$$N'_t \in O(|\Omega| \log^2 k).$$

The bound on N'_t is obtained assuming Conjecture 28.

The recursive block encoding is done by repeatedly applying Theorem 13, as explained before. We explain these in detail in Appendix C 2 b. In this case we have

$$\|H_{amp}\| \leq \frac{|\Omega|}{4} \left(k^2 \Delta^2 (2 + M^2 + d) + \frac{\Lambda}{12} k^4 \Delta^4 \right) + \frac{3}{4} |E_D| \Delta^2 k^2. \tag{161}$$

We again remember that this norm is sum of absolute value of the coefficients from the non-identity terms only. From [21], we require

$$\mathcal{R}' \in O \left(\|H_{amp}\| t + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)} \right) \tag{162}$$

calls to the block encoding of $\frac{H_{amp}}{\|H_{amp}\|}$ in order to implement an ϵ -precise block encoding of $e^{-iH_{amp}t}$. Thus the total gate complexity is obtained by multiplying \mathcal{R}' with the number of gates obtained in Theorem 29.

5. Phase estimation in the amplitude basis

As described earlier, we can perform phase estimation to compute scattering matrix elements. For simplicity, let us first focus on the $2 \rightarrow 2$ particle scattering. In the center of mass frame, the two incoming momenta can be described by $p_1 = p = -p_2$, while the Mandelstam variable is given by $s = E_{CM}^2 = 2(m^2 + p^2)$. The expression for the energy in Eq. (29) is given as $E = E_{2\text{-particle state}} - E_0$, where E_0 is the ground state. Although not equal to the spectral gap of the Hamiltonian, for values of the coupling away from the critical value, i.e. $\lambda \neq \lambda_c$, we can solve for E by extracting the difference between the first excited state and the ground state in the even-particle sector of the model's \mathbb{Z}_2 symmetry ($\phi \leftrightarrow -\phi$). Similarly, the difference between the first excited state and the ground state in the odd sector gives us the renormalized mass.

a. State preparation Although ground state preparation is in general a QMA-hard problem, it has been shown that the ground states of free scalar fields, namely a Gaussian in the amplitude basis, is efficient to prepare [59, 83]. In the presence of interactions, the ground state can in principle be prepared adiabatically. Here, we assume that we are provided with a state that has polynomial overlap with the two lowest interacting eigenstates. Provided such a state, we seek to prepare the even and the odd sectors of the model. In the field amplitude basis, the even and odd sectors are given by the states of the form

$$\begin{aligned} |\phi_{\text{even}}\rangle &= \frac{|\phi\rangle + |-\phi\rangle}{\sqrt{2}} \\ |\phi_{\text{odd}}\rangle &= \frac{|\phi\rangle - |-\phi\rangle}{\sqrt{2}} \end{aligned} \quad (163)$$

where given (for a single lattice site) $|\phi\rangle = \sum_j \alpha_j |j\rangle$, we define $|-\phi\rangle = \sum_j \alpha_{-j} |j\rangle$. Given our encoding, where $j \in \{-k+1, \dots, k\}$ for some cutoff k , we only exchange the coefficients $\alpha_{-j} \leftrightarrow \alpha_j$ for $j \in \{-k+1, \dots, k+1\}$, and assume that k is chosen large enough that $\alpha_k \approx 0$. Thus, the quantum circuit U we use to map $|\phi\rangle$ to $|-\phi\rangle$ leaves the basis states carrying the coefficients for both α_0 and α_k unchanged.

The discretized field values $j \in \{-k+1, \dots, k\}$ map to binary numbers $b = j+k-1$ in our encoding. For simplicity, we assume that the number of basis status used is some power of 2, i.e. $2k = 2^m$ for some positive integer m , which gives the number of qubits used in the encoding. The required transformation can then be implemented by flipping a bit only if all the bits to the right of this bit form any bitstring except for the all 1's bitstring. An example circuit that achieves this transformation for $k = 4$ is shown in Fig. 5.

To achieve the transformation $U : |\phi\rangle \rightarrow |-\phi\rangle$ more generally for some given k , setting $m = \log_2(2k)$, requires $m-2$ many ancillary qubits. In addition, we need two each of $C^{m-1}X$, $C^{m-2}X$, \dots , C^2X operations, as well as $m-1$ CNOT and X gates. In turn, each $C^m X$ operations, with targets initialized to $|0\rangle$, can be constructed using $n-2$ ancillas and $n-1$ logical AND operations, each of which require 4 T gates. The even sector can then be prepared by implementing the projector $I+U$, while the odd sector can similarly be prepared by applying the projector $I-U$.

These projectors can be realized using the Hadamard test. In the LCU picture, this is described simply as initializing an ancillary qubit in the $|+\rangle$ state, and applying a controlled- U operation controlled on the ancilla, and targeted on the field register. Upon measuring the ancilla in the X -basis, by applying a Hadamard gate before measurement, an outcome of 0 would project the field register onto the even sector while an outcome of 1 would project it onto the odd sector. This operation can be described as

$$(H \otimes I) (|0\rangle \langle 0| \otimes \mathbb{I} + |1\rangle \langle 1| \otimes U) (H \otimes I) |0\rangle |\phi\rangle = \frac{1}{\sqrt{2}} (|0\rangle |\phi_{\text{even}}\rangle + |1\rangle |\phi_{\text{odd}}\rangle) \quad (164)$$

We must apply one such operation for each of the lattice sites. With these considerations, we can gate cost the operation that produces an equal superposition of the even and odd sectors.

Lemma 30. *The total number of T gates required to simulate a controlled- U operation where $U : |\phi\rangle \rightarrow |-\phi\rangle$ is $2|\Omega| (\log_2 k + 1) (\log_2 k + 2) - 8 \in O(|\Omega| \log_2^2 k)$ where $k = \phi_{\text{max}}/\Delta_\phi$ and further obeys $k > 3/2$.*

Proof. From the above considerations, taking $m = \log_2 2k$, we have that the total T gate count for implementing $C-U$

for a single lattice site is given by

$$\begin{aligned}
\# \text{ T gates for } C-U &= \sum_{n=3}^m \text{Cost}(C^n X) + (m-1)\text{Cost}(\text{logical AND}) \\
&= \sum_{n=3}^m (n-1)\text{Cost}(\text{logical AND}) + 4(m-1) \\
&= 4 \sum_{n=3}^m (n-1) + 4(m-1) \\
&= 2m(m+1) - 8
\end{aligned} \tag{165}$$

The total number of T gates is then simply the product of the number of lattice sites $|\Omega|$ with the expression above. \square

Similarly, we can cost the total number of ancillary qubits needed to implement such a controlled- U operation

Lemma 31. *Under the assumptions of Lemma 30, the total number of ancillary qubits required to implement a controlled- U operation is*

$$\frac{1}{2} (\log_2^2 2k + \log_2 2k - 4) \in O(\log_2^2 k)$$

if we reuse ancillae, and

$$\frac{|\Omega|}{2} (\log_2^2 k + \log_2 k - 6) + 1 \in O(|\Omega| \log_2^2 k)$$

if we do not reuse ancillae and thereby allow the operations to be executed in parallel.

Proof. In addition to a single ancilla serving as the control qubit, we also need $m-2$ ancillary qubits to apply a controlled version of U to an m -qubit register where $m = \log_2 2k$ for a single lattice site. Moreover, each of the $\{C^n X\}_{n=3}^m$ operations required to implement $C-U$ require $n-2$ ancillae each. This gives a total number of ancillary qubits as

$$1 + (m-2) + \sum_{n=3}^m (n-2) = \frac{1}{2}(m^2 - m - 4) \tag{166}$$

If the ancillae are re-used across all the lattice sites, the total number of ancillary qubits is then simply given by the above expression. If instead the $C-U$ is performed in parallel across the entire lattice, then all except 1 of the ancillae, the one used to control the entire operation, have to be associated with each of the lattice sites, giving a total count of

$$|\Omega|(m-2) + 1 + \frac{|\Omega|}{2}(m^2 - 3m - 2) = \frac{|\Omega|}{2}(m^2 - m - 6) + 1 \tag{167}$$

Plugging in $m = \log_2 2k$ into these expressions gives the statement of the lemma. \square

b. Eigenvalue extraction for qubitization To extract the energy eigenvalues, we perform phase estimation on the walk operator

$$W = \left(2\text{PREP} |0\rangle \langle 0| \text{PREP}^\dagger \otimes \mathbb{I} - \mathbb{I} \right) \cdot \text{SELECT} \tag{168}$$

which provides a qubitized block encoding of the Hamiltonian, and furnishes a direct sum of two-dimensional irreducible representations, where each two-dimensional subspace is labelled by an eigenstate of the Hamiltonian.

$$W = \oplus_k \begin{pmatrix} \frac{E_k}{|\alpha|} & \sqrt{1 - \left(\frac{E_k}{|\alpha|}\right)^2} \\ -\sqrt{1 - \left(\frac{E_k}{|\alpha|}\right)^2} & \frac{E_k}{|\alpha|} \end{pmatrix} = \oplus_k e^{i \arccos(E_k/|\alpha|)Y} \tag{169}$$

with eigenvalues $\pm \arccos(E_k/|\alpha|)$, from which one may readily, with classical post-processing, obtain the desired eigenvalues E_k . In practice, we would run $\mathbb{I} \otimes W$ on the input state given in Eq. (164), and keep measurements whenever the ancilla measures out to be 0 (1) to infer eigenvalues in the even (odd) sector.

Due to compilation errors, the phase information we extract would be different from the ideal ones. In order to bound this error in the phase, and consequently the energy eigenvalues, we first note a series of observations below.

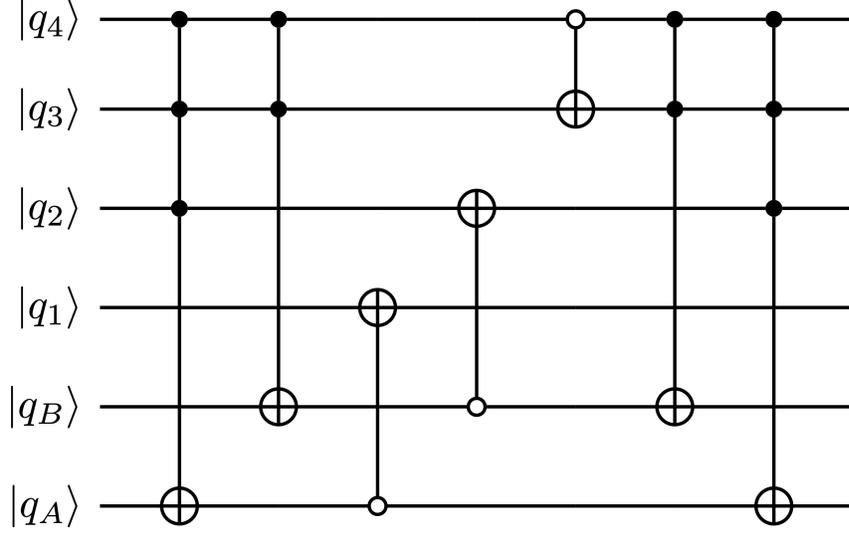


FIG. 5: Example circuit that maps $|\phi\rangle$ to $|\phi\rangle$ for $k = 4$.

Lemma 32. *Let $PREP'$ and $SELECT'$ denote the compiled versions of $PREP$ and $SELECT$, such that they can be written as*

$$\begin{aligned} PREP' |0\rangle^{\otimes \log_2 m} &= \sum_{j=1}^m \beta_j |j\rangle \\ SELECT' &= \sum_{j=1}^m |j\rangle \langle j| \otimes U_j \end{aligned} \quad (170)$$

where $\beta_j \in \mathbb{C}$. Then, $PREP'^{\dagger} SELECT' PREP'$ provides a $(\sum_j |\beta_j|^2, \log_2 m, 0)$ -block encoding of the operator $H' = \sum_{j=1}^m |\beta_j|^2 U_j$. Furthermore, H' is Hermitian.

Proof. The proof of the statement that

$$H' = (\langle 0| \otimes \mathbb{I}) PREP'^{\dagger} SELECT' PREP' (|0\rangle \otimes \mathbb{I}) = \sum_{j=1}^m |\beta_j|^2 U_j \quad (171)$$

directly follows from the LCU lemma of [3, 84]. To see that H' is also Hermitian, we note that all the U_j 's in the LCU are Hermitian diagonal signature matrices, except those that come from the LCU of the momentum (squared) term. The compilation errors do not affect the Hermiticity of the signature matrices, and only impact the Fourier transforms acting on each of the lattice site to diagonalize the momentum (squared) term. If we denote the true quantum Fourier transform circuit as F and its compiled version as F' , then it is straightforward to see that $F'VF'^{\dagger}$ is also Hermitian, given the (Hermitian) signature matrix V . Since the sum of Hermitian operators is Hermitian, H' is also Hermitian. \square

The Hermiticity of the block-encoded operator is an important property to ensure that the walk operator we construct using these (compiled) primitives provides a qubitized block-encoding of the same operator. Furthermore, we note that the $SELECT$ operation, as well as its compiled version $SELECT'$, squares to the identity. As a consequence, it follows from Lemma 8 and Corollary 9 of [73] that the walk operator

$$W' = \left(2PREP' |0\rangle \langle 0| PREP'^{\dagger} \otimes \mathbb{I} - \mathbb{I} \right) \cdot SELECT' \quad (172)$$

provides a qubitized block encoding of the perturbed Hamiltonian H' , and furnishes a direct sum of two-dimensional irreducible representations, where each two-dimensional subspace is labelled by an eigenstate of H' .

$$W' = \oplus_k \begin{pmatrix} \frac{E'_k}{|\alpha|} & \sqrt{1 - \left(\frac{E'_k}{|\alpha|}\right)^2} \\ -\sqrt{1 - \left(\frac{E'_k}{|\alpha|}\right)^2} & \frac{E'_k}{|\alpha|} \end{pmatrix} = \oplus_k e^{i \arccos(E'_k/|\alpha|)Y} \quad (173)$$

will yield eigenvalues $\pm \arccos(E'_k)$, where E'_k are the eigenvalues of H' . We can now bound the absolute difference between the eigenvalues extracted from the compiled walk operator and the ideal walk operator. The various sources of errors are the number of ancillary qubits used for phase estimation, the QFT part of the phase estimation circuit (which is negligible in gate cost), the synthesis error of the rotation gates and the error in the approximate quantum Fourier transforms (AQFTs) used. We bound these errors in terms of the target error in the estimate of energy eigenstates below, in analogy with Lemma 11.

Lemma 33. *To obtain an estimate of an eigenvalue of a Hamiltonian within error ϵ_E , it suffices to perform phase estimation of the compiled walk operator with the following bounds on the contributing errors*

$$\begin{aligned} 2^m &\geq \frac{\pi\alpha}{\sqrt{2}\epsilon_E} \\ \epsilon_r &\leq \frac{1}{3\sqrt{2}} \frac{\epsilon_E}{\alpha N_r} \\ \epsilon_f &\leq \frac{1}{3\sqrt{2}} \frac{\epsilon_E}{\alpha N_f} \end{aligned} \quad (174)$$

where m is the number of ancillary qubits used for phase estimation, ϵ_r is the synthesis error per RZ gate, ϵ_f is the approximation error per individual approximate quantum Fourier transform (AQFT), and N_r and N_f are respectively the total number of rotation gates and AQFTs used in the circuit compiling the walk operator.

Proof. We can approximate, similarly as in Lemma 11, the error in the phase estimate as

$$\epsilon_\theta \approx \sqrt{\left(\frac{\pi}{2^{m+1}}\right)^2 + \left(\pi\epsilon_{QFT}^2 + \epsilon_{synth} + \epsilon_{AQFT}^2\right)^2} \quad (175)$$

where m is the number of ancillary qubits used for phase estimation, ϵ_{QFT} is the error in the QFT part of the phase estimation circuit, ϵ_{synth} is the total synthesis error due to compiling single qubit RZ gates into T gates, and ϵ_{AQFT} is the total error due to the approximate quantum Fourier transforms. Distributing the errors roughly equally, we bound

$$\left(\frac{\pi}{2^{m+1}}\right)^2 \leq \frac{\epsilon_\theta^2}{2}, \quad \pi\epsilon_{QFT} = \epsilon_{synth} = \epsilon_{AQFT} \leq \frac{1}{3} \frac{\epsilon_\theta}{\sqrt{2}} \quad (176)$$

Performing phase estimation of the walk operator with error ϵ_θ induces an error ϵ_E in the estimate of the eigenvalue of the Hamiltonian where the two are related by $\epsilon_\theta = \epsilon_E/\alpha$ where α is the (coefficient) 1-norm of the Hamiltonian. Using this relation, we can readily solve the bounds for the synthesis error per rotation gate $\epsilon_r = \epsilon_{synth}/N_r$ and the error per approximate quantum Fourier transform (AQFT) $\epsilon_f = \epsilon_{AQFT}/N_f$, where N_r and N_f are respectively the total number of rotation gates and AQFTs used in the circuit. These bounds are reported in the statement of the lemma. \square

We now use the above results to report the T gate count of performing phase estimation using the qubitization based algorithms I, IIIa and IIIb as follows.

Theorem 34. *The total cost of performing phase estimation to estimate an eigenvalue of the Hamiltonian to within error ϵ_E is given by*

$$\begin{aligned} Cost(QPE)^{(I)} &\in O\left(\frac{|\Omega|^2}{\epsilon_E} [k^2\Lambda + kM^2] \log^2 k\right) \\ Cost(QPE)^{(IIIa)} &\in O\left(\frac{|\Omega|^2}{\epsilon_E} [k^2\Lambda + kM^2] \log^4 k\right) \end{aligned} \quad (177)$$

while the total number of logical qubits required, including those employed for phase estimation, are

$$\begin{aligned} \text{Count}(\text{Qubit})^{(I)} &\in O\left(|\Omega| \log k + \log^2 k + \log\left(\frac{|\Omega| [k^2 \Lambda + kM^2]}{\epsilon_E}\right)\right) \\ \text{Count}(\text{Qubit})^{(IIIa)} &\in O\left(|\Omega| \log k + \log\left[\frac{|\Omega| (k^2 \Lambda + kM^2)}{\epsilon_E}\right]\right) \end{aligned} \quad (178)$$

where the superscript denotes the algorithm employed.

Proof. The total T gate cost for performing quantum phase estimation (QPE) using any of the qubitization algorithms described above can be approximated as

$$\text{Cost}(\text{QPE}) \approx 2^m \left[N_r \cdot N^{(s)}(\epsilon_r) + N_f \cdot N^{(f)}(\epsilon_f) + N_+ \right] \quad (179)$$

where m is the total number of ancillary qubits used for QPE, N_r and N_f are respectively the total number of rotation gates and AQFTs used in the circuit compiling the walk operator, N_+ is the total number of other T gates used, and

$$N^{(s)}(\epsilon_r) = 3.067 \log_2(2/\epsilon_r) - 4.327 \quad (180)$$

is the T gate cost of compiling a single RZ gate from [32] to within synthesis error ϵ_r , and

$$N^{(f)}(\epsilon_f) = 8(\log_2(2k)) \log_2\left(\frac{(\log_2(2k))}{\epsilon_f}\right) + \log_2\left(\frac{(\log_2(2k))}{\epsilon_f}\right) \log_2\left(\frac{\log_2\left(\frac{(\log_2(2k))}{\epsilon_f}\right)}{\epsilon_f}\right) \quad (181)$$

is the cost of compiling a single AQFT to within error ϵ_f obtained simply by plugging in $n = \log_2(2k)$ in Eq. (60).

For Algorithm 1, we have

$$\begin{aligned} N_r &= 24 \\ N_f &= 2|\Omega| \\ N_+ &= 72|\Omega| \log_2^2 k + 168|\Omega| \log_2 k + 8|\Omega|D - 32|\Omega| - 16 \end{aligned} \quad (182)$$

Using the above, as well as the expression for the coefficient 1-norm α of the Hamiltonian provided in Eq. (102) for $\Delta_\phi = \sqrt{\pi/k}$, and Eq. (174) from Lemma 33, we find (for fixed spatial dimensionality)

$$\begin{aligned} 2^m &\in O\left(\frac{|\Omega|}{\epsilon_E} [k^2 \Lambda + kM^2]\right) \\ N_r \cdot N^{(s)}(\epsilon_r) &\in O\left(\log\left(\frac{\alpha}{\epsilon_E}\right)\right) \in O\left(\log\left(\frac{|\Omega| k(\Lambda + M^2)}{\epsilon_E}\right)\right) \\ N_f \cdot N^{(f)}(\epsilon_f) &\in O\left(|\Omega| \log k \log\left[\frac{\log k \cdot |\Omega| k(\Lambda + M^2)}{\epsilon_E}\right]\right) \\ N_+ &\in O(|\Omega| \log^2 k) \end{aligned} \quad (183)$$

The dominant cost is that of $2^m N_+$ and so, in all,

$$\text{Cost}(\text{QPE})^{(I)} \in O\left(\frac{|\Omega|^2}{\epsilon_E} [k^2 \Lambda + kM^2] \log^2 k\right) \quad (184)$$

Meanwhile, adding the qubit count from Eq. (125) and $m = \log_2 \frac{\pi\alpha}{\sqrt{2}\epsilon_E}$, we get

$$\text{Count}(\text{Qubit})^{(I)} \in O\left(|\Omega| \log k + \log^2 k + \log\left(\frac{|\Omega| [k^2 \Lambda + kM^2]}{\epsilon_E}\right)\right) \quad (185)$$

Similarly, we recall these gate counts for Algorithm 3a, borrowing from Eqs. (C82) and (C80) here for convenience,

$$\begin{aligned} N_r &= 4(L_1 + L_2) + 2(L_3 + L_4) + 2L_5 - 4 \\ N_f &= 2|\Omega| \\ N_+ &= |\Omega| \left[4\sqrt{L_1 + L_2} (\log_2(L_1 + L_2) - 2) + 12(L_1 + L_2) + 4\sqrt{L_3 + L_4} (\log_2(L_3 + L_4) - 2) + 12(L_3 + L_4) \right] \\ &\quad + |E_D| \left(4\sqrt{L_5} (\log_2 L_5 - 2) + 8L_5 \right) + 4\sqrt{|\Omega|} (\log_2 |\Omega| - 2) + 4|\Omega| + 4\sqrt{|E_D|} (\log_2 |E_D| - 2) + 4|E_D| \end{aligned} \quad (186)$$

and from Eq. (C92), we have that the coefficient 1-norm scales as $\alpha \in O(|\Omega| (k^2\Lambda + kM^2))$, similarly as in Algorithm 1. Using the results of Lemma 174 as before, we find

$$\begin{aligned}
2^m &\in O\left(\frac{|\Omega|}{\epsilon_E} [k^2\Lambda + kM^2]\right) \\
N_r \cdot N^{(s)}(\epsilon_r) &\in O\left(\log^4 k \cdot \log\left[\frac{|\Omega| \log k (k^2\Lambda + kM^2)}{\epsilon_E}\right]\right) \\
N_f \cdot N^{(f)}(\epsilon_f) &\in O\left(|\Omega| \log k \log\left[\frac{\log k \cdot |\Omega| k(\Lambda + M^2)}{\epsilon_E}\right]\right) \\
N_+ &\in O(|\Omega| \log^4 k)
\end{aligned} \tag{187}$$

The dominant cost is that of $2^m N_+$, and so we have

$$Cost(QPE)^{(IIIa)} \in O\left(\frac{|\Omega|^2}{\epsilon_E} [k^2\Lambda + kM^2] [\log^4 k + |\Omega|]\right) \tag{188}$$

We estimate the total number of logical qubits used in Algorithm IIIa as the sum total of $|\Omega| \log_2 2k$ many qubits used to represent the field itself, $\log_2 |\Omega|$ used to control the lattice site, $\log_2 \log_2 k$ used to control the unitary in the LCU expansion, and $m = \log_2 \frac{\pi\alpha}{\sqrt{2}\epsilon_E}$ many ancillary qubits used for QPE. Thus, the total qubit count used in Algorithm IIIa is estimated as

$$\begin{aligned}
Count(Qubit)^{(IIIa)} &\approx |\Omega| \log_2 2k + \log_2 |\Omega| + \log_2 \log_2 k + \log_2 \frac{\pi\alpha}{\sqrt{2}\epsilon_E} \\
&\in O\left(|\Omega| \log k + \log\left[\frac{|\Omega| (k^2\Lambda + kM^2)}{\epsilon_E}\right]\right)
\end{aligned} \tag{189}$$

□

Proposition 35. Assuming Conjecture 28, the T gate and qubit cost of the qubitization based algorithm IIIb in the amplitude basis is given by

$$\begin{aligned}
Cost(QPE)^{(IIIb)} &\in O\left(\frac{|\Omega|^2}{\epsilon_E} [k^2\Lambda + kM^2] \log^2 k\right) \\
Count(Qubit)^{(IIIb)} &\in O\left(|\Omega| \log k + \log\left[\frac{|\Omega| (k^2\Lambda + kM^2)}{\epsilon_E}\right]\right)
\end{aligned} \tag{190}$$

Proof. As in Theorem 34, the total T gate cost for performing QPE using this qubitization based algorithm can be approximated as

$$Cost(QPE) \approx 2^m [N_r \cdot N^{(s)}(\epsilon_r) + N_f \cdot N^{(f)}(\epsilon_f) + N_+] \tag{191}$$

where m is the total number of ancillary qubits used for QPE, N_r and N_f are respectively the total number of rotation gates and AQFTs used in the circuit compiling the walk operator, N_+ is the total number of other T gates used, and $N^{(s)}(\epsilon_r)$ and $N^{(f)}(\epsilon_f)$ are respectively the T gate cost of compiling a single RZ gate within synthesis error ϵ_r , and the T gate cost of compiling a single AQFT within error ϵ_f .

For convenience, we recall the relevant number of gates from Appendix C 2 b

$$\begin{aligned}
N_r &= 12 \log_2 k + 2(\log_2 k + 1)^2 - 3 \\
N_f &= 2|\Omega| \\
N_+ &= |\Omega| \left(4\sqrt{2\log_2 k}(\log_2(2\log_2 k) - 2) + 8\log_2 k + 4\sqrt{4\log_2 k}(\log_2(4\log_2 k) - 2) + 16\log_2 k\right) \\
&\quad + 4\sqrt{|\Omega|}(\log_2 |\Omega| - 2) + 4|\Omega| + |E_D| (8(\log_2 k + 1)(\log_2(\log_2 k + 1) - 1) + 4(\log_2 k + 1)^2 + 4) \\
&\quad + 4\sqrt{|E_D|}(\log_2 |E_D| - 2) + O(|\Omega| \log_2^2 k)
\end{aligned} \tag{192}$$

Using the expression for the coefficient 1-norm given in Eq. (161), we find

$$\begin{aligned}\alpha = \|H_{amp}\| &\in O(|\Omega| (k^2\Lambda + kM^2)) \\ \Rightarrow 2^m &\in O\left(\frac{|\Omega| (k^2\Lambda + kM^2)}{\epsilon_E}\right)\end{aligned}\quad (193)$$

The dominant cost is now that of $2^m N_+$, and so we have

$$Cost(QPE)^{(IIIb)} \in O\left(\frac{|\Omega|^2}{\epsilon_E} [k^2\Lambda + kM^2] \log^2 k\right)\quad (194)$$

For the qubit counts, we require a total of $O(|\Omega| \log k)$ many qubits to hold the field values, and an additional $O(\log \log k)$ qubits for the implementation of the signature matrices in the $O(\log k)$ LCU decomposition, a cost which is sub-dominant. In addition, we would require an additional number of $m \in O(\|H_{amp}\|/\epsilon_E)$ ancillary qubits used for phase estimation, so that in all we have

$$Count(Qubit)^{(IIIb)} \in O\left(|\Omega| \log k + \log \left[\frac{|\Omega| (k^2\Lambda + kM^2)}{\epsilon_E}\right]\right)\quad (195)$$

□

In order to perform a similar analysis for Algorithm II in the amplitude basis, which is a Trotter based algorithm that also employs the approximate QFT, which would incur its own error, we make use of the following lemma.

Lemma 36. *To obtain an estimate of an eigenvalue of a Hamiltonian within error ϵ_E , in the presence of Trotter, synthesis, and approximate QFT errors, it suffices to perform phase estimation on a single Trotter step with the following bounds on the contributing errors*

$$\begin{aligned}2^m &\geq \pi 2^{3/4} \frac{\alpha_{comm}^{1/2}}{\epsilon_E^{3/2}} \\ \epsilon_r &\leq 2^{4/15} \frac{\epsilon_E^{3/2}}{N_r \alpha_{comm}^{1/2}} \\ \epsilon_f &\leq 2^{4/15} \frac{\epsilon_E^{3/2}}{N_f \alpha_{comm}^{1/2}}\end{aligned}\quad (196)$$

Proof. The proof proceeds similarly to Lemmas 11 and 174, except that we now also account for both Trotter and approximate QFT errors here. Specifically, we approximate the error in the phase as

$$\epsilon_\theta \approx \sqrt{\left(\frac{\pi}{2^{m+1}}\right)^2 + \left(\pi \epsilon_{QFT}^2 + \epsilon_{Trotter} + \epsilon_{synth} + \epsilon_{AQFT}\right)^2}\quad (197)$$

The error in the phase error ϵ_θ is related to the error in the energy estimate ϵ_E as $\epsilon_\theta = \epsilon_E \tau$, where τ is the total simulation time. Distributing the errors roughly equally, we bound

$$\left(\frac{\pi}{2^{m+1}}\right)^2 \leq \frac{\epsilon_\theta^2}{2}, \quad \pi \epsilon_{QFT} = \epsilon_{Trotter} = \epsilon_{synth} = \epsilon_{AQFT} \leq \frac{1}{4} \frac{\epsilon_\theta}{\sqrt{2}}\quad (198)$$

Using the Trotter error bound, we obtain

$$\begin{aligned}\epsilon_{Trotter} &\leq \frac{1}{4} \frac{\epsilon_E \tau}{\sqrt{2}} \leq \alpha_{comm} \tau^3 \\ \Rightarrow \tau &\geq \sqrt{\frac{\epsilon_E}{2^{5/2} \alpha_{comm}}}\end{aligned}\quad (199)$$

Therefore,

$$\begin{aligned}\left(\frac{\pi}{2^{m+1}}\right)^2 &\leq \frac{\epsilon_\theta^2}{2} \\ \Rightarrow 2^m &\geq \pi 2^{3/4} \frac{\alpha_{comm}^{1/2}}{\epsilon_E^{3/2}}\end{aligned}\quad (200)$$

Similarly, we can derive

$$\begin{aligned}\epsilon_{synth} = \epsilon_r N_r &\leq \frac{1}{4} \frac{\epsilon_\theta}{\sqrt{2}} \\ \Rightarrow \epsilon_r &\leq 2^{4/15} \frac{\epsilon_E^{3/2}}{N_r \alpha_{comm}^{1/2}}\end{aligned}\quad (201)$$

and similarly,

$$\begin{aligned}\epsilon_{AQFT} = \epsilon_f N_f &\leq \frac{1}{4} \frac{\epsilon_\theta}{\sqrt{2}} \\ \Rightarrow \epsilon_f &\leq 2^{4/15} \frac{\epsilon_E^{3/2}}{N_f \alpha_{comm}^{1/2}}\end{aligned}\quad (202)$$

□

The total number of T gates for phase estimation using Algorithm II has already been detailed in Theorem 21. The additional number of ancillary qubits required for phase estimation is described in the following lemma.

Lemma 37. *The total number of qubits required for phase estimation using the amplitude basis Algorithm II is given by*

$$Count(Qubit)^{(II)} \approx O\left(\log_2\left(\frac{|\Omega|\Lambda k(\Lambda k + N^2)}{\epsilon_E}\right)\right)\quad (203)$$

Proof. As already noted in Theorem 21, the total number of qubits used for block-encoding is $O(|\Omega| \log k)$. A further number of ancillary qubits used for phase estimation is given by Lemma 36 and Eq. (144) as

$$\begin{aligned}m &\in O\left(\log_2\left(\frac{\alpha_{comm}^{1/2}}{\epsilon_E^{3/2}}\right)\right) \\ &\in O\left(\log_2\left(\frac{|\Omega|\Lambda k(\Lambda k + N^2)}{\epsilon_E}\right)\right)\end{aligned}\quad (204)$$

where we have repeatedly used $\log\left(\frac{A^m}{B^n}\right) \in O\left(\log\left(\frac{A}{B}\right)\right)$ for constant $m, n > 0$. □

6. Raw gate costs for phase estimation

We have reported the asymptotic T gate costs of all the algorithms discussed above in Table 1. Here, we numerically compute the raw T gate costs, taking care of pre-factors and other factors missing from the asymptotic expressions. We estimate the total cost of phase estimation as

$$Cost(QPE) \approx 2^m \left[N_r \cdot N^{(s)}(\epsilon_r) + N_f \cdot N^{(f)}(\epsilon_f) + N_+ \right]\quad (205)$$

where m is the total number of ancillary qubits used for QPE, N_r and N_f are respectively the total number of rotation gates and AQFTs used in the circuit compiling the walk operator, N_+ is the total number of other T gates used, and

$$N^{(s)}(\epsilon_r) = 3.067 \log_2(2/\epsilon_r) - 4.327\quad (206)$$

is the T gate cost of compiling a single RZ gate from [32] to within synthesis error ϵ_r , and

$$N^{(f)}(\epsilon_f) = 8(\log_2(2k)) \log_2\left(\frac{(\log_2(2k))}{\epsilon_f}\right) + \log_2\left(\frac{(\log_2(2k))}{\epsilon_f}\right) \log_2\left(\frac{\log_2\left(\frac{(\log_2(2k))}{\epsilon_f}\right)}{\epsilon_f}\right)\quad (207)$$

is the cost of compiling a single AQFT to within error ϵ_f obtained simply by plugging in $n = \log_2(2k)$ in Eq. (60).

The bounds for the errors, and the expressions for N_r , N_f and N_+ for the various algorithms have been recalled for convenience in Lemma 11, Theorem 12, Theorem 21, Lemma 33, Theorem 34, Proposition 35, and Lemmas 36 and 37 above.

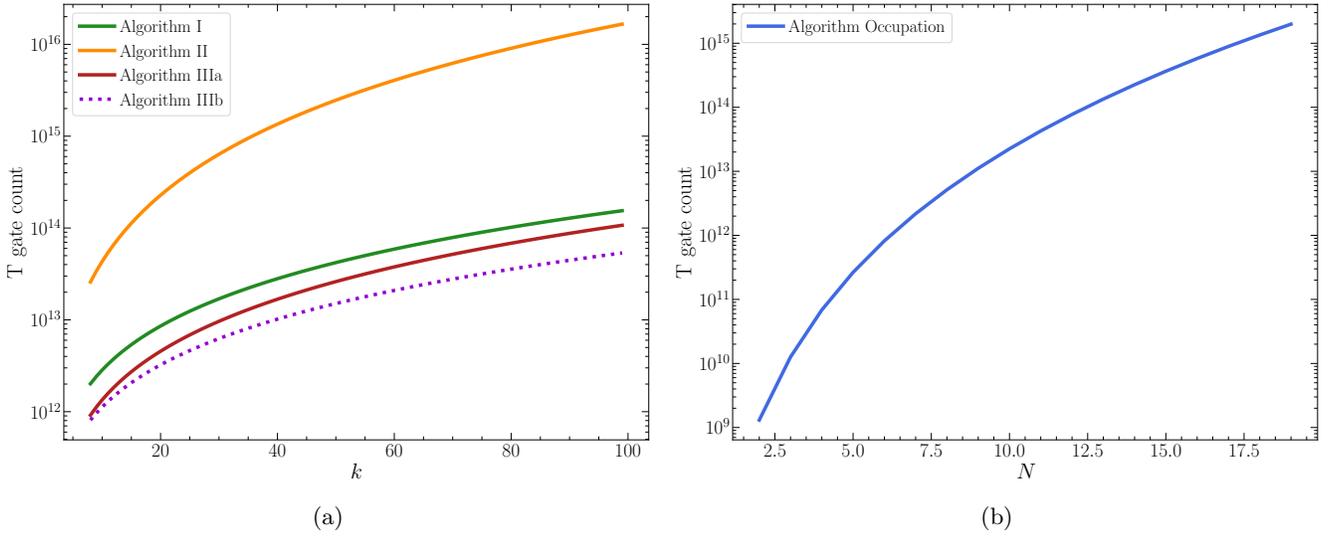


FIG. 6: a) The T gate count on a log axis as a function of the field amplitude cutoff k . Algorithm IIIb is plotted with a dotted line as the precise scaling rests upon conjectured behaviour of the field operator binary decomposition (Conjecture 28)). Unknown constant prefactors have been set to 1 here. b) The T gate count on a log axis, as a function of the field occupation cutoff N . For both, we consider a strong-coupling regime $\lambda = M = 1$, with $|\Omega| = 10^2$.

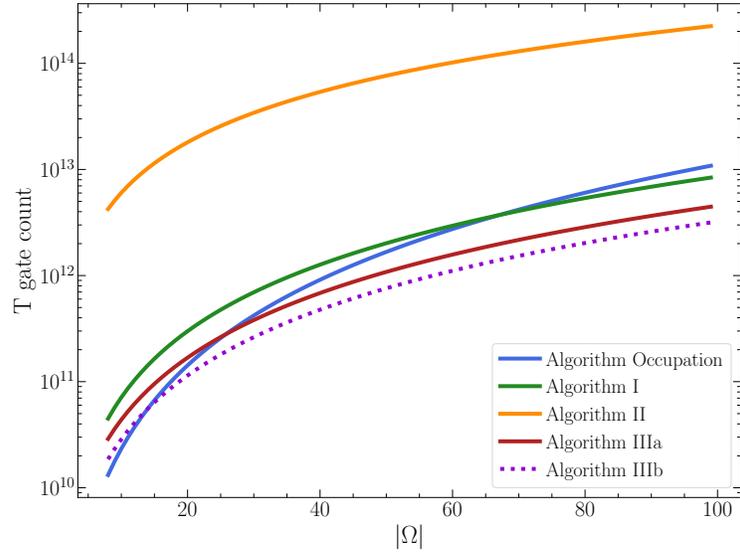


FIG. 7: The T gate count on a log axis as a function of the momentum volume cutoff $|\Omega|$. Here we consider a strong-coupling regime $\lambda = M = 1$ and $\epsilon_E = 10^{-2}$. Here we have $k = 20$. By dimensional analysis we expect an approximate scaling relation of $N \sim \sqrt{k}$. In order to illustrate the proper scaling relations (and optimum nature of the amplitude basis) with respect to $|\Omega|$, we select a larger (and therefore more accurate to the physics) value for $N = 9$. Algorithm IIIb is again plotted with a dotted line as the precise scaling rests upon conjectured behaviour (Conjecture 28).

F. Resource Estimates for Simulation Algorithms

We will now compare the T -count and logical qubit estimates for implementing the various algorithms described so far. A summary of the cost analysis has been provided in Table I (Section I) and Fig. 6, 7, 8, 9.

In Fig. 6(a) we show the variation of T-gate count of the amplitude basis algorithms with respect to the field amplitude cutoff k and in Fig. 7(b) we show the variation of the T-gate count of the occupation basis algorithm with respect to the field occupation cutoff N . We consider the strong coupling regime $\lambda = M = 1$ with $|\Omega| = 10^2$. We

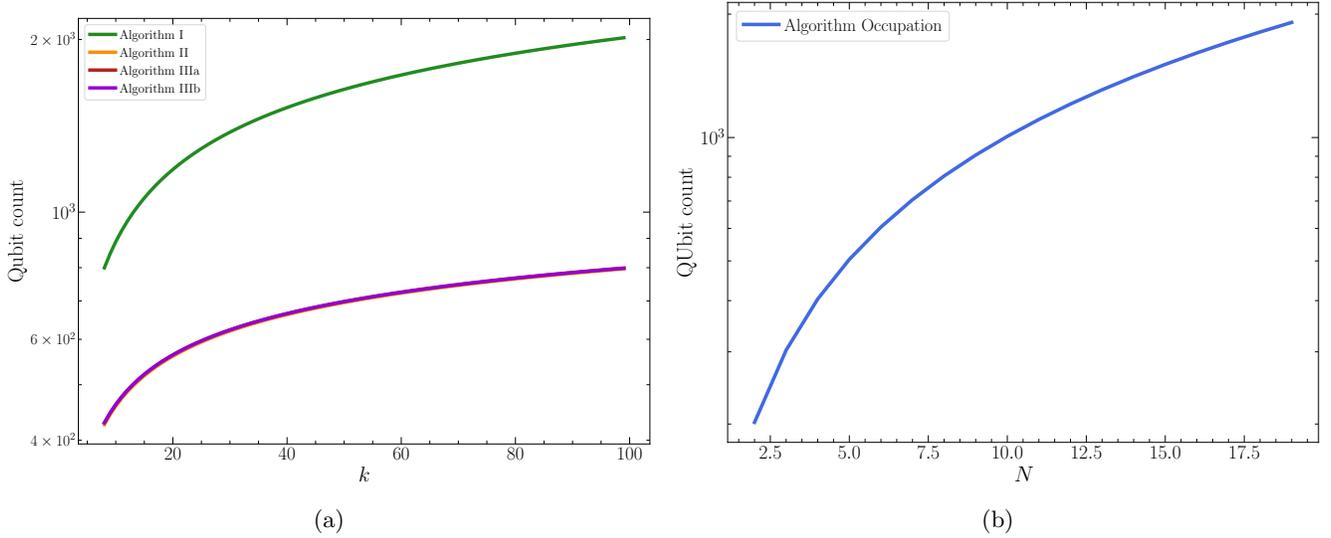


FIG. 8: a) The logical qubit count on a log axis as a function of the field amplitude cutoff k . b) The logical qubit count on a log axis as a function of the field occupation cutoff N . For both, we consider a strong-coupling regime $\lambda = M = 1$, with $|\Omega| = 10^2$ and $\epsilon_E = 10^{-2}$.

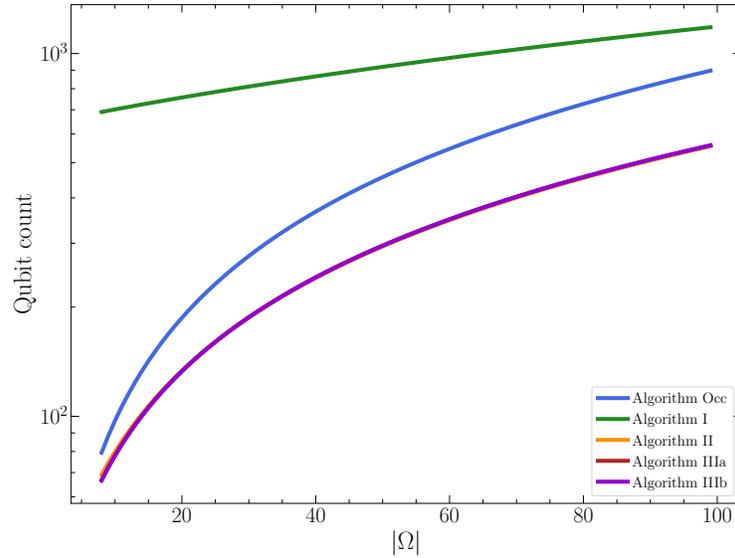


FIG. 9: The logical qubit count on a log axis as a function of the momentum volume cutoff $|\Omega|$. Here we consider a strong-coupling regime $\lambda = M = 1$ and $\epsilon_E = 10^{-2}$. Here we have $k = 20$. By dimensional analysis we expect an approximate scaling relation of $N \sim \sqrt{k}$. In order to illustrate the proper scaling relations (and optimum nature of the amplitude basis) with respect to $|\Omega|$, we select a larger (and therefore more accurate to the physics) value for $N = 9$.

observe that the Trotter based algorithms, both for the occupation basis and the amplitude basis (Algorithm II), have higher T-count than the qubitization based algorithms (Algorithms I, IIIa and IIIb). One reason for this is the use of more number of rotation gates in Trotter-based algorithms. For a complete fault-tolerant implementation we decompose each of them further into Clifford+T. One solution to circumvent this problem can be the use of partial fault-tolerant implementation where the rotations are implemented non-fault-tolerantly but the Clifford gates have a fault-tolerant implementation [85]. Among the qubitization based algorithms, Algorithm IIIb has the minimum T-gate count, but its estimates depend on Conjecture 28. Algorithm IIIa has the minimum rigorously proved T-gate-count estimate.

In Fig. 7 we plot the T-gate-count as a function of the momentum volume cutoff $|\Omega|$ in the strong coupling regime

$\lambda = M = 1$ and $\epsilon_E = 10^{-2}$. We consider the field amplitude cutoff $k = 20$ and an expected relation $N \sim \sqrt{k}$ with the field occupation cutoff. Here, we observe that the occupation basis Trotter algorithm performs better than all the amplitude basis algorithms in a small range. It performs better than Algorithms I and II for a larger range. For nearly the complete range of values considered, amplitude basis Algorithms IIIa and IIIb (with Conjecture 28) has the minimum T-gate-count.

In Fig. 8(a) we plot the logical qubit count of the amplitude basis algorithms as a function of the field amplitude cutoff k , while in Fig. 8(b) we show the logical qubit count of the occupation basis algorithm with respect to the occupation basis cutoff N . Again we consider the strong coupling regime $\lambda = M = 1$ and $|\Omega| = 10^2$. We observe that the qubit count of Algorithms II, IIIa and IIIb is much better than the others. This is due to the LCU decomposition of Φ , Φ^2 and Φ^4 operators based on binary representation of integers. We remember that the qubit cost of Algorithm IIIb does not depend on any conjecture.

In Fig. 9 we show the variation of the logical qubit count of all the algorithms with respect to the momentum volume cutoff $|\Omega|$. Again, as explained before we consider the strong coupling regime $\lambda = M = 1$ and $\epsilon_E = 10^{-2}$, $k = 20$, $N \sim \sqrt{k}$. Here we observe that the qubit count of the occupation basis Trotter algorithm is much less than the amplitude basis LCU based algorithm I. However, again the qubit counts of the other amplitude basis algorithms - both Trotter based Algorithm II and LCU based Algorithms IIIa and IIIb, are the minimum among all the algorithms considered.

In summary, the amplitude basis qubitization based Algorithm IIIa and its improved version Algorithm IIIb have better cost estimates i.e. the minimum T-gate-count and logical qubit count compared to other algorithms. One reason for this is the particular binary representation based LCU decomposition of the operators. The Trotterization algorithm using this decomposition, though enjoys the benefit of lower qubit count, but has higher T-gate count because of more rotation gates, as mentioned earlier. The other amplitude basis qubitization based Algorithm I has much lower T-gate-count in most cases compared to the Trotter algorithms - both in the occupation basis as well as Algorithm II. But it has much higher qubit count compared to all the algorithms, in nearly all the cases considered. The qubit cost of the occupation basis Trotter algorithm is somewhat intermediate and its T-gate-count, though low in a small regime, soon becomes higher than the others.

V. FAULT TOLERANT IMPLEMENTATION

To create fault-tolerant non-Clifford T gates within the surface code, one needs to prepare highly accurate magic states $|A_L\rangle$ which are then fed into the logical computation circuit to affect logical T gates, $|A_L\rangle = TH|0\rangle$ [86]. Note that these states are prepared in an area of the surface code that is separate from the area where the logical computation is taking place. The process of magic state distillation is resource intensive and is often responsible for the highest footprint of surface code-based fault-tolerant computation.

In this section we provide an estimate for the resources required to generate sufficiently accurate T gates, following Fowler's treatment in [40]. Assuming $N_T = 10^{12}$ logical T gates and a physical gate measurement time of $t_m = 10^{-7}$ seconds, the shortest possible time to compute the N_T consecutive gates is $t_c = t_m N_T = 10^5$ seconds, or around 28 hours.

To calculate the physical qubit footprint, note that each logical T gate needs one highly distilled ancilla (magic) state $|A_L\rangle$ injected into the logical circuit. The tolerable error rate per $|A_L\rangle$ state must then satisfy

$$P_{A_L} < 1/N_T = 10^{-12}. \quad (208)$$

Assuming an injection error rate of $P_I = 10^{-3}$, and an error-free distillation circuit, the error rate associated with the first layer of distillation will be,

$$P_1 = 35P_I^3 \sim 3.5 \times 10^{-8}, \quad (209)$$

which is greater than the required P_{A_L} , thus a second layer of distillation will be required. The error probability in this case will be,

$$P_2 = 35P_1^3 \sim 1.5 \times 10^{-21}, \quad (210)$$

which is less than P_{A_L} , therefore two layers of distillation will be sufficient to purify an ancilla state with the desired accuracy.

Using the 15-qubit Reed-Muller encoding for the distillation of $|A_L\rangle$, the first stage requires 15 sets of distillation circuits, each acting on 16 logical qubits, for a total of 240 logical qubits, operating in $8 \times 5d_1/4 = 10 d_1$ surface code

cycles where d_1 is the distance associated with the first layer of distillation. To estimate d_1 , note that the error rate for $|A_L\rangle$ after the first distillation is

$$P_{A_{L_1}} = 1800 d_1 P_{L_1}. \quad (211)$$

Here the coefficient 1800 results from 240 logical qubits, two types of logical qubits, three types of error chains, and $5d_1/4$ surface code cycles. The logical error rate for the first layer P_{L_1} is related to the distance with the following relation,

$$P_{L_1} \simeq 3 \times 10^{-2} \left(\frac{P}{P_{th}} \right)^{\frac{d_1}{2}}, \quad (212)$$

where $P_{th} = 10^{-2}$ is the threshold error and we assume the a physical error rate of $P \simeq 10^{-3}$.

We need to find d_1 such that

$$35(P_{A_{L_1}})^3 < P_{A_L} < 10^{-12}. \quad (213)$$

We find that the shortest distance for which this requirement is satisfied is for $d_1 = 15$. Assuming the rate of $r_1 = 5/2 \times 5d_1^2/4$ physical qubits per logical qubit, we need $n_1 = 1.7 \times 10^5$ physical qubits and $t_1 = 10 d_1 = 150$ surface code cycles.

The second stage of distillation requires another encoding with 16 logical qubits and $t_2 = 10 d_2$ surface code cycles, where again d_2 is the corresponding code distance for the second distillation. To find this distance, we need to satisfy,

$$P_{A_{L_2}} = 120 d_2 P_{L_2} < P_{A_L} < 10^{-12}, \quad (214)$$

where the coefficient, 120, stems from 16 logical qubits, two types of logical qubits, three types of error chains and $5d_2/4$ surface code cycles and,

$$P_{L_2} \simeq 3 \times 10^{-2} \left(\frac{P}{P_{th}} \right)^{\frac{d_2}{2}}. \quad (215)$$

We find that for $d_2 = 29$ this requirement is satisfied. Given the rate of $r_2 = 5/2 \times 5d_2^2/4$ physical qubits per logical qubit, we need a total of $n_2 = 4.2 \times 10^4$ physical qubits and $t_2 = 10d_2 = 290$ surface code cycles for the second round of distillation.

Note that the qubits used in the first round of distillation can be reused during the second round, thus total resources required for two rounds of distillation will be $n_1 = 1.7 \times 10^5$ physical qubits and $t_1 + t_2 = 440$ surface code cycles or $t_{AAA} = 4.4 \times 10^{-5}$ seconds, assuming a surface code cycle time of 100 nanoseconds. Following the argument in [40], this space-time footprint is enough to distill three ancilla states, or an AAA factory.

If we use only one AAA factory, the time to distill all 10^{12} $|A_L\rangle$ states will be $t = 1.5 \times 10^7$ seconds or ~ 6 months. To achieve the minimal computation time of $t_c = 10^5$ seconds, we need to create parallel AAA factories. To calculate this number, note that each AAA factory prepares 3 ancilla states in time $t_{AAA} = 4.4 \times 10^{-5}$ seconds. Therefore, in time $t_c = 10^5$ seconds we can create $3t_c/t_{AAA} =$ states. Thus to achieve the desired number of states $N_T = 10^{12}$ in $t_c = 10^5$ seconds we need to create $N_{AAA} = N_T/(3t_c/t_{AAA})$ or ~ 147 parallel AAA factories. This parallelization will save us time but instead increases our space footprint to $147 \times 1.7 \times 10^5 \simeq 2.5 \times 10^7$ physical qubits.

If we use the distance of the second layer of distillation for our logical computation, the cost of encoding a logical qubit will be $5/2 \times 5 \times (16)^2/4 = 2.6 \times 10^6$ physical qubits, which is $\sim 10\%$ of the number of physical qubits needed for the distillation of all the required ancilla states in the desired time. Thus the total footprint for the computation will be 2.8×10^7 physical qubits.

If we reduce the error per gate to $P \simeq 10^{-4}$, we can reduce the spacial footprint of the distillation circuit. Following the same logic as before we find that we still need two layers of distillations, where for the first layer we need a code with distance $d_1 = 8$, $n_1 \simeq 4.8 \times 10^4$ physical qubits and $t_1 = 80$ cycles. For the second round we need $d_2 = 14$, $n_2 \simeq 9.8 \times 10^3$ physical qubits and $t_2 = 140$ cycles. As before the qubits in the first layer can be recycled in the second layer so the total required resources per distilled state will be $n = n_1 = 4.8 \times 10^4$ qubits and $t = t_1 + t_2 = 220$ cycles or 2.2×10^{-5} seconds for an AAA factory. If we were to use only a single AAA factory, the time to prepare $N_T = 10^{12}$ state would be 7.3×10^6 seconds or ~ 3 months. To keep pace with the computational time $t_c = 10^{-5}$ seconds, we need to use $10^{12}/(3 \times 10^5/(2.2 \times 10^{-5})) \sim 74$ parallel AAA factories with the footprint of $\sim 3.5 \times 10^6$ physical qubits.

A logical computational qubit in this case will cost $3.125 \times 14^2 = 612$ physical qubits, hence for 1000 logical qubits we need $\sim 6.2 \times 10^5$ physical qubits, which is $\sim 17\%$ of the distillation footprint, bringing the total number of physical qubits to 4.2×10^6 . A summary of these results is given in table III.

The calculations in this section follow closely the arguments of [87]. Depending on the need to optimize either time or space, the methods discussed in more recent references, e.g. [42], can be used to modify the estimates.

p/p_{th}	10^{-1}	10^{-2}
First distillation distance d_1	15	8
Second distillation distance d_2	29	14
Time per AAA (s)	4.4×10^{-5}	2.2×10^{-5}
Total time for 10^{12} A states without parallelization (s)	1.5×10^7	7.3×10^6
Total time for 10^{12} A states with parallelization (s)	1.0×10^5	1.0×10^5
Number of parallel AAA factories	147	74
Number of Physical qubits per AAA factory	1.7×10^5	4.8×10^4
Total Physical qubits for distillation with parallelization	2.5×10^7	3.5×10^6
Physical qubits for 1000 logical computational qubits	2.6×10^6	7.4×10^5
Total physical qubits including computational qubits	2.8×10^7	4.2×10^6

TABLE III: Summary of the estimated resources required for the fault-tolerant implementation of the algorithm using the surface code. Note that the best possible logical computation time, which is limited by the gate time of $t_m = 10^{-7}$ seconds, is $t_c = 10^5$ seconds or ~ 28 hours. This time can be achieved by parallelizing magic state factories whose footprint is partially determined by the ratio p/p_{th}

VI. CONCLUSION

A central challenge that has remained unanswered within the quantum simulation community involves deciding whether simulation of scalar field theories can be practically done on a quantum computer. Here we have addressed this by providing a new method that uses phase estimation to estimate elements of the S -matrix for elastic collisions and designing optimized circuits for Qubitization and Trotter based simulation of scalar field theory in amplitude or occupation basis. We find that simulation of scalar field theory in 1 + 1D with a occupation cutoff of 9 or field cutoffs of 20 with a field volume $\Omega = 100$ leads to a number of T gates that is on the order of 10^{12} and the number of logical qubits that are on the order of 1000 for the occupation basis and in on the order of 500 logical qubits for the qubitization based amplitude basis algorithm. We show that the calculation can be performed in just over a day.

These estimates are predicated on 5 new algorithms considered in this paper. In the occupation basis we describe a Trotter-based simulation algorithm. Of the 4 algorithms described in the amplitude basis, the three qubitization based ones perform better. Among these, the most efficient one in terms of T-gate-count and number of logical qubits, is the one where the LCU decomposition of operators have been done using binary representation of integers. We can prove that the number of T gates needed for the Trotter-based algorithm, for coupling strength λ , occupation cutoff N , mass M , and energy uncertainty ϵ_E , is

$$N_{T,occ} \in \tilde{O} \left(\frac{\lambda N^7 |\Omega|^3}{M^{5/2} \epsilon_E^{3/2}} \right). \quad (216)$$

We compute this by evaluating commutator bounds on the second-order Trotter formula. This algorithm performs best in circumstances where the particle mass is large, when the number of particles in occupation basis is dramatically lower than the maximum field or when the coupling strength is relatively weak. Higher order formulas can give better scaling, but we focus on the low-order formulas because the tightest bounds available are for the second order formula [30, 67]. It is worth noting however, that even the tightest bounds on Trotter error are pessimistic [28, 30, 88, 89]. Nonetheless, we note that, if one is interested in high energy scattering, this basis can be ideally suited for such problems. This results from the fact at high energies the particles participating in scattering events can be thought as being nearly free (i.e. weak coupling), as long as the occupation number is small, while a field based picture would fail to efficiently describe such a scenario. Further, the occupation or particle based pictures have advantages in the extraction of physical observables such as particle number, while in the field basis such quantities are harder to obtain. For further discussion on these issues, see [22, 43, 90].

The most performant field amplitude basis algorithms are based on qubitization and for field cutoff k require a number of T -gates that scale as

$$N_{T,amp} = \tilde{O} \left(\frac{|\Omega|^2}{\epsilon_E} [k^2 \Lambda + k M^2] \right) \quad (217)$$

where Λ is a rescaled interaction strength. These bounds provide much better scaling with the error tolerance and slightly better scaling with the volume of the simulation. It is worth noting that this scaling is likely to be close to

the empirical performance of the algorithm because the error estimates for qubitization tend to be much tighter than those of the Trotter formula [30].

This work opens up a number of additional questions. In particular the development of improved methods for computation of the elements of the S -matrix in higher dimensions or for inelastic collisions. Further more detailed estimates of the Trotter error may reveal that the cost of these methods are substantially lower than current estimates. In a similar vein, identification of algorithms that may only require partial error correction can allow these applications to become feasible in early fault-tolerant quantum computers. For example, one possible candidate may be the Trotter based algorithm in the amplitude basis.

Perhaps the most important issue that needs to be addressed involves the theory itself. Scalar ϕ^4 theory is often used as a toy theory rather than one that accurately models realistic scattering events in colliders. Further studies such as ours are needed to consider gate count estimates for more realistic field theories. The direct extension of this work would be to consider complex scalar theories, or vector field theories. The next step towards realistic theories would then be to develop algorithms for boson-fermion coupled theories. It is our hope that this will not only reveal that field theory can be practically simulated on a quantum computer, but also help us understand the ultimate goal of figuring out whether quantum computers can simulate all physically realistic processes in polynomial time.

VII. ACKNOWLEDGMENTS

This work was primarily supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Co-design Center for Quantum Advantage (C2QA) under Contract Number DE-SC0012704. AH acknowledges support from a NSERC Graduate Fellowship. MSA, ER, and SH acknowledge support from the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Superconducting Quantum Materials and Systems Center (SQMS) under contract No. DE-AC02-07CH11359. MSA and SH acknowledge support from USRA NASA Academic Mission Services under contract No. NNA16BD14C with NASA, with this work funded under the NASA-DOE interagency agreement SAA2-403602 governing NASA's work as part of the SQMS center. RK acknowledges support by the U.S. Department of Energy, Office of Basic Energy Sciences, under Contract No. DE-SC0012704. RV is supported by the U.S. Department of Energy, Office of Science, under contract DE- SC0012704. We thank Erik Gustafson, Vladimir Korepin, Robert Pisarski, and Ted Yoder for useful discussions. We thank Marton Lajer both for discussion and for providing the data for the reproduced figures from [47] found in Figs. 1 and 2.

Appendix A: Quantum circuit for simulating in the Field Occupation Basis

In this section we derive the quantum circuits required to simulate the interacting Hamiltonian H_φ (Eq. 48). As explained in Section IV A, we divide the terms in the sum into 4 groups, i.e. $H_\varphi := H_{1\varphi} + H_{2\varphi} + H_{3\varphi} + H_{4\varphi}$. We map the resulting bosonic expressions into qubit space. We use the following two lemmas repeatedly in order to derive the Pauli expressions for the sum in each of these groups (Eq. 49, 50, 52, 53).

Lemma 38. *If \hat{n}_p is the number operator on momentum mode p then for any integer $r \geq 1$ we have,*

$$(\hat{n}_p)^r = \sum_n \frac{n^r}{2} (I_n - Z_n)_p.$$

Proof. We prove the lemma by considering the action of the operators on the LHS and RHS in the Fock basis. The action of the number operator on the LHS is given as

$$\hat{n}_p \sum_n |p, n\rangle = \sum_n n |p, n\rangle, \quad \text{and so} \quad (\hat{n}_p)^r \sum_n |p, n\rangle = \sum_n n^r |p, n\rangle.$$

Since $(I - Z)|0\rangle = 0$ and $(I - Z)|1\rangle = 2|1\rangle$, so given our unary encoding in Eq. 37 we have

$$\left(\sum_n \frac{n^r}{2} (I_n - Z_n)_p \right) \left(\sum_{n'} |p, n'\rangle \right) = \sum_n \frac{n^r}{2} (I_n - Z_n)_p |p, n\rangle = \sum_n n^r |p, n\rangle,$$

which proves the lemma. □

Lemma 39. *If $m \geq 1$ and $r \geq 0$ are integers then we have,*

$$(a_p^\dagger)^m (n_p)^r + (n_p)^r (a_p)^m = \frac{1}{2} \sum_n \sqrt{\frac{(n+m)!}{n!}} n^r (X_n X_{n+m} + Y_n Y_{n+m})_p. \quad (\text{A1})$$

Proof. First, let $r = 0$ and we prove the fact that,

$$(a_p^\dagger)^m + (a_p)^m = \frac{1}{2} \sum_n \sqrt{\frac{(n+m)!}{n!}} (X_n X_{n+m} + Y_n Y_{n+m})_p. \quad (\text{A2})$$

For $m = 1$ we can use the explicit qubit mapping in Eq.38, plug in the operators in Eq. 39 to get the above equation. For $m > 1$ we consider the action of the operators on the LHS and RHS in the Fock basis.

The nonzero matrix elements of a, a^\dagger in this basis are given by $a_{n,n-1} = \sqrt{n}, (a^\dagger)_{n,n+1} = \sqrt{n+1}$. The corresponding action is given as

$$(a_p + a_p^\dagger) \sum_n |p, n\rangle = \sum_n (\sqrt{n+1} |p, n+1\rangle + \sqrt{n} |p, n-1\rangle)$$

The nonzero matrix elements of the ladder operators, raised to power $m \geq 1$, can be explicitly computed to be

$$(a^m)_{n, n-m} = \sqrt{\frac{n!}{(n-m)!}}, \quad ((a^\dagger)^m)_{n, n+m} = \sqrt{\frac{(n+m)!}{n!}}.$$

This gives the action on the Fock basis as

$$((a_p)^m + (a_p^\dagger)^m) \sum_n |n\rangle = \sum_n \left(\sqrt{\frac{(n+m)!}{n!}} |p, n+m\rangle + \sqrt{\frac{n!}{(n-m)!}} |p, n-m\rangle \right). \quad (\text{A3})$$

Also, the action of the Pauli operator on the RHS of Eq. A2 is,

$$\begin{aligned} & \frac{1}{2} \left(\sum_n \sqrt{\frac{(n+m)!}{n!}} (X_n X_{n+m} + Y_n Y_{n+m})_p \right) \sum_n |p, n\rangle \\ &= \frac{1}{2} \sum_{n, n'} \sqrt{\frac{(n+m)!}{n!}} (X_n X_{n+m} + Y_n Y_{n+m}) |p, n'\rangle \\ &= \frac{1}{2} \sum_n \sqrt{\frac{(n+m)!}{n!}} (X_n X_{n+m} + Y_n Y_{n+m}) (|p, n\rangle + |p, n+m\rangle) \\ &= \sum_n \sqrt{\frac{(n+m)!}{n!}} (|p, n+m\rangle + |p, n\rangle) \\ &= \sum_n \left(\sqrt{\frac{(n+m)!}{n!}} |p, n+m\rangle + \sqrt{\frac{n!}{(n-m)!}} |p, n-m\rangle \right). \end{aligned}$$

The second last line is obtained from the fact that $(X_n X_{n+m} + Y_n Y_{n+m})_p |p, n\rangle = 2 |p, n+m\rangle$, $(X_n X_{n+m} + Y_n Y_{n+m})_p |p, n+m\rangle = 2 |p, n\rangle$ and $(X_n X_{n+m} + Y_n Y_{n+m})_p |p, n'\rangle = 0$ if $n' \neq n, n+m$. The last line follows by relabeling the indices in the second summation. Thus when $r = 0$, we have $\varphi(p, n, m, r) = \frac{1}{2} \sqrt{\frac{(n+m)!}{n!}}$.

Now let us consider the case when $r > 0$. We prove that

$$(a_p^\dagger)^m (n_p)^r + (n_p)^r (a_p)^m = \frac{1}{2} \sum_n n^r \sqrt{\frac{(n+m)!}{n!}} (X_n X_{n+m} + Y_n Y_{n+m})_p. \quad (\text{A4})$$

Since $\hat{n}_p |p, n\rangle = n |p, n\rangle$, so the action of the operator on the LHS of the above equation, in the Fock basis is as follows.

$$((a_p)^m (\hat{n}_p)^r + (\hat{n}_p)^r (a_p^\dagger)^m) \sum_n |p, n\rangle = \sum_n \left(\sqrt{\frac{(n+m)!}{n}} n^r |p, n+m\rangle + \sqrt{\frac{n!}{(n-m)!}} (n-m)^r |p, n-m\rangle \right)$$

Similar to the analysis before, we can show that the action of the Pauli operator on the RHS of Eq. A4, in the Fock basis is as follows.

$$\begin{aligned}
& \frac{1}{2} \sum_n n^r \sqrt{\frac{(n+m)!}{n}} (X_n X_{n+m} + Y_n Y_{n+m}) \sum_n |p, n\rangle \\
&= \sum_n \sqrt{\frac{(n+m)!}{n}} n^r (|p, n+m\rangle + |p, n\rangle) \\
&= \sum_n \sqrt{\frac{(n+m)!}{n}} n^r |p, n+m\rangle + \sqrt{\frac{n!}{(n-m)!}} (n-m)^r |p, n-m\rangle
\end{aligned}$$

Thus in this case $\varphi(p, n, m, r) = \frac{1}{2} \sqrt{\frac{(n+m)!}{n}} n^r$ and this proves the lemma. \square

Now we explain how we derive quantum circuits to implement the exponentiated sum in each group i.e. $e^{-iH_{1\varphi t}}$, $e^{-iH_{2\varphi t}}$, $e^{-iH_{3\varphi t}}$ and $e^{-iH_{4\varphi t}}$. We follow the methods described in [26]. Very briefly, we first divide the Pauli terms into mutually commuting sets. For each such set we derive an eigenbasis and from the diagonalized operator we calculate the number of distinct non-zero eigenvalues (ignoring sign), which is equal to the number of (controlled)-rotations we require. We apply some logical reasoning and optimizations to derive the remaining elements of circuits. The resulting Trotter error due to such splitting, has been calculated in Appendix B. We would like to emphasize that the quantum circuits and hence resource estimates depend on the grouping into commuting Paulis and we do not claim to give the optimal grouping in this paper.

Before we consider our 4 Hamiltonian groups, we derive quantum circuits for the exponential of some specific summation of Paulis. We fix some convention and notation. We denote $P_0 := X$, $P_1 := Y$ and for any binary variable v we denote its complement by $\bar{v} := 1 \oplus v$. Consider the following two sums of Paulis that act on $2n$ and $2n+1$ qubits respectively, which we index by $1, 2, \dots, 2n+1$. We denote $Z_{(j)}$ to imply that the operator Z acts on qubit j . When we denote a Pauli by P_j , $j \in \{0, 1\}$ then we do not mention explicitly in the subscript the qubit it acts on, in order to avoid clutter. We assume that the left-most operator is applied on qubit 1, then next one on qubit 2 and so on and this should be clear from the context.

$$\begin{aligned}
T_1 &= \theta \sum_{a_1, \dots, a_n \in \{0, 1\}} P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} \\
T_2 &= \theta \left(\sum_{a_1, \dots, a_n \in \{0, 1\}} P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} \right) (I_{(2n+1)} - Z_{(2n+1)})
\end{aligned} \tag{A5}$$

It is quite clear that the above two terms are sum of mutually commuting Paulis belonging to the following two sets, respectively.

$$\begin{aligned}
\mathcal{G}_1 &= \{P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} : a_j \in \{0, 1\}, j = 1, \dots, n\} \\
\mathcal{G}_2 &= \{P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} Z_{(2n+1)}^b : a_j, b \in \{0, 1\}, j = 1, \dots, n\}
\end{aligned} \tag{A6}$$

Now we derive the eigenbasis for each of these terms.

Lemma 40 (Eigenbasis for \mathcal{G}_1 and \mathcal{G}_2). *Let $w, v_2, \dots, v_{2n} \in \{0, 1\}$. Then the eigenvectors of the Paulis in \mathcal{G}_1 and \mathcal{G}_2 are of the following form, respectively.*

$$|\mathbf{v}_{1, \pm}\rangle = \frac{1}{\sqrt{2}} (|0v_2 \dots v_{2n}\rangle \pm |1\bar{v}_2, \dots, \bar{v}_{2n}\rangle), \quad |\mathbf{v}_{2, \pm}\rangle = \frac{1}{\sqrt{2}} (|0v_2 \dots v_{2n}w\rangle \pm |1\bar{v}_2, \dots, \bar{v}_{2n}w\rangle)$$

Specifically, if $\beta_1 = a_1 v_2 + a_2(v_3 + v_4) + \dots + a_n(v_{2n-1} + v_{2n})$ and $\beta_2 = a_1 v_2 + a_2(v_3 + v_4) + \dots + a_n(v_{2n-1} + v_{2n}) + wb$, then we have the following.

$$\begin{aligned}
P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} |\mathbf{v}_{1, \pm}\rangle &= \pm (-1)^{a_1 + a_2 + \dots + a_n + \beta_1} |\mathbf{v}_{1, \pm}\rangle \\
P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} Z_{(2n+1)}^b |\mathbf{v}_{2, \pm}\rangle &= \pm (-1)^{a_1 + a_2 + \dots + a_n + \beta_2} |\mathbf{v}_{2, \pm}\rangle
\end{aligned}$$

Proof. We have the following.

$$\begin{aligned}
P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} |0v_2 \dots v_{2n}\rangle &= i^{2(a_1+\dots+a_n)} (-1)^{a_1 v_2 + a_2(v_3+v_4) + \dots + a_n(v_{2n-1}+v_{2n})} |1\overline{v_2 \dots v_{2n}}\rangle \\
&= (-1)^{a_1+\dots+a_n+\beta_1} |1\overline{v_2 \dots v_{2n}}\rangle \\
P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} |1\overline{v_2 \dots v_{2n}}\rangle &= i^{2(a_1+\dots+a_n)} (-1)^{a_1+a_1(v_2+1)+a_2(v_3+1+v_4+1) + \dots + a_n(v_{2n-1}+1+v_{2n}+1)} |0v_2 \dots v_{2n}\rangle \\
&= (-1)^{a_1+\dots+a_n+\beta_1} |0v_2 \dots v_{2n}\rangle
\end{aligned}$$

Therefore,

$$P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} |\mathbf{v}_{1,\pm}\rangle = \pm (-1)^{a_1+\dots+a_n+\beta_1} |\mathbf{v}_{1,\pm}\rangle,$$

making $|\mathbf{v}_{1,\pm}\rangle$ an eigenvector for the Paulis in \mathcal{G}_1 . Similarly we can prove that the eigenvectors of the Paulis in \mathcal{G}_2 are of the form $|\mathbf{v}_{2,\pm}\rangle$. \square

It is easy to see that there are $2^{2n-1} \cdot 2 = 2^{2n}$ mutually orthogonal vectors of the form $|\mathbf{v}_{1,\pm}\rangle$ and $2^{2n} \cdot 2 = 2^{2n+1}$ mutually orthogonal vectors of the form $|\mathbf{v}_{2,\pm}\rangle$, and so we have complete eigenbases for the Paulis in \mathcal{G}_1 and \mathcal{G}_2 . Now, we derive diagonalizing circuits for the set of Paulis in \mathcal{G}_1 and \mathcal{G}_2 .

Theorem 41 (Diagonalizing circuit for \mathcal{G}_1 and \mathcal{G}_2). Let $W = \left(\prod_{j=2}^{2n} CNOT_{(1,j)} \right) H_{(1)}$ and $\widetilde{\mathbf{Z}}_1 = Z_{(1)} Z_{(2)}^{a_1} Z_{(3)}^{a_2} Z_{(4)}^{a_2} \dots Z_{(2n-1)}^{a_n} Z_{(2n)}^{a_n}$, $\widetilde{\mathbf{Z}}_2 = Z_{(1)} Z_{(2)}^{a_1} Z_{(3)}^{a_2} Z_{(4)}^{a_2} \dots Z_{(2n-1)}^{a_n} Z_{(2n)}^{a_n} Z_{(2n+1)}^b$, where $a_1, \dots, a_n, b \in \{0, 1\}$. Then,

$$\begin{aligned}
(-1)^{a_1+\dots+a_n} W \widetilde{\mathbf{Z}}_1 W^\dagger &= P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} \in \mathcal{G}_1 \\
(-1)^{a_1+\dots+a_n} W \widetilde{\mathbf{Z}}_2 W^\dagger &= P_{a_1} P_{a_1} P_{a_2} P_{a_2} \dots P_{a_n} P_{a_n} Z_{(2n+1)}^b \in \mathcal{G}_2
\end{aligned}$$

Proof. We prove the theorem by showing that the operators on the LHS and RHS have equivalent actions on an eigenbasis of the Paulis in \mathcal{G}_1 and \mathcal{G}_2 , respectively. Let us first consider the operators in \mathcal{G}_1 . We show the evolution of the eigenvectors in Lemma 40 when the operator on the LHS is applied. We first apply W^\dagger . Let $\beta_1 = a_1 v_2 + a_2(v_3 + v_4) + \dots + a_n(v_{2n-1} + v_{2n})$ and $U_c = \prod_{j=2}^{2n} CNOT_{(1,j)}$.

$$\begin{aligned}
|\mathbf{v}_{1,+}\rangle &\xrightarrow{U_c} \frac{|0v_2 \dots v_{2n}\rangle + |1v_2 \dots v_{2n}\rangle}{\sqrt{2}} \xrightarrow{H_{(1)}} |0v_2 \dots v_{2n}\rangle \xrightarrow{\widetilde{\mathbf{Z}}_1} (-1)^{\beta_1} |0v_2 \dots v_{2n}\rangle \\
&\xrightarrow{H_{(1)}} (-1)^{\beta_1} \frac{|0v_2 \dots v_{2n}\rangle + |1v_2 \dots v_{2n}\rangle}{\sqrt{2}} \xrightarrow{U_c} |\mathbf{v}_{1,+}\rangle
\end{aligned}$$

Also,

$$\begin{aligned}
|\mathbf{v}_{1,-}\rangle &\xrightarrow{U_c} \frac{|0v_2 \dots v_{2n}\rangle - |1v_2 \dots v_{2n}\rangle}{\sqrt{2}} \xrightarrow{H_{(1)}} |1v_2 \dots v_{2n}\rangle \xrightarrow{\widetilde{\mathbf{Z}}_1} -(-1)^{\beta_1} |1v_2 \dots v_{2n}\rangle \\
&\xrightarrow{H_{(1)}} (-1)^{\beta_1} \frac{|0v_2 \dots v_{2n}\rangle - |1v_2 \dots v_{2n}\rangle}{\sqrt{2}} \xrightarrow{U_c} |\mathbf{v}_{1,-}\rangle
\end{aligned}$$

Thus the Paulis in \mathcal{G}_1 and the operators on the LHS of the first equation in the statement of the theorem, have same eigenvalues for the eigenvectors in Lemma 40.

Similarly, we can prove that the operators on the RHS of the second equation in the statement of the theorem have similar action on an eigenbasis (shown in Lemma 40) of the Paulis in \mathcal{G}_2 . This proves the theorem. \square

Using the above theorem, we diagonalize the terms in T_1 and T_2 and re-write them as follows.

$$\begin{aligned}
T_1 &= W \left(\theta Z_{(1)} \sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2+\dots+a_n} Z_{(3)}^{a_2} Z_{(4)}^{a_2} \dots Z_{(2n)}^{a_n} - \theta Z_{(1)} Z_{(2)} \sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2+\dots+a_n} Z_{(3)}^{a_2} Z_{(4)}^{a_2} \dots Z_{(2n)}^{a_n} \right) W^\dagger \\
T_2 &= W \left(\theta Z_{(1)} \sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2+\dots+a_n} Z_{(3)}^{a_2} Z_{(4)}^{a_2} \dots Z_{(2n)}^{a_n} (I_{(2n+1)} - Z_{(2n+1)}) \right. \\
&\quad \left. - \theta Z_{(1)} Z_{(2)} \sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2+\dots+a_n} Z_{(3)}^{a_2} Z_{(4)}^{a_2} \dots Z_{(2n)}^{a_n} (I_{(2n+1)} - Z_{(2n+1)}) \right) W^\dagger
\end{aligned}$$

Then using Lemma 2.3 in [26], the eigenvalues of the sum of Z-operators can be expressed as functions of Boolean variables $x_1, \dots, x_{2n}, x_{2n+1}$, as follows.

$$\begin{aligned}
\phi_1 &= \theta(-1)^{x_1} \sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2 + \dots + a_n} (-1)^{x_3^{a_2} + x_4^{a_2} + \dots + x_{2n}^{a_n}} - \theta(-1)^{x_1 + x_2} \sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2 + \dots + a_n} (-1)^{x_3^{a_2} + x_4^{a_2} + \dots + x_{2n}^{a_n}} \\
&= \theta(-1)^{x_1} (1 - (-1)^{x_2}) \left(\sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2 + \dots + a_n} (-1)^{x_3^{a_2} + x_4^{a_2} + \dots + x_{2n}^{a_n}} \right) \\
\phi_2 &= \theta(-1)^{x_1} (1 - (-1)^{x_2}) (1 - (-1)^{x_{2n+1}}) \left(\sum_{a_2, \dots, a_n \in \{0,1\}} (-1)^{a_2 + \dots + a_n} (-1)^{x_3^{a_2} + x_4^{a_2} + \dots + x_{2n}^{a_n}} \right) \tag{A7}
\end{aligned}$$

Lemma 42. *Let y_1, y_2, \dots, y_{2m} are Boolean variables. Then*

$$\sum_{a_1, \dots, a_m \in \{0,1\}} (-1)^{a_1 + \dots + a_m} (-1)^{y_1^{a_1} + y_2^{a_1} + \dots + y_{2m}^{a_m}} = \prod_{j=1}^m (1 - (-1)^{y_{2j-1} + y_{2j}}).$$

Proof. We use induction to prove this lemma. The base case corresponds to the case when $m = 1$. Then, it is easy to see that

$$(-1)^0 (-1)^{y_1^0 + y_2^0} + (-1)^1 + (-1)^1 (-1)^{y_1^1 + y_2^1} = 1 - (-1)^{y_1 + y_2}.$$

Now we assume that the lemma is true when $m = k - 1$. That is

$$\sum_{a_1, \dots, a_{k-1} \in \{0,1\}} (-1)^{a_1 + \dots + a_{k-1}} (-1)^{y_1^{a_1} + y_2^{a_1} + \dots + y_{2k-2}^{a_{k-1}}} = \prod_{j=1}^{k-1} (1 - (-1)^{y_{2j-1} + y_{2j}}).$$

Then,

$$\begin{aligned}
&\sum_{a_1, \dots, a_k \in \{0,1\}} (-1)^{a_1 + \dots + a_k} (-1)^{y_1^{a_1} + y_2^{a_1} + \dots + y_{2k}^{a_k}} \\
&= \sum_{a_1, \dots, a_{k-1} \in \{0,1\}} (-1)^{a_1 + \dots + a_{k-1}} (-1)^{y_1^{a_1} + \dots + y_{2k-2}^{a_{k-1}}} - \sum_{a_1, \dots, a_{k-1} \in \{0,1\}} (-1)^{a_1 + \dots + a_{k-1}} (-1)^{y_1^{a_1} + \dots + y_{2k-2}^{a_{k-1}} + y_{2k-1} + y_{2k}} \\
&= (1 - (-1)^{y_{2k-1} + y_{2k}}) \prod_{j=1}^{k-1} (1 - (-1)^{y_{2j-1} + y_{2j}}) = \prod_{j=1}^k (1 - (-1)^{y_{2j-1} + y_{2j}}),
\end{aligned}$$

thus proving the lemma. □

Applying the above lemma in Eq. A7 we obtain,

$$\begin{aligned}
\phi_1 &= \theta(-1)^{x_1} (1 - (-1)^{x_2}) \prod_{j=2}^n (1 - (-1)^{x_{2j-1} + x_{2j}}) \\
\phi_2 &= \theta(-1)^{x_1} (1 - (-1)^{x_2}) (1 - (-1)^{x_{2n+1}}) \prod_{j=2}^n (1 - (-1)^{x_{2j-1} + x_{2j}}) \tag{A8}
\end{aligned}$$

Now, $\phi_1 = (-1)^{x_1} 2^n \theta$ when $x_2 = x_{2j-1} \oplus x_{2j} = 1$, where $j = 2, \dots, n$; else it is 0. Similarly, $\phi_2 = (-1)^{x_1} 2^{n+1} \theta$ when $x_2 = x_{2n+1} = x_{2j-1} \oplus x_{2j} = 1$, where $j = 2, \dots, n$; else it is 0. Thus, using Lemma 2.4 in [26] we can implement a circuit for $e^{-iT_1 t}$ and $e^{-iT_2 t}$, using one controlled- R_z . The complete circuits (for one time-step) have been shown in Figure 10 and 11, respectively. In both these circuits we require $2n - 1$ CNOT, controlled on qubit 1 and target on qubit $2 \leq j \leq 2n$, and 2 H gates to implement the diagonalizing circuit W . Then we use $n - 1$ CNOT with control on qubit $2j$ and target on qubit $2j - 1$, where $2 \leq j \leq n$, to test the parity constraints. Then we apply the multi-controlled- R_z to implement the rotation when the parity constraints are satisfied. Thus, the total number of gates required for implementing these exponentials per time step, can be summarized in the following lemma.

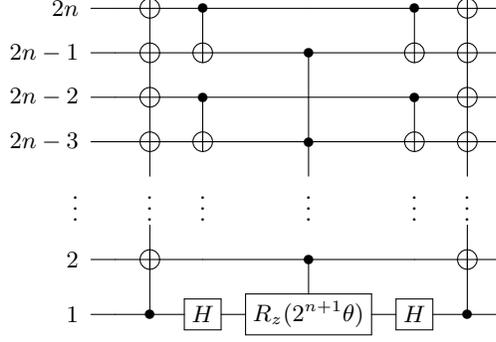


FIG. 10: Circuit implementing the exponential of the sum T_1 (Eq. A5) i.e. $e^{-iT_1 t}$.

Theorem 43. Suppose T_1 and T_2 are sum of Pauli terms, as defined in Eq. A5. Then it is possible to implement $e^{-iT_1 t}$ and $e^{-iT_2 t}$ using $6n - 4$ CNOT and 2 H gates per time step. Additionally, we require one $C^n R_z$ gate (per time step) for $e^{-iT_1 t}$ and one $C^{n+1} R_z$ gate for $e^{-iT_2 t}$.

Here $C^k R_z$ refers to a R_z gate controlled on k qubits. We can decompose $C^k R_z$ into compute-uncompute $C^k X$ pairs, cR_z (single-qubit controlled R_z) and a single ancilla. We can then further decompose $C^k X$ using $4k - 4$ T and $4k - 3$ CNOT gates [75]. If we use the logical AND construction in [76] then we require similar number of T and CNOT for both compute-uncompute pair, but at the cost of using measurement and additional classical resources.

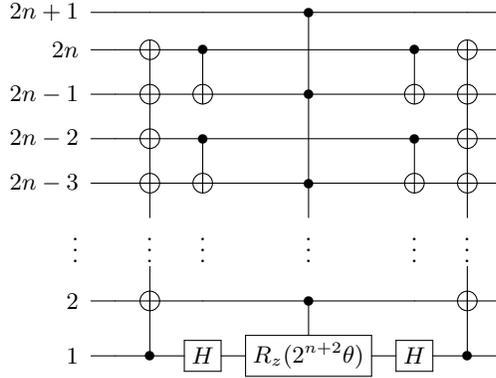


FIG. 11: Circuit implementing the exponential of the sum T_2 (Eq. A5) i.e. $e^{-iT_2 t}$.

We consider another specific sum of Paulis and give the cost of implementing its exponential per time step.

Theorem 44. Let T_3 is a sum of Paulis, as defined below.

$$T_3 = \theta \prod_{j=1}^n (I_{(n)} - Z_{(n)})$$

Then it is possible to implement $e^{-iT_3 t}$ using one $C^n R_z$ gate and an extra ancilla, per time step.

Proof. Using Lemma 2.3 in [26], the eigenvalues of T_3 can be expressed as functions of Boolean variables x_1, \dots, x_n , as follows.

$$\phi_3 = \theta \prod_{j=1}^n (1 - (-1)^{x_j})$$

It is easy to see that $\phi_3 = 2^n \theta$ if and only if $x_j = 1$ for $j = 1, \dots, n$; else it is zero. Thus using Lemma 2.4 in [26], we can implement a circuit for $e^{-iT_3 t}$, using one controlled- R_z . Specifically, we require a $C^n R_z$ gate and an extra ancilla, initialized to $|0\rangle$. The controls select the proper state of the qubits. The target R_z gate is applied on the ancilla. \square

Now, we consider the 4 Hamiltonian groups - $H_{1\varphi}$ (Eq. 49), $H_{2\varphi}$ (Eq. 50), $H_{3\varphi}$ (Eq. 52) and $H_{4\varphi}$ (Eq. 53); and estimate the number of gates and qubits required to implement their exponentials per time step, thus proving Lemma 5-8. As discussed, we have two sources of T-gates, one that comes from the approximately implementable (controlled)-rotation gates and the other that comes from the other exactly implementable components, for example, multi-controlled-X gates. When we report the number of T-gates we do not include the ones in the implementation of the rotation gates. The overall T-count can be easily obtained by plugging in the T-count estimates of (controlled)-rotation gates. More discussion on these bounds have been provided in Section I A.

a. Group I : $\mathbf{H}_{1\varphi}$: We re-write Eq. 49, for convenience. $S_{4\vec{p}} = \{\vec{p} = (p_1, p_2, p_3, p_4) : p_i \in \Gamma; i = 1, \dots, 4 \text{ and } \exists k \in \mathbb{Z} \text{ s.t. } p_3 = p_1 + k, p_4 = p_2 - k\}$ and $S_{4\vec{n}} = \{\vec{n} = (n_1, n_2, n_3, n_4) : n_i = 1, \dots, M; i = 1, \dots, 4\}$ are ordered 4-tuples of momentum modes and momentum states.

$$: H_{1\varphi} := \frac{\lambda}{24 \cdot 64 |\Omega|^3} \sum_{\vec{p} \in S_{4\vec{p}}} \sum_{\vec{n} \in S_{4\vec{n}}} \prod_{(p_j, n_j) \in (\vec{p}, \vec{n})} \sqrt{\frac{n_j + 1}{w_{p_j}}} (X_{p_j, n_j} X_{p_j, n_j+1} + Y_{p_j, n_j} Y_{p_j, n_j+1})$$

We see that for a given \vec{p}, \vec{n} , the sum of the 16 terms in the innermost summation is of the form T_1 (Eq. A5), with $n = 4$. Thus we can apply Theorem 43 and conclude that we require 20 CNOT, 2 H and one $C^4 R_z$ gate to implement its exponential. The $C^4 R_z$ gate can be decomposed further into 1 cR_z , 12 T and 13 additional CNOT. In this case $|S_{4\vec{p}}| \leq \binom{V}{2} \cdot V = \frac{V^2(V-1)}{2}$ and $|S_{4\vec{n}}| = M^4$. The 4! permutations of the elements in \vec{n} lead to the same coefficient, and hence can be summed together, resulting a form similar to T_1 , with $n = 4$. Hence, to implement $e^{-iH_{1\varphi}t}$ we require at most $\frac{M^4 V^2 (V-1)}{48} cR_z$, $\frac{12M^4 V^2 (V-1)}{48}$ T, $\frac{33M^4 V^2 (V-1)}{48}$ CNOT and $\frac{2M^4 V^2 (V-1)}{48}$ H gates per time step. This proves Lemma 5

b. Group II : $\mathbf{H}_{2\varphi}$: The sum of the terms in this group (Eq. 50) is as follows.

$$\begin{aligned} & : H_{2\varphi} : \\ &= \frac{\lambda}{96 |\Omega|} \sum_{p,k} \frac{1}{\omega_p \sqrt{\omega_{p+k} \omega_{p-k}}} \left(\sum_{n_1, n_2, n_3} c_n^{(1)} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_p (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \right. \\ & \quad \left. + \sum_{n_1, n_2, n_3} c_n^{(2)} (I_{n_1} - Z_{n_1})_p (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \right) \end{aligned}$$

where $c_n^{(1)} = \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8}$, $c_n^{(2)} = \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4}$.

For every p, k, n_1, n_2, n_3 there are two sums, one of the form T_1 , with $n = 3$ and the other of the form T_2 , with $n = 2$ (Eq. A5). Using Theorem 43 we require 14+8=22 CNOT, 4 H gates and two $C^3 R_z$ gates to implement the exponential of these sums. Each $C^3 R_z$ can be implemented with 1 cR_z , 8 T and 9 additional CNOT. In this case, $|S_{4\vec{p}}| \leq V^2$, $|S_{4\vec{n}}| \leq M^3$ and there are 3! permutations of the momentum states i.e. n_1, n_2, n_3 that can be summed together, because they have the same coefficients. Hence, to implement $e^{-iH_{2\varphi}t}$ we require at most $\frac{M^3 V^2}{3} cR_z$, $\frac{8M^3 V^2}{3}$ T, $\frac{20M^3 V^2}{3}$ CNOT and $\frac{2M^3 V^2}{3}$ H gates, per time step t , thus proving Lemma 6

c. Group III : $\mathbf{H}_{3\varphi}$: From Eq. 52 we can re-write the following.

$$\begin{aligned} & : H_{3\varphi} : \\ &= \frac{\lambda}{96 |\Omega|} \sum_{p_1, p_2} \frac{1}{\omega_{p_1} \omega_{p_2}} \left(\sum_{n_1, n_2} c_n^{(3)} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1} (X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2} \right. \\ & \quad + \sum_{n_1, n_2} c_n^{(4)} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1} (I_{n_2} - Z_{n_2})_{p_2} + \sum_{n_1, n_2} c_n^{(5)} (X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2} (I_{n_1} - Z_{n_1})_{p_1} \\ & \quad \left. + \sum_{n_1, n_2} n_1 n_2 (I_n - Z_{p_1, n_1} - Z_{p_2, n_2} + Z_{p_1, n_1} Z_{p_2, n_2}) \right) \end{aligned}$$

where $c_n^{(3)} = \frac{\sqrt{(n_1+2)(n_1+1)(n_2+2)(n_2+1)}}{4}$, $c_n^{(4)} = \frac{n_2 \sqrt{(n_1+2)(n_1+1)}}{2}$, $c_n^{(5)} = \frac{n_1 \sqrt{(n_2+2)(n_2+1)}}{2}$.

Here we see that for a given p_1, p_2, n_1, n_2 , there is one sum of the form T_1 with $n = 2$ (Eq. A5), two sums of the form T_2 with $n = 1$ (Eq. A5) and one sum of the form T_3 with $n = 2$ (Theorem 44). Using Theorem 43 we can implement the exponential of the first sum using 8 CNOT, 2 H and one $C^2 R_z$ gates; each of the second type of exponentiated sums (T_2) using 2 CNOT, 2 H and one $C^2 R_z$ gates. Using Theorem 44 we can implement the exponential of the last type of sum (T_3) using one $C^2 R_z$ gate per time step. We can decompose each of the $C^2 R_z$ using 4 T gates, 1 cR_z and

5 CNOT gates. In this case, $|S_{4\bar{p}}| \leq \frac{V(V-1)}{2}$ and $|S_{4\bar{n}}| \leq M^2$ and the $2!$ permutations of each (n_1, n_2) can be summed together. Thus, overall we require $2M^2V(V-1)$ cR_z , $8M^2V(V-1)$ T, $16M^2V(V-1)$ CNOT and $3M^2V(V-1)$ H gates. This proves Lemma 7.

d. Group IV : $H_{4\varphi}$: We re-write the terms in $H_{4\varphi}$ from Eq. 53.

$$\begin{aligned} & : H_{4\varphi} : \\ &= \frac{\lambda}{96} \sum_{p,n} \frac{1}{(\omega_p)^2} \left(\frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2} (X_n X_{n+4} + Y_n Y_{n+4})_p \right. \\ & \quad \left. + 2n \sqrt{(n+2)(n+1)} (X_n X_{n+2} + Y_n Y_{n+2})_p + 3(n^2 - n) (I_n - Z_n)_p \right) \end{aligned}$$

We see that for a given p, n there are two sums of the form T_1 with $n = 1$ (Eq. A5) and the exponentials of these can be implemented with 2 cR_z , 4 H and 4 CNOT (Theorem 43). The exponential of each term of the form $I_n - Z_n$ can be implemented with one R_z gate. In this case $|S_{4\bar{p}}| \leq V$ and $|S_{4\bar{n}}| \leq M$. Thus, overall we require $3MV$ (c)- R_z , $4MV$ H and $4MV$ CNOT gates, proving Lemma 8.

Appendix B: Trotter error when simulating in the Field Occupation Basis

We bound the Trotter error, resulting from approximating the exponentiated sum of Hamiltonian operators by a product of exponentiated Hamiltonian operators. We use the bound given in [67], which shows the dependence on nested commutators. If a Hamiltonian $H = \sum_{\gamma=1}^{\Gamma} H_{\gamma}$ is a sum of Γ fragment Hamiltonians, then $e^{-i\tau H}$ can be approximated by product of exponentials, using the p^{th} order Trotter-Suzuki formula [68], $\mathcal{S}_p(\tau) = e^{-i\tau H} + \mathcal{A}(\tau)$, where $\|\mathcal{A}(\tau)\| \in O(\tilde{\alpha}_{comm} \tau^{p+1})$ if each H_{γ} are Hermitian. Here $\tilde{\alpha}_{comm} = \sum_{\gamma_1, \gamma_2, \dots, \gamma_{p+1}=1}^{\Gamma} \|[H_{\gamma_{p+1}}, \dots, [H_{\gamma_2}, H_{\gamma_1}]]\|$. We use the following bound given in [27]. In this section, $\|\cdot\|$ refers to the spectral norm [67], which is defined as the induced Euclidean norm on the Hamming-Weight 1 subspace \mathcal{S}

$$\|O\| = \max_{|x\rangle \in \mathcal{S}: \|x\|_2=1} \|O|x\rangle\|_2. \quad (\text{B1})$$

This definition corresponds with the Schatten infinity-norm which yields the maximum singular value of a matrix. We also need the following result to simplify the calculation of our norms.

We also make use of the bound.

$$\tilde{\alpha}_{comm} \leq 2^{p-(\mathbf{p}'+1)} \sum_{\gamma_{i_1}, \gamma_{i_2}, \dots, \gamma_{i_{\mathbf{p}'+1}}} \|[H_{\gamma_{\mathbf{p}'+1}}, [\dots, [H_{\gamma_3}, [H_{\gamma_2}, H_{\gamma_1}]] \dots]]\| \left(\sum_{\gamma=1}^{\Gamma} \|H_{\gamma}\| \right)^{p-\mathbf{p}'} \quad [1 \leq \mathbf{p}' \leq p] \quad (\text{B2})$$

In this paper we derive expressions and rigorous bounds for the first order or innermost commutators, i.e. we take $\mathbf{p}' = 1$. In Section IV A we have expressed the occupation basis Hamiltonian as the sum of 5 other Hamiltonians (Eq. 46, 49, 50, 52, 53).

$$H_{occ} = H_0 + H_{1\varphi} + H_{2\varphi} + H_{3\varphi} + H_{4\varphi}$$

Each of these Hamiltonians has been expressed as sum of Pauli operators. First, we re-group these summands into sets of mutually commuting Paulis. Then we derive bounds on the norm of the Hamiltonians. Next, we derive bounds on the pair-wise commutators. Finally, we combine these to derive bound on $\tilde{\alpha}_{comm}$ and hence the p^{th} order Trotter error, thus formally proving Lemma 10.

We make the following observations. Let $P \in \{X, Y, Z\}$ is a Pauli operator. Then,

$$[P_{p,n} P_{p,n+j}, P_{\mathbf{p}',n'} P_{\mathbf{p}',n'+k}] = 0 \quad \text{if } p \neq \mathbf{p}' \text{ or } p = \mathbf{p}' \text{ but } n' \neq n, n+j, n-k, n+j-k. \quad (\text{B3})$$

Let \mathbb{Z} be the set of integers, including 0. We define the following sets.

$$\begin{aligned} S_{10} &= 2\mathbb{Z}; & S_{11} &= 2\mathbb{Z} + 1; \\ S_{20} &= \{1 + 4\mathbb{Z}\} \cup \{2 + 4\mathbb{Z}\}; & S_{21} &= \{3 + 4\mathbb{Z}\} \cup \{4\mathbb{Z}\}; \\ S_{40} &= \{1 + 8\mathbb{Z}\} \cup \{2 + 8\mathbb{Z}\} \cup \{3 + 8\mathbb{Z}\} \cup \{4 + 8\mathbb{Z}\}; & S_{41} &= \{5 + 8\mathbb{Z}\} \cup \{6 + 8\mathbb{Z}\} \cup \{7 + 8\mathbb{Z}\} \cup \{8\mathbb{Z}\}; \end{aligned}$$

We sum H_0 and $H_{4\varphi}$ into one Hamiltonian $H'_{4\varphi}$.

$$\begin{aligned}
& : H'_{4\varphi} : :=: H_{4\varphi} : + : H_0 : \\
& = \frac{\lambda}{96} \sum_{p,n} \frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2\omega_p^2} (X_n X_{n+4} + Y_n Y_{n+4})_p \\
& \quad + \left(\frac{\lambda}{48}\right) \sum_{p,n} \frac{n\sqrt{(n+2)(n+1)}}{\omega_p^2} (X_n X_{n+2} + Y_n Y_{n+2})_p \\
& \quad + \sum_{p,n} \left(\frac{n\omega_p}{2} + \frac{\lambda(n^2-n)}{32(\omega_p)^2}\right) (I_n - Z_n)_p := H_{41} + H_{42} + H_{43} + H_{44} + H_{45};
\end{aligned} \tag{B4}$$

where,

$$\begin{aligned}
H_{41} & = \frac{\lambda}{96} \sum_{p,n \in S_{40}} \frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2\omega_p^2} (X_n X_{n+4} + Y_n Y_{n+4})_p := \frac{\lambda}{96} \sum_{p,n \in S_{40}} A_{1pn}^{(4)} \\
H_{42} & = \frac{\lambda}{96} \sum_{p,n \in S_{41}} \frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2\omega_p^2} (X_n X_{n+4} + Y_n Y_{n+4})_p := \frac{\lambda}{96} \sum_{p,n \in S_{41}} A_{1pn}^{(4)} \\
H_{43} & = \left(\frac{\lambda}{48}\right) \sum_{p,n \in S_{20}} \frac{n\sqrt{(n+2)(n+1)}}{\omega_p^2} (X_n X_{n+2} + Y_n Y_{n+2})_p := \left(\frac{\lambda}{48}\right) \sum_{p,n \in S_{20}} A_{2pn}^{(4)} \\
H_{44} & = \left(\frac{\lambda}{48}\right) \sum_{p,n \in S_{21}} \frac{n\sqrt{(n+2)(n+1)}}{\omega_p^2} (X_n X_{n+2} + Y_n Y_{n+2})_p := \left(\frac{\lambda}{48}\right) \sum_{p,n \in S_{21}} A_{2pn}^{(4)} \\
H_{45} & = \sum_{p,n} \left(\frac{n\omega_p}{2} + \frac{\lambda(n^2-n)}{32(\omega_p)^2}\right) (I_n - Z_n)_p := \sum_{p,n} C_{pn}^{(4)}
\end{aligned} \tag{B5}$$

It is possible to check that each H_{4j} , where $j = 1, \dots, 5$, consists of sum of mutually commuting Pauli operators.

$S_{4n}^{(0)} = \{\vec{n} = (n_1, n_2, n_3, n_4) : \text{Any } 2, 4 \text{ or } 0 \text{ of the } n_j \text{ are in } S_{10} \text{ and the remaining are in } S_{11}; j = 1, \dots, 4\}$
and $S_{4n}^{(1)} = \{\vec{n} = (n_1, n_2, n_3, n_4) : \text{Any } 1 \text{ or } 3 \text{ of the } n_j \text{ are in } S_{10} \text{ and the remaining are in } S_{11}; j = 1, \dots, 4\}$.
Given a certain $\mathbf{p} \in S_{4\vec{p}}$ and $\vec{n} \in S_{4\vec{n}}$, let

$$A_{\mathbf{p}, \vec{n}}^{(1)} := \prod_{(p_j, n_j) \in (\mathbf{p}, \vec{n})} \sqrt{\frac{n_j + 1}{\omega_{p_j}}} (X_{p_j, n_j} X_{p_j, n_j+1} + Y_{p_j, n_j} Y_{p_j, n_j+1}), \tag{B6}$$

then from Eq. 49, we have that we can express the first term from the ϕ^4 term that arises in H_{occ} as the following.

$$\begin{aligned}
: H_{1\varphi} : & = \frac{\lambda}{1536} \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \sum_{\vec{n} \in S_{4\vec{n}}} A_{\mathbf{p}, \vec{n}}^{(1)} \\
& = \frac{\lambda}{1536} \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \sum_{\vec{n} \in S_{4n}^{(0)}} A_{\mathbf{p}, \vec{n}}^{(1)} + \frac{\lambda}{1536} \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \sum_{\vec{n} \in S_{4n}^{(1)}} A_{\mathbf{p}, \vec{n}}^{(1)} \\
& := H_{11} + H_{12}
\end{aligned} \tag{B7}$$

Again, both H_{11} and H_{12} consists of sum of mutually commuting Pauli operators. Let,

$$S_{3n}^{(0)} = \{(n_1, n_2, n_3) : n_1 \in S_{21}, n_2, n_3 \in S_{10}; \text{ or } n_1 \in S_{20}, n_2 \in S_{10}, n_3 \in S_{11}; \tag{B8}$$

$$\text{or } n_1 \in S_{20}, n_2 \in S_{11}, n_3 \in S_{10}; \text{ or } n_1 \in S_{21}, n_2, n_3 \in S_{11}\} \tag{B9}$$

$$S_{3n}^{(1)} = \{(n_1, n_2, n_3) : n_1 \in S_{20}, n_2, n_3 \in S_{11}; \text{ or } n_1 \in S_{21}, n_2 \in S_{11}, n_3 \in S_{10}; \tag{B10}$$

$$\text{or } n_1 \in S_{21}, n_2 \in S_{10}, n_3 \in S_{11}; \text{ or } n_1 \in S_{20}, n_2, n_3 \in S_{10}\} \tag{B11}$$

$$S_{2n}^{(0)} = \{(n_2, n_3) : n_2 \in S_{10}, n_3 \in S_{11}; \text{ or } n_2 \in S_{11}, n_3 \in S_{10}\} \tag{B12}$$

$$S_{2n}^{(1)} = \{(n_2, n_3) : n_2, n_3 \in S_{11}; \text{ or } n_2, n_3 \in S_{10}\} \tag{B13}$$

Given p, k, n_1, n_2, n_3 , let

$$A_{pk\bar{n}}^{(2)} := \frac{c_n^{(1)}}{\omega_p \sqrt{\omega_{p+k} \omega_{p-k}}} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_p (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \quad (\text{B14})$$

and

$$B_{pk\bar{n}}^{(2)} := \frac{c_n^{(2)}}{\omega_p \sqrt{\omega_{p+k} \omega_{p-k}}} (I_{n_1} - Z_{n_1})_p (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \quad (\text{B15})$$

where $c_n^{(1)} = \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8}$ and $c_n^{(2)} = \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4}$. Then, from Eq. 50, we have the following.

$$\begin{aligned} : H_{2\varphi} : &= \frac{\lambda}{96} \sum_{p,k} \sum_{n_1, n_2, n_3} A_{pk\bar{n}}^{(2)} + \frac{\lambda}{96} \sum_{p,k} \sum_{n_1, n_2, n_3} B_{pk\bar{n}}^{(2)} \\ &= \frac{\lambda}{96} \sum_{p,k} \sum_{\substack{(n_1, n_2, n_3) \\ \in S_{3n}^{(0)}}} A_{pk\bar{n}}^{(2)} + \frac{\lambda}{96} \sum_{p,k} \sum_{\substack{(n_1, n_2, n_3) \\ \in S_{3n}^{(1)}}} A_{pk\bar{n}}^{(2)} + \frac{\lambda}{96} \sum_{p,k} \sum_{\substack{n_1 \\ (n_2, n_3) \in S_{2n}^{(0)}}} B_{pk\bar{n}}^{(2)} \\ &\quad + \frac{\lambda}{96} \sum_{p,k} \sum_{\substack{n_1 \\ (n_2, n_3) \in S_{2n}^{(1)}}} B_{pk\bar{n}}^{(2)} \\ &:= H_{21} + H_{22} + H_{23} + H_{24} \end{aligned} \quad (\text{B16})$$

where each H_{2j} , for $j = 1, \dots, 4$, consists of sum of mutually commuting Pauli operators.

Let, $S_{2n}^{(2)} = \{(n_1, n_2) : n_1 \in S_{20}, n_2 \in S_{21}; \text{ or } n_1 \in S_{21}, n_2 \in S_{20}\}$;

and $S_{2n}^{(3)} = \{(n_1, n_2) : n_1, n_2 \in S_{20}; \text{ or } n_1, n_2 \in S_{21}\}$.

Given p_1, p_2, n_1, n_2 , let $A_{\mathbf{p}\bar{n}}^{(3)} = \frac{c_n^{(3)}}{\omega_{p_1} \omega_{p_2}} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1} (X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2}$;
 $B_{1\mathbf{p}\bar{n}}^{(3)} = \frac{c_n^{(4)}}{\omega_{p_1} \omega_{p_2}} (X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1} (I_{n_2} - Z_{n_2})_{p_2}$; $B_{2\mathbf{p}\bar{n}}^{(3)} = \frac{c_n^{(5)}}{\omega_{p_1} \omega_{p_2}} (X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2} (I_{n_1} - Z_{n_1})_{p_1}$; and
 $C_{\mathbf{p}\bar{n}}^{(3)} = \frac{n_1 n_2}{\omega_{p_1} \omega_{p_2}} (I_{n_1} - Z_{n_1})_{p_1} (I_{n_2} - Z_{n_2})_{p_2}$, where $c_n^{(3)} = \frac{\sqrt{(n_1+2)(n_1+1)(n_2+2)(n_2+1)}}{4}$, $c_n^{(4)} = \frac{n_2 \sqrt{(n_1+2)(n_1+1)}}{2}$, $c_n^{(5)} = \frac{n_1 \sqrt{(n_2+2)(n_2+1)}}{2}$. From Eq. 52 we get the following.

$$\begin{aligned} : H_{3\varphi} : &= \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{n_1, n_2} \left(A_{\mathbf{p}\bar{n}}^{(3)} + B_{1\mathbf{p}\bar{n}}^{(3)} + B_{2\mathbf{p}\bar{n}}^{(3)} + C_{\mathbf{p}\bar{n}}^{(3)} \right) \\ &= \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{\substack{(n_1, n_2) \\ \in S_{2n}^{(2)}}} A_{\mathbf{p}\bar{n}}^{(3)} + \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{\substack{(n_1, n_2) \\ \in S_{2n}^{(3)}}} A_{\mathbf{p}\bar{n}}^{(3)} + \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{\substack{n_1 \in S_{20} \\ n_2}} B_{1\mathbf{p}\bar{n}}^{(3)} + \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{\substack{n_1 \in S_{21} \\ n_2}} B_{1\mathbf{p}\bar{n}}^{(3)} \\ &\quad + \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{\substack{n_2 \in S_{20} \\ n_1}} B_{2\mathbf{p}\bar{n}}^{(3)} + \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{\substack{n_2 \in S_{21} \\ n_1}} B_{2\mathbf{p}\bar{n}}^{(3)} + \frac{\lambda}{96} \sum_{p_1, p_2} \sum_{n_1, n_2} C_{\mathbf{p}\bar{n}}^{(3)} \\ &:= H_{31} + H_{32} + H_{33} + H_{34} + H_{35} + H_{36} + H_{37} \end{aligned} \quad (\text{B17})$$

Again, each H_{3j} , where $j = 1, \dots, 7$, consists of sum of mutually commuting Pauli operators.

1. Norm of Hamiltonians

In this subsection we derive bounds on the norm of the operators that appear in the decomposition of H_{occ} . Since we use the triangle inequality it is sufficient to derive bounds on the norm of each $H_{j\varphi}$, where $j = 0, 1, \dots, 4$. Let

$\min_p \omega_p = \omega_{min}$ and $\max_p \omega_p = \omega_{max}$. Due to truncation we have $1 \leq n \leq N$. We first bound the norm of H_0 (Eq. 46).

$$\begin{aligned} \|H_0\| &\leq \left\| \frac{1}{|\Omega|} \sum_{n,p} \frac{n\omega_p}{2} (I_n - Z_n)_p \right\| \leq \max_{n,p} \frac{n\omega_p}{2} \|(I_n - Z_n)_p\| \\ &\leq |\Omega| \omega_{max} \max_n \frac{n}{2} \cdot 2 = \frac{\omega_{max} N}{2} \end{aligned} \quad (\text{B19})$$

Next, we bound the norm of $H_{1\varphi}$ from Eq. 49. We recall that $S_{4\mathbf{p}} = \{(p_1, p_2, p_1 + k, p_2 - k) : p_1, p_2, k \in \mathbb{Z}\}$ and $S_{4\vec{n}} = \{(n_1, n_2, n_3, n_4) : n_i = 0, \dots, N; i = 1, \dots, 4\}$ are 4-tuples of momentum modes and states. Also,

$$|S_{4\mathbf{p}}| \leq \frac{1}{2} \quad (\text{B20})$$

and $|S_{4\vec{n}}| \leq N^4$, as discussed in Appendix A.

$$\begin{aligned} \|H_{1\varphi}\| &\leq \left\| \frac{\lambda}{24 \cdot 64} \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \sum_{\vec{n} \in S_{4\vec{n}}} \prod_{(p_j, n_j) \in (\mathbf{p}, \vec{n})} \sqrt{\frac{n_j + 1}{\omega_{p_j}}} (X_{p_j, n_j} X_{p_j, n_j+1} + Y_{p_j, n_j} Y_{p_j, n_j+1}) \right\| \\ &\leq \frac{\lambda}{24 \cdot 64} \max_{\mathbf{p} \in S_{4\mathbf{p}}} \max_{\vec{n} \in S_{4\vec{n}}} \left\| \prod_{(p_j, n_j) \in (\mathbf{p}, \vec{n})} \sqrt{\frac{n_j + 1}{\omega_{p_j}}} (X_{p_j, n_j} X_{p_j, n_j+1} + Y_{p_j, n_j} Y_{p_j, n_j+1}) \right\| \\ &\leq \frac{\lambda |S_{4\mathbf{p}}|}{24 \cdot 64} \max_{\mathbf{p} \in S_{4\mathbf{p}}} \max_{\vec{n} \in S_{4\vec{n}}} \frac{(N+1)^2}{\omega_{min}^2} \cdot 16 \\ &\leq \frac{\lambda}{24 \cdot 64} \cdot \frac{(N+1)^2}{2\omega_{min}^2} \cdot 16 \\ &= \frac{\lambda(N+1)^2}{192\omega_{min}^2} \end{aligned} \quad (\text{B21})$$

Now, we bound the norm of $H_{2\varphi}$ (Eq. 50). In Appendix A we have shown that the number of possible pairs (p, k) , where $1 \leq p, k \leq |\Omega|$ are integers, is at most $|\Omega|^2$; while the number of tuples (n_1, n_2, n_3) , where $1 \leq n_i \leq N$ are integers, is at most N^3 .

$$\begin{aligned} \|H_{2\varphi}\| &\leq \frac{\lambda}{96} \max_{p,k} \max_{n_1, n_2, n_3} \frac{1}{\omega_p \sqrt{\omega_{p+k} \omega_{p-k}}} \left[\frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8} \|(X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_p \right. \\ &\quad \left. (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \right] + \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4} \|(I_{n_1} - Z_{n_1})_p \\ &\quad \left. (X_{n_2} X_{n_2+1} + Y_{n_2} Y_{n_2+1})_{p+k} (X_{n_3} X_{n_3+1} + Y_{n_3} Y_{n_3+1})_{p-k} \right] \\ &\leq \frac{\lambda}{96} \max_{p,k} \max_{n_1, n_2, n_3} \frac{1}{\omega_{min}^2} \left[\frac{(N+1)^{3/2} (N+2)^{1/2}}{8} \cdot 8 + \frac{N(N+1)}{4} \cdot 8 \right] \\ &\leq \frac{\lambda(N+1) \left(\sqrt{(N+1)(N+2)} + 2N \right)}{96\omega_{min}^2} \end{aligned} \quad (\text{B22})$$

Next, we bound the norm of $H_{3\varphi}$ (Eq. 52). In Appendix A we have shown that the number of possible pairs (p_1, p_2) , where $1 \leq p_1, p_2 \leq |\Omega|$ are unequal integers, is at most $|\Omega|(|\Omega| - 1)$; while the number of tuples (n_1, n_2) , where $1 \leq n_i \leq N$ are integers, is at most N^2 .

$$\begin{aligned}
\|H_{3\varphi}\| &\leq \frac{\lambda}{96|\Omega|^2} \max_{p_1, p_2} \max_{n_1, n_2} \frac{1}{\omega_{p_1} \omega_{p_2}} \left[\frac{\sqrt{(n_1+1)(n_1+2)(n_2+1)(n_2+2)}}{4} \|(X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1}\| \right. \\
&\quad \left. \|(X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2}\| + \frac{n_2 \sqrt{(n_1+2)(n_1+1)}}{2} \|(X_{n_1} X_{n_1+2} + Y_{n_1} Y_{n_1+2})_{p_1} (I_{n_2} - Z_{n_2})_{p_2}\| \right. \\
&\quad \left. + \frac{n_1 \sqrt{(n_2+2)(n_2+1)}}{2} \|(X_{n_2} X_{n_2+2} + Y_{n_2} Y_{n_2+2})_{p_2} (I_{n_1} - Z_{n_1})_{p_1}\| + n_1 n_2 \|(I_{n_1} - Z_{n_1})_{p_1} (I_{n_2} - Z_{n_2})_{p_2}\| \right] \\
&\leq \frac{\lambda}{96} \cdot \frac{1}{\omega_{min}^2} \\
&\quad \cdot \left[\frac{(N+1)(N+2)}{4} \cdot 4 + \frac{N \sqrt{(N+1)(N+2)}}{2} \cdot 4 + \frac{N \sqrt{(N+1)(N+2)}}{2} \cdot 4 + N^2 \cdot 4 \right] \\
&\leq \frac{\lambda}{96\omega_{min}^2} \left(\sqrt{(N+1)(N+2)} + 2N \right)^2 \tag{B23}
\end{aligned}$$

Finally, we bound the norm of $H_{4\varphi}$ given in Eq. 53. First note that as we are summing over p distinct momentum modes the norm can be bounded above by

$$\begin{aligned}
\|H_{4\varphi}\| &\leq \frac{\lambda}{96} \max_{p, n} \frac{1}{\omega_p^2} \left[\frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2} \|(X_n X_{n+4} + Y_n Y_{n+4})_p\| \right. \\
&\quad \left. + 2n \sqrt{(n+2)(n+1)} \|(X_n X_{n+2} + Y_n Y_{n+2})_p\| + 3(n^2 - n) \|(I_n - Z_n)_p\| \right] \\
&\leq \frac{\lambda}{96} \max_{p, n} \frac{1}{\omega_{min}^2} \left[\frac{\sqrt{(N+4)(N+3)(N+2)(N+1)}}{2} \cdot 2 + 2N \sqrt{(N+2)(N+1)} \cdot 2 + 3N^2 \cdot 2 \right] \\
&\leq \frac{\lambda}{96\omega_{min}^2} \left(\sqrt{(N+4)(N+3)(N+2)(N+1)} + 4N \sqrt{(N+2)(N+1)} + 6N^2 \right) \tag{B24}
\end{aligned}$$

2. First level commutators

In this sub-section we compute the first-level or innermost commutators between pairs of Hamiltonians $H_{1\varphi}, H_{2\varphi}, H_{3\varphi}, H_{4\varphi}$, including the commutator between the non-commutator terms of each, using Eq. B5-B18. We use the following results repeatedly. Let us define the adjoint operator $ad_x : y \rightarrow [x, y]$.

Lemma 45 (Decomposition of Commutators [27]). *Let $X_j = \sum_{i_j=1}^{m_j} A_{i_j}^{(j)}$, for $j = 1, \dots, p$, where $A_{i_j}^{(j)}$ are elements from the same ring. Then,*

$$ad_{X_p} ad_{X_{p-1}} \dots ad_{X_3} ad_{X_2} X_1 = \sum_{i_p=1}^{m_p} \sum_{i_{p-1}=1}^{m_{p-1}} \dots \sum_{i_2=1}^{m_2} \sum_{i_1=1}^{m_1} ad_{A_{i_p}^{(p)}} ad_{A_{i_{p-1}}^{(p-1)}} \dots ad_{A_{i_3}^{(3)}} ad_{A_{i_2}^{(2)}} A_{i_1}^{(1)}.$$

a. Commutators within $H_{1\varphi}$: The sum of the commutators within the terms of $H_{1\varphi}$ (Eq. B7) is as follows. In order to have two terms that do not commute, we need to ensure that at least one of the indices match for the momentum modes. The first such terms involve the commutator of terms of the form $[A_{\mathbf{p}, \vec{n}}^{(1)}, A_{\mathbf{p}', \vec{n}'}^{(1)}]$ where $A_{\mathbf{p}, \vec{n}}^{(1)}$ is defined in Eq. B6. We can see that unless at least one component of \mathbf{p} and \mathbf{p}' overlap then the commutator of $[X_{p,n} X_{p,n+1}, Y_{p,m} Y_{p,m+1}]$ is zero. Similarly, if there exists j, k such that none of the conditions $n_j = n'_k, n_j = n'_{k+1}, n_j n'_{k-1}$. Thus for each (\mathbf{p}, \vec{n}) there exists at most $3 \frac{|S_{4\mathbf{p}}|}{|\Omega|} A_{\mathbf{p}, \vec{n}}^{(1)}$ terms that do not commute with it in the commutator expansion. This is due to the fact that there are still two momentum modes to sum over other than the

overlapped mode. This observation leads to the following commutator bound.

$$\begin{aligned} \|[H_{11}, H_{12}]\| &= \frac{1}{|\Omega|^6} \left\| \sum_{\mathbf{p}, \mathbf{p}' \in S_{4\mathbf{p}}} \sum_{\vec{n} \in S_{4n}^{(0)}} \sum_{\vec{n}' \in S_{4n}^{(1)}} \left(\frac{\lambda}{1536} \right)^2 [A_{\mathbf{p}, \vec{n}}^{(1)}, A_{\mathbf{p}', \vec{n}'}^{(1)}] \right\| \\ &= 3 \frac{|S_{4\mathbf{p}}|^2}{|\Omega|^6} \left(\frac{\lambda}{1536} \right)^2 \max_{\ell=1}^4 \max_{\substack{\mathbf{p}, \mathbf{p}': \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\vec{n} \in S_{4n}^{(0)}; \\ \vec{n}' \in S_{4n}^{(1)}}} \|[A_{\mathbf{p}, \vec{n}}^{(1)}, A_{\mathbf{p}', \vec{n}'}^{(1)}]\| \end{aligned} \quad (\text{B25})$$

Now, $A_{\mathbf{p}, \vec{n}}^{(1)}$ and $A_{\mathbf{p}', \vec{n}'}^{(1)}$ are each sum of 16 Pauli operators and each of the Pauli operator $A_{\mathbf{p}, \vec{n}}^{(1)}$ can anti-commute with 8 Pauli operators from $A_{\mathbf{p}', \vec{n}'}^{(1)}$. So if $\omega_{min} = \min_p \omega_p$, then

$$\|[A_{\mathbf{p}, \vec{n}}^{(1)}, A_{\mathbf{p}', \vec{n}'}^{(1)}]\| \leq 16 \cdot 8 \cdot 2 \cdot \max_{\vec{n}, \vec{n}', \mathbf{p}, \beta \mathbf{p}'} \prod_{j=1}^4 \left(\sqrt{\frac{(n_j + 1)(n'_j + 1)}{\omega_{p_j} \omega_{p'_j}}} \right) = 2^8 \frac{(N+1)^4}{\omega_{min}^4}. \quad (\text{B26})$$

Hence, we have from the fact that $|S_{4\mathbf{p}}| \leq 1/2$,

$$\|[H_{11}, H_{12}]\| \leq \left(\frac{\lambda}{16 \cdot 96 \omega_{min}^2} \right)^2 3 \cdot 2^7 (N+1)^4 = \left(\frac{\lambda}{\omega_{min}^2} \right)^2 \left(\frac{(N+1)^4}{3 \cdot 2^{11}} \right). \quad (\text{B27})$$

b. Commutators within $\mathbf{H}_{2\varphi}$: Let $S_{3\mathbf{p}} = \{(p_1, p_2, p_3) : \exists k \in \mathbb{Z} \text{ s.t. } p_2 = p_1 + k, p_3 = p_1 - k\}$. From Eq. B16, norm of the sum of commutators within $H_{2\varphi}$ is.

$$\left\| \sum_{\substack{j,k=1 \\ j \neq k}}^4 [H_{2j}, H_{2k}] \right\| \leq \sum_{\substack{j,k=1 \\ j \neq k}}^4 \|[H_{2j}, H_{2k}]\|$$

We now bound each of the summands in the above equation. In this case, each \mathbf{p} is in $S_{3\mathbf{p}}$ and thus is of the form $(p, p+k, p-k)$, where $p, k \in \mathbb{Z}$. Two of the coordinates uniquely determines the third. So, two tuples $\mathbf{p} = (p, p+k, p-k)$, $\mathbf{p}' = (p', p'+k', p'-k')$ can have either 1 or 3 equal components. $\ell = |\mathbf{p} \cap \mathbf{p}'| = 1$ if either $p = p'$ or $k = k'$. Thus we can have at most $2 \cdot |\Omega| \cdot |\Omega|^2 = 2$ such pairs of $(\mathbf{p}, \mathbf{p}')$. $\ell = 3$ if $p = p'$ and $k = k'$ and thus we can have at most $|\Omega|^2$ such pairs. There is a normalization factor of $|\Omega|^2$ for if there are 1 or 2 overlap δ functions, so these factors reduce to factor of 3.

Also, each $A_{\mathbf{p}k\vec{n}}^{(2)}$ is sum of 8 Pauli operators and each of these operators can anti-commute with at most 4 Pauli operators in $A_{\mathbf{p}'k'\vec{n}'}^{(2)}$. Similarly, $B_{\mathbf{p}k\vec{n}}^{(2)}$ is sum of 8 Pauli operators and each can anti-commute with at most 4 Pauli operators of $A_{\mathbf{p}'k'\vec{n}'}^{(2)}$. Each of the 8 Pauli operators of $B_{\mathbf{p}k\vec{n}}^{(2)}$ can anti-commute with at most 4 Pauli operators of $B_{\mathbf{p}'k'\vec{n}'}^{(2)}$. So,

$$\begin{aligned} \|A_{\mathbf{p}k\vec{n}}^{(2)}, A_{\mathbf{p}'k'\vec{n}'}^{(2)}\| &\leq 8 \cdot 4 \cdot 2 \max_{\vec{n}, \vec{n}', \mathbf{p}, \mathbf{p}', k, k'} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \frac{\sqrt{(n'_1+2)(n'_1+1)(n'_2+1)(n'_3+1)}}{8\omega_{p'} \sqrt{\omega_{p'+k'}\omega_{p'-k'}}} \\ &\leq \frac{(N+1)^3(N+2)}{\omega_{min}^4} \\ \|A_{\mathbf{p}k\vec{n}}^{(2)}, B_{\mathbf{p}'k'\vec{n}'}^{(2)}\| &\leq 8 \cdot 4 \cdot 2 \max_{\vec{n}, \vec{n}', \mathbf{p}, \mathbf{p}', k, k'} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \frac{n'_1 \sqrt{(n'_2+1)(n'_3+1)}}{4\omega_{p'} \sqrt{\omega_{p'+k'}\omega_{p'-k'}}} \\ &\leq \frac{2N(N+1)^{5/2}(N+2)^{1/2}}{\omega_{min}^4} \\ \|B_{\mathbf{p}k\vec{n}}^{(2)}, B_{\mathbf{p}'k'\vec{n}'}^{(2)}\| &\leq 8 \cdot 4 \cdot 2 \max_{\vec{n}, \vec{n}', \mathbf{p}, \mathbf{p}', k, k'} \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \frac{n'_1 \sqrt{(n'_2+1)(n'_3+1)}}{4\omega_{p'} \sqrt{\omega_{p'+k'}\omega_{p'-k'}}} \leq \frac{4N^2(N+1)^2}{\omega_{min}^4}, \end{aligned} \quad (\text{B28})$$

where $\omega_{min} = \min_p \omega_p$. In each of the cases below, if $\ell = 1$ then there can be overlap of Pauli terms on any one of the 3 coordinates of \vec{n} and \vec{n}' ; and if $\ell = 3$ then also there can be overlap on any one of the 3 coordinates or on all the 3 coordinates.

In each of these cases we first apply triangle inequality to expand the sum and then count the number of pairs (\vec{n}, \vec{n}') that satisfy the constraints below the sum. We have already enumerated the number of pairs $(\mathbf{p}, \mathbf{p}')$ when $\ell = 1, 3$; and bounded the norm of the innermost commutator. We first consider the following commutator.

$$\| [H_{21}, H_{22}] \| \leq \left(\frac{\lambda}{96} \right)^2 \sum_{\ell \in \{1,3\}} \sum_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \left\| \sum_{\substack{\vec{n} \in S_{3n}^{(0)} \\ \vec{n}' \in S_{3n}^{(1)}}} [A_{\mathbf{p}k\vec{n}}^{(2)}, A_{\mathbf{p}'k'\vec{n}'}^{(2)}] \right\|. \quad (\text{B29})$$

Using the previous bounds on the number of non-zero commutators

$$\| [H_{21}, H_{22}] \| \leq \left(\frac{\lambda}{96} \right)^2 \frac{1}{\omega_{\min}^4} (N+1)^2 [(N+1)(N+2)]. \quad (\text{B30})$$

Next, consider the following commutator.

$$\| [H_{21}, H_{23}] \| \leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell \in \{1,3\}} \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\vec{n} \in S_{3n}^{(0)} \\ (n'_2, n'_3) \in S_{2n}^{(0)}, n'_1}} \| [A_{\mathbf{p}k\vec{n}}^{(2)}, B_{\mathbf{p}'k'\vec{n}'}^{(2)}] \| \quad (\text{B31})$$

Using Eq. B28, we have

$$\| [H_{21}, H_{23}] \| \leq \left(\frac{\lambda}{96\omega_{\min}^2} \right)^2 2N(N+1)^{5/2}(N+2)^{1/2}. \quad (\text{B32})$$

It is not hard to see that we can bound each of the terms below with arguments similar to $\| [H_{21}, H_{23}] \|$. ad

$$\| [H_{2i}, H_{2j}] \| \leq \left(\frac{\lambda}{96} \right)^2 3 \max_{\ell \in \{1,3\}} \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\vec{n} \in S_{3n}^{(i-1)} \\ (n'_2, n'_3) \in S_{2n}^{(j-3)} \\ n'_1}} \| [A_{\mathbf{p}k\vec{n}}^{(2)}, B_{\mathbf{p}'k'\vec{n}'}^{(2)}] \| \quad [i \in \{1, 2\}; j \in \{3, 4\}] \quad (\text{B33})$$

And hence,

$$\begin{aligned} & \| [H_{21}, H_{23}] \| + \| [H_{21}, H_{24}] \| + \| [H_{22}, H_{23}] \| + \| [H_{22}, H_{24}] \| \\ & \leq \left(\frac{\lambda}{96\omega_{\min}^2} \right)^2 8N \cdot 3(N+1)^{5/2}(N+2)^{1/2}. \end{aligned} \quad (\text{B34})$$

Now we bound the following.

$$\| [H_{23}, H_{24}] \| \leq \left(\frac{\lambda}{96} \right)^2 3 \max_{\ell \in \{1,3\}} \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\vec{n}, \vec{n}'} \| [B_{\mathbf{p}k\vec{n}}^{(2)}, B_{\mathbf{p}'k'\vec{n}'}^{(2)}] \| \quad (\text{B35})$$

$$\| [H_{23}, H_{24}] \| \leq \left(\frac{\lambda}{96\omega_{\min}^2} \right)^2 34N^2(N+1)^2. \quad (\text{B36})$$

Hence we get the following bound on the sum of the commutators within $H_{2\varphi}$.

$$\| \sum_{\substack{j,k=1 \\ j \neq k}}^4 [H_{2j}, H_{2k}] \| \leq \left(\frac{\lambda}{96\omega_{\min}^2} \right)^2 3 \left[4N^2(N+1)^2 + 8N(N+1)^{5/2}(N+2)^{1/2} \right] \quad (\text{B37})$$

c. *Commutators within $\mathbf{H}_{3\varphi}$* : Let $S_{2\mathbf{p}} = \{(p_1, p_2) : p_1 \neq p_2\}$. From Eq. B18, sum of commutators within $H_{3\varphi}$ are as follows.

$$\left\| \sum_{\substack{j,k=1 \\ j \neq k}}^7 [H_{3j}, H_{3k}] \right\| \leq \max_{\substack{j,k=1 \\ j \neq k}}^7 \|[H_{3j}, H_{3k}]\| \quad (\text{B38})$$

Now, $A_{\mathbf{p}\bar{n}}^{(3)}$ is sum of 4 Pauli operators and each of these can anti-commute with at most 2 Pauli operators in $A_{\mathbf{p}'\bar{n}'}^{(3)}$. Each of $B_{1\mathbf{p}\bar{n}}^{(3)}$ and $B_{2\mathbf{p}\bar{n}}^{(3)}$ is sum of 4 Pauli operators, each of which anti-commutes with at most 2 Pauli operators in $A_{\mathbf{p}'\bar{n}'}^{(3)}$. Also, each Pauli operator in $B_{1\mathbf{p}\bar{n}}^{(3)}$ anti-commutes with at most 2 Pauli operators in $B_{2\mathbf{p}'\bar{n}'}^{(3)}$ and $B_{1\mathbf{p}'\bar{n}'}^{(3)}$. Each of the 3 non-identity Pauli operators in $C_{\mathbf{p}\bar{n}}^{(3)}$ can anti-commute with at most 4 Pauli operators in $A_{\mathbf{p}'\bar{n}'}^{(3)}$. Two of the Pauli operators in $C_{\mathbf{p}\bar{n}}^{(3)}$ can anti-commute with at most 4 Pauli operators in each of $B_{1\mathbf{p}'\bar{n}'}^{(3)}$, $B_{2\mathbf{p}'\bar{n}'}^{(3)}$, while the third commutes with all. There are $|\Omega|$ overlaps. So, we have the following.

$$\begin{aligned} \|[A_{\mathbf{p}\bar{n}}^{(3)}, A_{\mathbf{p}'\bar{n}'}^{(3)}]\| &\leq 4 \cdot 2 \cdot 2 \max_{\bar{n}, \bar{n}', \mathbf{p}, \mathbf{p}'} \frac{\sqrt{(n_1+1)(n_1+2)(n_2+1)(n_2+2)}}{4\omega_{p_1}\omega_{p_2}} \frac{\sqrt{(n'_1+1)(n'_1+2)(n'_2+1)(n'_2+2)}}{4\omega_{p'_1}\omega_{p'_2}} \\ &\leq \frac{(N+1)^2(N+2)^2}{\omega_{min}^4} \\ \|[A_{\mathbf{p}\bar{n}}^{(3)}, B_{j\mathbf{p}'\bar{n}'}^{(3)}]\| &\leq 4 \cdot 2 \cdot 2 \max_{\bar{n}, \bar{n}', \mathbf{p}, \mathbf{p}'} \frac{\sqrt{(n_1+1)(n_1+2)(n_2+1)(n_2+2)}}{4\omega_{p_1}\omega_{p_2}} \frac{n'_1 \sqrt{(n'_2+1)(n'_2+2)}}{2\omega_{p'_1}\omega_{p'_2}} \quad [j \in \{1, 2\}] \\ &\leq \frac{2N(N+1)^{3/2}(N+2)^{3/2}}{\omega_{min}^4} \\ \|[B_{j\mathbf{p}\bar{n}}^{(3)}, B_{k\mathbf{p}'\bar{n}'}^{(3)}]\| &\leq 4 \cdot 2 \cdot 2 \max_{\bar{n}, \bar{n}', \mathbf{p}, \mathbf{p}'} \frac{n_2 \sqrt{(n_1+1)(n_1+2)}}{2\omega_{p_1}\omega_{p_2}} \frac{n'_1 \sqrt{(n'_2+1)(n'_2+2)}}{2\omega_{p'_1}\omega_{p'_2}} \\ &\leq \frac{4N^2(N+1)(N+2)}{\omega_{min}^4} \quad [j, k \in \{1, 2\}; j \neq k] \\ \|[A_{\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'\bar{n}'}^{(3)}]\| &\leq 3 \cdot 4 \cdot 2 \max_{\bar{n}, \bar{n}', \mathbf{p}, \mathbf{p}'} \frac{\sqrt{(n_1+1)(n_1+2)(n_2+1)(n_2+2)}}{4\omega_{p_1}\omega_{p_2}} \frac{n'_1 n'_2}{\omega_{p'_1}\omega_{p'_2}} \\ &\leq \frac{6N^2(N+1)(N+2)}{\omega_{min}^4} \\ \|[B_{j\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'\bar{n}'}^{(3)}]\| &\leq 2 \cdot 4 \cdot 2 \max_{\bar{n}, \bar{n}', \mathbf{p}, \mathbf{p}'} \frac{n_1 \sqrt{(n_2+1)(n_2+2)}}{2\omega_{p_1}\omega_{p_2}} \frac{n'_1 n'_2}{\omega_{p'_1}\omega_{p'_2}} \\ &\leq \frac{8N^3(N+1)^{1/2}(N+2)^{1/2}}{\omega_{min}^4} \quad [j \in \{1, 2\}] \end{aligned} \quad (\text{B39})$$

Now we bound each of the summands on the RHS of Inequality B38. First we consider the following.

$$\|[H_{31}, H_{32}]\| \leq \left(\frac{\lambda}{96}\right)^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\bar{n} \in S_{2n}^{(2)} \\ \bar{n}' \in S_{2n}^{(3)}}} \|[A_{\mathbf{p}\bar{n}}^{(3)}, A_{\mathbf{p}'\bar{n}'}^{(3)}]\|$$

using Eq. B39,

$$\|[H_{31}, H_{32}]\| = \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 (N+1)^2(N+2)^2$$

Now we consider the following.

$$\|[H_{31}, H_{33}]\| \leq \left(\frac{\lambda}{96}\right)^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\bar{n} \in S_{2n}^{(2)} \\ n'_1 \in S_{20; n'_2}}} \|[A_{\mathbf{p}\bar{n}}^{(3)}, B_{1\mathbf{p}'\bar{n}'}^{(3)}]\|$$

Using Eq. B39, we have

$$\| [H_{31}, H_{33}] \| = \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 2(N+1)^{3/2}(N+2)^{3/2}.$$

We can also check that the following terms can be bound with arguments similar to $\| [H_{31}, H_{33}] \|$.

$$\begin{aligned} \| [H_{3i}, H_{3j}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{\bar{n} \in S_{2^n}^{(i+2)} \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_1 \in S_{2(j-3)}; n'_2}} \| [A_{\mathbf{p}\bar{n}}^{(3)}, B_{\mathbf{1p}'n'}^{(3)}] \| \quad [i \in \{1, 2\}; j \in \{3, 4\}] \\ \| [H_{3i}, H_{3j}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{\bar{n} \in S_{2^n}^{(i+2)} \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_2 \in S_{2(j-5)}; n'_1}} \| [A_{\mathbf{p}\bar{n}}^{(3)}, B_{\mathbf{2p}'n'}^{(3)}] \| \quad [i \in \{1, 2\}; j \in \{5, 6\}] \end{aligned}$$

Hence we have,

$$\max_{j=1}^2 \max_{\ell=3}^6 \| [H_{3j}, H_{3\ell}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 16(N+1)^{3/2}(N+2)^{3/2}.$$

Now we consider the following two commutators that can be bound with similar arguments.

$$\| [H_{3i}, H_{37}] \| \leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{\bar{n}' \in S_{2^n}^{(i+2)} \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \| [A_{\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'\bar{n}'}^{(3)}] \| \quad [i \in \{1, 2\}]$$

$$\sum_{j=1}^2 \| [H_{3j}, H_{37}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 24N^2(N+1)(N+2). \quad (\text{B40})$$

Next, we consider the following commutators that can be bound with similar arguments.

$$\begin{aligned} \| [H_{33}, H_{34}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{n_1 \in S_{20}, n_2 \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_1 \in S_{21}, n'_2}} \| [B_{\mathbf{1p}\bar{n}}^{(3)}, B_{\mathbf{1p}'n'}^{(3)}] \| \\ \| [H_{33}, H_{35}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{n_1 \in S_{20}, n_2 \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_2 \in S_{20}, n'_1}} [B_{\mathbf{1p}\bar{n}}^{(3)}, B_{\mathbf{2p}'n'}^{(3)}] \\ \| [H_{33}, H_{36}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{n_1 \in S_{20}, n_2 \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_2 \in S_{21}, n'_1}} \| [B_{\mathbf{1p}\bar{n}}^{(3)}, B_{\mathbf{2p}'n'}^{(3)}] \| \\ \| [H_{34}, H_{35}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{n_1 \in S_{21}, n_2 \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_2 \in S_{20}, n'_1}} \| [B_{\mathbf{1p}\bar{n}}^{(3)}, B_{\mathbf{2p}'n'}^{(3)}] \| \\ \| [H_{34}, H_{36}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{n_1 \in S_{21}, n_2 \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_2 \in S_{21}, n'_1}} [B_{\mathbf{1p}\bar{n}}^{(3)}, B_{\mathbf{2p}'n'}^{(3)}] \\ \| [H_{35}, H_{36}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{n_2 \in S_{20}, n_1 \\ |\mathbf{p} \cap \mathbf{p}'| = \ell \\ n'_2 \in S_{21}, n'_1}} \| [B_{\mathbf{2p}\bar{n}}^{(3)}, B_{\mathbf{2p}'n'}^{(3)}] \| \end{aligned}$$

Again, the number of pairs (\vec{n}, \vec{n}') intersecting one one given coordinate is at most N^3 and these are the cases that give a non-zero commutator. Thus, using Eq. B39 we get the following.

$$\max_{j=3}^5 \max_{\ell=j+1}^6 \| [H_{3j}, H_{3\ell}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 48N^2(N+1)(N+2) \quad (\text{B41})$$

Finally, we consider the following group of commutators, which also can be bound with similar arguments.

$$\begin{aligned} \| [H_{33}, H_{37}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \left| \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{n_1 \in S_{20, n_2} \\ \vec{n}'}} \| [B_{1\mathbf{p}\vec{n}}^{(3)}, C_{\mathbf{p}'\vec{n}'}^{(3)}] \| \right| \\ \| [H_{34}, H_{37}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \left| \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{n_1 \in S_{21, n_2} \\ \vec{n}'}} \| [B_{1\mathbf{p}\vec{n}}^{(3)}, C_{\mathbf{p}'\vec{n}'}^{(3)}] \| \right| \\ \| [H_{35}, H_{37}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \left| \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{n_2 \in S_{20, n_1} \\ \vec{n}'}} \| [B_{2\mathbf{p}\vec{n}}^{(3)}, C_{\mathbf{p}'\vec{n}'}^{(3)}] \| \right| \\ \| [H_{36}, H_{37}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \left| \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{n_2 \in S_{21, n_1} \\ \vec{n}'}} \| [B_{2\mathbf{p}\vec{n}}^{(3)}, C_{\mathbf{p}'\vec{n}'}^{(3)}] \| \right| \end{aligned}$$

As before, the number of pairs (\vec{n}, \vec{n}') that give non-zero commutator are the ones that have overlap on a single coordinate and the number of such pairs that intersect on a given coordinate is at most N^3 . Hence, using Eq. B39 we have,

$$\max_{j=3}^6 \| [H_{3j}, H_{37}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 64N^3(N+2)^{1/2}(N+1)^{1/2} \quad (\text{B42})$$

Therefore, we get the following bound on the sum of the commutators within $H_{3\varphi}$.

$$\begin{aligned} \left\| \sum_{\substack{j,k=1 \\ j \neq k}}^7 [H_{3j}, H_{3k}] \right\| &\leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 [2N^2(N+1)(N+2) \\ &\quad + 16N^3(N+1)^{1/2}(N+2)^{1/2}] \end{aligned} \quad (\text{B43})$$

d. Commutators within $H'_{4\varphi}$: From Eq. B5, sum of commutators within $H_{4\varphi}$ is as follows.

$$\left\| \sum_{\substack{j,i=1 \\ j \neq \ell}}^5 [H_{4j}, H_{4i}] \right\| \leq \max_{\substack{j,i=1 \\ j \neq i}}^5 \| [H_{4j}, H_{4i}] \| \quad (\text{B44})$$

Here, each of $A_{1\mathbf{p}\mathbf{n}}^{(4)}$ and $A_{2\mathbf{p}\mathbf{n}}^{(4)}$ are sum of 2 Pauli operators and each can anti-commute with at most 1 Pauli operator of $A_{1\mathbf{p}'\mathbf{n}'}^{(4)}$ and $A_{2\mathbf{p}'\mathbf{n}'}^{(4)}$. $C_{\mathbf{p}\mathbf{n}}^{(4)}$ has a single non-identity Z operator and it can anti-commute with at most 2 Pauli operators in $A_{1\mathbf{p}'\mathbf{n}'}^{(4)}$ and $A_{2\mathbf{p}'\mathbf{n}'}^{(4)}$. Hence, we have the following. The Trotter error also requires $p = \mathbf{p}'$ which occurs $|\Omega|$ times.

$$\begin{aligned}
\| [A_{1pn}^{(4)}, A_{1p'n'}^{(4)}] \| &\leq 2 \cdot 1 \cdot 2 \max_{n,n',p,p'} \left| \frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2\omega_p^2} \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{p'}^2} \right| \\
&= \frac{(N+4)(N+3)(N+2)(N+1)}{2\omega_{min}^4} \\
\| [A_{1pn}^{(4)}, A_{2p'n'}^{(4)}] \| &\leq 2 \cdot 1 \cdot 2 \max_{n,n',p,p'} \left| \frac{\sqrt{(n+4)(n+3)(n+2)(n+1)}}{2\omega_p^2} \frac{n' \sqrt{(n'+2)(n'+1)}}{\omega_{p'}^2} \right| \\
&\leq \frac{2N(N+1)(N+2)(N+3)^{1/2}(N+4)^{1/2}}{\omega_{min}^4} \\
\| [A_{2pn}^{(4)}, A_{2p'n'}^{(4)}] \| &\leq 2 \cdot 1 \cdot 2 \max_{n,n',p,p'} \left| \frac{n \sqrt{(n+2)(n+1)}}{\omega_p^2} \frac{n' \sqrt{(n'+2)(n'+1)}}{\omega_{p'}^2} \right| \\
&= \frac{4N^2(N+1)(N+2)}{\omega_{min}^4}
\end{aligned} \tag{B45}$$

Also,

$$\begin{aligned}
\| [C_{pn}^{(4)}, A_{1p'n'}^{(4)}] \| &\leq 1 \cdot 2 \cdot 2 \max_{n,n',p,p'} \delta_{p,p'} \left[\frac{n\omega_p}{2} + \frac{\lambda(n^2-n)}{32\omega_p^2} \right] \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{p'}^2} \\
&\leq \frac{2(N+1)^{1/2}(N+2)^{1/2}(N+3)^{1/2}(N+4)^{1/2}}{\omega_{min}^2} \max_{n,p} \left[\frac{n\omega_p}{2} + \frac{\lambda n(n-1)}{32\omega_p^2} \right]
\end{aligned}$$

Let $\max_{n,p} \left[\frac{n\omega_p}{2} + \frac{\lambda n(n-1)}{32\omega_p^2} \right]$ occurs when $n = N\beta$ for some fraction $0 \leq \beta \leq 1$, that is a function of $p, \omega_p, \lambda, L, d$. That is,

$$\max_{n,p} \left[\frac{n\omega_p}{2} + \frac{\lambda n(n-1)}{32\omega_p^2} \right] = \frac{N\beta\omega_{max}}{2} + \frac{\lambda N\beta(N\beta-1)}{32\omega_{min}^2} \tag{B46}$$

In the rest of this section, unless stated we assume this fact for simplicity and convenience. We will discuss about our choice of β later, at appropriate place. So,

$$\begin{aligned}
\| [C_{pn}^{(4)}, A_{1p'n'}^{(4)}] \| &\leq \frac{N\sqrt{(N+1)(N+2)(N+3)(N+4)}}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right] \\
\| [C_{pn}^{(4)}, A_{2p'n'}^{(4)}] \| &\leq 1 \cdot 2 \cdot 2 \max_{n,n',p,p'} \left(\frac{n\omega_p}{2} + \frac{\lambda(n^2-n)}{32\omega_p^2} \right) \frac{n' \sqrt{(n'+2)(n'+1)}}{\omega_{p'}^2} \\
&\leq \frac{2N^2\sqrt{(N+1)(N+2)}}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right]
\end{aligned} \tag{B47}$$

Now we bound the following commutator, using the above inequalities.

$$\begin{aligned}
\| [H_{41}, H_{42}] \| &\leq \left(\frac{\lambda}{96} \right)^2 \max_p \max_{\substack{n \in S_{40} \\ n'=n+4}} \| [A_{1pn}^{(4)}, A_{1p'n'}^{(4)}] \| \\
&\leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 (N+1)(N+2)(N+3)(N+4)
\end{aligned} \tag{B48}$$

We consider the following commutators, which can be bounded with similar arguments, using Inequalities B45.

$$\| [H_{4i}, H_{4j}] \| \leq 2 \left(\frac{\lambda}{96} \right)^2 \max_p \max_{\substack{n \in S_{4(i-1)} \\ n' \in S_{2(j-3)}}} \| [A_{1pn}^{(4)}, A_{2p'n'}^{(4)}] \| \quad [i \in \{1, 2\}; j \in \{3, 4\}]$$

In each of the above cases, for a given n there can be only one n' for which there is an overlap between the Pauli terms and so,

$$\max_{j=1}^2 \max_{\ell=3}^4 \| [H_{4j}, H_{4\ell}] \| \leq 16 \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 |\Omega| N(N+1)(N+2) \sqrt{(N+3)(N+4)}.$$

Next, we consider the following and bound the commutators using inequalities B47.

$$\begin{aligned} \|[H_{41}, H_{45}] + [H_{42}, H_{45}]\| &\leq \left(\frac{\lambda}{96}\right)^2 \left| \max_p \max_{\substack{n \in S_{40} \\ n' = n, n+1}} \|[A_{1pn}^{(4)}, C_{pn'}^{(4)}]\| \right. + \left(\frac{\lambda}{96}\right)^2 \left| \max_p \max_{\substack{n \in S_{41} \\ n' = n, n+1}} \|[A_{1pn}^{(4)}, C_{pn'}^{(4)}]\| \right. \\ &\leq 2 \cdot \left(\frac{\lambda}{96\omega_{min}}\right)^2 N \sqrt{(N+1)(N+2)(N+3)(N+4)} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right] \end{aligned} \quad (\text{B49})$$

Now, we consider the following commutators.

$$\begin{aligned} \|[H_{43}, H_{45}] + [H_{44}, H_{45}]\| &\leq \left(\frac{\lambda}{48}\right) \left| \max_p \max_{\substack{n \in S_{20} \\ n' = n, n+1}} \|[A_{2pn}^{(4)}, C_{pn'}^{(4)}]\| \right. + \left(\frac{\lambda}{48}\right)^2 \max_p \max_{\substack{n \in S_{21} \\ n' = n, n+1}} \|[A_{2pn}^{(4)}, C_{pn'}^{(4)}]\| \\ &\leq 2 \cdot \left(\frac{\lambda}{48\omega_{min}^2}\right) 2N^2 \sqrt{(N+1)(N+2)} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right] \end{aligned}$$

Finally, we bound the following commutator using Inequality B45.

$$\begin{aligned} \|[H_{43}, H_{44}]\| &\leq \left(\frac{\lambda}{48}\right)^2 \sum_p \sum_{\substack{n \in S_{20} \\ n' = n+2}} \|[A_{2pn}^{(4)}, A_{2pn'}^{(4)}]\| \\ &\leq 16 \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 N^2 (N+1)(N+2) \end{aligned}$$

Therefore, we get the following bound on the sum of commutators within $H'_{4\varphi}$.

$$\begin{aligned} \left\| \sum_{\substack{j, i=1 \\ j \neq i}}^5 [H_{4j}, H_{4i}] \right\| &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 (N+1)(N+2) \left[(N+3)(N+4) + 16N \sqrt{(N+3)(N+4)} + 16N^2 \right] \\ &\quad + \left(\frac{\lambda}{48\omega_{min}^2}\right) \sqrt{(N+1)(N+2)} \left[\sqrt{(N+3)(N+4)} + 4N \right] \\ &\quad \cdot \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right] \end{aligned} \quad (\text{B50})$$

a. Intergroup Commutators

Now we bound the commutators between $H_{i\varphi}$ and $H_{j\varphi}$, where $i \neq j$. We keep in mind that when referring to Hamiltonians in $H_{1\varphi}, H_{2\varphi}, H_{3\varphi}$, then \mathbf{p}, \vec{n} are 4-tuples, 3-tuples and 2-tuples, respectively. This should be clear from the context.

a. $[\mathbf{H}_{1\varphi}, \mathbf{H}_{2\varphi}]$: Using Lemma 45 in Eq. B7 and B16 we have the following.

$$\|[H_{1\varphi}, H_{2\varphi}]\| \leq \sum_{i=1}^2 \sum_{j=1}^4 \|[H_{1i}, H_{2j}]\| \quad (\text{B51})$$

We now bound each of the summands in the above equation. We first bound the commutators between the terms $A_{\mathbf{p}, \vec{n}}^{(1)}$ in Eq. B7 and $A_{pk\vec{n}}, B_{pk\vec{n}}^{(2)}$ in Eq. B16. $A_{\mathbf{p}, \vec{n}}^{(1)}$ is a sum of 16 Pauli operators and each of these can anti-commute with at most 4 of the 8 Pauli operators in $A_{\mathbf{p}'\mathbf{k}'\vec{n}'}^{(2)}$ and $B_{\mathbf{p}'\mathbf{k}'\vec{n}'}^{(2)}$. We also require the overlap of at least one momentum

mode $\mathbf{p}_i = p' \pm k$. This occurs 2 times. So, we have the following.

$$\begin{aligned}
\| [A_{\mathbf{p}, \vec{n}}^{(1)}, A_{\mathbf{p}', \vec{k}', \vec{n}'} }^{(2)}] \| &\leq 16 \cdot 4 \cdot 2 \cdot \max_{\substack{\vec{n}, \vec{n}' \\ \mathbf{p}, \mathbf{p}', \mathbf{k}'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j + 1}{\omega_{p_j}}} \right) \frac{\sqrt{(n'_1 + 1)(n'_1 + 2)(n'_2 + 1)(n'_3 + 1)}}{8\omega_{\mathbf{p}'} \sqrt{\omega_{\mathbf{p}'+\mathbf{k}'} \omega_{\mathbf{p}'-\mathbf{k}'}}} \\
&\leq \frac{16(N+1)^{7/2}(N+2)^{1/2}}{\omega_{min}^4} \\
\| [A_{\mathbf{p}, \vec{n}}^{(1)}, B_{\mathbf{p}', \vec{k}', \vec{n}'} }^{(2)}] \| &\leq 16 \cdot 4 \cdot 2 \cdot \max_{\substack{\vec{n}, \vec{n}' \\ \mathbf{p}, \mathbf{p}', \mathbf{k}'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j + 1}{\omega_{p_j}}} \right) \frac{n'_1 \sqrt{(n'_2 + 1)(n'_3 + 1)}}{4\omega_{\mathbf{p}'} \sqrt{\omega_{\mathbf{p}'+\mathbf{k}'} \omega_{\mathbf{p}'-\mathbf{k}'}}} \\
&\leq \frac{32N(N+1)^3}{\omega_{min}^4}
\end{aligned} \tag{B52}$$

We can bound $\| [H_{1i}, H_{2j}] \|$, $i, j \in \{1, 2\}$ in a similar fashion where,

$$\| [H_{1i}, H_{2j}] \| \leq \frac{1}{16} \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^3 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ \vec{n}' \in S_{3n}^{(j-1)}}} \| A_{\mathbf{p}, \vec{n}}^{(1)}, A_{\mathbf{p}', \vec{n}'} }^{(2)} \| \quad [i \in \{1, 2\}; j \in \{1, 2\}].$$

We have

$$\| [H_{11}, H_{21}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 8(N+1)^{7/2}(N+2)^{1/2} \tag{B53}$$

and thus,

$$\sum_{i=1}^2 \sum_{j=1}^2 \| [H_{1i}, H_{2j}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 32(N+1)^{7/2}(N+2)^{1/2} \tag{B54}$$

We can also bound $\| [H_{1i}, H_{2j}] \|$, $i \in \{1, 2\}, j \in \{3, 4\}$ with similar arguments where,

$$\| [H_{1i}, H_{2j}] \| \leq \left(\frac{\lambda}{96} \right)^2 \sum_{\ell=1}^3 \sum_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \sum_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ (\vec{n}', n'_3) \in S_{2n}^{(j-3)} \\ n'_1}} \| A_{\mathbf{p}, \vec{n}}^{(1)}, B_{\mathbf{p}', \vec{n}'} }^{(2)} \| \quad [i \in \{1, 2\}; j \in \{3, 4\}]$$

Let us consider $\| [H_{11}, H_{23}] \|$. The conditions for the overlaps on the momentum modes and states that give non-zero commutators are similar to that of $\| [H_{11}, H_{21}] \|$. So,

$$\sum_{i=1}^2 \sum_{j=3}^4 \| [H_{1i}, H_{2j}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 32N(N+1)^3 \tag{B55}$$

and thus plugging in these bounds in Eq. B51 we obtain the following.

$$\| [H_{1\varphi}, H_{2\varphi}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 16N(N+1)^3 \cdot \left[\sqrt{(N+1)(N+2)} + 2N \right] \tag{B56}$$

b. $[H_{1\varphi}, H_{3\varphi}]$: We use Lemma 45 to obtain the commutator between $H_{1\varphi}$ (Eq. B7) and $H_{3\varphi}$ (Eq. B18).

$$\| [H_{1\varphi}, H_{3\varphi}] \| \leq \sum_{i=1}^2 \sum_{j=1}^7 \| [H_{1i}, H_{3j}] \| \tag{B57}$$

We first bound the commutators between $A_{\mathbf{p}, \vec{n}}^{(1)}$ (Eq. B7) and $A_{\mathbf{p}, \vec{n}}^{(3)}$, $B_{1\mathbf{p}, \vec{n}}^{(3)}$, $B_{2\mathbf{p}, \vec{n}}^{(3)}$, $C_{\mathbf{p}, \vec{n}}^{(3)}$ (Eq. B18). $A_{\mathbf{p}, \vec{n}}^{(1)}$ is a sum of 16 Pauli operators, each of which can anti-commute with at most 2 of the 4 Pauli operators in $A_{\mathbf{p}, \vec{n}'}^{(3)}$, $B_{1\mathbf{p}, \vec{n}'}^{(3)}$, $B_{2\mathbf{p}, \vec{n}'}^{(3)}$.

Also, each of them anti-commutes with at most 3 of the non-identity Pauli operators in $C_{\mathbf{p}'\vec{n}'}^{(3)}$. There are 2 momentum overlaps. So we have the following.

$$\begin{aligned}
\|A_{\mathbf{p}\vec{n}}^{(1)}, A_{\mathbf{p}'\vec{n}'}^{(3)}\| &\leq 16 \cdot 2 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \vec{n}, \vec{n}'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j + 1}{\omega_{p_j}}} \right) \cdot \frac{\sqrt{(n'_1 + 1)(n'_1 + 2)(n'_2 + 1)(n'_2 + 2)}}{4\omega_{p'_1}\omega_{p'_2}} \\
&\leq \frac{16(N + 1)^3(N + 2)}{\omega_{min}^4} \\
\|A_{\mathbf{p}\vec{n}}^{(1)}, B_{\mathbf{p}'\vec{n}'}^{(3)}\| &\leq 16 \cdot 2 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \vec{n}, \vec{n}'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j + 1}{\omega_{p_j}}} \right) \cdot \frac{n'_2 \sqrt{(n'_1 + 1)(n'_1 + 2)}}{2\omega_{p'_1}\omega_{p'_2}} \\
&\leq \frac{32N(N + 1)^{5/2}(N + 2)^{1/2}}{\omega_{min}^4} \quad [j \in \{1, 2\}] \\
\|A_{\mathbf{p}\vec{n}}^{(1)}, C_{\mathbf{p}'\vec{n}'}^{(3)}\| &\leq 16 \cdot 3 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \vec{n}, \vec{n}'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j + 1}{\omega_{p_j}}} \right) \cdot \frac{n'_1 n'_2}{\omega_{p'_1}\omega_{p'_2}} \leq \frac{96(N + 1)^2}{\omega_{min}^4}
\end{aligned} \tag{B58}$$

We can bound the following commutators in a similar way.

$$\|[H_{1i}, H_{3j}]\| \leq \frac{\lambda}{8} \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ \vec{n}' \in S_{2n}^{(j+1)}}} \| [A_{\mathbf{p}\vec{n}}^{(1)}, A_{\mathbf{p}'\vec{n}'}^{(3)}] \| \quad [i \in \{1, 2\}; j \in \{1, 2\}]$$

Let us consider $\|[H_{11}, H_{31}]\|$. Again we recap that \mathbf{p} is of the form $(p_1, p_2, p_1 + k, p_2 - k)$, where $p_1, p_2, k \in \mathbb{Z}$ and \mathbf{p}' is of the form (p_1, p_2) , where $p_1, p_2 \in \mathbb{Z}$. Let $\ell = |\mathbf{p} \cap \mathbf{p}'| = 1$ i.e. the momentum modes overlap on 1 index. Then a non-zero commutator can occur if there is intersection among the momentum states \vec{n}, \vec{n}' on that particular index. So,

$$\sum_{i=1}^2 \sum_{j=1}^2 \| [H_{1i}, H_{3j}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 32(N + 1)^3(N + 2) \quad [i \in \{1, 2\}; j \in \{1, 2\}]$$

We consider the following commutators, which have similar bound, using similar arguments.

$$\begin{aligned}
\|[H_{1i}, H_{3j}]\| &\leq \frac{\lambda}{8} \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ n'_1 \in S_{2(j-3)}; n'_2}} \| [A_{\mathbf{p}\vec{n}}^{(1)}, B_{\mathbf{p}'\vec{n}'}^{(3)}] \| \quad [i \in \{1, 2\}; j \in \{3, 4\}] \\
\|[H_{1i}, H_{3j}]\| &\leq \frac{\lambda}{8} \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ n'_2 \in S_{2(j-5)}; n'_1}} \| [A_{\mathbf{p}\vec{n}}^{(1)}, B_{\mathbf{p}'\vec{n}'}^{(3)}] \| \quad [i \in \{1, 2\}; j \in \{5, 6\}]
\end{aligned}$$

In each of the above sums, the conditions for overlap of momentum modes and states in order to give a non-zero commutator is similar to that of $\|[H_{11}, H_{31}]\|$. So,

$$\sum_{i=1}^2 \sum_{j=3}^6 \| [H_{1i}, H_{3j}] \| \leq \left(128 \frac{\lambda}{96\omega_{min}^2} \right)^2 2(N + 1)^{5/2}(N + 2)^{1/2} \tag{B59}$$

Next, we consider the following commutators with similar arguments on their bounds.

$$\|[H_{1i}, H_{36}]\| \leq \frac{\lambda}{8} \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ n'_1, n'_2}} \| [A_{\mathbf{p}\vec{n}}^{(1)}, C_{\mathbf{p}'\vec{n}'}^{(3)}] \| \quad [i \in \{1, 2\}]$$

Again, the conditions for overlap of momentum modes and states in order to give a non-zero commutator is similar to that of $\|[H_{11}, H_{31}]\|$. So,

$$\sum_{i=1}^2 \| [H_{1i}, H_{36}] \| \leq \frac{1}{48} \left(\frac{\lambda}{\omega_{min}^2} \right)^2 (N + 1)^2 \tag{B60}$$

Plugging these bounds in Eq.B57 we obtain,

$$\|[H_{1\varphi}, H_{3\varphi}]\| \leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 64\Omega^4(N+1)^2 \cdot \left[(N+1)(N+2) + 4N\sqrt{(N+1)(N+2)} + 3N^2\right] \quad (\text{B61})$$

c. $[H_{1\varphi}, H'_{4\varphi}]$: Using the definitions of $H_{1\varphi}$, $H_{4\varphi}$ from Eq. B7, B5, respectively and Lemma 45 we obtain,

$$\|[H_{1\varphi}, H_{4\varphi}]\| \leq \sum_{i=1}^2 \sum_{j=1}^5 \|[H_{1i}, H_{4j}]\|. \quad (\text{B62})$$

We first bound the commutators between $A_{\mathbf{p}\vec{n}}^{(1)}$ (Eq. B7) and $A_{1\mathbf{p}n}^{(4)}$, $A_{2\mathbf{p}n}^{(4)}$, $C_{\mathbf{p}n}^{(4)}$ (Eq. B5). $A_{\mathbf{p}\vec{n}}^{(1)}$ is a sum of 16 Pauli operators, each of which can anti-commute with at most 1 of the 2 Pauli operators in $A_{1\mathbf{p}'n'}^{(4)}$, $A_{2\mathbf{p}'n'}^{(4)}$ and the only non-identity Pauli operator in $C_{\mathbf{p}'n'}^{(4)}$. Thus, we have the following.

$$\begin{aligned} \|[A_{\mathbf{p}\vec{n}}^{(1)}, A_{1\mathbf{p}'n'}^{(4)}]\| &\leq 16 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \vec{n}, n'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j+1}{\omega_{p_j}}} \right) \cdot \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{\mathbf{p}'}} \\ &\leq \frac{16(N+1)^{5/2}(N+2)^{1/2}(N+3)^{1/2}(N+4)^{1/2}}{\omega_{min}^4} \\ \|[A_{\mathbf{p}\vec{n}}^{(1)}, A_{2\mathbf{p}'n'}^{(4)}]\| &\leq 16 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \vec{n}, n'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j+1}{\omega_{p_j}}} \right) \cdot \frac{n' \sqrt{(n'+2)(n'+1)}}{\omega_{\mathbf{p}'}} \\ &\leq \frac{32N(N+1)^{5/2}(N+2)^{1/2}}{\omega_{min}^4} \\ \|[A_{\mathbf{p}\vec{n}}^{(1)}, C_{\mathbf{p}'n'}^{(4)}]\| &\leq 16 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \vec{n}, n'}} \left(\prod_{j=1}^4 \sqrt{\frac{n_j+1}{\omega_{p_j}}} \right) \cdot \left(\frac{n'\omega_{\mathbf{p}'}}{2} + \frac{\lambda(n'^2-n)}{32(\omega_{\mathbf{p}'})^2} \right) \\ &\leq \frac{16N(N+1)^2}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right] \end{aligned} \quad (\text{B63})$$

We consider the following commutators which can be bound with similar arguments.

$$\sum_{i=1}^2 \sum_{j=1}^2 \|[H_{1i}, H_{4j}]\| \leq \frac{1}{16} \left(\frac{\lambda}{96}\right)^2 \sum_{i=1}^2 \sum_{j=1}^2 \max_{\mathbf{p} \in S_{4\mathbf{p}}} \max_{\ell=1}^4 \max_{\substack{\mathbf{p}' \\ \mathbf{p}'=p_\ell}} \max_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ n' \in S_{4(j-1)}}} \|[A_{\mathbf{p}\vec{n}}^{(1)}, A_{1\mathbf{p}'n'}^{(4)}]\|$$

In this case, an overlap in the momentum modes can occur if \mathbf{p}' is equal to any of the 3 unique coordinates of \mathbf{p} . There can be at most $4|\Omega|^2$ such pairs $(\mathbf{p}, \mathbf{p}')$.

A non-zero commutator can occur if there is intersection among the momentum states \vec{n}, n' on the index where the modes intersect. Number of such pairs (\vec{n}, n') with overlap on one given coordinate is at most $2N \cdot N^3 = 2N^4$. Thus using Eq. B63, we get

$$\sum_{i=1}^2 \sum_{j=1}^2 \|[H_{1i}, H_{4j}]\| \leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 32(N+1)^{5/2} \sqrt{(N+2)(N+3)(N+4)} \quad (\text{B64})$$

Now we consider the following commutators that can also be bound with similar arguments. The conditions for overlap of momentum modes and states, in order to have a non-zero commutator, is same as before. Thus, bounding the innermost commutators using Eq. B63 we get,

$$\begin{aligned} \sum_{i=1}^2 \sum_{j=3}^4 \|[H_{1i}, H_{4j}]\| &\leq \frac{1}{8} \left(\frac{\lambda}{96}\right)^2 \sum_{i=1}^2 \sum_{j=3}^4 \max_{\mathbf{p} \in S_{4\mathbf{p}}} \max_{\ell=1}^4 \max_{\substack{\mathbf{p}' \\ \mathbf{p}'=p_\ell}} \max_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ n' \in S_{2(j-3)}}} \|[A_{\mathbf{p}\vec{n}}^{(1)}, A_{2\mathbf{p}'n'}^{(4)}]\| \\ &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 128(N+1)^{5/2}(N+2)^{1/2}. \end{aligned}$$

Next, we consider the following commutators with similar arguments for bounding the norm. As before, the conditions for overlap of momentum modes and states, in order to have a non-zero commutator, remains the same. So, using Eq. B63 we get,

$$\begin{aligned} \sum_{i=1}^2 \|[H_{1i}, H_{45}]\| &\leq \frac{\lambda}{16 \cdot 96^4} \cdot \sum_{i=1}^2 \sum_{\mathbf{p} \in S_{4\mathbf{p}}} \sum_{\ell=1}^4 \sum_{\substack{\mathbf{p}' : \\ \mathbf{p}' = p\ell}} \sum_{\substack{\vec{n} \in S_{4n}^{(i-1)} \\ n'}} \|[A_{\mathbf{p}\vec{n}}^{(1)}, C_{\mathbf{p}'n'}^{(4)}]\| \\ &\leq \left(\frac{\lambda}{96}\right) \cdot \frac{16N(N+1)^2}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right] \end{aligned}$$

Finally, plugging the above bounds in Eq. B62 we obtain the following.

$$\begin{aligned} \|[H_{1\varphi}, H_{4\varphi}]\| &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 32(N+1)^{5/2}(N+2)^{1/2} \left[\sqrt{(N+3)(N+4)} + 4N \right] \\ &\quad + \left(\frac{\lambda}{96\omega_{min}^2}\right) 16N(N+1)^2 \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right] \end{aligned} \quad (\text{B65})$$

d. $[\mathbf{H}_{2\varphi}, \mathbf{H}_{3\varphi}]$: Applying Lemma 45 to the sum in Eq. B16 ($H_{2\varphi}$) and Eq. B18 ($H_{3\varphi}$) we have,

$$\|[H_{2\varphi}, H_{3\varphi}]\| \leq \sum_{i=1}^4 \sum_{j=1}^7 \|[H_{2i}, H_{3j}]\| \quad (\text{B66})$$

As before, we first bound the following commutators between the following operators, obtained from Eq. B16 and Eq. B18. We observe that $A_{pk\vec{n}}^{(2)}, B_{pk\vec{n}}^{(2)}$ is a sum of 8 Pauli operators, each of which anti-commutes with at most 2 of the 4 Pauli operators in $A_{\mathbf{p}'\vec{n}'}^{(3)}, B_{1\mathbf{p}'\vec{n}'}^{(3)}, B_{2\mathbf{p}'\vec{n}'}^{(3)}$. Also, each of them anti-commutes with at most 3 of the non-identity

Pauli operators in $C_{\mathbf{p}'\vec{n}'}^{(3)}$.

$$\begin{aligned}
\| [A_{pk\vec{n}}^{(2)}, A_{\mathbf{p}'\vec{n}'}^{(3)}] \| &\leq 8 \cdot 2 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \vec{n},\vec{n}'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p\sqrt{\omega_{p+k}\omega_{p-k}}} \cdot \frac{\sqrt{(n'_1+2)(n'_1+1)(n'_2+2)(n'_2+1)}}{4\omega_{p'_1}\omega_{p'_2}} \\
&\leq \frac{(N+1)^{5/2}(N+2)^{3/2}}{\omega_{min}^4} \\
\| [A_{pk\vec{n}}^{(2)}, B_{j\mathbf{p}'\vec{n}'}^{(3)}] \| &\leq 8 \cdot 2 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \vec{n},\vec{n}'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p\sqrt{\omega_{p+k}\omega_{p-k}}} \cdot \frac{n'_1\sqrt{(n'_2+2)(n'_2+1)}}{2\omega_{p'_1}\omega_{p'_2}} \quad [j \in \{1,2\}] \\
&\leq \frac{2N(N+1)^2(N+2)}{\omega_{min}^4} \\
\| [A_{pk\vec{n}}^{(2)}, C_{\mathbf{p}'\vec{n}'}^{(3)}] \| &\leq 8 \cdot 3 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \vec{n},\vec{n}'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p\sqrt{\omega_{p+k}\omega_{p-k}}} \cdot \frac{n'_1 n'_2}{\omega_{p'_1}\omega_{p'_2}} \\
&\leq \frac{6N^2(N+1)^{3/2}(N+2)^{1/2}}{\omega_{min}^4} \\
\| [B_{pk\vec{n}}^{(2)}, A_{\mathbf{p}'\vec{n}'}^{(3)}] \| &\leq 8 \cdot 2 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \vec{n},\vec{n}'}} \frac{n_1\sqrt{(n_2+1)(n_3+1)}}{4\omega_p\sqrt{\omega_{p-k}\omega_{p+k}}} \cdot \frac{\sqrt{(n'_1+2)(n'_1+1)(n'_2+2)(n'_2+1)}}{4\omega_{p'_1}\omega_{p'_2}} \\
&\leq \frac{2N(N+1)^2(N+2)}{\omega_{min}^4} \\
\| [B_{pk\vec{n}}^{(2)}, B_{j\mathbf{p}'\vec{n}'}^{(3)}] \| &\leq 8 \cdot 2 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \vec{n},\vec{n}'}} \frac{n_1\sqrt{(n_2+1)(n_3+1)}}{4\omega_p\sqrt{\omega_{p-k}\omega_{p+k}}} \cdot \frac{n'_1\sqrt{(n'_2+2)(n'_2+1)}}{2\omega_{p'_1}\omega_{p'_2}} \quad [j \in \{1,2\}] \\
&\leq \frac{4N^2(N+1)^{3/2}(N+2)^{1/2}}{\omega_{min}^4} \\
\| [B_{pk\vec{n}}^{(2)}, C_{\mathbf{p}'\vec{n}'}^{(3)}] \| &\leq 8 \cdot 3 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \vec{n},\vec{n}'}} \frac{n_1\sqrt{(n_2+1)(n_3+1)}}{4\omega_p\sqrt{\omega_{p-k}\omega_{p+k}}} \cdot \frac{n'_1 n'_2}{\omega_{p'_1}\omega_{p'_2}} \\
&\leq \frac{12N^3(N+1)}{\omega_{min}^4} \tag{B67}
\end{aligned}$$

In this case we recall that the momentum modes of the operators in $H_{2\varphi}$ are 3-tuples of the form $\mathbf{p} = (p, p+k, p-k)$ and can be denoted by two integers p, k . The momentum modes of the operators in $H_{3\varphi}$ are 2-tuples of the form $\mathbf{p}' = (p_1, p_2)$, where $p_1 \neq p_2$, and can also be represented by two integers. So \mathbf{p} and \mathbf{p}' can intersect non-trivially or have common values in 1 or 2 coordinates. We consider the following sum. With one momentum overlap, there are still $|\Omega|^2$ possibilities for the two free momentums, giving possibilities.

$$\sum_{i=1}^2 \sum_{j=1}^2 \| [H_{2i}, H_{3j}] \| \leq \sum_{i=1}^2 \sum_{j=1}^2 \left(\frac{\lambda}{96} \right)^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| = \ell}} \max_{\substack{\vec{n} \in S_{3n}^{(i-1)} \\ \vec{n}' \in S_{2n}^{(j+1)}}} \| [A_{\mathbf{p}\vec{n}}^{(2)}, A_{\mathbf{p}'\vec{n}'}^{(3)}] \|$$

$$\sum_{i=1}^2 \sum_{j=1}^2 \| [H_{2i}, H_{3j}] \| \leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 48(N+1)^{5/2}(N+2)^{3/2}$$

Next, we consider the following sum of commutators, which be bound in a similar fashion. The conditions of overlap of the momentum modes and states, in order to have a non-zero commutator is same as before. So, using Eq. B67 to

bound the innermost commutators, we have,

$$\begin{aligned}
& \sum_{i=1}^2 \sum_{j=3}^4 \|[H_{2i}, H_{3j}]\| + \sum_{i=1}^2 \sum_{j=5}^6 \|[H_{2i}, H_{3j}]\| \\
& \leq \left(\frac{\lambda}{96}\right)^2 \sum_{i=1}^2 \sum_{\ell=1}^2 \left[\sum_{j=3}^4 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\bar{n} \in S_{3n}^{(i-1)} \\ n'_1 \in S_{2(j-3)}; n'_2}} \|[A_{\mathbf{p}\bar{n}}^{(2)}, B_{1\mathbf{p}'n'}^{(3)}]\| + \sum_{j=5}^6 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\bar{n} \in S_{3n}^{(i-1)} \\ n'_2 \in S_{2(j-5)}; n'_1}} \|[A_{\mathbf{p}\bar{n}}^{(2)}, B_{2\mathbf{p}'n'}^{(3)}]\| \right] \\
& \leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 192N(N+1)^2(N+2)
\end{aligned}$$

Now we consider the following commutators, which again can be bound with similar arguments and we use Eq. B67 to bound the innermost commutators.

$$\begin{aligned}
\sum_{i=1}^2 \|[H_{2i}, H_{37}]\| & \leq \left(\frac{\lambda}{96}\right)^2 \sum_{i=1}^2 \sum_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{\bar{n} \in S_{3n}^{(i-1)} \\ n'}} \|[A_{\mathbf{p}\bar{n}}^{(2)}, C_{\mathbf{p}'n'}^{(3)}]\| \\
& \leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 144N(N+1)^{3/2}(N+2)^{1/2}
\end{aligned} \tag{B68}$$

Till now, we count the commutators between H_{21} , H_{22} and H_{3j} , where $j = 1, \dots, 7$. Now, we consider the commutators between H_{23} , H_{24} and each H_{3j} . We observe that the grouping of commutators that can be bound with similar arguments are same as before. Also, the conditions for overlap of momentum modes and momentum states, in order to have a non-zero overlap, are also similar. The innermost commutators change and they can be bound using Eq. B67. Thus we have,

$$\begin{aligned}
\sum_{i=3}^4 \sum_{j=1}^2 \|[H_{2i}, H_{3j}]\| & \leq |\Omega| \sum_{i=3}^4 \sum_{j=1}^2 \left(\frac{\lambda}{96}\right)^2 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{(n_2, n_3) \in S_{2n}^{(i-3)} \\ n_1, n'_1 \in S_{2n}^{(j+1)}}} \|[B_{\mathbf{p}\bar{n}}^{(2)}, A_{\mathbf{p}'n'}^{(3)}]\| \\
& \leq \frac{1}{96} \left(\frac{\lambda}{\omega_{min}^2}\right)^2 |\Omega| N(N+1)^2(N+2);
\end{aligned} \tag{B69}$$

and

$$\begin{aligned}
& \sum_{i=3}^4 \sum_{j=3}^4 \|[H_{2i}, H_{3j}]\| + \sum_{i=3}^4 \sum_{j=5}^6 \|[H_{2i}, H_{3j}]\| \\
& \leq \left(\frac{\lambda}{96}\right)^2 \sum_{i=3}^4 \sum_{\ell=1}^2 \left[\sum_{j=3}^4 \sum_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \sum_{\substack{(n_2, n_3) \in S_{2n}^{(i-3)}; n_1 \\ n'_1 \in S_{2(j-3)}; n'_2}} \|[B_{\mathbf{p}\bar{n}}^{(2)}, B_{1\mathbf{p}'n'}^{(3)}]\| + \sum_{j=5}^6 \sum_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \sum_{\substack{(n_2, n_3) \in S_{3n}^{(i-3)}; n_1 \\ n'_2 \in S_{2(j-5)}; n'_1}} \|[B_{\mathbf{p}\bar{n}}^{(2)}, B_{2\mathbf{p}'n'}^{(3)}]\| \right] \\
& \leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 384N^2(N+1)^{3/2}(N+2)^{1/2};
\end{aligned} \tag{B70}$$

and finally

$$\begin{aligned}
\sum_{i=3}^4 \|[H_{2i}, H_{37}]\| & \leq \left(\frac{\lambda}{96}\right)^2 \sum_{i=3}^4 \max_{\substack{\mathbf{p}, \mathbf{p}' \\ |\mathbf{p} \cap \mathbf{p}'| \\ = \ell}} \max_{\substack{(n_2, n_3) \in S_{2n}^{(i-3)} \\ n_1; n'_1}} \|[B_{\mathbf{p}\bar{n}}^{(2)}, C_{\mathbf{p}'n'}^{(3)}]\| \\
& \leq \left(\frac{\lambda}{96}\right)^2 \frac{288N^3(N+1)}{\omega_{min}^4}
\end{aligned}$$

Therefore, plugging in Eq. B66 we have the following.

$$\|[H_{2\varphi}, H_{3\varphi}]\| \leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 \left[48N^2(N+1)^{3/2}(N+2)^{1/2} + 288N^3(N+1) \right] \tag{B71}$$

e. $[\mathbf{H}_{2\varphi}, \mathbf{H}'_{4\varphi}]$: We consider the definitions of $H_{2\varphi}$ and $H_{4\varphi}$, as given in Eq. B16 and B5, respectively. Applying Lemma 45 we obtain

$$\|[\mathbf{H}_{2\varphi}, \mathbf{H}_{4\varphi}]\| \leq \sum_{i=1}^4 \sum_{j=1}^5 \|[\mathbf{H}_{2i}, \mathbf{H}_{4j}]\|. \quad (\text{B72})$$

We first bound the commutators between the following operators in the definition of $H_{2\varphi}$ and $H_{4\varphi}$. Both $A_{pk\bar{n}}^{(2)}$ and $B_{pk\bar{n}}^{(2)}$ are sum of 8 Pauli operators, each of which anti-commute with at most 1 of the 2 Pauli operators in $A_{1\mathbf{p}'n'}^{(4)}$; $A_{2\mathbf{p}'n'}^{(4)}$ and the single non-identity Pauli operator in $C_{\mathbf{p}'n'}^{(4)}$. Hence, we have the following, where we leave the factors of ω till the final max expression for simplicity.

$$\begin{aligned} \| [A_{pk\bar{n}}^{(2)}, A_{1\mathbf{p}'n'}^{(4)}] \| &\leq 8 \cdot 1 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \bar{n},n'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \cdot \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{\mathbf{p}'}} \\ &\leq \frac{(N+1)^2(N+2)\sqrt{(N+3)(N+4)}}{\omega_{min}^4} \\ \| [A_{pk\bar{n}}^{(2)}, A_{2\mathbf{p}'n'}^{(4)}] \| &\leq 8 \cdot 1 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \bar{n},n'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \cdot \frac{n' \sqrt{(n'+2)(n'+1)}}{\omega_{\mathbf{p}'}} \\ &\leq \frac{2N(N+1)^2(N+2)}{\omega_{min}^4} \\ \| [A_{pk\bar{n}}^{(2)}, C_{\mathbf{p}'n'}^{(4)}] \| &\leq 8 \cdot 1 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \bar{n},n'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+1)(n_3+1)}}{8\omega_p \sqrt{\omega_{p+k}\omega_{p-k}}} \cdot \left(\frac{n'\omega_{\mathbf{p}'}}{2} + \frac{\lambda(n'^2 - n')}{32\omega_{\mathbf{p}'}} \right) \\ &\leq \frac{N(N+1)^{3/2}(N+2)^{1/2}}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta - 1)}{16\omega_{min}^2} \right] \\ \| [B_{pk\bar{n}}^{(2)}, A_{1\mathbf{p}'n'}^{(4)}] \| &\leq 8 \cdot 1 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \bar{n},n'}} \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4\omega_p \sqrt{\omega_{p-k}\omega_{p+k}}} \cdot \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{\mathbf{p}'}} \\ &\leq \frac{2N(N+1)^{3/2}\sqrt{(N+2)(N+3)(N+4)}}{\omega_{min}^4} \end{aligned} \quad (\text{B73})$$

$$\begin{aligned} \| [B_{pk\bar{n}}^{(2)}, A_{2\mathbf{p}'n'}^{(4)}] \| &\leq 8 \cdot 1 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \bar{n},n'}} \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4\omega_p \sqrt{\omega_{p-k}\omega_{p+k}}} \cdot \frac{n' \sqrt{(n'+2)(n'+1)}}{\omega_{\mathbf{p}'}} \\ &\leq \frac{4N^2(N+1)^{3/2}(N+2)^{1/2}}{\omega_{min}^4} \\ \| [B_{pk\bar{n}}^{(2)}, C_{\mathbf{p}'n'}^{(4)}] \| &\leq 8 \cdot 1 \cdot 2 \cdot \max_{\substack{p,k,\mathbf{p}' \\ \bar{n},n'}} \frac{n_1 \sqrt{(n_2+1)(n_3+1)}}{4\omega_p \sqrt{\omega_{p-k}\omega_{p+k}}} \cdot \left(\frac{n'\omega_{\mathbf{p}'}}{2} + \frac{\lambda(n'^2 - n')}{32\omega_{\mathbf{p}'}} \right) \\ &\leq \frac{2N^2(N+1)}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta - 1)}{16\omega_{min}^2} \right] \end{aligned} \quad (\text{B74})$$

We first bound the following commutators between H_{21} , H_{22} and H_{4j} , where $j = 1, \dots, 5$. Here there are $|\Omega|$ overlaps and $|\Omega|$ possible options with one free variable. Using Eq. B74 we have

$$\begin{aligned} \sum_{i=1}^2 \sum_{j=1}^2 \| [H_{2i}, H_{4j}] \| &\leq \sum_{i=1}^2 \sum_{j=1}^2 \left(\frac{\lambda}{96} \right)^2 \max_{\mathbf{p}} \max_{\ell=1}^3 \max_{\substack{\mathbf{p}': \\ \mathbf{p}'=\ell}} \max_{\substack{\bar{n} \in S_{3n}^{(i-1)} \\ n' \in S_{4(j-1)}}} \| [A_{\mathbf{p}\bar{n}}^{(2)}, A_{1\mathbf{p}'n'}^{(4)}] \| \\ &\leq \left(\frac{\lambda}{96\omega_{min}^2} \right)^2 24(N+1)^2(N+2)\sqrt{(N+3)(N+4)}; \end{aligned}$$

and

$$\begin{aligned} \sum_{i=1}^2 \sum_{j=3}^4 \|[H_{2i}, H_{4j}]\| &\leq |\Omega| \sum_{i=1}^2 \sum_{j=3}^4 \left(\frac{\lambda}{96}\right)^2 \max_{\mathbf{p}} \max_{\ell=1}^3 \max_{\substack{\mathbf{p}': \\ \mathbf{p}'=p\ell}} \max_{\substack{\bar{n} \in S_{3n}^{(i-1)} \\ n' \in S_{2(j-3)}}} \|[A_{\mathbf{p}\bar{n}}^{(2)}, A_{2\mathbf{p}'n'}^{(4)}]\| \\ &\leq \frac{1}{96} \left(\frac{\lambda}{\omega_{min}^2}\right)^2 N(N+1)^2(N+2); \end{aligned}$$

and finally

$$\begin{aligned} \sum_{i=1}^2 \|[H_{2i}, H_{45}]\| &\leq \sum_{i=1}^2 \left(\frac{\lambda}{96}\right) \max_{\mathbf{p}} \max_{\ell=1}^3 \max_{\substack{\mathbf{p}': \\ \mathbf{p}'=p\ell}} \max_{\substack{\bar{n} \in S_{3n}^{(i-1)} \\ n'}} \|[A_{\mathbf{p}\bar{n}}^{(2)}, C_{\mathbf{p}'n'}^{(4)}]\| \\ &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right) 12N^2(N+1)^{3/2}(N+2)^{1/2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2}\right]. \end{aligned}$$

Now, we consider the commutators between H_{23} , H_{24} and each H_{4j} , where $j = 1, \dots, 5$. We observe that the grouping of commutators that can be bound with similar arguments are same as before. Also, the conditions for overlap of momentum modes and momentum states, in order to have a non-zero overlap, are also similar. The innermost commutators change and they can be bound using Eq. B74. Hence we have

$$\begin{aligned} \sum_{i=3}^4 \sum_{j=1}^2 \|[H_{2i}, H_{4j}]\| &\leq \sum_{i=3}^4 \sum_{j=1}^2 \left(\frac{\lambda}{96}\right)^2 \max_{\mathbf{p}} \max_{\ell=1}^3 \max_{\substack{\mathbf{p}': \\ \mathbf{p}'=p\ell}} \max_{\substack{(n_2, n_3) \in S_{2n}^{(i-3)} \\ n_1, n' \in S_{4(j-1)}}} \|[B_{\mathbf{p}\bar{n}}^{(2)}, A_{1\mathbf{p}'n'}^{(4)}]\| \\ &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 48N(N+1)^{3/2} \sqrt{(N+2)(N+3)(N+4)}; \end{aligned}$$

and

$$\begin{aligned} \sum_{i=3}^4 \sum_{j=3}^4 \|[H_{2i}, H_{4j}]\| &\leq \sum_{i=3}^4 \sum_{j=3}^4 \left(\frac{\lambda}{96}\right)^2 \sum_{\mathbf{p}} \sum_{\ell=1}^3 \sum_{\substack{\mathbf{p}': \\ \mathbf{p}'=p\ell}} \sum_{\substack{(n_2, n_3) \in S_{2n}^{(i-3)} \\ n_1, n' \in S_{2(j-3)}}} \|[B_{\mathbf{p}\bar{n}}^{(2)}, A_{2\mathbf{p}'n'}^{(4)}]\| \\ &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 192|\Omega|N^2(N+1)^{3/2}(N+2)^{1/2}; \end{aligned}$$

and finally

$$\begin{aligned} \sum_{i=3}^4 \|[H_{2i}, H_{45}]\| &\leq \sum_{i=3}^4 \left(\frac{\lambda}{96}\right) \Omega^4 \max_{\mathbf{p}} \max_{\ell=1}^3 \max_{\substack{\mathbf{p}': \\ \mathbf{p}'=p\ell}} \max_{\substack{\bar{n} \in S_{3n}^{(i-1)} \\ n'}} \|[A_{\mathbf{p}\bar{n}}^{(2)}, C_{\mathbf{p}'n'}^{(4)}]\| \\ &\leq \left(\frac{\lambda}{96}\right) \frac{24N^3(N+1)}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2}\right]. \end{aligned}$$

Thus, plugging in Eq. B72 we have the following.

$$\begin{aligned} \|[H_{2\varphi}, H_{4\varphi}]\| &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 24N(N+1)^{3/2}(N+2)^{1/2} \left[\sqrt{(N+1)(N+2)} + 2N\right] \left[\sqrt{(N+3)(N+4)} + 4N\right] \\ &\quad + \left(\frac{\lambda}{96\omega_{min}^2}\right) 12N^2(N+1) \left[\sqrt{(N+1)(N+2)} + 2N\right] \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2}\right]. \end{aligned} \quad (\text{B75})$$

f. $[H_{3\varphi}, H'_{4\varphi}]$: Using Lemma 45 along with the definition of $H_{3\varphi}$ and $H_{4\varphi}$ in Eq. B18 and B5, respectively, we have

$$\|[H_{3\varphi}, H_{4\varphi}]\| \leq \sum_{i=1}^7 \sum_{j=1}^5 \|[H_{3i}, H_{4j}]\| \quad (\text{B76})$$

We first bound the following commutators between the operators in Eq. B18 and B5. Each of $A_{\mathbf{p}\bar{n}}^{(3)}$, $B_{1\mathbf{p}\bar{n}}^{(3)}$ and $B_{2\mathbf{p}\bar{n}}^{(3)}$ are sum of 4 Pauli operators. Each of these Pauli operators can anti-commute with at most 1 of the 2 Pauli operators in $A_{1\mathbf{p}'n'}^{(4)}$, $A_{2\mathbf{p}'n'}^{(4)}$ and the single non-identity Pauli operator in $C_{\mathbf{p}'n'}^{(4)}$. $C_{\mathbf{p}\bar{n}}^{(3)}$ is a sum of 3 non-identity Pauli operators, each of which can anti-commute with both the Pauli operators in each of $A_{1\mathbf{p}'n'}^{(4)}$, $A_{2\mathbf{p}'n'}^{(4)}$ and the single non-identity Pauli operator in $C_{\mathbf{p}'n'}^{(4)}$. So, we have the commutators for the inner terms, again we leave the factors of ω till the final

max expression for simplicity.

$$\begin{aligned}
\| [A_{\mathbf{p}\bar{n}}^{(3)}, A_{1\mathbf{p}'n'}^{(4)}] \| &\leq 4 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+2)(n_2+1)}}{4\omega_p^2} \cdot \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{\mathbf{p}'}^2} \\
&\leq \frac{(N+1)^{3/2}(N+2)^{3/2}\sqrt{(N+3)(N+4)}}{\omega_{min}^4} \\
\| [B_{j\mathbf{p}\bar{n}}^{(3)}, A_{1\mathbf{p}'n'}^{(4)}] \| &\leq 4 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{n_2\sqrt{(n_1+2)(n_1+1)}}{2\omega_p^2} \cdot \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{\mathbf{p}'}^2} \quad [j \in \{1, 2\}] \\
&\leq \frac{2N(N+1)(N+2)\sqrt{(N+3)(N+4)}}{\omega_{min}^4} \\
\| [C_{\mathbf{p}\bar{n}}^{(3)}, A_{1\mathbf{p}'n'}^{(4)}] \| &\leq 3 \cdot 2 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{n_1 n_2}{\omega_p^2} \cdot \frac{\sqrt{(n'+4)(n'+3)(n'+2)(n'+1)}}{2\omega_{\mathbf{p}'}^2} \\
&\leq \frac{6N^2\sqrt{(N+1)(N+2)(N+3)(N+4)}}{\omega_{min}^4} \\
\| [A_{\mathbf{p}\bar{n}}^{(3)}, A_{2\mathbf{p}'n'}^{(4)}] \| &\leq 4 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+2)(n_2+1)}}{4\omega_p^2} \cdot \frac{n'\sqrt{(n'+1)(n'+2)}}{\omega_{\mathbf{p}'}^2} \\
&\leq \frac{2N(N+1)^{3/2}(N+2)^{3/2}}{\omega_{min}^4} \\
\| [B_{j\mathbf{p}\bar{n}}^{(3)}, A_{2\mathbf{p}'n'}^{(4)}] \| &\leq 4 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{n_2\sqrt{(n_1+2)(n_1+1)}}{2\omega_p^2} \cdot \frac{n'\sqrt{(n'+1)(n'+2)}}{\omega_{\mathbf{p}'}^2} \quad [j \in \{1, 2\}] \\
&\leq \frac{4N^2(N+1)(N+2)}{\omega_{min}^4} \\
\| [C_{\mathbf{p}\bar{n}}^{(3)}, A_{2\mathbf{p}'n'}^{(4)}] \| &\leq 3 \cdot 2 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{n_1 n_2}{\omega_p^2} \cdot \frac{n'\sqrt{(n'+1)(n'+2)}}{\omega_{\mathbf{p}'}^2} \\
&\leq \frac{12N^3\sqrt{(N+1)(N+2)}}{\omega_{min}^4} \\
\| [A_{\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'n'}^{(4)}] \| &\leq 4 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{\sqrt{(n_1+2)(n_1+1)(n_2+2)(n_2+1)}}{4\omega_p^2} \cdot \left(\frac{n'\omega_{\mathbf{p}'}}{2} + \frac{\lambda(n'^2 - n')}{32\omega_{\mathbf{p}'}^2} \right) \\
&\leq \frac{N(N+1)(N+2)}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta - 1)}{16\omega_{min}^2} \right] \\
\| [B_{j\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'n'}^{(4)}] \| &\leq 4 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{n_2\sqrt{(n_1+2)(n_1+1)}}{2\omega_p^2} \cdot \left(\frac{n'\omega_{\mathbf{p}'}}{2} + \frac{\lambda(n'^2 - n')}{32\omega_{\mathbf{p}'}^2} \right) \quad [j \in \{1, 2\}] \\
&\leq \frac{2N^2\sqrt{(N+1)(N+2)}}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta - 1)}{16\omega_{min}^2} \right] \\
\| [C_{\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'n'}^{(4)}] \| &\leq 3 \cdot 1 \cdot 2 \cdot \max_{\substack{\mathbf{p}, \mathbf{p}' \\ \bar{n}, n'}} \frac{n_1 n_2}{\omega_p^2} \cdot \left(\frac{n'\omega_{\mathbf{p}'}}{2} + \frac{\lambda(n'^2 - n')}{32\omega_{\mathbf{p}'}^2} \right) \\
&\leq \frac{3N^3}{\omega_{min}^2} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta - 1)}{16\omega_{min}^2} \right] \tag{B77}
\end{aligned}$$

We first bound the commutators between H_{31} , H_{32} and H_{4j} , where $j = 1, \dots, 5$. We consider the following sum, where each commutator has the same bound. There are $|\Omega|^2$ commutators as before. Using Eq. B77 we have

$$\begin{aligned} \sum_{i=1}^2 \sum_{j=1}^2 \|[H_{3i}, H_{4j}]\| &\leq \left(\frac{\lambda}{96}\right)^2 \sum_{i=1}^2 \sum_{j=1}^2 \max_{\substack{\ell=1 \\ \mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}}^2 \max_{\substack{\bar{n} \in S_{2n}^{(i+1)} \\ n' \in S_{4(j-1)}}} \|[A_{\mathbf{p}\bar{n}}^{(3)}, A_{1\mathbf{p}'n'}^{(4)}]\| \\ &= \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 16(N+1)^{3/2}(N+2)^{3/2}\sqrt{(N+3)(N+4)}; \end{aligned}$$

and

$$\begin{aligned} \sum_{i=1}^2 \sum_{j=3}^4 \|[H_{3i}, H_{4j}]\| &\leq 2 \cdot \left(\frac{\lambda}{96}\right)^2 \sum_{i=1}^2 \sum_{j=3}^4 \max_{\substack{\ell=1 \\ \mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}}^2 \max_{\substack{\bar{n} \in S_{2n}^{(i+1)} \\ n' \in S_{2(j-3)}}} \|[A_{\mathbf{p}\bar{n}}^{(3)}, A_{2\mathbf{p}'n'}^{(4)}]\| \\ &= \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 64N(N+1)^{3/2}(N+2)^{3/2}; \end{aligned}$$

and finally

$$\begin{aligned} \sum_{i=1}^2 \|[H_{3i}, H_{45}]\| &\leq \left(\frac{\lambda}{96}\right)^2 \sum_{i=1}^2 \max_{\ell=1}^2 \max_{\substack{\mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}} \max_{\substack{\bar{n} \in S_{2n}^{(i+1)} \\ n'}} \|[A_{\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'n'}^{(4)}]\| \\ &= \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 8N^2(N+1)(N+2) \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right]. \end{aligned}$$

Now we have

$$\begin{aligned} &\sum_{i=3}^4 \sum_{j=1}^2 \|[H_{3i}, H_{4j}]\| + \sum_{i=5}^6 \sum_{j=1}^2 \|[H_{3i}, H_{4j}]\| \\ &\leq \left(\frac{\lambda}{96}\right)^2 \sum_{j=1}^2 \max_{\ell=1}^2 \left[\sum_{i=3}^4 \max_{\substack{\mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}} \max_{\substack{n_1 \in S_{2(i-3)}; n_2 \\ n' \in S_{4(j-1)}}} \|[B_{1\mathbf{p}\bar{n}}^{(3)}, A_{1\mathbf{p}'n'}^{(4)}]\| + \max_{i=5}^6 \max_{\substack{\mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}} \max_{\substack{n_2 \in S_{2(i-5)}; n_1 \\ n' \in S_{4(j-1)}}} \|[B_{2\mathbf{p}\bar{n}}^{(3)}, A_{1\mathbf{p}'n'}^{(4)}]\| \right] \\ &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 64N(N+1)(N+2)\sqrt{(N+3)(N+4)}; \end{aligned} \tag{B78}$$

and

$$\begin{aligned} &\sum_{i=3}^4 \sum_{j=3}^4 \|[H_{3i}, H_{4j}]\| + \sum_{i=5}^6 \sum_{j=3}^4 \|[H_{3i}, H_{4j}]\| \\ &\leq 2 \cdot \left(\frac{\lambda}{96}\right)^2 \sum_{j=3}^4 \max_{\ell=1}^2 \left[\sum_{i=3}^4 \max_{\substack{\mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}} \max_{\substack{n_1 \in S_{2(i-3)}; n_2 \\ n' \in S_{2(j-3)}}} \|[B_{1\mathbf{p}\bar{n}}^{(3)}, A_{2\mathbf{p}'n'}^{(4)}]\| + \sum_{i=5}^6 \max_{\substack{\mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}} \max_{\substack{n_2 \in S_{2(i-5)}; n_1 \\ n' \in S_{2(j-3)}}} \|[B_{2\mathbf{p}\bar{n}}^{(3)}, A_{2\mathbf{p}'n'}^{(4)}]\| \right] \\ &\leq \left(\frac{\lambda}{96\omega_{min}^4}\right)^2 256N^2(N+1)(N+2); \end{aligned}$$

again

$$\begin{aligned} &\sum_{i=3}^4 \|[H_{3i}, H_{45}]\| + \sum_{i=5}^6 \|[H_{3i}, H_{45}]\| \\ &\leq \left(\frac{\lambda}{96}\right)^2 \sum_{\ell=1}^2 \left[\sum_{i=3}^4 \sum_{\substack{\mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}} \sum_{\substack{n_1 \in S_{2(i-3)}; n_2 \\ n'}} \|[B_{1\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'n'}^{(4)}]\| + \sum_{i=5}^6 \sum_{\substack{\mathbf{p}, \mathbf{p}': \\ \mathbf{p}'=p_\ell}} \sum_{\substack{n_2 \in S_{2(i-5)}; n_1 \\ n'}} \|[B_{2\mathbf{p}\bar{n}}^{(3)}, C_{\mathbf{p}'n'}^{(4)}]\| \right] \\ &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 32N^2\sqrt{(N+1)(N+2)} \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right]. \end{aligned}$$

We now consider the commutators between H_{33} , H_{34} , H_{35} , H_{36} and each H_{4j} , where $j = 1, \dots, 5$. We observe that the conditions for overlap of the momentum modes and momentum states, in order to have a non-zero overlap, are also similar. The innermost commutators change and they can be bound using Eq. B77. Next, we consider the commutators between H_{37} and each H_{4j} , where $j = 1, \dots, 5$. We observe that the conditions for overlap of the momentum modes and momentum states, in order to have a non-zero overlap, are also similar. The innermost commutators change and they can be bound using Eq. B77. Thus we have

$$\begin{aligned} \sum_{j=1}^2 \|[H_{37}, H_{4j}]\| &\leq \left(\frac{\lambda}{96}\right)^2 \sum_{j=1}^2 \max_{\ell=1}^2 \max_{\mathbf{p}, \mathbf{p}'} \max_{\substack{\tilde{n}; \\ \mathbf{p}'=p_\ell \ n' \in S_{4(j-1)}}} \|[C_{\mathbf{p}\tilde{n}}^{(3)}, A_{1\mathbf{p}'n'}^{(4)}]\| \\ &= \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 24N^2 \sqrt{(N+1)(N+2)(N+3)(N+4)}; \end{aligned}$$

and

$$\begin{aligned} \sum_{j=3}^4 \|[H_{37}, H_{4j}]\| &\leq 2 \cdot \left(\frac{\lambda}{96}\right)^2 \sum_{j=3}^4 \sum_{\ell=1}^2 \sum_{\mathbf{p}, \mathbf{p}'} \sum_{\substack{\tilde{n}; \\ \mathbf{p}'=p_\ell \ n' \in S_{2(j-3)}}} \|[C_{\mathbf{p}\tilde{n}}^{(3)}, A_{2\mathbf{p}'n'}^{(4)}]\| \\ &= \frac{\lambda}{96} \left(\frac{\lambda}{\omega_{min}^2}\right)^2 N^3 \sqrt{(N+1)(N+2)}; \end{aligned}$$

and finally

$$\begin{aligned} \|[H_{37}, H_{45}]\| &\leq \left(\frac{\lambda}{96}\right)^2 \sum_{\ell=1}^2 \sum_{\mathbf{p}, \mathbf{p}'} \sum_{\substack{\tilde{n}, n' \\ \mathbf{p}'=p_\ell}} \|[C_{\mathbf{p}\tilde{n}}^{(3)}, C_{\mathbf{p}'n'}^{(4)}]\| \\ &= \left(\frac{12N^3\lambda}{96\omega_{min}^4}\right) \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right]. \end{aligned}$$

Therefore, plugging in Eq. B76 we have

$$\begin{aligned} \|[H_{3\varphi}, H_{4\varphi}]\| &\leq \left(\frac{\lambda}{96\omega_{min}^2}\right)^2 16\sqrt{(N+1)(N+2)} \left[\sqrt{(N+1)(N+2)} + N \right] \\ &\quad \cdot \left[\sqrt{(N+1)(N+2)} + 3N \right] \left[\sqrt{(N+1)(N+2)} + 4N \right] \\ &\quad + \left(\frac{\lambda}{96\omega_{min}^2}\right) 4N \left[2(N+1)(N+2) + 8N\sqrt{(N+1)(N+2)} + 3N^2 \right] \\ &\quad \cdot \left[\beta\omega_{max} + \frac{\lambda\beta(N\beta-1)}{16\omega_{min}^2} \right]. \end{aligned} \tag{B79}$$

3. Bounding $\tilde{\alpha}_{comm}$ and hence the Trotter error

Now we are in a position to bound $\tilde{\alpha}_{comm}$ according to Eq. B2. First we sum Eq. B19-B24 to get a bound on the sum of norms, which is

$$\leq \frac{\omega_{max}N(N+1)}{2} + \left(\frac{\lambda}{96\omega_{min}^2}\right) |\Omega| \left[\frac{(N+1)^2}{2} + (N+2)^2 + (N+2) + (N+4)^2 \right]. \tag{B80}$$

Next, we bound the sum of first-level commutators by adding Eq. B27, B37, B43, B50, B56, B61, B65, B71, B75 and B79 and it is as follows.

$$\in O\left(\frac{\lambda^2 N^4}{\omega_{min}^4} + \frac{\lambda N^4}{\omega_{min}^4} (\beta\omega_{max}\omega_{min}^2 + \lambda\beta^2 N)\right)$$

Since $\beta < 1$, $\omega_{min} = M$ and $\omega_{max} = \sqrt{M^2 + P_{max}^2} \sim M$, so we have since these commutators only fire when there is a momentum overlap, we have

$$\begin{aligned}
\tilde{\alpha}_{comm} &\in O\left(\frac{\lambda^3 N^6}{\omega_{min}^6} + \frac{\lambda^2 \omega_{max} N^6}{\omega_{min}^6} + \frac{\lambda^2 N^6}{\omega_{min}^6} (\beta \omega_{max} \omega_{min}^2 + \lambda \beta^2 N) \left(1 + \frac{\omega_{max}}{\lambda}\right)\right) \\
&\in O\left(\frac{\lambda^2 N^6 (M + \lambda)}{M^6}\right), \\
&\in O\left(\frac{\lambda^2 N^6}{M^5}\right), \quad \text{assuming } \lambda < N|\Omega|;
\end{aligned} \tag{B81}$$

and hence the Trotter error is in $O\left(\frac{\lambda^2 N^6 \tau^3}{M^5}\right)$.

Appendix C: Field Amplitude Basis Algorithms

1. Equal weight LCU

Here, we provide proofs of Lemmas and Theorems stated in Section IV E 1.

Proof of Lemma 19. For integers i, j and k , we note the following identities:

$$\begin{aligned}
\Theta(i - j - 1) &= (j < i) \\
\Theta(-k - 1) &= 1 - \Theta(k) \\
Z|k\rangle &= (1 - 2k)|k\rangle
\end{aligned} \tag{C1}$$

We then have

$$\begin{aligned}
&\text{CMP}^\dagger (\mathbb{I} \otimes \mathbb{I} \otimes Z) \text{CMP} |i\rangle |j\rangle |0\rangle \\
&= \text{CMP}^\dagger (\mathbb{I} \otimes \mathbb{I} \otimes Z) |i\rangle |j\rangle |j < i\rangle \\
&= \text{CMP}^\dagger (\mathbb{I} \otimes \mathbb{I} \otimes Z) |i\rangle |j\rangle |\Theta(i - j - 1)\rangle \\
&= \text{CMP}^\dagger (1 - 2\Theta(i - j - 1)) |i\rangle |j\rangle |\Theta(i - j - 1)\rangle \\
&= \text{CMP}^\dagger (1 - 2(1 - \Theta(j - i))) |i\rangle |j\rangle |\Theta(i - j - 1)\rangle \\
&= \text{CMP}^\dagger (2\Theta(j - i) - 1) |i\rangle |j\rangle |j < i\rangle \\
&= (2\Theta(j - i) - 1) |i\rangle |j\rangle |0\rangle
\end{aligned} \tag{C2}$$

□

Proof of Corollary 20. Given some state on a single lattice site $|\psi\rangle = \sum_{j=0}^{2k-1} c_j |j\rangle \in \mathcal{H}_\phi$, an ancillary register $|\bar{0}\rangle =$

$|0\rangle^{\otimes m} \in \mathcal{H}_P$ where $m = \lceil \log_2 k + 1 \rceil$, and a single additional ancillary qubit $|0\rangle_{anc} \in \mathcal{H}_{scratch}$, we have

$$\begin{aligned}
U_{SUB-PREP_\phi} |\bar{0}\rangle |\psi\rangle &= (H^{\otimes m} |0\rangle^{\otimes m}) |\psi\rangle = |\overline{+}\rangle |\psi\rangle \\
\left(U_{SUB-SELECT_\phi} |\overline{+}\rangle |\psi\rangle \right) |0\rangle_{anc} &= \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} |\overline{+}\rangle |\psi\rangle |0\rangle_{anc} \\
&= \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} \left(\frac{1}{\sqrt{2k}} \sum_{i=0}^{2k-1} |i\rangle \right) \left(\sum_{j=0}^{2k-1} c_j |j\rangle \right) |0\rangle_{anc} \\
&= \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \frac{1}{\sqrt{2k}} \sum_{i,j=0}^{2k-1} c_j |i\rangle |j\rangle |\Theta(i-j-1)\rangle_{anc} \\
&= \text{CMP}_{P,\phi,anc}^\dagger \frac{1}{\sqrt{2k}} \sum_{i,j=0}^{2k-1} c_j (1 - 2\Theta(i-j-1)) |i\rangle |j\rangle |\Theta(i-j-1)\rangle_{anc} \\
&= \frac{1}{\sqrt{2k}} \sum_{i,j=0}^{2k-1} c_j (2\Theta(j-i) - 1) |i\rangle |j\rangle |0\rangle_{anc} \\
&= \frac{1}{\sqrt{2k}} \sum_{i=0}^{2k-1} |i\rangle \otimes \sum_{m=0}^{2k-1} (2\Theta(m-i) - 1) |m\rangle \langle m| \left(\sum_{j=0}^{2k-1} c_j |j\rangle \right) |0\rangle_{anc} \\
&= \frac{1}{\sqrt{2k}} \left(\sum_{i=0}^{2k-1} |i\rangle \otimes U^{(i)} |\psi\rangle \right) |0\rangle_{anc} \tag{C3}
\end{aligned}$$

□

Note that

$$\begin{aligned}
\left(|\overline{+}\rangle \otimes I \right) \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} |\overline{+}\rangle |\psi\rangle |0\rangle_{anc} &= \left(\frac{1}{2k} \sum_{i=0}^{2k-1} U^{(i)} |\psi\rangle \right) |0\rangle_{anc} \\
&= \frac{\hat{\phi}}{k\Delta\phi} |\psi\rangle |0\rangle_{anc} \\
&= \frac{\hat{\phi}}{\phi_{max}} |\psi\rangle |0\rangle_{anc} \tag{C4}
\end{aligned}$$

and so we have

$$\left(H^{\otimes m} \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} H^{\otimes m} \right) |\bar{0}\rangle |\psi\rangle |0\rangle_{anc} = \left(|\bar{0}\rangle \frac{\hat{\phi}}{\phi_{max}} |\psi\rangle + |\Phi^\perp\rangle \right) |0\rangle_{anc} \tag{C5}$$

where $(|\bar{0}\rangle \langle \bar{0}| \otimes I) |\Phi^\perp\rangle = 0$, and we obtain the state $\hat{\phi}|\psi\rangle$ with probability $\frac{|\hat{\phi}|\psi\rangle|^2}{|\phi_{max}|^2}$.

a. Arithmetic primitives

a. Addition: Our first result provides the resources needed to implement a generic addition circuit in the Clifford + T gate set. Our construction can be thought of as an elaboration on top of the Gidney adder given in [76].

Lemma 46 (Addition Circuit). *Let $U \in \mathbb{C}^{2^{3n-1} \times 2^{3n-1}}$ for $n \in \mathbb{Z}_+$ be a unitary matrix such that U acts as a reversible adder meaning that for any $x, y \in \mathbb{Z}_{2^n}$, $U |x\rangle |y\rangle |0\rangle = |x + y \bmod 2^n\rangle |y\rangle |0\rangle$ can be constructed using the following*

resources

$$\begin{aligned}
T\text{-count} &= 4n \\
\text{Ancilla count} &= n \\
\text{Measurement depth} &= n \\
\text{CNOT count} &= 12n - 3 \\
\text{CZ count} &= n \\
S\text{ count} &= n \\
H\text{ count} &= 2n
\end{aligned} \tag{C6}$$

Proof. Let us consider addition of two n -bit binary numbers a and b using the classical full-adder, which takes two single-bit inputs a_i and b_i and $c_{i;in}$, the carry-in bit from the previous bit addition, and outputs the sum s_i and the carry-out $c_{i;out}$. In Reed-Muller form, these are given by

$$\begin{aligned}
s_i &= a_i \oplus b_i \oplus c_{i;in} \\
c_{i;out} &= a_i b_i \oplus a_i c_{i;in} \oplus b_i c_{i;in}
\end{aligned} \tag{C7}$$

Let us assume that we are allowed to over-write one of the registers storing the binary representations of a and b with that of $(a + b)$. Let us also assume that we can selectively measure out and reset a subset of the total qubits available to us. Then, we carry out this primitive using n ancillas, $n - 1$ to store the values of the carry bits, and 1 to hold the value of the left-most (most significant) digit of $(a + b)$, if we can selectively reset the ancillas to perform a similar subsequent computation. A bit more explicitly,

$$\begin{aligned}
c_0 &= 0 \quad , \quad s_0 = a_0 \oplus b_0 \\
c_{i+1} &= a_i b_i \oplus a_i c_i \oplus b_i c_i \quad , \quad s_i = a_i \oplus b_i \oplus c_i \quad \forall 0 \leq i \leq n - 1 \\
s_n &= c_n
\end{aligned} \tag{C8}$$

For actually carrying out the binary addition, we follow the method described in [76]. The first step is to note a construction of the logical AND gate that transforms $|x\rangle |y\rangle |0\rangle \rightarrow |x\rangle |y\rangle |xy\rangle$, and that uses only 4 T gates, as shown in Fig. 12.

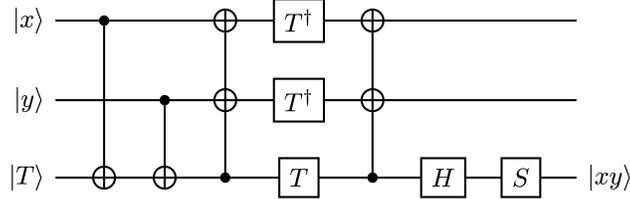


FIG. 12: Logical AND gate construction using 4 T gates.

Step by step, we can prove that this construction computes $|xy\rangle$ given input bits $|x\rangle$ and $|y\rangle$ as follows

$$\begin{aligned}
|x\rangle |y\rangle |T\rangle &\rightarrow |x\rangle |y\rangle \frac{1}{\sqrt{2}} \left(|x\rangle + e^{i\pi/4} |x \oplus 1\rangle \right) \\
&\rightarrow |x\rangle |y\rangle \frac{1}{\sqrt{2}} \left(|x \oplus y\rangle + e^{i\pi/4} |x \oplus y \oplus 1\rangle \right) \\
&\rightarrow \frac{1}{\sqrt{2}} \left(|y\rangle |x\rangle |y \oplus x\rangle + e^{i\pi/4} |y \oplus 1\rangle |x \oplus 1\rangle |x \oplus y \oplus 1\rangle \right) \\
&\rightarrow \frac{1}{\sqrt{2}} \left(e^{i\frac{\pi}{4}(x \oplus y - x - y)} |y\rangle |x\rangle |x \oplus y\rangle + e^{i\frac{\pi}{4}(x \oplus y \oplus 1 - x \oplus 1 - y \oplus 1)} |y \oplus 1\rangle |x \oplus 1\rangle |x \oplus y \oplus 1\rangle \right) \\
&\rightarrow \frac{1}{\sqrt{2}} \left(e^{i\frac{\pi}{4}(x \oplus y - x - y)} |x\rangle |y\rangle |x \oplus y\rangle + e^{i\frac{\pi}{4}(x \oplus y \oplus 1 - x \oplus 1 - y \oplus 1)} |x\rangle |y\rangle |x \oplus y \oplus 1\rangle \right) \\
&\rightarrow |x\rangle |y\rangle ((1 - xy) |0\rangle - i(xy) |1\rangle) \\
&\rightarrow |x\rangle |y\rangle ((1 - xy) |0\rangle + (xy) |1\rangle) \\
&= |x\rangle |y\rangle |xy\rangle
\end{aligned} \tag{C9}$$

where we have used $CNOT |a\rangle |b\rangle = |a\rangle |a \oplus b\rangle$, $T |a\rangle = e^{i\pi/4} |a\rangle$, $T^\dagger |a\rangle = e^{-i\pi/4} |a\rangle$, $H |a\rangle = \frac{|0\rangle + e^{i\pi a} |1\rangle}{\sqrt{2}}$, in the third last step used the truth tables

x	y	$x \oplus y - x - y$	$x \oplus y \oplus 1 - x \oplus 1 - y \oplus 1$
0	0	0	0
0	1	0	0
1	0	0	0
1	1	-2	2

and

x	y	$\left(\frac{x \oplus y - x - y}{4}\right) + (x \oplus y)$	$\left(\frac{x \oplus y \oplus 1 - x \oplus 1 - y \oplus 1}{4}\right) + (x \oplus y \oplus 1)$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	-1/2	3/2

so that

$$\begin{aligned} & \frac{1}{2} |x\rangle |y\rangle |0\rangle \left(e^{i\pi/4(x \oplus y - x - y)} + e^{i\pi/4(x \oplus y \oplus 1 - x \oplus 1 - y \oplus 1)} \right) = (1 - xy) |x\rangle |y\rangle |0\rangle \\ & \frac{1}{2} |x\rangle |y\rangle |1\rangle \left(e^{i\pi\left(\frac{x \oplus y - x - y}{4}\right) + (x \oplus y)} + e^{i\pi\left(\frac{x \oplus y \oplus 1 - x \oplus 1 - y \oplus 1}{4}\right) + (x \oplus y \oplus 1)} \right) = -i(xy) |x\rangle |y\rangle |0\rangle \end{aligned} \quad (C10)$$

While this allows us to compute the carry-out bit using 4 T gates, we must also uncompute this operation if we are to leave the sum register unentangled with all the ancillae at the end of the computation. To this end, [76] constructs the (irreversible) uncomputation step depicted in Fig. 13 that notably uses no T gates.

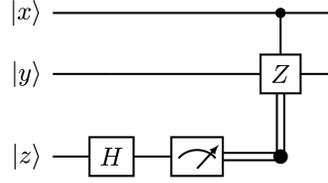


FIG. 13: Uncomputing the logical AND gate with no T gates.

To see that this works, let us start with some arbitrary 2-qubit superposition $|\psi\rangle = \sum_{i,j=0}^1 \alpha_{ij} |i\rangle |j\rangle$ and suppose that we have applied the Toffoli gate above in the first step. Following this, we have the series of transformations

$$\begin{aligned} |\psi\rangle |0\rangle & \rightarrow \alpha_{00} |00\rangle |0\rangle + \alpha_{01} |01\rangle |0\rangle + \alpha_{10} |10\rangle |0\rangle + \alpha_{11} |11\rangle |1\rangle \\ & \xrightarrow{I \otimes I \otimes H} (\alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle) \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) + \alpha_{11} \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) \\ & = \frac{1}{\sqrt{2}} (\alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + \alpha_{11} |11\rangle) |0\rangle \\ & \quad + \frac{1}{\sqrt{2}} (\alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle - \alpha_{11} |11\rangle) |1\rangle \\ & \xrightarrow{\text{Measure and correct}} (\alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + (CZ)^x (-1)^x \alpha_{11} |11\rangle) |x\rangle \\ & = (\alpha_{00} |00\rangle + \alpha_{01} |01\rangle + \alpha_{10} |10\rangle + \alpha_{11} |11\rangle) |x\rangle = |\psi\rangle |x\rangle \end{aligned} \quad (C11)$$

We can use this in an adder by first computing a carry-out bit given a carry-in, and the two bits to be added as depicted in Fig. 14.

Note that this uses a single AND gate (and therefore 4 T gates), and we have used the identity $(i_k \oplus c_k)(t_k \oplus c_k) = i_k t_k \oplus i_k c_k \oplus t_k c_k \oplus c_k$. We can then use the carry-out bit for later computation. After it is used, we need to uncompute it and over-write one of the bits to hold the addition computation. This is performed by first carrying out a single CNOT between the carry-in qubit and the carry-out qubit as depicted in Fig. 15.

The bottom-most qubit now holds the logical AND of the two qubits above it, so that we can now measure and discard this qubit while keeping the above qubits in the state shown in Fig. 15 using the uncomputation technique

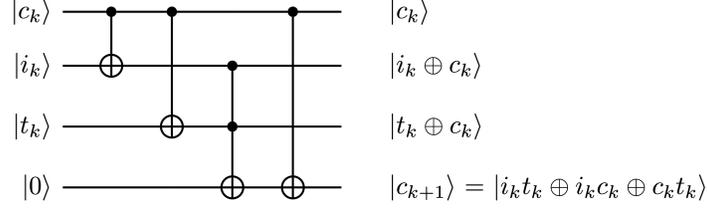


FIG. 14: Computing the carry-out bit in the first part of the adder primitive.

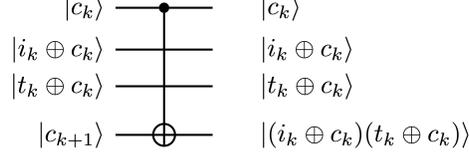


FIG. 15: The first stage in the uncomputation part of the adder.

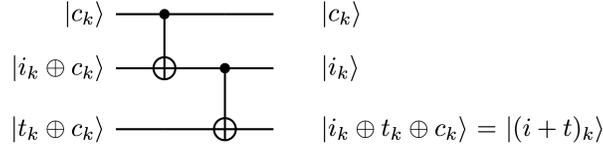


FIG. 16: The last stage in the uncomputation part of the adder, completing the addition computation.

described above. After this measure and fix part, we simply carry out two CNOTs as depicted in Fig. 16 to complete the addition. Note that the Gidney adder is reversible.

To make everything a bit more explicit, this method of addition performs the following series of transformations

$$\begin{aligned}
 U_{A+B} |a\rangle |b\rangle |scratch\rangle & : (|0\rangle |a_{n-1}\rangle \dots |a_0\rangle) (|b_{n-1}\rangle \dots |b_0\rangle) (|0\rangle^{\otimes n-1}) \\
 & \rightarrow (|c_n\rangle |a_{n-1} \oplus c_{n-1}\rangle \dots |a_1 \oplus c_1\rangle |a_0\rangle) (|b_{n-1} \oplus c_{n-1}\rangle \dots |b_1 \oplus c_1\rangle |b_0\rangle) (|c_{n-1}\rangle \dots |c_1\rangle) \\
 & \rightarrow (|c_n\rangle |s_{n-1}\rangle \dots |s_0\rangle) (|b_{n-1}\rangle \dots |b_0\rangle) (|0\rangle^{\otimes n-1}) \\
 & = |a + b\rangle |b\rangle |scratch\rangle
 \end{aligned} \tag{C12}$$

Only the first arrow above requires any AND operations. In all, we require $4n$ T gates, as well as n ancillary qubits. We also have 6 CNOTs per AND gate, $3(n-1)$ CNOTs during the computation part, and an additional $3n$ CNOTs during the uncomputation part. Lastly, we have 1 S gate, 1 CZ gate and 2 H gates for every AND compute/uncompute pair. These results are summarized in the statement of the lemma and thus this concludes our proof. \square

b. Subtraction: Given a modular adder it is often straight forward to construct a modular subtraction circuit. Such subtraction circuits are extremely helpful for LCU decompositions because performing a subtraction between two numbers and examining the sign allows us to find the maximum of two numbers. Below we outline a ones complement approach for constructing such a subtractor / comparator.

Corollary 47 (Subtraction Circuit). *Let $U \in \mathbb{C}^{2^{3n-1} \times 2^{3n-1}}$ for $n \in \mathbb{Z}_+$ be a unitary matrix such that U acts as a reversible subtractor meaning that for any $x, y \in \mathbb{Z}_{2^n}$, $U |x\rangle |y\rangle |0\rangle = |x - y \bmod 2^n\rangle |y\rangle |0\rangle$ can be constructed using the following resources*

$$\begin{aligned}
T\text{-count} &= 4n \\
\text{Ancilla count} &= n \\
\text{Measurement depth} &= n \\
\text{CNOT count} &= 12n - 3 \\
\text{CZ count} &= n \\
\text{S count} &= n \\
\text{H count} &= 2n \\
\text{Pauli-X count} &= 2n
\end{aligned}
\tag{C13}$$

Using that the ones' complement, i.e. bitwise complementation, of a is given by $\bar{a} = 2^n - 1 - a$, we arrive at the identity

$$a - b = 2^n - 1 - (2^n - 1 - a + b) = \overline{(a + b)} \tag{C16}$$

which allows us to implement subtraction in essentially the same way as addition, plus some single qubit X gates. For ease of notation, let us define

$$\begin{aligned}
c'_0 &= 0 \quad , \quad s'_0 = a_0 \oplus b_0 \oplus 1 \\
c'_{k+1} &= a_k b_k \oplus b_k c'_k \oplus a_k c'_k \oplus b_k \oplus c'_k \quad , \quad s'_k = a_k \oplus b_k \oplus c'_k \oplus 1 \\
&\quad \quad \quad s'_n = c'_n
\end{aligned}
\tag{C17}$$

Explicitly, we run the following series of transformations

$$\begin{aligned}
U_{A-B} |a\rangle |b\rangle |scratch\rangle &: (|0\rangle |a_{n-1}\rangle \dots |a_0\rangle) (|b_{n-1}\rangle \dots |b_0\rangle) (|0\rangle^{\otimes n-1}) \\
&\rightarrow (|0\rangle |a_{n-1} \oplus 1\rangle \dots |a_0 \oplus 1\rangle) (|b_{n-1}\rangle \dots |b_0\rangle) (|0\rangle^{\otimes n-1}) \\
&\rightarrow (|c'_n\rangle |a_{n-1} \oplus 1 \oplus c'_{n-1}\rangle \dots |a_1 \oplus 1 \oplus c'_1\rangle |a_0 \oplus 1\rangle) (|b_{n-1} \oplus c'_{n-1}\rangle \dots |b_1 \oplus c'_1\rangle |b_0\rangle) (|c'_{n-1}\rangle \dots |c'_1\rangle) \\
&\rightarrow (|c'_n\rangle |s'_{n-1}\rangle \dots |s'_0\rangle) (|b_{n-1}\rangle \dots |b_0\rangle) (|0\rangle^{\otimes n-1}) \\
&\rightarrow (|c'_n \oplus 1\rangle |s'_{n-1} \oplus 1\rangle \dots |s'_0 \oplus 1\rangle) (|b_{n-1}\rangle \dots |b_0\rangle) (|0\rangle^{\otimes n-1}) \\
&= |a - b\rangle |b\rangle |scratch\rangle
\end{aligned}
\tag{C18}$$

Only the second arrow above (computing the carry-bits) requires any ANDs. The resource counts are the same as in Eq. (C6), and also additionally $2n$ X gates.

c. Incrementer: In the special case where we want to add only 1 to some other number, we have the much simpler case

$$\begin{aligned}
c_0 &= 0 \quad , \quad s_0 = a_0 \oplus 1 \\
c_1 &= a_0 \quad , \quad s_1 = a_1 \oplus a_0
\end{aligned}
\tag{C19}$$

and for $n \geq 2$,

$$\begin{aligned}
c_n &= a_{n-1} a_{n-2} \dots a_0 \\
s_n &= a_n \oplus c_n
\end{aligned}
\tag{C20}$$

In the worst case, to add 1 to an n -bit number, we need $n+1$ bits to express the answer. This requires first computing $n-2$ many logical ANDs to compute all the necessary carry bits c_2, \dots, c_{n+1} with $c_{n+1} = s_{n+1}$. We then perform CNOTs controlled on the carry bits, targeted on the input register to over-write it with the value of the sum, i.e. $a_i \rightarrow c_i \oplus a_i$ for every $i \geq 2$. To handle the least significant two bits, we first simply perform a CNOT controlled on a_0 and targeted on a_1 , and then perform a NOT operation on a_0 . After this, we uncompute all the logical ANDs expressing the carry bits except the very high bit c_{n+1} . Thus, to add 1 to an n -qubit register, we need $n-1$

ancillas, and $n - 1$ ANDs. Each of these AND operations requires 4 T gates, 6 CNOTs, 1 S gate, 1 CZ gate and 2 H gates. There are also an additional $n - 1$ CNOTs and a single X gate. The total resource count is then given below.

$$\begin{aligned}
\text{T-count} &= 4(n - 1) \\
\text{Ancilla count} &= n - 1 \\
\text{Measurement depth} &= n - 1 \\
\text{CNOT count} &= 7(n - 1) \\
\text{CZ count} &= n - 1 \\
\text{S count} &= n \\
\text{H count} &= 2(n - 1) \\
\text{X count} &= 1
\end{aligned} \tag{C21}$$

d. Comparator: In order to implement the comparator, i.e.

$$\text{CMP}_{A,B,C} |i\rangle |j\rangle |0\rangle = |i\rangle |j\rangle |j < i\rangle \tag{C22}$$

we can compute the high-bit of $j - i$, which is 1 if and only if $j < i$. Essentially, we can run the same circuit as subtraction, use a CNOT to write out the value of $c'_n \oplus 1$ onto an ancillary qubit, and then reversibly undo the entire subtraction circuit.

$$\begin{aligned}
\text{CMP}_{A,B,C} |i\rangle |j\rangle |0\rangle &: |i\rangle |j\rangle |0\rangle |scratch\rangle \\
&\rightarrow |i\rangle |j - i\rangle |0\rangle |scratch\rangle \\
&= |i_{n-1}\rangle \dots |i_0\rangle |(j - i)_n\rangle \dots |(j - i)_0\rangle |0\rangle |scratch\rangle \\
&\rightarrow |i_{n-1}\rangle \dots |i_0\rangle |(j - i)_n\rangle \dots |(j - i)_0\rangle |(j - i)_n\rangle |scratch\rangle \\
&= |i\rangle |j - i\rangle |(j - i)_n\rangle |scratch\rangle \\
&\rightarrow |i\rangle |j\rangle |(j - i)_n\rangle |scratch\rangle \\
&= |i\rangle |j\rangle |j < i\rangle |scratch\rangle
\end{aligned} \tag{C23}$$

However, this scheme would require running the subtraction circuit twice, once in the forward direction, and once in the reverse direction. This would result in a gate count of $2n$ ANDs, and therefore $8n$ T gates.

We can do better, by simply running the circuit until computing the carry-bits, running a CNOT to write out the high-bit onto an ancillary (scratch) qubit, and then undoing the carry-bit computations. This way, the entire operation requires n ANDs, and therefore $4n$ T gates. It is this operation that is referred to as CMP' in the main text. The main use of this operation is in applying $\text{CMP}'^\dagger Z_{anc} \text{CMP}$. Apart from the Z_{anc} , the cost of this composite operation is the same as that of CMP . Equivalently, the cost of this is the same as that of the subtraction circuit, which in turn is the same as that of the addition circuit Eq. (C6), plus an additional $2n$ X gates.

e. Multiplication: To perform binary multiplication, we can use bitwise shift operations (by simply adding ancilla qubits), as well as the addition technique described above. As an example, consider adding two 3-bit numbers

$$\begin{array}{rcccc}
& a_2 & a_1 & a_0 & \\
& \underline{b_2} & \underline{b_1} & \underline{b_0} & \\
0 & 0 & (b_0 a_2) & (b_0 a_1) & (b_0 a_0) \\
0 & (b_1 a_2) & (b_1 a_1) & (b_1 a_0) & 0 \\
(b_2 a_2) & (b_2 a_1) & (b_2 a_0) & 0 & 0
\end{array} \tag{C24}$$

Each of the products in parantheses above can be computed using a total of n^2 AND gates. In general, we need $n + (n - 1)$ many additional qubits to hold the values of each of the binary numbers in each of the lines above. We then need to add all these numbers together. Since some of the bits in these numbers are 0, the circuits for as well as the number of ancillary qubits required for addition can be simplified. However, we simply report the upper bound, where we assume that we require n registers of $2n - 1$ many qubits, contributing to an ancillary qubit count of $n(2n - 1)$. These registers must be added in a sequence of n additions. We also require 1 additional ancilla to hold the value of the highest bit for the very last addition.

Each of the n additions require $2n - 1$ ancillary qubits for the computation. Since these additions must be carried out sequentially, these $2n - 1$ ancillae can be recycled. Apart from this ancilla count, the cost associated will be n times the resources reported in Eq. (C6). In all, we require $2n^2 - 1$ ANDs. Putting everything together, we find the following resource counts for performing multiplication of two n -qubit registers.

$$\begin{aligned}
\text{T-count} &= 8n^2 - 4 \\
\text{Ancilla count} &= (n + 1)(2n - 1) + 1 \\
\text{Measurement depth} &= 2n^2 - 1 \\
\text{CNOT count} &= 12n^2 - 6 \\
\text{CZ count} &= 2n^2 - 1 \\
\text{S count} &= 2n^2 - 1 \\
\text{H count} &= 4n^2 - 2
\end{aligned} \tag{C25}$$

b. Sub-SELECT circuits

a. Sub-PREPARE and SELECT circuits for $\hat{\phi}$ (toy example) Then, given some state $|\psi\rangle = \sum_{j=0}^{2k-1} c_j |j\rangle \in \mathcal{H}_\phi$, an ancillary register $|\bar{0}\rangle = |0\rangle^{\otimes m} \in \mathcal{H}_P$ where $m = \lceil \log_2 k + 1 \rceil$, and a single additional ancillary qubit $|0\rangle_{anc} \in \mathcal{H}_{scratch}$, we have

$$\begin{aligned}
U_{SUB-PREP_\phi} |\bar{0}\rangle |\psi\rangle &= (H^{\otimes m} |0\rangle^{\otimes m}) |\psi\rangle = |\bar{+}\rangle |\psi\rangle \\
\left(U_{SUB-SELECT_\phi} |\bar{+}\rangle |\psi\rangle \right) |0\rangle_{anc} &= \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} |\bar{+}\rangle |\psi\rangle |0\rangle_{anc} \\
&= \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} \left(\frac{1}{\sqrt{2k}} \sum_{i=0}^{2k-1} |i\rangle \right) \left(\sum_{j=0}^{2k-1} c_j |j\rangle \right) |0\rangle_{anc} \\
&= \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \frac{1}{\sqrt{2k}} \sum_{i,j=0}^{2k-1} c_j |i\rangle |j\rangle |\Theta(i-j-1)\rangle_{anc} \\
&= \text{CMP}_{P,\phi,anc}^\dagger \frac{1}{\sqrt{2k}} \sum_{i,j=0}^{2k-1} c_j (1 - 2\Theta(i-j-1)) |i\rangle |j\rangle |\Theta(i-j-1)\rangle_{anc} \\
&= \frac{1}{\sqrt{2k}} \sum_{i,j=0}^{2k-1} c_j (2\Theta(j-i) - 1) |i\rangle |j\rangle |0\rangle_{anc} \\
&= \frac{1}{\sqrt{2k}} \sum_{i=0}^{2k-1} |i\rangle \otimes \sum_{m=0}^{2k-1} (2\Theta(m-i) - 1) |m\rangle \langle m| \left(\sum_{j=0}^{2k-1} c_j |j\rangle \right) |0\rangle_{anc} \\
&= \frac{1}{\sqrt{2k}} \left(\sum_{i=0}^{2k-1} |i\rangle \otimes U^{(i)} |\psi\rangle \right) |0\rangle_{anc}
\end{aligned} \tag{C26}$$

Note that

$$\begin{aligned}
\left(|\bar{+}\rangle \otimes I \right) \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} |\bar{+}\rangle |\psi\rangle |0\rangle_{anc} &= \left(\frac{1}{2k} \sum_{i=0}^{2k-1} U^{(i)} |\psi\rangle \right) |0\rangle_{anc} \\
&= \frac{\hat{\phi}}{k\Delta\phi} |\psi\rangle |0\rangle_{anc} \\
&= \frac{\hat{\phi}}{\phi_{max}} |\psi\rangle |0\rangle_{anc}
\end{aligned} \tag{C27}$$

and so we have

$$\left(H^{\otimes m} \text{CMP}_{P,\phi,anc}^\dagger Z_{anc} \text{CMP}_{P,\phi,anc} H^{\otimes m} \right) |\bar{0}\rangle |\psi\rangle |0\rangle_{anc} = \left(|\bar{0}\rangle \frac{\hat{\phi}}{\phi_{max}} |\psi\rangle + |\Phi^\perp\rangle \right) |0\rangle_{anc} \tag{C28}$$

where $(|\bar{0}\rangle\langle\bar{0}| \otimes I) |\Phi^\perp\rangle = 0$, and we obtain the state $\hat{\phi}|\psi\rangle$ with probability $\frac{|\hat{\phi}|\psi\rangle|^2}{|\phi_{max}|^2}$. Moreover, instead of using the CMP primitive, we are also free to use any other primitive CMP' that instead applies any unitary transformation of the form

$$\text{CMP}' |i\rangle |j\rangle |\text{scratch}\rangle |0\rangle = |\Psi_{i,j,\text{scratch}}\rangle |j < i\rangle \quad (\text{C29})$$

which may significantly reduce the gate cost. An example of such an operation would be to compute the high carry bit of $\bar{j} + i = j - i$, which is 1 iff $j < i$, but leaves the state of the $|i\rangle$, $|j\rangle$ and other ancillary qubits $|\text{scratch}\rangle$ entangled. Once the appropriate phase has been extracted from the $|j < i\rangle$ qubit, we simply run the inverse of this operation. Noting also that

$$|\bar{x}\rangle \langle\bar{x}| \otimes (U^\dagger V U) + |\bar{x}^\perp\rangle \langle\bar{x}^\perp| \otimes \mathbb{I} = (\mathbb{I} \otimes U^\dagger) (|\bar{x}\rangle \langle\bar{x}| \otimes V + |\bar{x}^\perp\rangle \langle\bar{x}^\perp| \otimes \mathbb{I}) (\mathbb{I} \otimes U) \quad (\text{C30})$$

this means that the controlled version of the entire operation $H^{\otimes m} \text{CMP}'^\dagger Z_{anc} \text{CMP}' H^{\otimes m}$ simply needs to control the Z_{anc} operation. We use $\log_2 2k$ ancillary qubits to compare against, an additional $\log_2 2k$ ancillary qubits to compute the carry bits, with an arbitrary state at a single site on the lattice $|\phi_i\rangle = \sum_{a_0, a_1, a_2=0}^1 c_{a_2, a_1, a_0} |a_2, a_1, a_0\rangle$. In general, this circuit involves $\log_2 2k$ logical ANDs (and just as many uncomputations of those ANDs), $6(\log_2 2k - 1)$ many CNOTs, $2\log_2 2k$ many X gates and a single Z gate. The controlled version of this circuit only requires a control on the single Z gate. The circuit for $-\frac{\hat{\phi}}{\Delta\phi}$ is essentially the same, except that we replace Z with XZX.

At this point, we can use these primitives to use the Taylor series method to construct an approximation to $e^{i\hat{\phi}t}$, but this approach does not yield optimal scaling, so we instead adopt the QSVT approach.

Note that in order to apply all $\hat{\phi}_i$ where i denotes the lattice site, we would need to run something like the circuit on top part of Fig. 3 where we control on the family of terms in the Hamiltonian, the lattice site as well as the particular unitary on that lattice site. With this construction however, we can simply control on the family and site indices, as shown in Fig. 3. While in principle we might need to control on the state of some ancillary qubit register to apply each unitary in the LCU for the entire family of $\hat{\phi}$ operators, this becomes unnecessary with the use of the comparator. We now apply much fewer controlled operations, controlled on the state of much fewer qubits. In the above, we have used different ancillary registers for the field registers at sites 0 and $|\Omega| - 1$ (in which case the computation can be trivially parallelized) though in principle we could use the same ancillary registers again. This is just a toy demo for a fictitious Hamiltonian $H = \sum_{\bar{x} \in \Omega} \phi(\bar{x}) + \dots$, but similar constructions, and consequent simplifications, hold for the actual ϕ^4 Hamiltonian we consider.

To compute the LCU for $-\frac{\hat{\phi}}{\Delta\phi}$, we simply run XZX on the ancillary qubit instead of Z to extract the phase (see next section). We can run this circuit in parallel to the one for $\frac{\hat{\phi}}{\Delta\phi}$, for the same gate depth but twice the gate count, to run the sub-select circuit for a single $-\hat{\phi}_a \hat{\phi}_b$ term in the Hamiltonian. A single such term requires $4 \log_2 2k$ ancillary qubits, $2 \log_2 2k$ logical AND computations, $2 \log_2 2k$ logical AND uncomputations (along with any additional ancillas these require), $12(\log_2 2k - 1)$ CNOTs, $4 \log_2 2k$ X gates and 2 Z gates. All such terms in the Hamiltonian will require $D|\Omega|$ times this number many gates.

The circuit depth is the circuit depth of $\log_2 2k$ logical ANDs + circuit depth of $\log_2 2k$ logical AND uncomputations + $8 \log_2 2k - 5$. The circuit depth for the entire group of such terms is at least $2D$, since the same site has to couple to all its spatial neighbors (in the positive direction for each of the D directions), and any site is also the neighbor of some other site (and hence the factor of 2), but this parallelization requires $\frac{|\Omega|}{2}(4 \log_2 2k)$ ancillary qubits.

b. Sub-SELECT circuit for $-\hat{\phi}(\bar{x})\hat{\phi}(\bar{x}')$ In order to keep the coefficients of the (families of) terms in the Hamiltonian positive, we simply absorb the negative sign of the nearest-neighbor coupling site into the LCUs themselves. This means the action of such a term on sites a and b is given on some general state $|\phi\rangle = \sum_{j_1, \dots, j_N} c_{j_1 \dots j_N} |j_1\rangle \otimes \dots \otimes |j_N\rangle$ (where $1 \dots N$ denote the lattice sites) as

$$-\hat{\phi}_a \hat{\phi}_b |\phi\rangle = \left(\frac{\Delta\phi}{2}\right)^2 \sum_{m, m'=0}^{2k-1} \sum_{j_1 \dots j_a \dots j_b \dots j_N=0}^{2k-1} [2\Theta(j_a - m) - 1] [1 - 2\Theta(j_b - m')] c_{j_1 \dots j_a \dots j_b \dots j_N} |j_1\rangle \dots |j_a\rangle \dots |j_b\rangle \dots |j_N\rangle \quad (\text{C31})$$

We assume we are given some arbitrary field state $|\phi\rangle = \sum_{j_1, \dots, j_N=0}^{2k-1} c_{j_1 \dots j_N} |j_1\rangle \dots |j_N\rangle$, with ancillary registers $|\overline{+}\rangle = |+\rangle^{\otimes \log_2 2k} \in \mathcal{H}_{P_a}$, $|\overline{+}\rangle = |+\rangle^{\otimes \log_2 2k} \in \mathcal{H}_{P_b}$, $|0\rangle \in \mathcal{H}_{anc,a}$ and $|0\rangle \in \mathcal{H}_{anc,b}$. Then, using

$$\begin{aligned} \text{CMP}_{A,B,C} |i\rangle |j\rangle |0\rangle &= |i\rangle |j\rangle |\Theta(i-j-1)\rangle \\ 1 - 2\Theta(x-1) &= 2\Theta(-x) - 1 \\ Z|\Theta(i-j-1)\rangle &= [2\Theta(j-i) - 1] |\Theta(i-j-1)\rangle \\ XZX|\Theta(i-j-1)\rangle &= [1 - 2\Theta(j-i)] |\Theta(i-j-1)\rangle \end{aligned} \quad (\text{C32})$$

we have

$$\begin{aligned} & \text{CMP}_{P_a, \phi_a, anc_a}^\dagger \text{CMP}_{P_b, \phi_b, anc_b}^\dagger Z_{anc_a} X_{anc_b} Z_{anc_b} X_{anc_b} \text{CMP}_{P_b, \phi_b, anc_b} \text{CMP}_{P_a, \phi_a, anc_a} |\overline{+}\rangle |\overline{+}\rangle |\phi\rangle |0\rangle_{anc_a} |0\rangle_{anc_b} \\ &= \text{CMP}_{P_a, \phi_a, anc_a}^\dagger \text{CMP}_{P_b, \phi_b, anc_b}^\dagger Z_{anc_a} X_{anc_b} Z_{anc_b} X_{anc_b} \text{CMP}_{P_b, \phi_b, anc_b} \text{CMP}_{P_a, \phi_a, anc_a} |\overline{+}\rangle |\overline{+}\rangle \\ & \quad \left(\sum_{j_1, \dots, j_N=0}^{2k-1} c_{j_1 \dots j_N} |j_1\rangle \dots |j_N\rangle \right) |0\rangle_{anc_a} |0\rangle_{anc_b} \\ &= \text{CMP}_{P_a, \phi_a, anc_a}^\dagger \text{CMP}_{P_b, \phi_b, anc_b}^\dagger Z_{anc_a} X_{anc_b} Z_{anc_b} X_{anc_b} \\ & \quad \left(\sum_{j_1, \dots, j_N=0}^{2k-1} \sum_{m_a, m_b=0}^{2k-1} \frac{1}{2k} c_{j_1 \dots j_a \dots j_b \dots j_N} [2\Theta(j_a - m_a) - 1] [1 - 2\Theta(j_b - m_b)] \right. \\ & \quad \left. |m_a\rangle |m_b\rangle |j_1 \dots j_N\rangle |\Theta(m_a - j_a - 1)\rangle |\Theta(m_b - j_b - 1)\rangle \right) \\ &= \sum_{j_1, \dots, j_N=0}^{2k-1} \sum_{m_a, m_b=0}^{2k-1} \frac{1}{2k} c_{j_1 \dots j_N} [2\Theta(j_a - m_a) - 1] [1 - 2\Theta(j_b - m_b)] |m_a\rangle |m_b\rangle |j_1 \dots j_N\rangle |0\rangle_{anc_a} |0\rangle_{anc_b} \\ &= \sum_{j_1, \dots, j_N=0}^{2k-1} \sum_{m_a, m_b=0}^{2k-1} \frac{1}{2k} c_{j_1 \dots j_N} \left\{ \sum_{j'_a=0}^{2k-1} [2\Theta(j'_a - m_a) - 1] |j'_a\rangle \langle j'_a|_a \right\} \left\{ \sum_{j'_b=0}^{2k-1} [1 - 2\Theta(j'_b - m_b)] |j'_b\rangle \langle j'_b|_b \right\} \\ & \quad |m_a\rangle |m_b\rangle |j_1 \dots j_N\rangle |00\rangle_{anc} \\ &= \left(\sum_{m_a, m_b=0}^{2k-1} \frac{1}{2k} |m_a\rangle |m_b\rangle U_{\phi_a}^{(m_a)} U_{\phi_b}^{(m_b)} \right) \left(\sum_{j_1, \dots, j_N=0}^{2k-1} c_{j_1 \dots j_N} |j_1 \dots j_N\rangle \right) |00\rangle_{anc} \\ &= \sum_{m_a, m_b=0}^{2k-1} \frac{1}{2k} |m_a\rangle |m_b\rangle \left(U_{\phi_a}^{(m_a)} U_{\phi_b}^{(m_b)} |\phi\rangle \right) |00\rangle_{anc} \\ &\equiv |\phi'\rangle \end{aligned} \quad (\text{C33})$$

Note that

$$\begin{aligned} \left(|\overline{+}\rangle |\overline{+}\rangle \otimes I \right) |\phi'\rangle &= -\frac{1}{(2k)^2} \left(\sum_{m_a=0}^{2k-1} U_{\phi_a}^{(m_a)} \right) \left(\sum_{m_b=0}^{2k-1} U_{\phi_b}^{(m_b)} \right) |\phi\rangle |00\rangle_{anc} \\ &= -\frac{\hat{\phi}_a \hat{\phi}_b}{|\phi_{max}|^2} |\phi\rangle |00\rangle_{anc} \end{aligned} \quad (\text{C34})$$

Thus, we have

$$\begin{aligned} & \left(H^{\otimes 2 \log_2 2k} \text{CMP}_{P_a, \phi_a, anc_a}^\dagger \text{CMP}_{P_b, \phi_b, anc_b}^\dagger Z_{anc_a} X_{anc_b} Z_{anc_b} X_{anc_b} \text{CMP}_{P_b, \phi_b, anc_b} \text{CMP}_{P_a, \phi_a, anc_a} H^{\otimes 2 \log_2 2k} \right) |0\rangle^{\otimes 2 \log_2 2k} |\phi\rangle |0\rangle^{\otimes 2} \\ &= \left(|0\rangle^{\otimes 2 \log_2 2k} \left(-\frac{\hat{\phi}_a \hat{\phi}_b}{|\phi_{max}|^2} \right) |\phi\rangle + |\Phi^\perp\rangle \right) |00\rangle_{anc} \\ &= U_{PREP}^\dagger U_{SUB-SELECT_{\phi_a \phi_b}} U_{PREP} |0\rangle^{\otimes 2 \log_2 2k} |\phi\rangle |0\rangle^{\otimes 2} \end{aligned} \quad (\text{C35})$$

The entire sub-select unitary for this family of terms is then given by

$$\begin{aligned}
U_{SUB-SELECT_{\phi\phi}} = & \prod_{\vec{x} \in \Omega} \prod_{i=1}^D \text{CMP}'_{P_a, \phi_{\vec{x}}, anc_a} \left(|\vec{x}\rangle \langle \vec{x}| \otimes |i\rangle \langle i| \otimes Z_{anc,a} + |\Phi^\perp\rangle \langle \Phi^\perp| \otimes \mathbb{I} \right) \text{CMP}'_{P_a, \phi_{\vec{x}}, anc_a} \\
& \cdot \text{CMP}'_{P_b, \phi_{\vec{x}+\delta\hat{x}_i}, anc_b} X_{anc,b} \left(|\vec{x}\rangle \langle \vec{x}| \otimes |i\rangle \langle i| \otimes Z_{anc,b} + |\Phi^\perp\rangle \langle \Phi^\perp| \otimes \mathbb{I} \right) X_{anc,b} \text{CMP}'_{P_b, \phi_{\vec{x}+\delta\hat{x}_i}, anc_b}
\end{aligned} \tag{C36}$$

Thus, we require $2|\Omega|D$ many controlled Z gates (here, the factor of 2 comes from the fact that we have to apply $\hat{\phi}$ at each of the neighboring sites), each controlled on the state of $\log_2 |\Omega|D + 2 = \log_2 4|\Omega|D$ many qubits (here, the factor of 2 comes from the fact that there are 4 groups of families in the Hamiltonian). Each of these can in turn be performed using [76, 80] $\log_2 4|\Omega|D - 1 = \log_2 2|\Omega|D$ many logical AND computations (and as many uncomputations) using $\log_2 4|\Omega|D - 2 = \log_2 |\Omega|D$ ancillary qubits initialized in the $|0\rangle$ state. Thus, these controlled operations contribute a total T gate count of $4 \times 2|\Omega|D \times \log_2 (2|\Omega|D) = 8|\Omega|D \log_2 (2|\Omega|D)$, independent of k (or equivalently, ϕ_{max}).

In addition, each CMP' operation requires $\log_2 2k$ many logical ANDs (and just as many uncomputations of these ANDs via the CMP'^\dagger operations). There are $2|\Omega|D$ such operations (here, the factor of 2 comes from the 2 sites). These contribute a total T gate count of $4 \times 2|\Omega|D \times \log_2 2k = 8|\Omega|D \log_2 (2k)$.

The total T gate count for the $\phi\phi$ part of the Hamiltonian is therefore

$$\begin{aligned}
\text{Count(T)}_{\phi\phi} &= 8|\Omega|D (\log_2 (2|\Omega|D) + \log_2 (2k)) \\
&= 8|\Omega|D (\log_2 |\Omega|Dk + 2) \\
&= 8|\Omega|D \left(\log_2 \frac{|\Omega|\phi_{max}|D}{\Delta\phi} + 2 \right)
\end{aligned} \tag{C37}$$

c. SELECT $_{\Phi^2}$ term For $L = 2k$, $n_{max} = k^2$ and $n_i = (k - i - 1)^2$ in Eq. (87), we get

$$\begin{aligned}
\left(\frac{\hat{\phi}}{\Delta\phi} \right)^2 &= \frac{1}{2} \sum_{i=0}^{2k^2-1} U^{(i)}, \quad \text{where} \\
U^{(i)} &= \sum_{j=0}^{2k-1} [2\Theta(k^2 + (k - j - 1)^2 - i - 1) - 1] |j\rangle \langle j|
\end{aligned} \tag{C38}$$

Then, we run the following series of operations

$$\begin{aligned}
& |+\rangle^{\otimes \log_2 2k^2} |\phi\rangle |k\rangle |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |\text{scratch}\rangle \\
= & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a \dots j_{|\Omega|}\rangle \right) |k\rangle \\
& |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |\text{scratch}\rangle \\
\stackrel{U_{P+1}}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a \dots j_{|\Omega|}\rangle \right) |k\rangle \\
& |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |\text{scratch}\rangle \\
\stackrel{U_{\phi_a+1}}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 \dots j_{|\Omega|}\rangle \right) |k\rangle \\
& |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |\text{scratch}\rangle \\
\stackrel{U_{\phi-k}}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right) \\
& |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |\text{scratch}\rangle \\
\stackrel{\text{CNOT}_{\phi, \phi_{anc}} \otimes \text{CNOT}_{K, K_{anc}}}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right) \\
& |j_a + 1 - k\rangle |k\rangle |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |\text{scratch}\rangle \\
\stackrel{U_{\phi \times \phi_{anc}, \phi_{anc}^2} \otimes U_{K \times K_{anc}, K_{anc}^2}}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right) \\
& |j_a + 1 - k\rangle |k\rangle |(j_a + 1 - k)^2\rangle |k^2\rangle |\text{scratch}\rangle \\
\stackrel{\phi_{anc}^2 + K_{anc}^2}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right) \\
& |j_a + 1 - k\rangle |k\rangle |k^2 + (j_a + 1 - k)^2\rangle |k^2\rangle |\text{scratch}\rangle
\end{aligned} \tag{C39}$$

Let us denote this entire sequence of operations as $U_{initial}$ for succinctness. The point of these operations is to bring the registers to a form where a comparison of values can be made followed by an extraction of a phase, similar to the toy example of the $\hat{\phi}$ operator discussed previously. Let $n = \log_2 2k$. The two incrementers above require $2n - 3$ and $n - 2$ ANDs respectively[91], the subtraction n ANDs, the CNOTs none, the multiplications $(2n^2 - 1) + (2(n - 1)^2 - 1)$ ANDs[92], and the addition $2n - 1$ ANDs. Adding all these, the cost of $U_{initial}$ is $4n^2 + 2n - 6$ ANDs. Since the cost of $U_{initial}^\dagger$ is the same, these operations cost a total of $2|\Omega| (4n^2 + 2n - 6)$ ANDs, which equals $8|\Omega| (4n^2 + 2n - 6)$ T gates.

The ancilla count for $U_{initial}$ is given by the number of ancillae used to store values, as well as those used in the $|\text{scratch}\rangle$ register actually used to compute the values. Those used to store values are given by

- $\log_2(2k^2)$ (for the comparison)
- $\log_2(k)$ (store $|k\rangle$)
- $\log_2(2k)$ (copy $|\phi\rangle_a$)
- $\log_2(k)$ (copy $|k\rangle$)
- $2 \log_2(2k)$ (product of two $\log_2(2k)$ numbers)
- $2 \log_2(k)$ (product of two $\log_2(k)$ numbers)

In addition, the number of ancillae used in $|\textit{scratch}\rangle$ to actually carry out the various computations in $U_{\textit{initial}}$ is given by

- $\log_2(2k^2) - 1$ (U_{P+1})
- $\log_2(2k) - 1$ (U_{ϕ_a+1})
- $\log_2(2k)$ ($U_{\phi-k}$)
- $[\log_2(2k) + 1][2\log_2(2k) - 1] + 1$ ($U_{\phi \times \phi_{anc}, \phi_{anc}^2}$)
- $[\log_2(k) + 1][2\log_2(k) - 1] + 1$ ($U_{K \times K_{anc}, K^2}$)
- $2\log_2(2k)$ ($\phi_{anc}^2 + K_{anc}^2$)

Once we have run $U_{\textit{initial}}$, we then run the following series of operations.

$$\begin{aligned}
& \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad \left. |j_a + 1 - k\rangle |k\rangle |k^2 + (j_a + 1 - k)^2\rangle |k^2\rangle \right) |0\rangle |\textit{scratch}\rangle \\
\stackrel{\text{CMP}}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad \left. |j_a + 1 - k\rangle |k\rangle |k^2 + (j_a + 1 - k)^2\rangle |k^2\rangle |\Theta(i - k^2 - (k - j - 1)^2)\rangle_{anc} \right) |\textit{scratch}\rangle \\
\stackrel{Z_{anc}}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} [1 - 2\Theta(i - k^2 - (k - j - 1)^2)] |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad \left. |j_a + 1 - k\rangle |k\rangle |k^2 + (j_a + 1 - k)^2\rangle |k^2\rangle |\Theta(i - k^2 - (k - j - 1)^2)\rangle_{anc} \right) |\textit{scratch}\rangle \\
= & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} [2\Theta(k^2 + (k - j - 1)^2) - i - 1] |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad \left. |j_a + 1 - k\rangle |k\rangle |k^2 + (j_a + 1 - k)^2\rangle |k^2\rangle |\Theta(i - k^2 - (k - j - 1)^2)\rangle_{anc} \right) |\textit{scratch}\rangle \\
\stackrel{\text{CMP}^\dagger}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} [2\Theta(k^2 + (k - j - 1)^2) - i - 1] |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad \left. |j_a + 1 - k\rangle |k\rangle |k^2 + (j_a + 1 - k)^2\rangle |k^2\rangle \right) |0\rangle_{anc} |\textit{scratch}\rangle \\
\stackrel{U_{\textit{initial}}^\dagger}{\longrightarrow} & \left(\frac{1}{\sqrt{2k^2}} \sum_{i=0}^{2k^2-1} |i\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} [2\Theta(k^2 + (k - j - 1)^2) - i - 1] |j_1 \dots j_a \dots j_{|\Omega|}\rangle \right) |k\rangle \\
& \quad |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2\log_2 2k-1} |0\rangle^{\otimes 2\log_2 k-1} |\textit{scratch}\rangle
\end{aligned} \tag{C40}$$

With a similar analysis as before, we see that with the identifications

$$\begin{aligned}
U_{PREP} &= H^{\otimes [1+2\log_2 k]} \otimes \mathbb{I} \\
U_{SELECT} &= U_{\textit{initial}}^\dagger \text{CMP}^\dagger Z_{anc} \text{CMP} U_{\textit{initial}}
\end{aligned} \tag{C41}$$

we have

$$U_{PREP}^\dagger U_{SELECT} U_{PREP} |\bar{0}\rangle |\phi\rangle |0\rangle_{anc} = \left(|\bar{0}\rangle \frac{\hat{\phi}^2}{|\phi_{max}|^2} |\phi\rangle + |\Phi^\perp\rangle \right) |0\rangle_{anc} \tag{C42}$$

where $(|\bar{0}\rangle \langle \bar{0}| \otimes I) |\Phi^\perp\rangle = 0$, and we obtain the state $\hat{\phi}^2 |\phi\rangle$ with probability $\frac{|\hat{\phi}^2 |\phi\rangle|^2}{|\phi_{max}|^4}$ upon post-selection.

The CMP' operation (and its inverse), replacing the CMP operation as before, compares two $\log_2 2k^2$ -qubit numbers and costs $\log_2 2k^2$ many logical ANDs. Thus, the comparisons require $|\Omega| \log_2 2k^2$ many logical ANDs across the entire lattice, contributing to a T gate count of $4|\Omega| \log_2 2k^2$.

In addition, we have $|\Omega|$ many multi-controlled Z gates to apply across the entire lattice, which incurs a cost of $4|\Omega| - 4$ T gates using the unary iteration method of [70].

In all, the total T gate count for this family of terms is given by the following, as also noted in the main text in Eq. (115)

$$\text{Count}(T)_{\phi^2} = 8|\Omega| (4 \log_2^2 k + 11 \log_2 k + 1) - 4 \quad (\text{C43})$$

d. SELECT $_{\Phi^4}$ term In much the same way as for the ϕ^2 term, we can construct an LCU for $\hat{\phi}^4$ as

$$\begin{aligned} \left(\frac{\hat{\phi}}{\Delta\phi} \right)^4 &= \frac{1}{2} \sum_{i=0}^{2k^4-1} U^{(i)}, \quad \text{where} \\ U^{(i)} &= \sum_{j=0}^{2k-1} [2\Theta(k^4 + (k-j-1)^4 - i - 1) - 1] |j\rangle \langle j| \end{aligned} \quad (\text{C44})$$

with essentially the same method for $\hat{\phi}^2$ as described above, except that U_{initial} is modified as follows.

$$\begin{aligned} &|+\rangle^{\otimes \log_2 2k^4} |\phi\rangle |k\rangle |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |0\rangle^{\otimes 2 \log_2 k-1} \\ &|0\rangle^{\otimes 4 \log_2 2k-3} |0\rangle^{\otimes 4 \log_2 k-2} \quad |\text{scratch}\rangle \\ &= \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a \dots j_{|\Omega|}\rangle \right) |k\rangle \\ &\quad |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |0\rangle^{\otimes 2 \log_2 k-1} \\ &\quad |0\rangle^{\otimes 4 \log_2 2k-3} |0\rangle^{\otimes 4 \log_2 k-2} \quad |\text{scratch}\rangle \\ &\xrightarrow{U_{P+1}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a \dots j_{|\Omega|}\rangle \right) |k\rangle \\ &\quad |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |0\rangle^{\otimes 2 \log_2 k-1} \\ &\quad |0\rangle^{\otimes 4 \log_2 2k-3} |0\rangle^{\otimes 4 \log_2 k-2} \quad |\text{scratch}\rangle \\ &\xrightarrow{U_{\phi_a+1}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 \dots j_{|\Omega|}\rangle \right) |k\rangle \\ &\quad |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |0\rangle^{\otimes 2 \log_2 k-1} \\ &\quad |0\rangle^{\otimes 4 \log_2 2k-3} |0\rangle^{\otimes 4 \log_2 k-2} \quad |\text{scratch}\rangle \\ &\xrightarrow{U_{\phi-k}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|=0}}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right) \\ &\quad |0\rangle^{\otimes \log_2 2k} |0\rangle^{\otimes \log_2 k} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |0\rangle^{\otimes 2 \log_2 k-1} \end{aligned} \quad (\text{C45})$$

$$\begin{aligned}
& \xrightarrow{\text{CNOT}_{\phi, \phi_{anc}} \otimes \text{CNOT}_{K, K_{anc}}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad |j_a + 1 - k\rangle |k\rangle |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 2k-1} |0\rangle^{\otimes 2 \log_2 k-1} |0\rangle^{\otimes 2 \log_2 k-1} \\
& \quad \left. |0\rangle^{\otimes 4 \log_2 2k-3} |0\rangle^{\otimes 4 \log_2 k-2} |scratch\rangle \right) \\
& \xrightarrow{U_{\phi \times \phi_{anc}, \phi_{anc}^2} \otimes U_{K \times K_{anc}, K_{anc}^2}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad |j_a + 1 - k\rangle |k\rangle |(j_a + 1 - k)^2\rangle |0\rangle^{\otimes 2 \log_2 2k-1} |k^2\rangle |0\rangle^{\otimes 2 \log_2 k-1} \\
& \quad \left. |0\rangle^{\otimes 4 \log_2 2k-3} |0\rangle^{\otimes 4 \log_2 k-2} |scratch\rangle \right) \\
& \xrightarrow{\text{CNOT ladder} \otimes \text{CNOT ladder}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad |j_a + 1 - k\rangle |k\rangle |(j_a + 1 - k)^2\rangle |(j_a + 1 - k)^2\rangle |k^2\rangle |k^2\rangle \\
& \quad \left. |0\rangle^{\otimes 4 \log_2 2k-3} |0\rangle^{\otimes 4 \log_2 k-2} |scratch\rangle \right) \\
& \xrightarrow{U_{\phi^2 \times \phi^2, \phi_{anc}^4} \otimes U_{k^2 \times k^2, k_{anc}^4}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad |j_a + 1 - k\rangle |k\rangle |(j_a + 1 - k)^2\rangle |(j_a + 1 - k)^2\rangle |k^2\rangle |k^2\rangle \\
& \quad \left. |(j_a + 1 - k)^4\rangle |k^4\rangle |scratch\rangle \right) \\
& \xrightarrow{U_{\phi_{anc}^4 + K_{anc}^4}} \left(\frac{1}{\sqrt{2k^4}} \sum_{i=0}^{2k^4-1} |i+1\rangle \right) \left(\sum_{j_1 \dots j_a \dots j_{|\Omega|}=0}^{2k-1} c_{j_1 \dots j_a \dots j_{|\Omega|}} |j_1 \dots j_a + 1 - k \dots j_{|\Omega|}\rangle |k\rangle \right. \\
& \quad |j_a + 1 - k\rangle |k\rangle |(j_a + 1 - k)^2\rangle |(j_a + 1 - k)^2\rangle |k^2\rangle |k^2\rangle \\
& \quad \left. |k^4 + (j_a + 1 - k)^4\rangle |k^4\rangle |scratch\rangle \right)
\end{aligned} \tag{C46}$$

As before, the ancilla count for $U_{initial}$ is given by the number of ancillae used to store values, as well as those used in the $|scratch\rangle$ register actually used to compute the values. Those used to store values are now given by

- $\log_2(2k^4)$ (for the comparison)
- $\log_2(k)$ (store $|k\rangle$)
- $\log_2(2k)$ (copy $|\phi\rangle_a$)
- $\log_2(k)$ (copy $|k\rangle$)
- $2 \log_2(2k)$ (product of two $\log_2(2k)$ numbers)
- $2 \log_2(k)$ (product of two $\log_2(k)$ numbers)
- $2 \log_2(2k)$ (copy $(j_a + 1 - k)^2$)
- $2 \log_2(k)$ (copy k^2)
- $4 \log_2(2k)$ (product of two $2 \log_2(2k)$ numbers)
- $4 \log_2(k)$ (product of two $2 \log_2(k)$ numbers)

In addition, the number of ancillae used in $|scratch\rangle$ to actually carry out the various computations in $U_{initial}$ is given by

- $\log_2(2k^2) - 1$ (U_{P+1})

- $\log_2(2k) - 1$ ($U_{\phi_{a+1}}$)
- $\log_2(2k)$ ($U_{\phi_{-k}}$)
- $[\log_2(2k) + 1][2\log_2(2k) - 1] + 1$ ($U_{\phi \times \phi_{anc}, \phi_{anc}^2}$)
- $[\log_2(k) + 1][2\log_2(k) - 1] + 1$ ($U_{K \times K_{anc}, K^2}$)
- $[2\log_2(2k) + 1][4\log_2(2k) - 1] + 1$ ($U_{\phi^2 \times \phi^2, \phi_{anc}^4}$)
- $[2\log_2(k) + 1][4\log_2(k) - 1] + 1$ ($U_{K^2 \times K^2, K_{anc}^2}$)
- $4\log_2(2k)$ ($U_{\phi_{anc}^4 + K_{anc}^4}$)

With this redefinition of $U_{initial}$ and the resultant definitions using Eq. (C41), we obtain

$$U_{PREP}^\dagger U_{SELECT} U_{PREP} \overline{|0\rangle} |\phi\rangle |0\rangle_{anc} = \left(\overline{|0\rangle} \frac{\hat{\phi}^4}{|\phi_{max}|^4} |\phi\rangle + |\Phi^\perp\rangle \right) |0\rangle_{anc} \quad (C47)$$

where $(\overline{|0\rangle} \langle \overline{0}| \otimes I) |\Phi^\perp\rangle = 0$, and we obtain the state $\hat{\phi}^4 |\phi\rangle$ with probability $\frac{|\hat{\phi}^4 |\phi\rangle|^2}{|\phi_{max}|^8}$ upon post-selection.

Just as before, let us denote this entire sequence as $U_{initial}$. Let $n = \log_2 2k$. The two incrementers cost $(n-2) + (4n-3)$ ANDs, the subtraction n , the (first round of) multiplications $(2n^2-1) + (2(n-1)^2-1)$, the (first round of) CNOTs none, the (second round of) multiplications $(8n^2-1) + (8(n-1)^2-1)$, and the addition $4n$ ANDs. Adding all these, the cost of $U_{initial}$ is now $20n^2 - 10n + 1$ ANDs. Since the cost of $U_{initial}^\dagger$ is the same, these operations cost a total of $2|\Omega| (20n^2 - 10n + 1)$ ANDs, or $8|\Omega| (20n^2 - 10n + 1)$ T gates.

Just as before, we follow this with a CMP' operation, followed by a (multi)-controlled Z , then the CMP'^\dagger operation. The CMP' operation (and its inverse) compares two $\log_2 2k^4$ -qubit numbers and costs $\log_2 2k^4$ many ANDs. Thus, the comparisons require $|\Omega| \log_2 2k^4$ many logical ANDs across the entire lattice, contributing to a T gate count of $4|\Omega| \log_2 2k^4$.

In addition, each Z gate must be controlled on the state of $2 + \log_2 |\Omega| = \log_2 4|\Omega|$ many qubits. Each of these can be performed using $\log_2 4|\Omega| - 1 = \log_2 2|\Omega|$ many logical ANDs (and as many uncomputations). Thus, these controlled gates contribute a total of $|\Omega| \log_2 2|\Omega|$ many logical ANDs, and therefore $4|\Omega| \log_2 2|\Omega|$ many T gates.

In all, the total T gate count for this family of terms is given by

$$\text{Count}(T)_{\phi^4} = 2|\Omega| (20 \log_2^2 k + 38 \log_2 k + 2 \log_2 |\Omega| + 15) \quad (C48)$$

2. Block encoding of H_{amp} using binary decompositions

In this section first we describe in detail Algorithm IIIa, outlined in Section IVE3, that simulates H_{amp} (Eq. 11) using an LCU decomposition of Φ , obtained from the binary decomposition of integers. This helps in expressing Φ as sum of Z operators and hence we obtain LCU decomposition of Φ^2 and Φ^4 as sum of Z operators. The ℓ_1 norm of the coefficients of an LCU is referred to as the ℓ_1 norm of the decomposition or operator. We recall that $H_{amp} = c\mathbb{I} + H'_{amp}$, for some constant c . Thus it is sufficient to implement $e^{-iH'_{amp}\tau}$ because it is equal to $e^{-iH_{amp}\tau}$ up to some global phase. We describe a block-encoding of H'_{amp} by first block encoding each of its partitions and then combining them using Theorem 13. We implement the circuits using Clifford+T and rotation gates and apply optimization techniques (for example Theorem 14). Thus we prove Theorem 22 in Section IVE3. Next, we prove a bound on the ℓ_1 norm of the Hamiltonian in Section C2a. Finally, we provide a more detailed explanation about Algorithm IIIb outlined in Section IVE4, where we obtain LCU decomposition of Φ^2 and Φ^4 from binary representation of integers.

a. Decomposition of Φ : We decompose $\frac{\Phi}{\Delta\Phi}$ as follows, as done in [15].

$$\begin{aligned} \frac{\Phi}{\Delta\Phi} &= \text{diag}(-k, \dots, -1, 0, 1, \dots, k-1) = \frac{1}{2} \text{diag}(-2k+1, \dots, -1, 1, \dots, 2k-1) - \frac{1}{2} \mathbb{I} \\ &= -\frac{1}{2} \sum_{j=0}^{\log_2 k} 2^j Z_j - \frac{1}{2} \mathbb{I} \end{aligned} \quad (C49)$$

Number of non-identity unitaries in the above LCU is $(\log_2 k + 1)$ and the ℓ_1 norm is $\frac{1}{2} \left[1 + \sum_{j=0}^{\log_2 k} 2^j \right] = k$.

b. *Decomposition of Φ^2* : From Eq. (C49), we have

$$\begin{aligned}
\left(\frac{\Phi}{\Delta\Phi}\right)^2 &= \left(\frac{\Phi}{\Delta\Phi} + \frac{\mathbb{I}}{2}\right)^2 - \left(\frac{\Phi}{\Delta\Phi} + \frac{\mathbb{I}}{2}\right) + \frac{\mathbb{I}}{4} \\
&= \frac{1}{12} (4^{\log_2 k+1} - 1) \mathbb{I} + \sum_{j=0}^{\log_2 k-1} \sum_{k>j}^{\log_2 k} 2^{j+k-1} Z_j Z_k + \frac{1}{2} \sum_{j=0}^{\log_2 k} 2^j Z_j + \frac{\mathbb{I}}{4} \\
&= \frac{1}{6} (2^{2\log_2 k+1} + 1) \mathbb{I} + \sum_{j=0}^{\log_2 k-1} \sum_{k>j}^{\log_2 k} 2^{j+k-1} Z_j Z_k + \frac{1}{2} \sum_{j=0}^{\log_2 k} 2^j Z_j.
\end{aligned} \tag{C50}$$

Number of non-identity unitaries in this expansion is

$$1 + \frac{\log_2 k (\log_2 k + 1)}{2} + (\log_2 k + 1) = 1 + \frac{(\log_2 k + 1)(\log_2 k + 2)}{2}. \tag{C51}$$

If $\log_2 k = \zeta$, then the ℓ_1 norm is

$$\begin{aligned}
&\frac{1}{6} (2^{2\zeta+1} + 1) + \frac{1}{2} \sum_{j=0}^{\zeta} 2^j Z_j + \frac{1}{2} \sum_{j=0}^{\zeta-1} 2^j \sum_{k>j}^{\zeta} 2^k \\
&= \frac{1}{6} (2k^2 + 1) + \frac{1}{2} (2^{\zeta+1} - 1) + \frac{1}{2} \sum_{j=0}^{\zeta-1} 2^{j+j+1} \sum_{k=0}^{\zeta-(j+1)} 2^k \\
&= \frac{k^2}{3} + \frac{1}{6} + k - \frac{1}{2} + \frac{1}{2} \sum_{j=0}^{\zeta-1} 2^{2j+1} (2^{\zeta-j} - 1) \\
&= \frac{k^2}{3} + k - \frac{1}{3} + \sum_{j=0}^{\zeta-1} 2^{\zeta+j} - \sum_{j=0}^{\zeta-1} 4^j \\
&= \frac{k^2}{3} + k - \frac{1}{3} + 2^\zeta (2^\zeta - 1) - \frac{4^\zeta - 1}{3} \\
&= \frac{k^2}{3} + k - \frac{1}{3} + k^2 - k - \frac{k^2}{3} + \frac{1}{3} = k^2.
\end{aligned} \tag{C52}$$

c. *Decomposition of Φ^4* : Since $\left(\frac{\Phi}{\Delta\Phi}\right)^4 = \left(\left(\frac{\Phi}{\Delta\Phi}\right)^2\right)^2$, so we can express it as sum of terms with 1, 2, 3 and up to 4 Z gates. So number of unitaries in the decomposition can be at most

$$\begin{aligned}
&1 + \binom{\zeta}{1} + \binom{\zeta}{2} + \binom{\zeta}{3} + \binom{\zeta}{4} \in \\
&= \frac{1}{24} (\zeta^4 - 2\zeta^3 + 11\zeta^2 + 14\zeta + 24) \quad [\zeta = \log_2 k] \\
&\in O(\zeta^4) \quad \text{or} \quad O(\log_2^4 k).
\end{aligned} \tag{C53}$$

To derive the ℓ_1 norm of this decomposition we first observe that

$$\left(\frac{\Phi}{\Delta\Phi}\right)^4 = \left(\left(\frac{\Phi}{\Delta\Phi}\right)^2 - \frac{2k^2+1}{6}\mathbb{I}\right)^2 + 2\frac{2k^2+1}{6} \left(\left(\frac{\Phi}{\Delta\Phi}\right)^2 - \frac{2k^2+1}{6}\mathbb{I}\right) + \left(\frac{2k^2+1}{6}\right)^2 \mathbb{I}.$$

From Eq. C52 we know that that the ℓ_1 norm of $\left(\left(\frac{\Phi}{\Delta\Phi}\right)^2 - \frac{2k^2+1}{6}\mathbb{I}\right)$ is $k^2 - \frac{2k^2+1}{6} = \frac{2k^2}{3} - \frac{1}{6}$. Thus the ℓ_1 norm of $\left(\frac{\Phi}{\Delta\Phi}\right)^4$ is at most

$$\left(\frac{2k^2}{3} - \frac{1}{6}\right)^2 + 2 \cdot \frac{2k^2+1}{6} \left(\frac{2k^2}{3} - \frac{1}{6}\right) + \left(\frac{2k^2+1}{6}\right)^2 = k^4. \tag{C54}$$

Proof of Theorem 22. We re-collect that we have partitioned the Hamiltonian H'_{amp} (i.e. H_{amp} without the \mathbb{I} part) as follows.

$$H_{1\bar{x}} = \sum_j \alpha_j Z_j + \sum_{j,k} \alpha_{jk} Z_j Z_k + \mathcal{F} \left(\sum_j \gamma_j Z_j + \sum_{j,k} \gamma_{jk} Z_j Z_k \right) \mathcal{F}^\dagger \quad (\text{C55})$$

$$H_{2\bar{x}} = \sum_{j,k,l} \alpha_{jkl} Z_j Z_k Z_l + \sum_{j,k,l,m} \alpha_{jklm} Z_j Z_k Z_l Z_m \quad (\text{C56})$$

$$H_{12} = \sum_{\bar{x} \in \Omega} H_{1\bar{x}} + H_{2\bar{x}} \quad (\text{C57})$$

$$H_{3\bar{x}\vec{x}'} = \sum_{j,j'} \beta_{jj'} (Z_j)_{\bar{x}} (Z_{j'})_{\vec{x}'} \quad (\text{C58})$$

$$H_3 = \sum_{(\bar{x}, \vec{x}') \in E_D} H_{3\bar{x}\vec{x}'} \quad (\text{C59})$$

$$H'_{amp} = H_{12} + H_3 \quad (\text{C60})$$

Let $L_1 = \binom{\log_2 k+1}{1}$, $L_2 = \binom{\log_2 k+1}{2}$, $L_3 = \binom{\log_2 k+1}{3}$, $L_4 = \binom{\log_2 k+1}{4}$ and $L_5 = (\log_2 k + 1)^2$.

d. Block encoding of $H_{1\bar{x}}$: The ancilla preparation sub-routine is as follows. We assume a bijective map between $j' = (j, k)$ to some integer in $[L_1 + 1, L_1 + L_2]$.

$$\begin{aligned} \text{PREP}_{1\bar{x}} |0\rangle^{1+\log_2(L_1+L_2)} &= \frac{1}{\mathcal{N}_{1\bar{x}}} \left(\sum_{j=1}^{L_1} \sqrt{\alpha_j} |j, 0\rangle + \sum_{j'=L_1+1}^{L_1+L_2} \sqrt{\alpha_{j'}} |j', 0\rangle + \sum_{j=1}^{L_1} \sqrt{\gamma_j} |j, 1\rangle \right. \\ &\quad \left. + \sum_{j'=L_1+1}^{L_1+L_2} \sqrt{\gamma_{j'}} |j', 0\rangle \right) \quad [\mathcal{N}_{1\bar{x}} \text{ is normalization constant.}] \quad (\text{C61}) \end{aligned}$$

The last qubit is used to select the QFT. We require $1 + \log_2(L_1 + L_2)$ qubits. For the state preparation we require $1 + \log_2(L_1 + L_2)$ H, $4(L_1 + L_2) - 2$ rotation gates and $4(L_1 + L_2) + 3 \log_2(L_1 + L_2) - 4$ CNOT.

The select sub-routine does the following.

$$\begin{aligned} \text{SELECT}_{1\bar{x}} |j, 0\rangle |\psi\rangle &\rightarrow |j, 0\rangle Z_j |\psi\rangle \\ \text{SELECT}_{1\bar{x}} |j, 1\rangle |\psi\rangle &\rightarrow |j, 1\rangle \mathcal{F} Z_j \mathcal{F}^\dagger |\psi\rangle \end{aligned} \quad (\text{C62})$$

If $j > L_1 + 1$ then we apply two Z gates depending on the mapping. The last qubit is used to selectively apply the pairs of $(\log_2 k + 1)$ -qubit QFT. We require $L_1 + L_2$ compute-uncompute pairs of $C^{\log_2(L_1+L_2)} X$ gates, which can be synthesized efficiently using Theorem 14 [27]. If we divide the control qubits into M_1 groups such that the j^{th} group has $\frac{1}{r_j}$ fraction of the qubits, then we require

$$\sum_{i=1}^{M_1} (L_1 + L_2)^{\frac{1}{r_i}} C^{\frac{\log(L_1+L_2)}{r_i}} X + (L_1 + L_2) \cdot C^{M_1} X$$

(compute-uncompute) pairs of gates. We assume equal partitioning into 2 groups i.e. $M_1 = 2$ and $r_1 = \frac{1}{2}$. Then, using the constructions in [75, 76], i.e. from Eq. 85 we require

$$4\sqrt{L_1 + L_2} (\log_2(L_1 + L_2) - 2) + 4(L_1 + L_2)$$

T gates and

$$\sqrt{L_1 + L_2} (4 \log_2(L_1 + L_2) - 6) + 5(L_1 + L_2)$$

CNOT gates. Additionally we require $(\log_2 k + 1) + \frac{\log_2 k (\log_2 k + 1)}{2} = \frac{(\log_2 k + 1)(\log_2 k + 2)}{2}$ number of CZ gates and two $(\log_2 k + 1)$ -qubit QFT. To implement the QFTs we require approximately [71]

$$8(\log_2 2k) \log_2 \left(\frac{\log_2 2k}{\epsilon} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon} \right)}{\epsilon} \right)$$

T gates and almost an equal number of CNOT gates.

e. Block encoding of $H_{2\vec{x}}$: The ancilla preparation sub-routine is as follows.

$$\text{PREP}_{2\vec{x}} |0\rangle^{\log_2(L_3+L_4)} = \frac{1}{\sqrt{\mathcal{N}_{2\vec{x}}}} \sum_{j=1}^{L_3+L_4} \sqrt{\alpha_j''} |j\rangle \quad (\text{C63})$$

where α_j'' are the weights obtained while expressing $H_{2\vec{x}}$ as sum of Z operators (Eq. C56). For the state preparation we require $\log_2(L_3 + L_4)$ H, $2(L_3 + L_4) + 3 \log_2(L_3 + L_4) - 7$ CNOT and $2(L_3 + L_4) - 2$ rotation gates.

The select sub-routines does the following :

$$\text{SELECT}_{2\vec{x}} |j\rangle |\psi\rangle \rightarrow |j\rangle U_j |\psi\rangle, \quad (\text{C64})$$

where U_j is the j^{th} unitary in the LCU decomposition of $H_{2\vec{x}}$ in Eq. C56. To selectively implement the unitaries we require $L_3 + L_4$ compute-uncompute pairs of $C^{\log_2(L_3+L_4)} X$ gates. Using Theorem 14 and assuming that the control qubits have been divided into two equal groups, we require

$$2(L_3 + L_4)^{\frac{1}{2}} C^{\frac{\log(L_3+L_4)}{2}} X + (L_3 + L_4) \cdot C^2 X \quad (\text{C65})$$

(compute-uncompute) pairs. These in turn can be decomposed into

$$4\sqrt{L_3 + L_4} (\log_2(L_3 + L_4) - 2) + 4(L_3 + L_4) \quad (\text{C66})$$

T gates and

$$\sqrt{L_3 + L_4} (4 \log_2(L_3 + L_4) - 6) + 5(L_3 + L_4) \quad (\text{C67})$$

CNOT gates. Additionally, we require

$$\binom{\log_2 k + 1}{3} + \binom{\log_2 k + 1}{4} = \binom{\log_2 k + 2}{4} \quad (\text{C68})$$

CZ gates.

f. Block encoding of $H_{3\vec{x}\vec{x}'}$: The ancilla preparation sub-routine is as follows.

$$\text{PREP}_{3\vec{x}\vec{x}'} |0\rangle^{\log_2 L_5} = \frac{1}{\mathcal{N}_{3\vec{x}\vec{x}'}} \sum_{j=1}^{L_5} \sqrt{\beta_j'} |j\rangle \quad (\text{C69})$$

where β_j' are the weights given in Eq. C58. For state preparation we require $\log_2 L_5$ H gates, $2L_5 + 3 \log_2 L_5 - 7$ CNOT and $2L_5 - 2$ rotation gates.

The selection sub-routine is as follows.

$$\text{SELECT}_{3\vec{x}\vec{x}'} |j\rangle |\psi\rangle \rightarrow |j\rangle U_j |\psi\rangle \quad (\text{C70})$$

where U_j is the corresponding unitary. We require L_5 compute-uncompute pairs of $C^{\log_2 L_5} X$ gates. Again we apply Theorem 14 and assume that the control qubits have been divided into two equal groups. Then we require

$$2L_5^{\frac{1}{2}} C^{\frac{\log L_5}{2}} X + L_5 \cdot C^2 X \quad (\text{C71})$$

(compute-uncompute) pairs of gates. These can be further decomposed [75, 76] into

$$4\sqrt{L_5} (\log_2 L_5 - 2) + 4L_5 \quad (\text{C72})$$

T gates and

$$\sqrt{L_5} (4 \log_2 L_5 - 6) + 5L_5 \quad (\text{C73})$$

CNOT gates. Additionally we require $(\log_2 k + 1)^2$ CZ gates.

g. Block encoding of H_{12} : We use the recursive block encoding Theorem 13. We can block encode $H_{1\vec{x}} + H_{2\vec{x}}$ using ancilla preparation sub-routine that has 1 H and 2 rotation gates. The unitary selection sub-routine adds an extra control to each unitary. For H_{12} we prepare an equal superposition of $\log_2 |\Omega|$ qubits, using $\log_2 |\Omega|$ H gates and use these to select an ancilla of each subspace. The rest of the operations are controlled on this. Thus this adds another control. We require $|\Omega|$ number of $C^{\log_2 |\Omega|} X$ compute-uncompute pairs of gates. Using Theorem 14 and assuming an equal partitioning of the control qubits into two equal groups, we can implement the multi-controlled-X gates using

$$|\Omega|^{\frac{1}{2}} C^{\frac{\log_2 |\Omega|}{2}} + |\Omega| \cdot C^{M_4} X \quad (C74)$$

(compute-uncompute) pairs, that can be further decomposed [75, 76] into

$$4\sqrt{|\Omega|} (\log_2 |\Omega| - 2) + 4|\Omega| \quad (C75)$$

T gates and

$$\sqrt{|\Omega|} (4 \log_2 |\Omega| - 6) + 5|\Omega| \quad (C76)$$

CNOT gates.

h. Block encoding of H_3 : We use Theorem 13. We prepare $\log_2 |E_D|$ qubits in equal superposition, using $\log_2 |E_D|$ H gates. We use these to select two sub-spaces. Specifically, each superimposed state selects an ancilla. From this ancilla we use two CNOTs to select an ancilla in each of the two corresponding sub-spaces. The rest of the operations are controlled on these ancillae. So, each unitary of $H_{3\vec{x}\vec{x}'}$ has 1 extra control. Using Theorem 14 and assuming an equal partitioning into two groups, for the selection we require

$$2|E_D|^{\frac{1}{2}} C^{\frac{\log_2 |E_D|}{2}} X + |E_D| \cdot C^{M_5} X \quad (C77)$$

(compute-uncompute) pairs. These, in turn can be decomposed [75, 76] into

$$4\sqrt{|E_D|} (\log_2 |E_D| - 2) + 4|E_D| \quad (C78)$$

T gates and

$$\sqrt{|E_D|} (4 \log_2 |E_D| - 6) + 5|E_D| \quad (C79)$$

CNOT gates.

i. Block encoding of H : We prepare 1 qubit in equal superposition using 1 H. The rest of the operations are controlled on this. So it adds an extra control. Thus overall, unitaries in $H_{1\vec{x}}, H_{2\vec{x}}$ have 3 extra controls and unitaries in $H_{3\vec{x}\vec{x}'}$ have 2 extra controls. Thus for each unitary in $H_{1\vec{x}}, H_{2\vec{x}}$ we require a (compute-uncompute) pair of $C^3 X$, that can be implemented with 8 T and 9 CNOT gates. Each unitary in $H_{3\vec{x}\vec{x}'}$ require a (compute-uncompute) pair of $C^2 X$, that can be implemented with 4 T and 5 CNOT gates.

Overall, we require following numbers of T gates,

$$\begin{aligned} N_t \leq & |\Omega| \left[4\sqrt{L_1 + L_2} (\log_2 (L_1 + L_2) - 2) + 12(L_1 + L_2) + 4\sqrt{L_3 + L_4} (\log_2 (L_3 + L_4) - 2) + 12(L_3 + L_4) \right] \\ & + |E_D| \left(4\sqrt{L_5} (\log_2 L_5 - 2) + 8L_5 \right) + 4\sqrt{|\Omega|} (\log_2 |\Omega| - 2) + 4|\Omega| + 4\sqrt{|E_D|} (\log_2 |E_D| - 2) + 4|E_D| \\ & + 2|\Omega| \left(8(\log_2 2k) \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right)}{\epsilon_{QFT}} \right) \right) \end{aligned} \quad (C80)$$

following number of CNOT gates,

$$\begin{aligned} N_{cx} \in & |\Omega| \left[\sqrt{L_1 + L_2} (4 \log_2 (L_1 + L_2) - 6) + 14(L_1 + L_2) + \sqrt{L_3 + L_4} (4 \log_2 (L_3 + L_4) - 6) + 14(L_3 + L_4) \right] \\ & + |E_D| \left(\sqrt{L_5} (4 \log_2 L_5 - 6) + 10L_5 \right) + \sqrt{|\Omega|} (4 \log_2 |\Omega| - 6) + 5|\Omega| + \sqrt{|E_D|} (4 \log_2 |E_D| - 6) + 5|E_D| \\ & + 2|\Omega| \left(8(\log_2 2k) \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right)}{\epsilon_{QFT}} \right) \right) \\ & + 2(2(L_1 + L_2) + L_3 + L_4 + L_5) + 3(\log_2 (L_1 + L_2) + \log_2 (L_3 + L_4) + \log_2 L_5) - 21 \end{aligned} \quad (C81)$$

following number of rotation gates,

$$N_r \leq 4(L_1 + L_2) + 2(L_3 + L_4) + 2L_5 - 4 \quad (\text{C82})$$

following number of CZ gates

$$N_{cz} \leq |\Omega| (L_1 + L_2 + L_3 + L_4) + |E_D| L_5, \quad (\text{C83})$$

and the following number of H gates,

$$N_h \leq 3 + \log_2(L_1 + L_2) + \log_2(L_3 + L_4) + \log_2 L_5 + \log_2 |E_D| + \log_2 |\Omega|, \quad (\text{C84})$$

where $L_1 = \log_2 k + 1$, $L_2 = \binom{\log_2 k + 1}{2}$, $L_3 = \binom{\log_2 k + 1}{3}$, $L_4 = \binom{\log_2 k + 1}{4}$ and $L_5 = (\log_2 k + 1)^2$. Hence we prove Theorem 22. \square

a. ℓ_1 norm of H'_{amp}

In this section we prove a bound on the ℓ_1 norm of the implemented Hamiltonian i.e. $H_{amp'}$, as decomposed in Eq C55-C60.

Proof of Lemma 23. We try to shave off most of the identity factors, as these contribute to a global phase only. Let $|\Delta\Phi| := \Delta$ and $\log_2 k := \zeta$.

$$\begin{aligned} B_1 &:= \sum_{j=0}^{\zeta} 2^j Z_j \\ B_2 &:= \sum_{j=0}^{\zeta-1} \sum_{k>j}^{\zeta} 2^{j+k} Z_j Z_k \\ A_1 &:= \left(\frac{\hat{\Phi}}{\Delta} \right)^2 - \frac{2k^2 + 1}{6} \mathbb{I} = \frac{B_1 + B_2}{2} \end{aligned}$$

Each one of the above matrices is a sum of non-identity operators. Now we express each summand as a function of the above operators and identity.

$$\left(\frac{\hat{\Phi}}{\Delta} \right)^2 = A_1 + \frac{2k^2 + 1}{6} \mathbb{I} \quad (\text{C85})$$

$$\left(\frac{\hat{\Phi}}{\Delta} \right)^4 = A_1^2 + \frac{2k^2 + 1}{3} A_1 + \left(\frac{2k^2 + 1}{6} \right)^2 \mathbb{I} + \left(\frac{2k^2 + 1}{3} \right) A_1 \quad (\text{C86})$$

$$\left(\frac{\Pi}{\Delta} \right)^2 = \mathcal{F} \left(\frac{\hat{\Phi}}{\Delta} \right)^2 \mathcal{F}^\dagger = \mathcal{F} A_1 \mathcal{F}^\dagger + \frac{2k^2 + 1}{6} \mathbb{I} \quad (\text{C87})$$

$$B_1^2 = \left(\sum_{j=0}^{\zeta} 2^j Z_j \right)^2 = \frac{4k^2 - 1}{3} \mathbb{I} + \frac{1}{2} B_2 \quad (\text{C88})$$

$$A_1^2 = \frac{1}{4} (B_1^2 + B_2^2 + 2B_1 B_2) = \frac{4k^2 - 1}{12} \mathbb{I} + \frac{1}{8} B_2 + \frac{1}{4} B_2^2 + \frac{1}{2} B_1 B_2 \quad (\text{C89})$$

Now consider the following term.

$$\begin{aligned}
& \sum_{\vec{x} \in \Omega} \sum_{\vec{x}' \in \mathcal{N}_x} \Phi(\vec{x}) \Phi(\vec{x}') \\
&= \left(\frac{\Delta}{2}\right)^2 \sum_{\vec{x}} \sum_{\vec{x}' \in \mathcal{N}_x} \left(\mathbb{I} + \sum_{j=0}^{\zeta} 2^j Z_{j,x} + \sum_{j'=0}^{\zeta} 2^{j'} Z_{j',x'} + \sum_{j,j'=0}^{\zeta} 2^{j+j'} Z_{j,x} Z_{j',x'} \right) \\
&= \left(\frac{\Delta}{2}\right)^2 \sum_{\vec{x}} \sum_{\vec{x}' \in \mathcal{N}_x} \left(\mathbb{I} + B_{1,x} + B_{1,x'} + \sum_{j,j'=0}^{\zeta} 2^{j+j'} Z_{j,x} Z_{j',x'} \right) \\
&:= \left(\frac{\Delta}{2}\right)^2 \sum_{\vec{x}} \sum_{\vec{x}' \in \mathcal{N}_x} (\mathbb{I} + B_{1,x} + B_{1,x'} + B_{3,x,x'}) \tag{C90}
\end{aligned}$$

Plugging Eq. C85-C90 in the expression for H_{amp} we obtain the following.

$$\begin{aligned}
H_{amp} &= \sum_{\vec{x}} c_{\mathbb{I}} \mathbb{I} + \sum_{\vec{x}} \left(\frac{\lambda \Delta^4}{24} A_1^2 + \frac{\Delta^2}{2} \mathcal{F} A_1 \mathcal{F}^\dagger + \left(\frac{(M^2+1)\Delta^2}{4} + \frac{\lambda \Delta^4(2k^2+1)}{144} \right) B_1 \right. \\
&\quad \left. \left(\frac{(M^2+d)\Delta^2}{4} + \frac{\lambda \Delta^4(2k^2+1)}{144} \right) B_2 \right) - \frac{\Delta^2}{2} \sum_{\vec{x}} \sum_{\vec{x}' \in \mathcal{N}_x} B_{3,x,x'} \\
&= \sum_{\vec{x}} c'_{\mathbb{I}} \mathbb{I} + \sum_{\vec{x}} \frac{\Delta^2}{2} \mathcal{F} A_1 \mathcal{F}^\dagger - \frac{\Delta^2}{2} \sum_{\vec{x}, \vec{x}'} B_{3,x,x'} + \sum_{\vec{x}} \left(\frac{(M^2+1)\Delta^2}{4} + \frac{\lambda \Delta^4(2k^2+1)}{144} \right) B_1 \\
&\quad + \sum_{\vec{x}} \frac{\lambda \Delta^4}{48} B_1 B_2 + \sum_{\vec{x}} \left(\frac{(M^2+d)\Delta^2}{4} + \frac{\lambda \Delta^4(2k^2+1)}{144} + \frac{\lambda \Delta^4}{192} \right) B_2 + \sum_{\vec{x}} \frac{\lambda \Delta^4}{96} B_2^2 \tag{C91}
\end{aligned}$$

Now we compute the norms of the following.

$$\begin{aligned}
\|A_1\| &\leq \frac{2k^2}{3} - \frac{1}{6} \\
\|B_{3,x,x'}\| &\leq (2k-1)^2 \\
\|B_1\| &\leq 2k-1 \\
\|B_2\| &\leq 2 \left(\frac{2k^2}{3} - k + \frac{1}{3} \right) \\
\|B_2^2\| &\leq 4 \left(\frac{4k^4}{9} - \frac{4k^3}{3} + \frac{13k^2}{9} - \frac{2k}{3} + \frac{1}{9} \right) \\
\|B_1 B_2\| &\leq 2 \left(\frac{4k^3}{3} - \frac{8k^2}{3} + \frac{5k}{3} - \frac{1}{3} \right)
\end{aligned}$$

Substituting the above inequalities in Eq. C91 we have the following, thus proving Lemma 23.

$$\begin{aligned}
\|H'_{amp}\| &\leq |\Omega| \left(\frac{\lambda \Delta^4}{27} k^4 + k^2 \left(\left(\frac{M^2+7d+1}{3} \right) \Delta^2 - 0.048611 \lambda \Delta^4 \right) + k(-3d\Delta^2 + 0.03125 \lambda \Delta^4) \right. \\
&\quad \left. + \Delta^2 \left(\frac{-M^2+8d-4}{6} \right) - 0.0081019 \lambda \Delta^4 \right) \tag{C92}
\end{aligned}$$

□

b. Block encoding using binary decomposition for all operators

In this section we first prove Theorem 29 in detail and then we explicitly construct some circuits for the signature matrices arising in the decomposition of Φ^2 and Φ^4 .

Proof of Theorem 29. In this case we partition H_{amp} as follows.

$$\begin{aligned}
H_{\phi^2 \vec{x}} &= \Pi^2(\vec{x}) + (M^2 + d + 1)\Phi^2(\vec{x}) \\
H_{\phi^4 \vec{x}} &= \Phi^4(\vec{x}) \\
H_{\phi \vec{x} \vec{x}'} &= \Phi(\vec{x})\Phi(\vec{x}') = H_{3\vec{x} \vec{x}'} \\
H_3 &= \sum_{\vec{x}, \vec{x}' \in E_D} H_{3\vec{x} \vec{x}'} \\
H'_{12} &= \frac{1}{2} \sum_{\vec{x} \in \Omega} H_{\phi^2 \vec{x}} + \frac{\Lambda}{4!} H_{\phi^4 \vec{x}} \\
H_{amp} &= H'_{12} + H_3
\end{aligned} \tag{C93}$$

The recursive block encoding is done in a similar fashion as before. We give brief descriptions with gate complexity.

a. Block encoding of $H_{\phi^2 \vec{x}}$: Let $\Phi^2 = \sum_{j=0}^{2 \log_2 k} U_j$, where U_j is a signature matrix. These can be obtained using Lemma 16, as described in Section IV E 4. The ancilla preparation sub-routine is,

$$\text{PREP}_{\phi^2 \vec{x}} = \frac{1}{\mathcal{N}_{\phi^2 \vec{x}}} \left(\sum_{j=0}^{2 \log_2 k} \sqrt{\alpha_j} |j, 0\rangle + \sum_{l=0}^{2 \log_2 k} \sqrt{\beta_l} |l, 1\rangle \right), \tag{C94}$$

where the weights α_j and β_l include the coefficients from LCU decomposition of operators Φ^2 and $\Pi^2 = \mathcal{F}\Phi^2\mathcal{F}^\dagger$, as well as the coefficients of Φ^2 and Π^2 in the definition of the Hamiltonian. The last qubit is used to select the QFT. We require $1 + \log_2(2 \log_2 k)$ qubits. For the state preparation we require $1 + \log_2(2 \log_2 k)$ H, $4 \log_2(k) - 2$ rotations, $4 \log_2(k) + 3 \log_2(2 \log_2 k) - 7$ CNOT gates.

The SELECT sub-routine does the following.

$$\begin{aligned}
\text{SELECT}_{\phi^2 \vec{x}} |j, 0\rangle |\psi\rangle &\mapsto |j, 0\rangle U_j |\psi\rangle \\
\text{SELECT}_{\phi^2 \vec{x}} |j, 1\rangle |\psi\rangle &\mapsto |j, 0\rangle \mathcal{F} U_j \mathcal{F}^\dagger |\psi\rangle
\end{aligned} \tag{C95}$$

As explained, we require $2 \log_2 k$ number of $C^{\log(2 \log k)} X$ gates to select the unitaries. Using Theorem 14, if we divide the control qubits into 2 equal groups then we require the following number of T gates to implement the control that selects the signature matrices

$$4\sqrt{2 \log_2 k} (\log(2 \log_2 k) - 2) + 8 \log_2 k, \tag{C96}$$

and the following number of CNOT gates

$$\sqrt{2 \log_2 k} (4 \log(2 \log_2 k) - 6) + 10 \log_2 k. \tag{C97}$$

Additionally, assuming Conjecture 28 we require $O(\log^2 k)$ number of T-gates for the implementation of the signature matrices. We also require two $\log_2 k + 1$ -qubit QFT.

b. Block encoding of $H_{\phi^4 \vec{x}}$: Let $\Phi^4 = \sum_{j=0}^{4 \log_2 k} U_j$, where U_j is a signature matrix obtained using Lemma 16, as described in Section IV E 4. The ancilla preparation sub-routine is as follows.

$$\text{PREP}_{\phi^4 \vec{x}} = \frac{1}{\mathcal{N}_{\phi^4 \vec{x}}} \sum_{j=0}^{4 \log_2 k} \sqrt{\alpha_j} |j\rangle \tag{C98}$$

where the weights α_j include the coefficients from LCU decomposition of operators Φ^4 . We require $1 + \log_2(4 \log_2 k)$ qubits. For the state preparation we require $1 + \log_2(4 \log_2 k)$ H, $8 \log_2(k) - 2$ rotations, $8 \log_2(k) + 3 \log_2(4 \log_2 k) - 7$ CNOT gates.

The SELECT sub-routine does the following.

$$\text{SELECT}_{\phi^4 \vec{x}} |j, 0\rangle |\psi\rangle \mapsto |j\rangle U_j |\psi\rangle \tag{C99}$$

As explained, we require $4 \log_2 k$ number of $C^{\log(4 \log k)} X$ gates to select the unitaries. Using Theorem 14, if we divide the control qubits into 2 equal groups then we require the following number of T gates

$$4\sqrt{4 \log_2 k} (\log(4 \log_2 k) - 2) + 16 \log_2 k, \tag{C100}$$

and the following number of CNOT gates

$$\sqrt{4\log_2 k}(4\log(4\log_2 k) - 6) + 20\log_2 k. \quad (\text{C101})$$

Assuming Conjecture 28 we require $O(\log^2 k)$ T-gates to implement the signature matrices.

c. Block encoding of H'_{12} : We use the recursive block encoding Theorem 13. We can block encode $H_{\phi^2\vec{x}} + \frac{\Lambda}{24}H_{\phi^4\vec{x}}$ using ancilla preparation sub-routine that has 1 H and 2 rotation gates. For H'_{12} we prepare an equal superposition of $\log_2 |\Omega|$ qubits, using $\log_2 |\Omega|$ H gates and use these to select an ancilla of each subspace. The rest of the operations are controlled on this. Thus this adds another control. We require $|\Omega|$ number of $C^{\log_2 |\Omega|} X$ compute-uncompute pairs of gates. Using Theorem 14 and assuming an equal partitioning of the control qubits into two equal groups, we can implement the multi-controlled-X gates using

$$|\Omega|^{\frac{1}{2}} C^{\frac{\log |\Omega|}{2}} X + |\Omega| \cdot C^2 X \quad (\text{C102})$$

(compute-uncompute) pairs, that can be further decomposed [75, 76] into

$$4\sqrt{|\Omega|}(\log_2 |\Omega| - 2) + 4|\Omega| \quad (\text{C103})$$

T gates and

$$\sqrt{|\Omega|}(4\log_2 |\Omega| - 6) + 5|\Omega| \quad (\text{C104})$$

CNOT gates.

d. Block encoding of $H_{3\vec{x}\vec{x}}$ and H_3 : The block encoding of $\Phi(\vec{x})\Phi(\vec{x}') = H_{3\vec{x}\vec{x}}$ is provided earlier. For the preparation sub-routine we require $\log_2 |E_D| + \log_2(\log_2 k + 1)^2$ H, $2(\log_2 k + 1)^2 - 2$ rotation gates, $2(\log_2 k + 1)^2 + 3\log_2(\log_2 k + 1)^2 - 7$ CNOT gates. For the selection sub-routine we require

$$\begin{aligned} & |E_D|(8(\log_2 k + 1)(\log_2(\log_2 k + 1) - 1) + 4(\log_2 k + 1)^2) \\ & + 4\sqrt{|E_D|}(\log_2 |E_D| - 2) + 4|E_D| \end{aligned} \quad (\text{C105})$$

T gates and

$$\begin{aligned} & |E_D|((\log_2 k + 1)(8\log_2(\log_2 k + 1) - 6) + 5(\log_2 k + 1)^2) \\ & + \sqrt{|E_D|}(4\log_2 |E_D| - 6) + 5|E_D| \end{aligned} \quad (\text{C106})$$

CNOT gates and $(\log_2 k + 1)^2$ CZ gates.

e. Block encoding of H_{amp} : Using Theorem 13 we can block encode H_{amp} using a PREP sub-routine with 1 H and 2 rotation gates and a SELECT sub-routine that adds an extra control. Thus under the assumptions of Conjecture 28 we require

$$\begin{aligned} N'_t \leq & |\Omega| \left(4\sqrt{2\log_2 k}(\log_2(2\log_2 k) - 2) + 8\log_2 k + 4\sqrt{4\log_2 k}(\log_2(4\log_2 k) - 2) + 16\log_2 k \right) \\ & + 4\sqrt{|\Omega|}(\log_2 |\Omega| - 2) + 4|\Omega| + |E_D| \left(8(\log_2 k + 1)(\log_2(\log_2 k + 1) - 1) + 4(\log_2 k + 1)^2 + 4 \right) \\ & + 4\sqrt{|E_D|}(\log_2 |E_D| - 2) \\ & + 2|\Omega| \left(8(\log_2 2k) \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right)}{\epsilon_{QFT}} \right) \right) + O(|\Omega| \log_2^2 k) \end{aligned} \quad (\text{C107})$$

T gates,

$$\begin{aligned} N'_{cx} \leq & |\Omega| \left(\sqrt{2\log_2 k}(4\log_2(2\log_2 k) - 6) + 10\log_2 k + \sqrt{4\log_2 k}(4\log_2(4\log_2 k) - 6) + 20\log_2 k + 5 \right) \\ & + \sqrt{|\Omega|}(4\log_2 |\Omega| - 6) + |E_D| \left((\log_2 k + 1)(8\log_2(\log_2 k + 1) - 6) + 5(\log_2 k + 1)^2 + 5 \right) + \sqrt{|E_D|}(4\log_2 |E_D| - 6) \\ & + 2|\Omega| \left(8(\log_2 2k) \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) + \log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right) \log_2 \left(\frac{\log_2 \left(\frac{\log_2 2k}{\epsilon_{QFT}} \right)}{\epsilon_{QFT}} \right) \right) \end{aligned} \quad (\text{C108})$$

CNOT gates,

$$N_r \leq 12\log_2 k + 2(\log_2 k + 1)^2 - 3 \quad (\text{C109})$$

rotation gates. This proves Theorem 29 \square

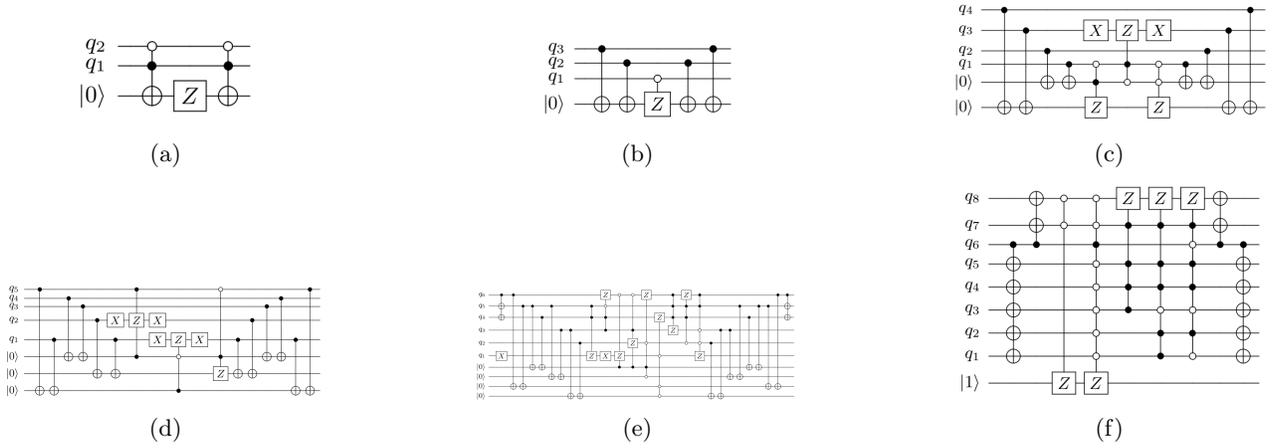


FIG. 17: Quantum circuits for (a) U_3 , (b) U_4 , (c) U_5 , (d) U_6 , (e) U_7 and (f) U_{14} . Qubits not affected by gate operations have not been shown.

c. *Quantum circuits for signature matrices in the decomposition of Φ^2*

In this section we describe the explicit constructions of signature matrices appearing in the LCU decomposition of Φ^2 , as discussed in Section IV E 4. We recall that

$$\Phi^2 = \text{diag}((k-1)^2, (k-2)^2, \dots, 1, 0, 1, \dots, (k-1)^2, k^2) = +c_0\mathbb{I} + \sum_{\ell=1}^{\zeta'} c_\ell U_\ell, \quad (\text{C110})$$

where c_0, c_ℓ are real coefficients and U_ℓ is a signature matrix obtained from the ℓ^{th} bit in the binary decomposition of $0, 1, \dots, k^2$ and then replacing the 0 with 1 and 1 with -1, as explained earlier. $\zeta' = 1 + 2 \log k$ is the maximum number of bits in the binary expansion of k^2 . We assume $k = 2^{k'}$ for some integer k' since we have a unitary. As mentioned earlier, in Table IV we have listed all integers n such that $n \leq 2^7 = 128$ and the b_ℓ^{th} ($1 \leq \ell \leq 15$) bit in the binary decomposition of n^2 has 1. In Table VI we enumerate all integers $n \leq 2^x$ (where $x = 2, \dots, 12$) such that n^2 has a binary decomposition with 1 at a particular bit position.

We consider the case where $k = 2^7$. So our circuits consist of 8 qubits and we label them as q_8, \dots, q_1 , corresponding to the 8 bits - b_8, \dots, b_1 , in the binary decomposition of any number with value at most 2^7 . b_8 and b_1 are the most and the least significant bits, respectively. In Fact 26 we have mentioned that U_1 can be implemented with a single Z gate on qubit q_1 and U_2 is an all 0 matrix.

$\ell = 3$: From Lemma 24 we know that -1 occurs whenever $n \in \{8k+4, 8k+5, 8k+6, 8k+7\}$ and it can be square of an integer if it is of the form $8k+4$ (Eq. 154). Now, $8k+4 = 4(2k+1) = 2^2(2k+1)$ and so it is a square of the form $(2(2k''+1))^2$, where k'' is an integer. If $(2(2k''+1))^2 = j^2$ then it can occur at row $(k-1)-j$ and $k-1+j$. If we expand all integers of the form $2(2k''+1)$ we observe that they have a trailing 10. Adding or subtracting $k-1$ changes these two bits to 01. Thus all such integers appear in those rows whose index has a trailing 01. This implies whenever the state of the first qubit is 1 and second qubit is 0, a -1 phase is incurred. Thus we can implement U_3 with a (compute-uncompute) pair of Toffoli, X, and a Z on an extra ancilla, as shown in Fig. 17a. To implement controlled- U_3 we add one control to Z, which does not contribute to additional T gates.

$\ell = 4$: From Lemma 24 we know that -1 occurs whenever $n \in \{16k+8, 16k+9, \dots, 16k+15\}$ and it can be square of an integer if it is of the form $16k+9$ (Equation 154). The integers whose squares are of this form have trailing 011 or 101, for example $3^3, 5^2, 11^2, 13^2$, etc. So they occur at positions with trailing 100 or 010. U_4 can be implemented with 4 CNOT and 1 CZ, as shown in Fig. 17b. To implement controlled- U_4 we add an extra control to CZ, that contributes to T gates.

$\ell = 5$: Integers, the binary decomposition of whose squares have 1 in the 5^{th} position are of the form $16n+m$, where n is a non-negative integer and $m \in \{4, 5, 7, 9, 11, 12\}$ (Table IV) and in binary $m \in \{0100, 0101, 0111, 1001, 1100, 1011\}$. The trailing bits in the binary decomposition of integers of the form $16n+m$ are in $\{0100, 0101, 0111, 1001, 1100, 1011\}$ and so in U_5 they appear at positions whose binary decomposition have the following trailing bits - 1011, 1010, 1000, 0110, 0011, 0100. So we can implement U_5 with the circuit shown in Fig. 17c, consisting of 3 double-controlled-Z and 4 pairs of CNOT. To implement controlled- U_5 we add extra controls to the CZs.

$\ell = 6$: In Table IV we have listed the integers such that the binary decomposition of their squares have 1 in the 6th position. These are of the form $32n + m$ where n is a non-negative integer and

$$m \in S_m = \{6, 7, 10, 11, 13, 15, 17, 19, 21, 22, 25, 26\}.$$

In binary the integers in S_m are as follows.

$$S_m \in \{00110, 00111, 01010, 01011, 01101, 01111, 10001, 10011, 10101, 10110, 11001, 11010\}$$

We obtain the row positions by adding and subtracting $11\dots111$, which implies that the trailing 5 bits get complemented. Thus U_6 can be implemented with the circuit shown in Fig. 17d, consisting of 6 (compute-uncompute) pairs of CNOT and 3 2-qubit-controlled-Z. To implement controlled- U_6 we need to add one control to each multi-controlled Z.

$\ell = 7$: In Table IV we have listed the integers such that the binary decomposition of their squares have 1 in the 7th position. These are of the form $64n + m$ where n is a non-negative integer and $m \in S_m$, a set specified in Table IV. As explained, the gates in this unitary act on 6 qubits, corresponding to the last significant 6 bits in the binary representation of an integer. We calculate the binary representation of integers in S_m . We obtain the row positions with -1 by adding and subtracting $11\dots111$. We group appropriate binary strings in order to induce "Don't care" conditions, in order to optimize the circuit. Thus we obtain the circuit in Fig. 17e, that implements U_7 .

$\ell = 14$: From Table IV we observe that squares of integers from 91 (01011011) to 127 (01111111) are such that their binary expansions have 1 in the 14th bit position. These integers appear at rows 0-36 and 218-254. The binary encoding of these rows range from 00000000 (0) - 00100100 (36) and 11011010 (218) - 11111110 (254). Thus when the state of the qubits are within the stated range the circuit incurs a phase of -1. So U_{14} can be implemented with the circuit shown in Fig. 17f. Again, we have optimized the circuit by identifying "Don't care" conditions among groups of binary strings.

d. Quantum circuits for signature matrices in the decomposition of ϕ^4

In this section we describe the explicit constructions of signature matrices appearing in the LCU decomposition of Φ^4 , as discussed in Section IVE 4. We recall that

$$\Phi^4 = \text{diag}((k-1)^4, (k-2)^4, \dots, 1, 0, 1, \dots, (k-1)^4, k^4) = +c'_0\mathbb{I} + \sum_{\ell=1}^{\zeta''} c'_\ell U'_\ell, \quad (\text{C111})$$

where c'_0, c'_ℓ are real coefficients and U'_ℓ is a signature matrix obtained from the ℓ^{th} bit in the binary decomposition of $0, 1, \dots, k^4$ and then replacing the 0 with 1 and 1 with -1, as explained earlier. $\zeta'' = 1 + 4 \log k$ is the maximum number of bits in the binary expansion of k^4 . As mentioned earlier, in Table V we have listed all integers n such that $n \leq 2^7 = 128$ and the b_ℓ^{th} ($1 \leq \ell \leq 29$) bit in the binary decomposition of n^4 has 1. In Table VII we enumerate all integers $n \leq 2^x$ (where $x = 2, \dots, 12$) such that n^4 has a binary decomposition with 1 at a particular bit position. As in previous section, we consider the case where $k = 2^7$. So our circuits consist of 8 qubits and we label them as q_8, \dots, q_1 , corresponding to the 8 bits - b_8, \dots, b_1 , in the binary decomposition of any number with value at most 2^7 . b_8 and b_1 are the most and the least significant bits, respectively. In Fact 26 we have mentioned that U_1 can be implemented with a single Z gate on qubit q_1 and U_2, U_3, U_4 are all 0 matrix.

$\ell = 5$: In Table V we have enlisted all the integers such that the binary decomposition of their fourth power has 1 in the 5th bit position. These are of the form $8n + 2, 8n + 3, 8n + 5, 8n + 6$, where n is a non-negative integer. The binary decomposition of 2, 3, 5 and 6 are 010, 011, 101 and 110, respectively. So the trailing bits in the binary decomposition of integers of the above form are either 010, 011, 101 or 110. These appear in rows whose binary decomposition has any of the four trailing bits - 101, 100, 010, 001. Therefore, U'_5 can be implemented with the circuit shown in Figure 18a, consisting of 4 (compute-uncompute) pairs of CNOT and 2 CZ. Controlled- U'_5 can be implemented by adding extra controls to each CZ.

$\ell = 6$: Consider integers of the form $16n + m$, where n is a non-negative integer and

$$m \in S_m = \{5(0101), 7(0111), 9(1001), 11(1011)\}.$$

From Table V, we observe that the binary decomposition of the fourth power of these integers has 1 in the 6th bit position. In Φ^4 these integers appear at rows such that the binary decomposition of their index has trailing 4 bits which are complements of those in S_m . Thus, U'_6 can be implemented with the circuit in Figure 18b, consisting of 2

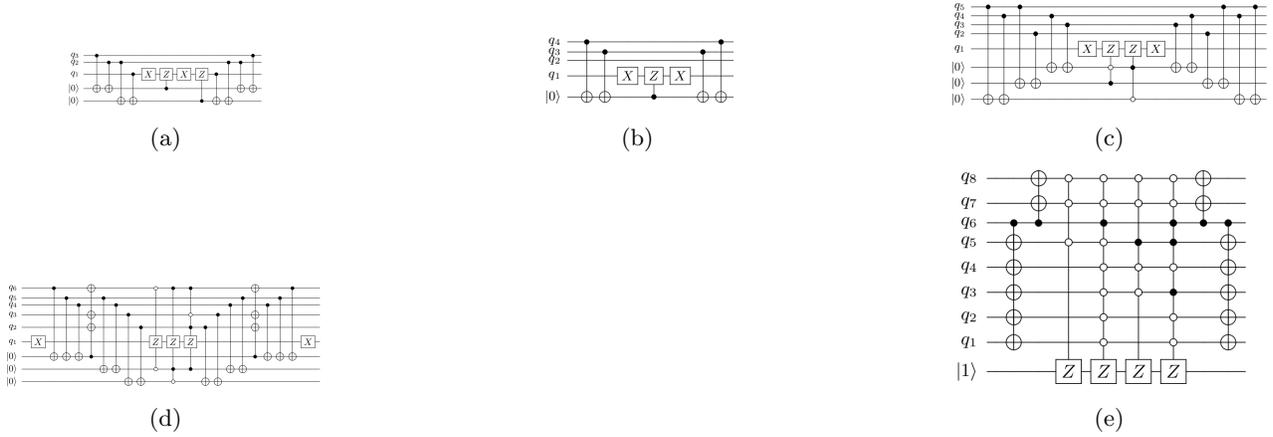


FIG. 18: Quantum circuits for (a) U'_5 , (b) U'_6 , (c) U'_7 , (d) U'_8 and (e) U'_{28} . Qubits not affected by gate operations have not been shown.

(compute-uncompute) pairs of CNOTs, and 1 CZ. Controlled- U'_6 can be implemented by adding an extra control to CZ.

$\ell = 7$: Consider integers of the form $32n + m$, where n is a non-negative integer and

$$m \in S_m = \{3(00011), 5(00101), 7(00111), 15(01111), 17(10001), 25(11001), 27(11011), 29(11101)\}.$$

From Table V we can say that the binary decomposition of the fourth power of these integers has 1 in the 7^{th} bit position. These integers appear at rows whose binary decomposition has the last 5 bits that are complements of the ones in S_m . So, U'_7 can be implemented with the circuit in Figure 18c, consisting of 6 (compute-uncompute) pairs of CNOT and two double-controlled-Z. Controlled- U'_7 can be implemented by adding extra control to the controlled-Z.

$\ell = 8$: In Table V we have listed the integers such that the binary decomposition of their fourth power have 1 in the 8^{th} position. These are of the form $64n + m$ where n is a non-negative integer and $m \in S_m$, a set specified in Table V. As explained, the gates in this unitary act on 6 qubits, corresponding to the last significant 6 bits in the binary representation of an integer. We calculate the binary representation of integers in S_m . We obtain the row positions with -1 by adding and subtracting $11\dots 111$. We group appropriate binary strings in order to induce "Don't care" conditions, in order to optimize the circuit. Thus we obtain the circuit in Fig. 18d, that implements U'_8 .

$\ell = 28$: From Table V we observe that the fourth power of integers from 108 to 127 are such that their binary expansions have 1 in the 14^{th} bit position. These integers appear at rows 0-19 and 235-254. The binary encoding of these rows range from 00000000 (0) - 00010011 (19) and 11101011 (218) - 11111110 (254). Thus when the state of the qubits are within the stated range the circuit incurs a phase of -1. So U'_{28} can be implemented with the circuit shown in Figure 18e. Again, we have optimized the circuit significantly by identifying appropriate "Don't care" conditions.

-
- [1] Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996.
 - [2] Dominic W Berry, Graeme Ahokas, Richard Cleve, and Barry C Sanders. Efficient quantum algorithms for simulating sparse Hamiltonians. *Communications in Mathematical Physics*, 270:359–371, 2007.
 - [3] Andrew M Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Information and Computation*, 12(11&12):901–924, 2012.
 - [4] Andrew M Childs, Robin Kothari, and Rolando D Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6):1920–1950, 2017.
 - [5] Guang Hao Low and Isaac L. Chuang. Hamiltonian Simulation by Qubitization. *Quantum*, 3:163, July 2019.
 - [6] Yuan Su, Dominic W Berry, Nathan Wiebe, Nicholas Rubin, and Ryan Babbush. Fault-tolerant quantum simulations of chemistry in first quantization. *PRX Quantum*, 2(4):040332, 2021.
 - [7] Christian W. Bauer, Zohreh Davoudi, A. Baha Balantekin, Tanmoy Bhattacharya, Marcela Carena, Wibe A. de Jong, Patrick Draper, Aida El-Khadra, Nate Gemelke, Masanori Hanada, Dmitri Kharzeev, Henry Lamm, Ying-Ying Li, Junyu Liu, Mikhail Lukin, Yannick Meurice, Christopher Monroe, Benjamin Nachman, Guido Pagano, John Preskill, Enrico Rinaldi, Alessandro Roggero, David I. Santiago, Martin J. Savage, Irfan Siddiqi, George Siopsis, David Van Zanten, Nathan Wiebe, Yukari Yamauchi, Kübra Yeter-Aydeniz, and Silvia Zorzetti. Quantum Simulation for High Energy Physics. *arXiv*, April 2022.

Bit	Integers	#Integers
b_1	$2n + 1 : 0 \leq n \leq 63$	64
b_2	\emptyset	0
b_3	$2(2n + 1) : 0 \leq 31$	32
b_4	$8n + 3, 8n + 5 : 0 \leq 15$	32
b_5	$16n + m : 0 \leq n \leq 7, m \in \{4, 5, 7, 9, 11, 12\}$	48
b_6	$32n + m : 0 \leq n \leq 3, m \in \{6, 7, 10, 11, 13, 15, 17, 19, 21, 22, 25, 26\}$	48
b_7	$64n + m : n = 0, 1, m \in \{8, 9, 10, 11, 14, 15, 18, 19, 22, 24, 25, 27, 29, 31, 33, 35, 37, 39, 40, 42, 45, 46, 49, 50, 53, 54, 55, 56\}$	56
b_8	12, 13, 14, 15, 20, 21, 22, 26, 27, 30, 31, 34, 35, 38, 39, 41, 42, 44, 45, 47, 50, 52, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 76, 78, 81, 83, 84, 86, 87, 89, 90, 93, 94, 97, 98, 101, 102, 106, 107, 108, 113, 114, 115, 116	56
b_9	16, 17, 18, 19, 20, 21, 22, 28, 29, 30, 31, 36, 37, 38, 39, 43, 44, 45, 48, 49, 50, 54, 55, 58, 59, 62, 63, 66, 67, 70, 71, 74, 75, 77, 78, 80, 81, 84, 87, 90, 92, 93, 95, 98, 100, 101, 103, 105, 106, 108, 110, 112, 113, 115, 117, 119, 121, 123, 125, 127	60
b_{10}	23, 24, 25, 26, 27, 28, 29, 30, 31, 40, 41, 42, 43, 44, 45, 51, 52, 53, 54, 55, 60, 61, 62, 63, 68, 69, 70, 71, 76, 77, 78, 82, 83, 84, 88, 89, 90, 94, 95, 99, 100, 101, 104, 105, 106, 109, 110, 114, 115, 118, 119, 122, 123, 126, 127	55
b_{11}	32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 56, 57, 58, 59, 60, 61, 62, 63, 72, 73, 74, 75, 76, 77, 78, 85, 86, 87, 88, 89, 90, 96, 97, 98, 99, 100, 101, 107, 108, 109, 110, 116, 117, 118, 119, 124, 125, 126, 127	53
b_{12}	46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 102, 103, 104, 105, 106, 107, 108, 109, 110, 120, 121, 122, 123, 124, 125, 126, 127	47
b_{13}	64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127	44
b_{14}	91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127	37
b_{15}	128	1

TABLE IV: The leftmost column stores bit positions, indicated by b_i . The second column stores integers such that the binary decomposition of their square has 1 in the b_i^{th} bit position. The third column stores the number of integers whose square has 1 in the b_i^{th} bit position. We have listed integers until 128.

- [8] John Kogut and Leonard Susskind. Hamiltonian formulation of Wilson’s lattice gauge theories. *Phys. Rev. D*, 11:395–408, Jan 1975.
- [9] Sidney D. Drell, Helen R. Quinn, Benjamin Svetitsky, and Marvin Weinstein. Quantum electrodynamics on a lattice: A Hamiltonian variational approach to the physics of the weak-coupling region. *Phys. Rev. D*, 19:619–638, Jan 1979.
- [10] Stephen P. Jordan, Hari Krovi, Keith S. M. Lee, and John Preskill. BQP-completeness of scattering in scalar quantum field theory. *Quantum*, 2:44, January 2018.
- [11] Natalie Klco and Martin J. Savage. Digitization of scalar fields for quantum computing. *Physical Review A*, 99(5):052335, May 2019.
- [12] Yuri Alexeev, Dave Bacon, Kenneth R Brown, Robert Calderbank, Lincoln D Carr, Frederic T Chong, Brian DeMarco, Dirk Englund, Edward Farhi, Bill Fefferman, et al. Quantum computer systems for scientific discovery. *PRX quantum*, 2(1):017001, 2021.
- [13] Christian W Bauer, Zohreh Davoudi, A Baha Balantekin, Tanmoy Bhattacharya, Marcela Carena, Wibe A De Jong, Patrick Draper, Aida El-Khadra, Nate Gemelke, Masanori Hanada, et al. Quantum simulation for high-energy physics. *PRX quantum*, 4(2):027001, 2023.
- [14] Nhung H Nguyen, Minh C Tran, Yingyue Zhu, Alaina M Green, C Huerta Alderete, Zohreh Davoudi, and Norbert M Linke. Digital quantum simulation of the Schwinger model and symmetry protection with trapped ions. *PRX Quantum*, 3(2):020324, 2022.
- [15] Alexander F Shaw, Pavel Lougovski, Jesse R Stryker, and Nathan Wiebe. Quantum algorithms for simulating the lattice Schwinger model. *Quantum*, 4:306, 2020.
- [16] Stephen P. Jordan, Keith S. M. Lee, and John Preskill. Quantum algorithms for quantum field theories. *Science*, 336(6085):1130–1133, 2012.
- [17] Stephen P. Jordan, Keith S. M. Lee, and John Preskill. Quantum Algorithms for Fermionic Quantum Field Theories. *arXiv:1404.7115 [hep-th, physics:quant-ph]*, April 2014.
- [18] Panayotis G. Kevrekidis and Jesús Cuevas-Maraver. *A dynamical perspective on the Φ^4 model: Past, present and future*. Springer, 2019.
- [19] Richard P. Feynman. *Photon-hadron Interactions*. Taylor and Francis, 5 2019.
- [20] J. D. Bjorken and Emmanuel A. Paschos. Inelastic Electron Proton and gamma Proton Scattering, and the Structure of the Nucleon. *Phys. Rev.*, 185:1975–1982, 1969.
- [21] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 193–204, 2019.

Bit	Integers	#Integers
b_1	$2n + 1 : 0 \leq n \leq 63$	64
b_2	\emptyset	0
b_3	\emptyset	0
b_4	\emptyset	0
b_5	$8n + 2, 8n + 3, 8n + 5, 8n + 6 : 0 \leq n \leq 15$	64
b_6	$16n + m : 0 \leq n \leq 7, m \in \{5, 7, 9, 11\}$	32
b_7	$32n + m : 0 \leq n \leq 3, m \in \{3, 5, 7, 15, 17, 25, 27, 29\}$	32
b_8	$64n + m : n = 0, 1, m \in \{9, 13, 15, 21, 25, 27, 29, 31, 33, 35, 37, 39, 43, 49, 51, 55\}$	32
b_9	4, 6, 7, 9, 10, 11, 12, 13, 15, 19, 20, 21, 22, 23, 25, 26, 27, 28, 31, 35, 36, 38, 42, 44, 47, 52, 54, 58, 59, 60, 61, 63, 65, 67, 68, 69, 70, 74, 76, 81, 84, 86, 90, 92, 93, 97, 100, 101, 102, 103, 105, 106, 107, 108, 109, 113, 115, 116, 117, 118, 119, 121, 122, 124	64
b_{10}	5, 10, 13, 14, 17, 18, 21, 22, 27, 29, 31, 41, 42, 43, 45, 46, 49, 50, 51, 53, 54, 57, 63, 67, 69, 73, 74, 78, 81, 82, 86, 89, 91, 93, 95, 103, 105, 106, 109, 110, 113, 114, 117, 118, 119, 121, 125, 127	48
b_{11}	6, 10, 13, 14, 15, 17, 19, 21, 23, 25, 30, 31, 34, 35, 39, 41, 47, 49, 50, 53, 54, 58, 59, 61, 63, 69, 70, 74, 75, 77, 78, 79, 81, 85, 87, 89, 91, 93, 94, 95, 98, 101, 105, 107, 109, 111, 113, 114, 115, 118, 119, 122, 127	53
b_{12}	7, 9, 11, 13, 18, 19, 26, 27, 29, 30, 33, 37, 39, 41, 42, 43, 50, 51, 54, 58, 62, 63, 66, 67, 69, 70, 74, 75, 78, 83, 86, 87, 89, 91, 93, 97, 98, 102, 107, 110, 121, 123, 125, 127	44
b_{13}	8, 9, 11, 12, 14, 18, 19, 20, 21, 22, 24, 25, 26, 27, 30, 31, 33, 37, 38, 40, 41, 42, 44, 45, 46, 47, 49, 50, 51, 52, 54, 56, 57, 62, 63, 67, 69, 70, 72, 73, 76, 79, 81, 84, 88, 89, 91, 94, 95, 97, 101, 104, 105, 107, 108, 116, 117, 118, 120, 121, 122, 126, 127	63
b_{14}	10, 11, 13, 19, 20, 21, 25, 26, 28, 34, 35, 36, 42, 43, 44, 47, 49, 51, 53, 54, 55, 58, 59, 62, 65, 67, 77, 82, 83, 84, 86, 87, 90, 92, 93, 98, 100, 102, 103, 105, 106, 108, 109, 111, 113, 114, 119, 126, 127	49
b_{15}	12, 13, 15, 17, 19, 20, 21, 23, 25, 26, 28, 29, 30, 34, 35, 38, 39, 42, 46, 47, 49, 50, 53, 59, 60, 61, 62, 63, 65, 67, 68, 69, 70, 71, 73, 75, 77, 78, 79, 81, 82, 89, 91, 93, 94, 95, 97, 98, 99, 100, 101, 103, 106, 108, 109, 111, 113, 115, 116, 117, 118, 119, 121, 122, 125, 126, 127	67
b_{16}	14, 15, 18, 19, 21, 22, 25, 26, 29, 35, 36, 37, 38, 45, 49, 52, 54, 55, 58, 59, 60, 66, 69, 71, 74, 75, 78, 81, 82, 84, 85, 86, 95, 97, 99, 100, 101, 102, 105, 108, 109, 113, 115, 116, 119, 121, 123, 124, 126	49
b_{17}	16, 17, 18, 19, 22, 24, 25, 28, 36, 38, 39, 40, 41, 42, 44, 48, 49, 50, 51, 52, 54, 55, 57, 60, 61, 62, 66, 67, 69, 71, 73, 74, 76, 80, 82, 84, 88, 89, 90, 92, 93, 94, 98, 99, 100, 101, 102, 103, 104, 108, 109, 112, 113, 114, 117, 119, 124, 125, 126, 127	60
b_{18}	20, 21, 22, 26, 29, 31, 33, 35, 38, 39, 40, 41, 42, 45, 47, 49, 50, 51, 52, 55, 56, 61, 67, 68, 70, 71, 72, 75, 79, 84, 86, 87, 88, 91, 94, 95, 97, 98, 101, 102, 105, 106, 108, 110, 113, 116, 117, 118, 119, 121, 124	51
b_{19}	23, 24, 25, 26, 30, 31, 34, 35, 37, 38, 40, 42, 43, 45, 46, 49, 50, 51, 52, 56, 58, 60, 68, 70, 76, 78, 83, 84, 85, 89, 91, 92, 93, 94, 97, 98, 100, 103, 105, 106, 111, 113, 115, 118, 120, 121, 122, 123, 124, 125, 126	51
b_{20}	27, 28, 29, 30, 31, 36, 37, 38, 41, 42, 44, 45, 47, 50, 52, 53, 55, 58, 59, 69, 70, 72, 74, 76, 77, 85, 87, 89, 90, 95, 98, 99, 104, 105, 108, 109, 110, 111, 115, 116, 117, 118, 120, 125	44
b_{21}	32, 33, 34, 35, 36, 37, 38, 43, 44, 45, 48, 49, 50, 53, 56, 59, 61, 63, 65, 67, 69, 72, 73, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 88, 89, 91, 93, 95, 96, 98, 99, 100, 101, 102, 103, 104, 105, 107, 108, 110, 113, 114, 119, 120, 122, 124	56
b_{22}	39, 40, 41, 42, 43, 44, 45, 51, 52, 53, 57, 58, 59, 62, 63, 66, 67, 70, 73, 75, 76, 78, 80, 82, 84, 87, 89, 90, 93, 94, 98, 99, 100, 101, 102, 103, 104, 105, 109, 110, 112, 113, 115, 117, 119, 122, 123	47
b_{23}	46, 47, 48, 49, 50, 51, 52, 53, 60, 61, 62, 63, 68, 69, 70, 74, 75, 76, 79, 80, 83, 84, 86, 87, 90, 92, 93, 95, 97, 98, 100, 102, 104, 107, 109, 112, 115, 116, 119, 120, 121	41
b_{24}	54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 71, 72, 73, 74, 75, 76, 81, 82, 83, 84, 88, 89, 90, 94, 95, 99, 100, 103, 104, 106, 107, 110, 113, 116, 118, 119, 121, 123, 125, 127	40
b_{25}	64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 85, 86, 87, 88, 89, 90, 96, 97, 98, 99, 100, 105, 106, 107, 111, 112, 113, 117, 118, 119, 122, 123, 126, 127	37
b_{26}	77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 101, 102, 103, 104, 105, 106, 107, 114, 115, 116, 117, 118, 119, 124, 125, 126, 127	31
b_{27}	91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 120, 121, 122, 123, 124, 125, 126, 127	25
b_{28}	108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127	20
b_{29}	128	1

TABLE V: The leftmost column stores bit positions, indicated by b_i . The second column stores integers such that the binary decomposition of their fourth power has 1 in the b_i^{th} bit position. The third column stores the number of integers whose fourth power has 1 in the b_i^{th} bit position. We have listed integers until 128.

	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}	2^{11}	2^{12}
\mathbf{b}_1	2	4	8	16	32	64	128	256	512	1024	2048
\mathbf{b}_2	0	0	0	0	0	0	0	0	0	0	0
\mathbf{b}_3	1	2	4	8	16	32	64	128	256	512	1024
\mathbf{b}_4	1	2	4	8	16	32	64	128	256	512	1024
\mathbf{b}_5	1	3	6	12	24	48	96	192	384	768	1536
\mathbf{b}_6	0	2	6	12	24	48	96	192	384	768	1536
\mathbf{b}_7	0	1	6	14	28	56	112	224	448	896	1792
\mathbf{b}_8	0	0	4	11	28	56	112	224	448	896	1792
\mathbf{b}_9	0	0	1	11	27	60	120	240	480	960	1920
\mathbf{b}_{10}	0	0	0	9	24	55	120	240	480	960	1920
\mathbf{b}_{11}	0	0	0	1	22	53	121	248	496	992	1984
\mathbf{b}_{12}	0	0	0	0	18	47	108	233	496	992	1984
\mathbf{b}_{13}	0	0	0	0	1	44	106	236	493	1008	2016
\mathbf{b}_{14}	0	0	0	0	0	37	97	221	478	995	2016
\mathbf{b}_{15}	0	0	0	0	0	1	88	212	469	983	2017
\mathbf{b}_{16}	0	0	0	0	0	0	74	195	446	957	1990
\mathbf{b}_{17}	0	0	0	0	0	0	1	175	422	932	1952
\mathbf{b}_{18}	0	0	0	0	0	0	0	149	389	891	1915
\mathbf{b}_{19}	0	0	0	0	0	0	0	1	350	847	1864
\mathbf{b}_{20}	0	0	0	0	0	0	0	0	299	779	1786
\mathbf{b}_{21}	0	0	0	0	0	0	0	0	1	699	1692
\mathbf{b}_{22}	0	0	0	0	0	0	0	0	0	599	1560
\mathbf{b}_{23}	0	0	0	0	0	0	0	0	0	1	1397
\mathbf{b}_{24}	0	0	0	0	0	0	0	0	0	0	1199
\mathbf{b}_{25}	0	0	0	0	0	0	0	0	0	0	1

TABLE VI: Number of integers less than 2^x , $x = 2, 3, \dots, 12$, such that the binary decomposition of their square has 1 at a certain bit position. That is, the cell at the intersection of row labeled by bit b_i and column labeled by integer 2^x stores the number of integers $n \leq 2^x$ such that the binary decomposition of n^2 has 1 at bit position b_i .

- [22] João Barata, Niklas Mueller, Andrey Tarasov, and Raju Venugopalan. Single-particle digitization strategy for quantum computation of a ϕ^4 scalar field theory. *Physical Review A*, 103(4):042410, 2021.
- [23] Zohreh Davoudi, Alexander F. Shaw, and Jesse R. Stryker. General quantum algorithms for Hamiltonian simulation with applications to a non-Abelian lattice gauge theory. *Quantum*, 7:1213, December 2023.
- [24] Mason Rhodes, Michael Kreshchuk, and Shivesh Pathak. Exponential improvements in the simulation of lattice gauge theories using near-optimal techniques, 2024.
- [25] Siddharth Hariprakash, Neel S. Modi, Michael Kreshchuk, Christopher F. Kane, and Christian W Bauer. Strategies for simulating time evolution of Hamiltonian lattice field theories. *arXiv*, 2024.
- [26] Priyanka Mukhopadhyay, Nathan Wiebe, and Hong Tao Zhang. Synthesizing efficient circuits for Hamiltonian simulation. *npj Quantum Information*, 9(31), 2023.
- [27] Priyanka Mukhopadhyay, Torin F Stetina, and Nathan Wiebe. Quantum simulation of the first-quantized Pauli-Fierz Hamiltonian. *PRX Quantum*, 5:010345, 2024.
- [28] Markus Reiher, Nathan Wiebe, Krysta M Svore, Dave Wecker, and Matthias Troyer. Elucidating reaction mechanisms on quantum computers. *Proceedings of the national academy of sciences*, 114(29):7555–7560, 2017.
- [29] Ryan Babbush, Nathan Wiebe, Jarrod McClean, James McClain, Hartmut Neven, and Garnet Kin-Lic Chan. Low-depth quantum simulation of materials. *Physical Review X*, 8(1):011044, 2018.
- [30] Andrew M Childs, Dmitri Maslov, Yunseong Nam, Neil J Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, 2018.
- [31] Brett Giles and Peter Selinger. Exact synthesis of multiqubit Clifford+T circuits. *Physical Review A Atomic, Molecular, and Optical Physics*, 87(3):032332, 2013.
- [32] Vadym Kliuchnikov, Dmitri Maslov, and Michele Mosca. Practical approximation of single-qubit unitaries by single-qubit quantum Clifford and T circuits. *IEEE Transactions on Computers*, 65(1):161–172, 2015.
- [33] Neil J Ross and Peter Selinger. Optimal ancilla-free Clifford+T approximation of Z-rotations. *Quantum Inf. Comput.*, 16(11&12):901–953, 2016.
- [34] Michele Mosca and Priyanka Mukhopadhyay. A polynomial time and space heuristic algorithm for T-count. *Quantum Science and Technology*, 7(1):015003, 2021.
- [35] Vlad Gheorghiu, Michele Mosca, and Priyanka Mukhopadhyay. A (quasi-) polynomial time heuristic algorithm for synthesizing T-depth optimal circuits. *npj Quantum Information*, 8(1):110, 2022.
- [36] Vlad Gheorghiu, Michele Mosca, and Priyanka Mukhopadhyay. T-count and T-depth of any multi-qubit unitary. *npj Quantum Information*, 8(1):1–10, 2022.

	2^2	2^3	2^4	2^5	2^6	2^7	2^8	2^9	2^{10}	2^{11}	2^{12}
b_1	2	4	8	16	32	64	128	256	512	1024	2048
b_2	0	0	0	0	0	0	0	0	0	0	0
b_3	0	0	0	0	0	0	0	0	0	0	0
b_4	0	0	0	0	0	0	0	0	0	0	0
b_5	2	4	8	16	32	64	128	256	512	1024	2048
b_6	0	2	4	8	16	32	64	128	256	512	1024
b_7	1	3	4	8	16	32	64	128	256	512	1024
b_8	0	0	3	8	16	32	64	128	256	512	1024
b_9	1	3	9	19	32	64	128	256	512	1024	2048
b_{10}	0	1	4	11	23	48	96	192	384	768	1536
b_{11}	0	1	5	12	25	53	96	192	384	768	1536
b_{12}	0	1	4	10	22	44	91	192	384	768	1536
b_{13}	0	1	5	16	35	63	135	263	512	1024	2048
b_{14}	0	0	3	9	24	49	102	213	439	896	1792
b_{15}	0	0	3	13	28	67	122	226	451	905	1792
b_{16}	0	0	2	9	21	49	102	217	447	886	1763
b_{17}	0	0	1	8	26	60	116	261	507	1029	2073
b_{18}	0	0	0	6	22	51	106	216	458	935	1915
b_{19}	0	0	0	6	22	51	123	248	487	987	1970
b_{20}	0	0	0	5	19	44	105	222	465	935	1876
b_{21}	0	0	0	1	18	56	118	235	503	1023	2080
b_{22}	0	0	0	0	15	47	108	226	458	961	1941
b_{23}	0	0	0	0	12	41	103	241	494	995	2026
b_{24}	0	0	0	0	10	40	96	225	465	958	1928
b_{25}	0	0	0	0	1	37	104	231	478	1019	2038
b_{26}	0	0	0	0	0	31	93	208	447	927	1936
b_{27}	0	0	0	0	0	25	89	211	475	990	2020
b_{28}	0	0	0	0	0	20	83	202	448	963	1965
b_{29}	0	0	0	0	0	1	73	194	448	943	1993
b_{30}	0	0	0	0	0	0	63	190	440	942	1926
b_{31}	0	0	0	0	0	0	51	178	435	952	2008
b_{32}	0	0	0	0	0	0	40	162	411	918	1911
b_{33}	0	0	0	0	0	0	1	146	393	904	1922
b_{34}	0	0	0	0	0	0	0	125	377	878	1870
b_{35}	0	0	0	0	0	0	0	103	352	856	1854
b_{36}	0	0	0	0	0	0	0	81	323	828	1875
b_{37}	0	0	0	0	0	0	0	1	291	793	1821
b_{38}	0	0	0	0	0	0	0	0	251	756	1755
b_{39}	0	0	0	0	0	0	0	0	208	707	1737
b_{40}	0	0	0	0	0	0	0	0	162	645	1665
b_{41}	0	0	0	0	0	0	0	0	1	579	1592
b_{42}	0	0	0	0	0	0	0	0	0	503	1515
b_{43}	0	0	0	0	0	0	0	0	0	416	1412
b_{44}	0	0	0	0	0	0	0	0	0	325	1294
b_{45}	0	0	0	0	0	0	0	0	0	1	1159
b_{46}	0	0	0	0	0	0	0	0	0	0	1004
b_{47}	0	0	0	0	0	0	0	0	0	0	832
b_{48}	0	0	0	0	0	0	0	0	0	0	651
b_{49}	0	0	0	0	0	0	0	0	0	0	1

TABLE VII: Number of integers less than 2^x , $x = 2, 3, \dots, 12$, such that the binary decomposition of their fourth power has 1 at a certain bit position. That is, the cell at the intersection of row labeled by bit b_i and column labeled by integer 2^x stores the number of integers $n \leq 2^x$ such that the binary decomposition of n^4 has 1 at bit position b_i .

- [37] Priyanka Mukhopadhyay. Synthesis of V-count-optimal quantum circuits for multiqubit unitaries. *Physical Review A*, 109(5):052619, 2024.
- [38] Priyanka Mukhopadhyay. CS-count-optimal quantum circuits for arbitrary multi-qubit unitaries. *Scientific Reports*, 14(1):13916, 2024.
- [39] Priyanka Mukhopadhyay. Synthesizing Toffoli-optimal quantum circuits for arbitrary multi-qubit unitaries. *arXiv preprint arXiv:2401.08950*, 2024.
- [40] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Phys. Rev. A*, 86:032324, Sep 2012.
- [41] Austin G. Fowler and Craig Gidney. Low overhead quantum computation using lattice surgery, 2019.
- [42] Daniel Litinski. Magic State Distillation: Not as Costly as You Think. *Quantum*, 3:205, December 2019.
- [43] Meijian Li, Tuomas Lappi, and Xingbo Zhao. Scattering and gluon emission in a color field: A light-front Hamiltonian approach. *Phys. Rev. D*, 104(5):056014, 2021.
- [44] Niklas Mueller, Andrey Tarasov, and Raju Venugopalan. Deeply inelastic scattering structure functions on a hybrid quantum computer. *Phys. Rev. D*, 102(1):016007, 2020.
- [45] Michael Kreshchuk, William M. Kirby, Gary Goldstein, Hugo Beauchemin, and Peter J. Love. Quantum simulation of quantum field theory in the light-front formulation. *Phys. Rev. A*, 105(3):032418, 2022.
- [46] Heinz J. Rothe. *Lattice Gauge Theories*. World Scientific Publishing Company, 2012.
- [47] Zoltan Bajnok and Marton Lajer. Truncated Hilbert space approach to the 2d ϕ^4 theory. *Journal of High Energy Physics*, 2016(10), October 2016.
- [48] Barak Gabai and Xi Yin. On the S-matrix of Ising field theory in two dimensions. *Journal of High Energy Physics*, 2022(10), October 2022.
- [49] Martin Lüscher. Two-particle states on a torus and their relation to the scattering matrix. *Nuclear Physics B*, 354(2):531–578, 1991.
- [50] Raúl A. Briceño, Jozef J. Dudek, and Ross D. Young. Scattering processes and resonances from lattice QCD. *Rev. Mod. Phys.*, 90:025001, Apr 2018.
- [51] Maxwell T. Hansen and Stephen R. Sharpe. Lattice QCD and three-particle decays of resonances. *Annual Review of Nuclear and Particle Science*, 69(Volume 69, 2019):65–107, 2019.
- [52] Akaki Rusetsky. Three particles on the lattice, 2019.
- [53] Maxim Mai, Michael Döring, and Akaki Rusetsky. Multi-particle systems on the lattice and chiral extrapolations: a brief review. *The European Physical Journal Special Topics*, 230(6):16231643, June 2021.
- [54] Fernando Romero-Lpez. Multi-hadron interactions from lattice QCD, 2023.
- [55] Richard D Mattuck and Börje Johansson. Quantum field theory of phase transitions in Fermi systems. *Advances in Physics*, 17(68):509–562, 1968.
- [56] JR Espinosa, M Quiros, and Fabio Zwirner. On the phase transition in the scalar theory. *Physics Letters B*, 291(1-2):115–124, 1992.
- [57] Matthew Hagan and Nathan Wiebe. Composite quantum simulations. *Quantum*, 7:1181, 2023.
- [58] Abhishek Rajput, Alessandro Roggero, and Nathan Wiebe. Hybridized methods for quantum simulation in the interaction picture. *Quantum*, 6:780, 2022.
- [59] Rolando D. Somma. Quantum simulations of one dimensional quantum systems. *arXiv*, August 2016.
- [60] Alexandru Macridin, Panagiotis Spentzouris, James Amundson, and Roni Harnik. Digital quantum computation of fermion-boson interacting systems. *Physical Review A*, 98(4):042312, October 2018.
- [61] Alexandru Macridin, Panagiotis Spentzouris, James Amundson, and Roni Harnik. Electron-Phonon Systems on a Universal Quantum Computer. *Physical Review Letters*, 121(11):110504, September 2018.
- [62] Enrico Rinaldi, Xizhi Han, Mohammad Hassan, Yuan Feng, Franco Nori, Michael McGuigan, and Masanori Hanada. Matrix Model simulations using Quantum Computing, Deep Learning, and Lattice Monte Carlo. *PRX Quantum*, 3(1):010324, February 2022.
- [63] Masanori Hanada, Junyu Liu, Enrico Rinaldi, and Masaki Tezuka. Estimating truncation effects of quantum bosonic systems using sampling algorithms. *arXiv*, March 2023.
- [64] Jacky Huyghebaert and Hans De Raedt. Product formula methods for time-dependent Schrodinger problems. *Journal of Physics A: Mathematical and General*, 23(24):5777, 1990.
- [65] Nathan Wiebe, Dominic Berry, Peter Høyer, and Barry C Sanders. Higher order decompositions of ordered operator exponentials. *Journal of Physics A: Mathematical and Theoretical*, 43(6):065203, 2010.
- [66] Dave Wecker, Matthew B Hastings, Nathan Wiebe, Bryan K Clark, Chetan Nayak, and Matthias Troyer. Solving strongly correlated electron models on a quantum computer. *Physical Review A*, 92(6):062318, 2015.
- [67] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu. Theory of Trotter error with commutator scaling. *Phys. Rev. X*, 11:011020, Feb 2021.
- [68] Masuo Suzuki. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Journal of Mathematical Physics*, 32(2):400–407, 1991.
- [69] Andrew M. Childs, Dmitri Maslov, Yunseong Nam, Neil J. Ross, and Yuan Su. Toward the first quantum simulation with quantum speedup. *Proceedings of the National Academy of Sciences*, 115(38):9456–9461, 2018.
- [70] Ryan Babbush, Craig Gidney, Dominic W. Berry, Nathan Wiebe, Jarrod McClean, Alexandru Paler, Austin Fowler, and Hartmut Neven. Encoding electronic spectra in quantum circuits with linear T complexity. *Phys. Rev. X*, 8:041015, Oct 2018.
- [71] Yunseong Nam, Yuan Su, and Dmitri Maslov. Approximate quantum Fourier transform with $O(n \log(n))T$ gates. *npj*

- Quantum Information*, 6(1):1–6, 2020.
- [72] Guang Hao Low and Isaac L Chuang. Optimal Hamiltonian simulation by quantum signal processing. *Physical Review Letters*, 118(1):010501, 2017.
 - [73] Guang Hao Low and Isaac L Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3:163, 2019.
 - [74] Andrew M Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *arXiv preprint arXiv:1202.5822*, 2012.
 - [75] Yong He, Ming-Xing Luo, E Zhang, Hong-Ke Wang, and Xiao-Feng Wang. Decompositions of n-qubit Toffoli gates with linear circuit complexity. *International Journal of Theoretical Physics*, 56(7):2350–2361, 2017.
 - [76] Craig Gidney. Halving the cost of quantum addition. *Quantum*, 2:74, 2018.
 - [77] Farrokh Vatan and Colin Williams. Optimal quantum circuits for general two-qubit gates. *Phys. Rev. A*, 69:032315, Mar 2004.
 - [78] Alex Bocharov, Martin Roetteler, and Krysta M. Svore. Efficient synthesis of universal repeat-until-success quantum circuits. *Phys. Rev. Lett.*, 114:080502, Feb 2015.
 - [79] Peter Selinger. Efficient Clifford+T approximation of single-qubit operators, 2014.
 - [80] Adriano Barenco, Charles H Bennett, Richard Cleve, David P DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical review A*, 52(5):3457, 1995.
 - [81] Matthew Amy, Parsiad Azimzadeh, and Michele Mosca. On the controlled-not complexity of controlled-not–phase circuits. *Quantum Science and Technology*, 4(1):015002, 2018.
 - [82] Vlad Gheorghiu, Jiaxin Huang, Sarah Meng Li, Michele Mosca, and Priyanka Mukhopadhyay. Reducing the CNOT count for Clifford+T circuits on NISQ architectures. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2022.
 - [83] Alexei Kitaev and William A. Webb. Wavefunction preparation and resampling using a quantum computer, 2009.
 - [84] Dominic W Berry, Andrew M Childs, and Robin Kothari. Hamiltonian simulation with nearly optimal dependence on all parameters. In *2015 IEEE 56th annual symposium on foundations of computer science*, pages 792–809. IEEE, 2015.
 - [85] Yutaro Akahoshi, Kazunori Maruyama, Hirotaka Oshima, Shintaro Sato, and Keisuke Fujii. Partially fault-tolerant quantum computing architecture with error-corrected Clifford gates and space-time efficient analog rotations. *PRX Quantum*, 5(1):010337, 2024.
 - [86] Earl T. Campbell, Barbara M. Terhal, and Christophe Vuillot. Roads towards fault-tolerant universal quantum computation. *Nature*, 549(7671):172179, September 2017.
 - [87] Austin G Fowler, Matteo Mariantoni, John M Martinis, and Andrew N Cleland. Surface codes: Towards practical large-scale quantum computation. *Physical Review A Atomic, Molecular, and Optical Physics*, 86(3):032324, 2012.
 - [88] Dave Wecker, Bela Bauer, Bryan K Clark, Matthew B Hastings, and Matthias Troyer. Gate-count estimates for performing quantum chemistry on small quantum computers. *Physical Review A*, 90(2):022305, 2014.
 - [89] Ryan Babbush, Jarrod McClean, Dave Wecker, Alán Aspuru-Guzik, and Nathan Wiebe. Chemical basis of Trotter-Suzuki errors in quantum chemistry simulation. *Physical Review A*, 91(2):022311, 2015.
 - [90] João Barata, Xiaojian Du, Meijian Li, Wenyang Qian, and Carlos A. Salgado. Quantum simulation of in-medium QCD jets: Momentum broadening, gluon production, and entropy growth. *Phys. Rev. D*, 108(5):056023, 2023.
 - [91] The cost of incrementing 1 to an n -bit number is $n - 2$ ANDs.
 - [92] The cost of multiplying 2 n -bit numbers is $2n^2 - 1$ many logical ANDs.