

Quantum Phase Estimation Without Controlled Unitaries

Laura Clinton^{1,*} Toby S. Cubitt^{1,2} Raul Garcia-Patron^{1,3} Ashley Montanaro^{1,4} Stasja Stanisic¹
and Maarten Stroeks^{1,5,†}

¹*Phasecraft Ltd., London, United Kingdom*

²*Department of Computer Science, University College London, London, United Kingdom*

³*School of Informatics, University of Edinburgh, Edinburgh, United Kingdom*

⁴*School of Mathematics at the University of Bristol, University of Bristol, Bristol, United Kingdom*

⁵*Delft Institute of Applied Mathematics & QuTech, Delft University of Technology, Delft, The Netherlands*

 (Received 18 August 2025; revised 19 November 2025; accepted 7 January 2026; published 5 March 2026)

In this work we demonstrate the use of adapted classical phase retrieval algorithms to perform control-free quantum phase estimation. We eliminate the costly controlled time evolution and Hadamard test commonly required to access the complex time-series needed to reconstruct the spectrum. This significant reduction of the number of coherent controlled-operations lowers the circuit depth and considerably simplifies the implementation of statistical quantum phase estimation in near-term devices. This seemingly impossible task can be achieved by extending the problem that one wishes to solve to one with a larger set of input signals while exploiting natural constraints on the signal and/or the spectrum. We leverage well-established algorithms that are widely used in the context of classical signal processing, demonstrating two complementary methods to do this, vectorial phase retrieval and two-dimensional phase retrieval. We numerically investigate the feasibility of both approaches for estimating the spectrum of the Fermi-Hubbard model and discuss their resilience to inherent statistical noise.

DOI: [10.1103/7qcr-znl2](https://doi.org/10.1103/7qcr-znl2)

I. INTRODUCTION

The prediction of spectral properties is a key aspect in the discovery of materials and advancing our understanding of chemical properties and reactions [1]. The evaluation of spectral properties of large many-body systems is a computationally intensive task, limited by numerical bottlenecks associated with classical simulation of time-evolution [2]. Quantum computers have the potential to make such simulations tractable, offering the opportunity to revolutionize material science and chemistry [3]. Algorithms based on quantum phase estimation (QPE) offer a route toward obtaining such spectral properties, albeit requiring one to perform many costly time-evolution simulation steps *controlled* on one or several ancillary qubit(s), making it difficult to run on current *fault-prone* quantum computers. Therefore, eliminating or reducing the need for controlled time-evolution in QPE has attracted a lot

of attention, as it could render the prediction of spectral properties accessible to near-term devices.

A significant step in this direction was the proposal of statistical phase estimation (see, e.g., Refs. [4–6]) which reduces QPE to the computation of a time series $f(t) = \langle \psi | e^{iHt} | \psi \rangle$ on a quantum device followed by classical postprocessing to obtain an estimate of the spectrum of H with similar resolution as standard QPE for most Hamiltonians. As shown in Fig. 1(a), statistical phase estimation reduces the required quantum circuit to two variants of a Hadamard test circuit, giving the real and imaginary parts of $f(t)$. This result has spurred a plethora of theoretical work [7–9], followed by experimental demonstrations [10,11], which can be adapted to estimation of more general spectral quantities such as linear response of quantum systems [12], operator-resolved density of states [13], or Green functions [14], to give some examples. Even if statistical phase-estimation represents indisputable progress toward making quantum phase estimation more reachable for near-term devices, a single-qubit control time-evolution remains a highly nontrivial operation to implement on near-term devices. This makes it often impractical to implement statistical phase estimation on currently existing hardware (beyond trivial-size problems). Its difficulty arises from the significant numbers of qubits and nontrivial circuit depth time-evolution operations that

*Contact author: laura@phasecraft.io

†Contact author: m.e.h.m.stroeks@tudelft.nl

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

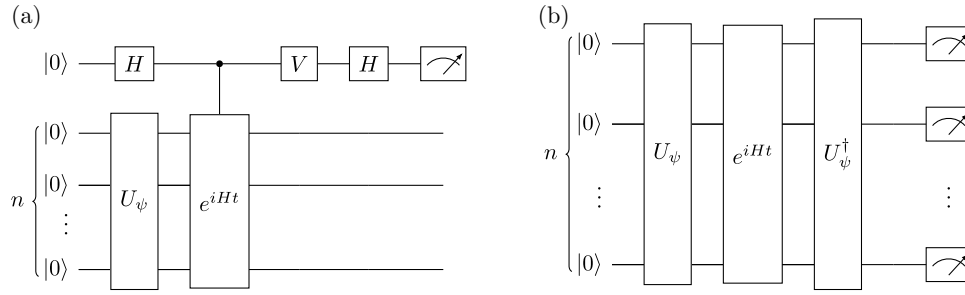


FIG. 1. (a) Single-ancillary-qubit QPE computing a time series $f(t) = \langle \psi | e^{iHt} | \psi \rangle$ for time t . The circuit U_ψ prepares the input state $|\psi\rangle$, which is followed by a circuit implementing e^{iHt} controlled by the single-ancillary-qubit. The gate V is an identity ($S^\dagger$ gate) when estimating the real (imaginary) part of the time series. (b) Ancillary-qubit-free circuit that allows one to estimate $|f(t)|^2 = |\langle \psi | e^{iHt} | \psi \rangle|^2$.

need to be controlled by one (or few) qubit(s), making it extremely sensitive to decoherence, increasing the cost of each time-evolution step, and also significantly increasing the circuit depth of implementing all of the required controlled-time-evolution operations.

The significant limitation imposed by the *controlled* time-evolution operation has led to an effort to develop algorithmic techniques to retrieve spectra using control-free time evolution operations. Note that the quantity $|\langle \psi | e^{iHt} | \psi \rangle|^2$ can be obtained with the circuit shown in Fig. 1(b), which does not make use of a controlled operation. Unfortunately, this only gives us access to the absolute values of the time-series $f(t)$, while its phases are key in recovering the spectrum. One way of exploiting the access to $|\langle \psi | e^{iHt} | \psi \rangle|^2$ is to compute its Fourier transform to retrieve a new spectrum composed of differences between energy levels, rather than obtaining the energies directly, as done in Ref. [15]. Alternatively, in Ref. [16], the authors use a complex-time evolution, and in Ref. [17], the authors use the knowledge of one eigenvector and eigenenergies to obtain information about the spectrum using control-free operations. As we detail in the next subsection, each technique has its own limitations and sometimes strong constraints, leaving room for significant improvements.

Retrieving a spectrum while not having (complete) access to the phases of its signal is a well-known problem in classical signal-processing called phase-retrieval [18], which has been extensively studied and offers a plethora of different solutions, each one having its own benefits and set of most appropriate applications. The time-series obtained on a digital quantum computer or quantum simulator can be understood as a classical signal and retrieving the spectrum of the many-body problem can therefore benefit from phase-retrieval ideas and algorithms.

In this work we show how the adaptation to quantum problems of two different techniques, *vectorial phase-retrieval* [19] and the *two-dimensional phase-retrieval* [20], allows us to remove the intricate controlled

time-evolution currently limiting the scalability of phase-estimation algorithms. Vectorial phase-retrieval works by including measurements of absolute values of additional time series, and—crucially—measurements of the absolute values of interferences of the target time series and these additional time series. In addition, we show how to recast the problem as a two-dimensional one, which can be solved with the well-known hybrid input-output algorithm (HIO) [20]. We numerically investigate the feasibility of both approaches for estimating the spectrum of the Fermi-Hubbard model and discuss the resilience of the algorithms to inherent statistical noise.

A. Related work

There are a couple of techniques that allow to compute time-series without having to implement a controlled time evolution. Both require as condition the knowledge and easy preparation of an eigenstate of the Hamiltonian of interest. If this can be easily satisfied for many physically motivated Hamiltonians, for example, for all Hamiltonians having vacuum as an eigenstate, the techniques will fail for arbitrary Hamiltonians. The first approach, detailed in Ref. [21], requires doubling the number of qubits and the implementation of a pair of control-SWAP operations over the two register, where the time-evolution is applied locally on one of them and without the needs of control. It is worth pointing that for current QPU devices, doubling the number of qubits may turn unfeasible what could be an initial good candidate to quantum advantage. If doubling the number of qubits is not a big limitation, the approach has a limited additional cost given by the implementation pair of control-SWAP, which will result in a extra cost of $O(N)$ CNOT gates. The second approach [17,22–24] requires the preparation of a superposition $|\psi\rangle + e^{i\theta} |R\rangle$, where $|R\rangle$ is an eigenvector of the Hamiltonian that will play the role of a phase reference. One can then exploit postprocessing algorithms to extract the value of our time-series of interest. Unfortunately, even in the case where $|R\rangle$

is trivial to prepare, for example, an all zeros state, creating the superposition would be in general cases nontrivial, as it will require a controlled implementation of $U_{\psi}^{\dagger}$. Again, removing of the control time-evolution shifts the burden to an extra cost that depends on the original gate count of $U_{\psi}^{\dagger}$, leading to a cost that would be of at least $O(N)$ CNOT gates but in many cases $O(N^2)$.

Last year, there were two works that removed those constraints on the input state requirements, making ancillary-free QPE closer to the range of applicability of traditional QPE. In Ref. [15], Chan *et al.* leverage the power of the recently proposed classical shadows technique to estimate energy gaps via the computation of expectation values of certain operators as a function of time. A downside of the technique is that it cannot provide the spectrum of $\langle \psi | e^{iHt} | \psi \rangle$, but rather the spectrum of $|\langle \psi | e^{iHt} | \psi \rangle|^2$, which leaves the challenge of recovering the eigenvalues of the Hamiltonian from the set of all pairwise energy gaps. Almost at the same time, Yang *et al.* showed that one can remove the ancillary qubit by combining real-time and imaginary-time evolution [16], exploiting ideas from complex function analysis. Its main limitation is the need for an additional implementation of imaginary-time evolution on a quantum device itself, which currently works only for input states with finite correlation length and short evolution times, but improvement in this direction could render the technique more widely applicable. Finally, in the recent work [13], the authors propose a hybrid classical-quantum algorithm for calculating quantities of interest in many-body spectroscopy that exploits ideas inspired by classical shadows that do not require a controlled time-evolution due to particular properties of the spectral objects of interest. Nonetheless, the approach requires the application of a (potentially shallow) controlled unitary gate independent of the time evolution. In addition, despite the promise of delegating part of the computation to a classical computer, the constraints it imposes in terms of having access to families of states that can be classically emulated may limit its use for arbitrary Hamiltonians.

B. Comparison of controlled vs uncontrolled time evolution

There are two key advantages of removing the controlled time-evolution: (1) It takes out a significant source

of errors resulting from the sensitivity of the original scheme to the decoherence of the single qubit controlling the dynamics; (2) it requires a smaller number of gates and less circuit depth to implement time evolution compared to its controlled counterpart. Here we summarize in Table I the reduction in circuit complexity achieved when removing the time-evolution for three simple examples that capture the essence of the discussion: (i) the 1D transverse-field Ising model (TFIM) implemented on hardware with all-to-all connectivity; (ii) 1D TFIM with matching hardware connectivity; (iii) the 2D Fermi-Hubbard model with square-lattice hardware connectivity. In each case we calculate the complexity of 1D vectorial phase retrieval. We do not take into account the cost of preparing or uncomputing the initial state $|\psi\rangle$, which we assume to be simple, nor the cost of preparing a superposition of $|\psi\rangle$ and another state [25]. However, we also make assumptions which significantly favor the controlled time evolution calculation which we use for comparison. We do not cost the extra control qubits used to reduce the circuit depth when implementing controlled time dynamics on 1D and 2D architectures. More details on the discussion and derivation can be found in Appendix D. These calculations do not take into account any Hamiltonian specific techniques which could be used to optimize further the gate counts.

In the case of quantum hardware with all-to-all connectivity, a quantum circuit computing a unitary operator U can be converted into a quantum circuit for computing controlled- U by replacing every CNOT gate with a Toffoli gate (requiring six CNOT gates each), and every single-qubit unitary V with a controlled- V operator (requiring two CNOT gates each). Despite being only a constant factor gain, this can make a significant difference to the ability of near-term quantum computers to obtain meaningful results, by increasing the number of accessible Trotter steps.

However, the difference in complexity between the two situations can be significantly greater if there are restrictions on hardware connectivity, as in 1D and 2D architectures. It is instructive to compare the bounds obtained in a realistic regime for near-term quantum algorithms. As an example, assume that we have a quantum computer with 100 qubits that can execute quantum circuits of CNOT depth at most 100 accurately. Then, based on the algorithmic complexities in Table I, we could not even implement 1

TABLE I. The cost of implementing k Trotter layers for several Hamiltonians, an n qubits transverse field Ising model (TFIM) and an $n \times n$ spinless Fermi-Hubbard model. “Depth” is CNOT depth. Note that the algorithms achieving the minimal CNOT depth and CNOT count may be different. “2D FH” is Fermi-Hubbard without spin, where results are approximate. In each case “PR” refers to the 1D vectorial phase retrieval approach, in which we do not consider the cost of preparing the initial state.

Model	H/w	CNOTs (PR)	CNOTs (no PR)	Depth (PR)	Depth (no PR)
1D TFIM	All-to-all	$(2n - 2)k$	$(6n - 4)k$	$4k$	$2\lceil \log_2 n \rceil + 10k$
1D TFIM	1D	$(2n - 2)k$	$6(n - 1) + (6n - 4)k$	$4k$	$6\lceil n/2 \rceil + 10k$
2D FH	2D	$32kn(n - 1)/2$	$48kn(n - 1)/2 + 6\lceil (n - 1)^2/2 \rceil$	$32k$	$48k + 3(n - 2)$

Trotter step for the 1D TFIM under 1D connectivity if the standard method is used but we could implement 25 Trotter steps using phase retrieval. The reductions in gate count are less significant than depth but are still relatively substantial (e.g., approximately a factor of 3 for the 1D TFIM).

A key aspect of making phase-retrieval work is to enlarge the problem that we wish to solve in a “cheap” way, either by enlarging the set of input states included in the procedure or by implementing additional (comparatively simple) time dynamics. The saving in terms of circuit depth therefore comes at a price of a larger number of circuits to be implemented and required shots.

C. Structure of paper

Section II summarizes our results on the adaptation of both vectorial phase-retrieval and two-dimensional phase-retrieval to statistical quantum phase-estimation with a final comparison of the performance of both techniques. This is followed by a conclusion and discussion (Sec. III). The paper also contains three Appendixes on the technical preliminaries (Appendix A), and the full details of our adaptation of vectorial phase-retrieval (Appendix B) and two-dimensional phase-retrieval (Appendix C) to statistical phase-estimation.

II. SUMMARY OF RESULTS

The classical postprocessing in statistical phase-estimation is a standard signal processing problem: One is provided with a noisy version of some complex-valued signal $f(t)$, which is typically measured at integer multiples of some time increment Δt (resulting in a discrete time-series [26] $f[t]$). The goal is to approximately determine the Fourier transform $F(\omega)$ of the continuous signal, typically corresponding to some quantity of interest such as the spectrum of a Hamiltonian. This is a nontrivial task due to noise and to finite measurement time T . There exists extensive literature on this problem with applications to multiple areas of engineering, such as optical imaging, spectroscopy, and audio processing [18,27–29]. We note that in the statistical phase-estimation scenario, the largest measurement time T is bounded by the largest circuit depth we can afford in the simulation of the *controlled* time evolution unitary e^{iHT} .

The goal of phase-retrieval as presented in classical signal processing literature is to reconstruct the signal from the absolute values of its discrete Fourier transform. At first sight, this is an impossible task, as the phases are key to achieve the reconstruction. As with any no-go result, the question is which rules one needs to change to circumvent it. This can be successfully achieved by extending the problem that one wishes to solve to one with a larger set of input signals and exploiting natural constraints on the signal and/or the spectrum. A key nontrivial insight is to understand when this generalized and constrained problem

has a unique solution that recovers all phases of the signals. If such an insight can be used to develop an algorithm that solves the problem in the ideal noiseless case, a nontrivial aspect is to design algorithms that work efficiently in realistic regimes of noise and have reasonable computational cost. This work is devoted to designing these algorithms in the scenario of ancillary-free QPE. It is important to note that phase-retrieval literature defines *unique solution* (for the spectrum) up to unavoidable trivial ambiguities corresponding to a multiplication by a global phase, a translation, and reflection with complex-conjugation of the recovered spectrum $F[k]$. The first two have no real physical impact, as experimentally we only have access to energy differences while the final ambiguity can reflect the spectrum, an effect that can be mitigated if handled with proper care.

The adaptation of phase-retrieval algorithms to the quantum-phase-estimation scenario is unfortunately not a trivial “plug & play”. First it requires a conceptual adaptation of the framework. The standard problem in classical phase-retrieval literature consists of reconstructing the phases $e^{i\theta[t]}$ of $f[t]$ while having access to $|F[\omega]|$, where in our case we know $|f[t]|$ and want to recover $e^{i\theta[t]}$ (and thereby $F[\omega]$) instead. Because of the symmetries of the standard classical signal processing problem where both signal and its spectrum are potentially complex functions, we can most of the time circumvent this slight difference by just exchanging the role of time and frequency in most phase-retrieval algorithms. Secondly, one needs to find an enlarged set of time-series, by enlarging the set input states or inducing additional dynamics, that combined with natural constraints on the spectrum, such as its non-negativity or the fact that it has a well-defined support, guarantee an efficient recovery of the phases $e^{i\theta[t]}$ (we shall come back to this point below and further details are provided in Appendixes B and C). This will finally allow approximate reconstruction of our initial spectrum of interest, but—importantly—also of the phases of the time series $f[t]$. In this paper, we show how to redesign two well-known phase-retrieval techniques, vectorial phase-retrieval and two-dimensional phase-retrieval, to address the problem of ancillary-free quantum phase-estimation. Vectorial phase-retrieval [19] achieves its goal by including measurements of the absolute values of multiple well-behaved time-series and—importantly—the time-series of their interferences. Where standard vectorial phase-retrieval makes use of two signals (the target signal and one additional signal) and their interferences, we generalize to multiple additional signals to improve the algorithm’s resilience to noise in the QPE framework. The total number of required samples equals $(1 + 3R)$ times the number of samples taken for each individual times series, where R denotes the number of additional signals used in the analysis. Secondly, we show how to transform a one-dimensional time series (corresponding to time evolution

of an input state under a Hamiltonian) into a two dimensional phase-retrieval problem by adding a dummy Hamiltonian H_D that commutes with H and produces nontrivial dynamics on the input state of interest. We then exploit the well-known *Hybrid Input Output* algorithm to retrieve the target spectra. The sampling cost then increases by a factor equal to the number of measurement points taken in this second “time” dimension.

It is important to point out that our objective is never to recover the individual eigenvalues in the spectrum of a given Hamiltonian, which is not practical even for a fault-tolerant computer due to the exponential increase of the number of spectral peaks with the size of the system, but to retrieve a realistic spectrum resulting from a time evolution of time T and time steps ΔT . The unavoidable prescience of error in near-term devices affects both statistical phase-estimation and phase-retrieval techniques, and reconstructing the initial exact signal is a complex task beyond the scope of this paper.

A fair goal for this work is to compare the phase-retrieval postprocessing to its statistical phase-estimation counterpart, where both have the same realistic constraint on the time-dynamics simulation errors (Trotter errors, for example) and choices of accessible time and size of time intervals. It is worth stressing we are not judging the quality of the output signal from a physicist, chemist or material scientist point of view, but rather given a time-series as reference, which quality can depend on the quality of time-dynamics simulation, how the trivial recovery by statistical phase-estimation compares to the nontrivial one using phase-retrieval techniques. We therefore focus on the errors resulting from the finite-size of samples that induces a shot noise in the evaluation of the expectation values of interest. Demonstrating a successful recovery by phase-retrieval algorithms in this regime is not trivial and is the focus of our work here. In a nutshell, the effect of Trotter errors or the effect of time-interval choices is to shift the ground truth for the post-processing analysis. On the other hand, the gate errors can have slightly different impacts, as statistical phase-estimation implements a different circuit than the ones we propose here. Statistical phase-estimation needs a control time-evolution that has a higher gate counts than standard time-dynamics, while phase-retrieval may need a reference state preparation (vectorial phase-retrieval) or a slightly longer time-evolution (2D phase-retrieval). We leave a more refined effect of error in the quantum circuit for future work, as our primary focus here is the recovery performance of our algorithm. However, errors due to quantum noise in the circuit can often be modeled as global depolarizing noise [30]. Under this model the time series will experience exponential decay as a function of time, this will become Lorentzian line broadening on the spectrum.

We stress that the goal of phase-retrieval is solely to reconstruct the phases of the time series and we could

use different metrics to quantify the quality of its recovery. Because our ultimate aim is to use the time-series to later generate a QPE spectrum related to a given many-body problem, we will use the recovery of the spectrum as a good metric of performance of the phase-retrieval problem. More concretely we use the 1-norm between the reconstruction and the exact solution. In order to keep the analysis simple we just implement a recovery relying on a discrete Fourier transform (DFT) of the time-series, however, we note that the discrete Fourier transform is not necessarily the most accurate reconstruction of the spectrum of a Hamiltonian. Typically, classical postprocessing techniques like the one developed in Ref. [5] can be used to obtain a more useful description of the spectrum.

Let us stress that we work in the scenario in which the available circuit depth is limited to some finite value. In that scenario, the presented circuit-depth reduction might yield the estimation of some previously unavailable many-body spectra tractable. As outlined above, this comes at some sampling cost. We do not explicitly bound this sampling cost in terms of the accuracy with which the spectrum is recovered. However, our numerical investigations suggest that the number of samples to be taken remains reasonable, even for larger system sizes.

In the rest of the document we will use dimensionless time and frequency variables, i.e., we will use the notation $f[j]$, where $j \in \mathbb{N}$ and $t_j = j \Delta t$, and $F[k]$, where $k \in \mathbb{N}$ and $\omega_k = 2\pi k/T$, and we have defined $\Delta t = T/N$. We will be interested in instances of the Fermi-Hubbard model over a line or square lattice graph G , where V is the set of vertices and E the edges, where each lattice site, i.e., vertex, has two spin modes $\sigma \in \{\uparrow, \downarrow\}$. The Hamiltonian reads

$$H = -\tau \sum_{(i,j) \in E, \sigma} \left(a_{i\sigma}^\dagger a_{j\sigma} + a_{j\sigma}^\dagger a_{i\sigma} \right) + U \sum_{v \in V} n_{v\uparrow} n_{v\downarrow},$$

where the first term of the Hamiltonian consists of hopping terms among modes of same spin while the second sum corresponds to interactions between particles of opposite spin at the same lattice site. Throughout this work we will take $\tau = 1$ and $U = 4$, which is an intermediate coupling regime where the model exhibits nontrivial behavior [31]. When referring to states, we are—throughout this work—referring to qubit states, which are related to the fermionic states by a Jordan-Wigner transformation. For later reference, we note that a standard basis qubit state corresponds to a position-spin-basis Slater determinant.

A. Vectorial phase retrieval

Vectorial phase retrieval [19] is a one-dimensional phase retrieval technique. One-dimensional phase retrieval techniques aim to solve a problem that, clearly, is ambiguous in general. Namely, without any further constraints, *every*

assignment of phases, denoted by $\{x_j\}_{j=0}^{N-1}$, constitutes a time series $\{f[j]|x_j\}_{j=0}^{N-1}$ that is consistent with the absolute value measurements $\{|f[j]|\}_{j=0}^{N-1}$. Therefore, in order to (approximately) solve the one-dimensional phase retrieval problem, one has to include additional constraints or measurements. In the vectorial phase retrieval framework, this is done by including absolute value measurements of interferences between the target time series f (which we shall denote by f_1 in the remainder of this section) and another (secondary) time series f_2 . To be more precise, vectorial phase-retrieval resolves the ambiguity in the retrieval of the phases of the signal f_1 by measuring not only its absolute values $|f_1[j]|$, but also those of a secondary signal $|f_2[j]|$ and their interferences $|f_1[j] + f_2[j]|$ and $|f_1[j] + if_2[j]|$. The vectorial phase retrieval problem has a *unique* solution—up to trivial ambiguities—in the noiseless scenario, provided that (1) f_1 and f_2 are *spectrally independent* (i.e., their z -transforms have no common roots in the complex plane [32]) and (2) the discrete Fourier transforms of f_1 and f_2 (denoted, respectively, by F_1 and F_2) have no support outside some finite interval (of size at most $N/2$ or, equivalently, of size at most π in terms of frequency) [19].

In the setting of control-free QPE, we take $f_1[j] = \langle \Phi | \exp(ijH\Delta t) | \Phi \rangle$ and $f_2[j] = \langle \Phi | \exp(ijH\Delta t) | \psi \rangle$ (for some state $|\psi\rangle$ different from $|\Phi\rangle$), so that $f_1[j] + f_2[j] = \langle \Phi | \exp(ijH\Delta t) (|\Phi\rangle + |\psi\rangle)$ and $f_1[j] + if_2[j] = \langle \Phi | \exp(ijH\Delta t) (|\Phi\rangle + i|\psi\rangle)$. The constraint of F_1 and F_2 only being supported on a frequency interval of size π can be satisfied by an appropriate choice of Δt . Note that for standard statistical phase estimation routines, such a constraint should also be satisfied, except that there the support should be of size 2π . We stress that the aforementioned condition (1) of f_1 and f_2 being spectrally independent does

not mean that states $|\Phi\rangle$ and $|\psi\rangle$ can only have support on disjoint subsets of eigenvalues of H , but rather that the z -transforms of the time-series have no common roots in the complex plane.

Remark that in our scenario we are interested in retrieving the phases of the time series f_1 , which can then—for instance—be used to obtain F_1 . Two important factors that lead to one only obtaining an approximation of the correct assignment of phases $\{x_j\}_{j=0}^{N-1}$ in a practical setting are (1) the noise on the learned signals $|f_1|$ and $|f_2|$ and (2) the fact that F_1 and F_2 fail to have their support restricted to a well-defined finite interval. This second factor is caused by spectral leakage of F_1 and F_2 away from the eigenvalues of H , which may spread outside the expected restricted support. This can be particularly aggravated when the state $|\Phi\rangle$ and distribution of supported eigenvalues might be such that F_1 and F_2 decay slowly near either end of the $[0, \pi]$ interval.

We find that to ensure that the approximate solution obtained through vectorial phase retrieval is accurate, one has to ensure that the scenario is such that F_1 and F_2 have *approximately* well-defined support. This can generally be done by for instance increasing the total evolution time (at the expense of deeper circuits implementing the time evolution) or making informed choices for the reference states or for $|\Phi\rangle$ (when the problem we aim to solve allows for some freedom in the choice of the input state). Our numerical investigations suggest that vectorial phase retrieval performs relatively well even when F_1 and F_2 do not have well-defined support.

In what follows we summarize our implementation of the vectorial phase retrieval technique (and its application to control-free QPE), more details and derivations can be found in Appendix B.

1. Vectorial phase retrieval as an optimization problem

The key aspect behind the vectorial phase-retrieval technique is the realization that the phases of $f_1[j]$ and $f_2[j]$ are related by the relation $e^{\theta_1[j]} = e^{\theta_2[j]} G[j]$, where [33]

$$G[j] := \frac{|f_1[j] + f_2[j]|^2 + i|f_1[j] + if_2[j]|^2 - (1+i)(|f_1[j]|^2 + |f_2[j]|^2)}{2|f_1[j]||f_2[j]|} \quad (1)$$

is fully accessible from the measurement data. This allows us to define an optimization problem over the variable $\mathbf{y} \in \mathbb{C}^{2N}$ with unit absolute value entries, encoding the estimates of the phases $\{e^{\theta_1[j]}, e^{\theta_2[j]}\}$. The cost function to be optimized consists of two contributions. The first contribution quantifies how close we are from satisfying the

relative phase constraints $e^{\theta_1[j]} = e^{\theta_2[j]} G[j]$ and reads

$$\mathcal{Q}_{\text{interference}}(\mathbf{y}) := \sum_{j=0}^{N-1} |y_j - y_{N+j} G[j]|^2. \quad (2)$$

As will be argued in Appendix B (and in Ref. [19]), imposing these phase constraints is not sufficient for (approximate) recovery of the phases in general. Therefore, the cost function also contains a second contribution. This additional contribution quantifies how close we are from F_1 and F_2 being zero outside the support $\{0, 1, \dots, \sigma\}$ (for some integer $\sigma \in \{0, 1, \dots, \lfloor N/2 \rfloor - 1\}$, the *exact* value of which is unknown in a general setting). In an ideal noiseless scenario, this contribution is minimized if F_1 and F_2 are zero on the domain $\{\sigma + 1, \sigma + 2, \dots, N - 1\}$. In reality, one of the factors making this constraint not fully satisfied is the inevitable spectral leakage of the discrete Fourier transform, that will lead to F_1 and F_2 having support outside of the supported frequency range. Nonetheless, as we will demonstrate, the approach still recovers the phases relatively accurately provided that $|F_1[k]|$ and $|F_2[k]|$ decay quickly for $k > \sigma$. To impose that the solution (approximately) has the correct support size, we define a family of support cost functions, one for each $s \in \{0, 1, \dots, N - 1\}$ (because, again, the true support size σ might be unknown, or not even well defined), defined as

$$\begin{aligned} Q_{\text{support}}^{(s)}(\mathbf{y}) = & \sum_{k=s}^{N-1} \left| \sum_{j=0}^{N-1} |f_1[j]| y_j \exp[-i2\pi jk/N] \right|^2 \\ & + \sum_{k=s}^{N-1} \left| \sum_{j=0}^{N-1} |f_2[j]| y_{N+j} \exp[-i2\pi jk/N] \right|^2, \end{aligned} \quad (3)$$

quantifying how far the spectrum is from being zero on the region $\{s, s + 1, \dots, N - 1\}$.

As we explain in more detail in Appendix B, one can define a family of total cost functions, combining the support component and the interference component, which can be expressed as a non-negative quadratic form

$$Q^{(s)}(\mathbf{y}) := Q_{\text{support}}^{(s)}(\mathbf{y}) + Q_{\text{interference}}(\mathbf{y}) = \mathbf{y}^\dagger A_s^\dagger A_s \mathbf{y}, \quad (4)$$

where $A_s \in \mathbb{C}^{(2(N-s)+N) \times 2N}$ entries are detailed in Appendix B. Remark that $Q^{(s)}(\mathbf{y})$ is equal to zero for all $s \in \{\sigma + 1, \sigma + 2, \dots, N - 1\}$ if $\mathbf{y}$ is the unique solution of the ideal noiseless vectorial phase-retrieval problem (i.e., for which $|F_1[k]|$ and $|F_2[k]|$ are indeed equal to zero for $k > \sigma$). When $|F_1[k]|$ and $|F_2[k]|$ are not exactly equal to zero for $k > \sigma$, the minimum does not equal zero for any $s \in \{0, 1, \dots, N - 1\}$. Typically, however, it will decay quickly as a function of s around the optimal choice of s . As presented in Fig. 2, finding the solution to the optimization problems at different values of s typically allows one to obtain an estimate of an approximate σ , which in turn can be used to approximately retrieve the phases of f_1 .

In any case, the optimal assignment of phases $\mathbf{y}$ for a given s can be found by minimizing the function $Q^{(s)}(\mathbf{y})$ over all assignments of *phases* $\mathbf{y}$, which is not a convex optimization problem due to the constraints on the entries of $\mathbf{y}$ (i.e., that each entry has unit absolute value). To approximate the optimal solution in a practical setting, one will have to relax the nonconvex optimization over phase assignments. The relaxation that we will employ is the problem of optimizing $Q^{(s)}(\mathbf{y})$ over all vectors $\mathbf{y} \in \mathbb{C}^{2N}$ for which $\|\mathbf{y}\|_2 = \sqrt{2N}$, instead of optimization over $\mathbf{y}$ whose entries have unit absolute value [34]. Hence, we relax the problem to the problem determining the smallest eigenvalue of a matrix.

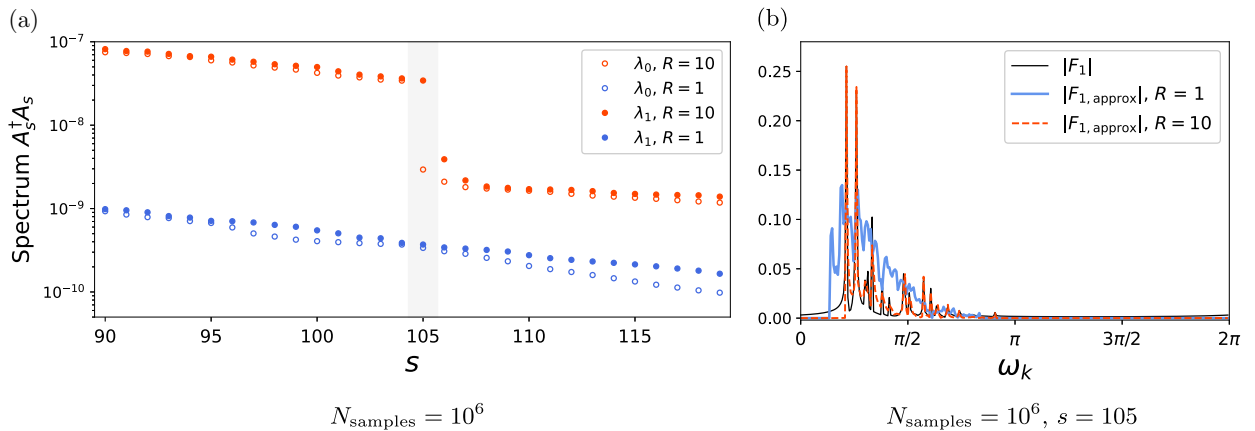


FIG. 2. (a) The smallest two eigenvalues of $A_s^\dagger A_s$ for $R = 1$ (blue) and $R = 10$ (red) in the noisy setting corresponding to $N_{\text{samples}} = 10^6$ as a function of s ; (b) Comparison between the reconstructed spectrum for $R = 1$ (blue) and $R = 10$ (red) versus the exact solution (black) (obtained by evaluating the smallest eigenvector of $A_s^\dagger A_s$ at $s = 105$). Note that in (a) it can be seen that, in the $R = 10$ scenario, the smallest eigenvalue drops rapidly around $s = 105$ while it is the opposite for $R = 1$. We show the spectrum on the range π to 2π , to validate there is no support between π to 2π , as required by condition (2) discussed at the start of Sec. II A.

Empirical investigation, illustrated in Fig. 2, has shown us that in order to make the phase-retrieval more resilient to noise, it is convenient to generalize the problem by including $R > 1$ *secondary* signals and their respective interferences with the target signal (as opposed to including just one such secondary signal). This leads to an equivalent expression for the cost function, now expressed in terms of $\tilde{A}_s \in \mathbb{C}^{((R+1)(N-s)+RN) \times (R+1)N}$ (where $\tilde{\cdot}$ is used to stress that noisy input data is used in the definition of $\tilde{A}_s$) and $\mathbf{y}_R \in \mathbb{C}^{(R+1)N}$. Employing an equivalent relaxation as the one just discussed, we optimize over all vectors $\mathbf{y} \in \mathbb{C}^{(R+1)N}$ for which $\|\mathbf{y}\|_2 = \sqrt{(R+1)N}$, corresponding to the following eigenvalue problem:

$$\min_{\mathbf{y} \in \mathbb{C}^{2N} \text{ s.t. } \|\mathbf{y}\|_2 = \sqrt{(R+1)N}} \tilde{\mathcal{Q}}^{(s)}(\mathbf{y}) = (R+1)N \lambda_{\min}(\tilde{A}_s^\dagger \tilde{A}_s). \quad (5)$$

Our numerical investigations suggest that for relatively small noise magnitudes, the vector $\mathbf{y}$ obtained by solving Eq. (5) will generally be close to the ideal solution. We do note that the entries of the optimal vector $\mathbf{y}$ are not necessarily phases (i.e., have unit absolute value), nonetheless, we approximate $\{f_1[j]\}_{j=0}^{N-1}$ by $\{|f_1[j]|y_j|\}_{j=0}^{N-1}$ because our numerical investigations suggest that this generally gives a more accurate reconstruction compared to the solution that is obtained when rounding to a vector whose entries are phases. For a more detailed discussion of this latter fact, we refer to Appendix B 3. We note that the sampling effort for running the $R > 1$ version of VPR scales linearly in R . How large R needs to be to obtain an accurate reconstruction depends on the problem instance and noise magnitude.

2. Implementation

To demonstrate the technique, we apply the vectorial phase retrieval framework to the state evolution signals generated by a (1×5) spinful Fermi-Hubbard Hamiltonian with $|U/\tau| = 4$. The target initial state is taken to be a (standard-basis) state at half filling, where we map to qubits under a Jordan-Wigner transformation. We consider a scenario in which the number of secondary states is one (i.e., $R = 1$), and a scenario in which we include ten secondary states (i.e., $R = 10$). The *secondary* states are created as follows. Take the target state and select at most $p < n/2$ (distinct) pairs of qubits that are unequally occupied, then flip the states on these qubits. We obtain the time series at $N = 300$ points in time (with $N_{\text{samples}} = 10^6$ shots per point in time), with a time increment $\Delta t = 0.133$.

Note that superpositions of the target state and a given secondary state can be produced using similar circuit to those generating a multiqubit GHZ state followed by local X gates, which requires $O(\log p)$ layers of two-qubit gates (given all-to-all connectivity). This is similar to a method that can be used to reduce the cost of controlled

time-evolution on hardware with restricted connectivity, or the depth on hardware with all-to-all connectivity (see Table I). While one should take this additional cost into account when comparing complexities, we believe that it is not a fundamental limitation of the method, as different secondary states could be used.

Figure 2(a) depicts the smallest two eigenvalues of the $\tilde{A}_s^\dagger \tilde{A}_s$ matrix, whose smallest eigenvector is used as an estimate for the assignment of phases of the time series entries in both scenarios. Our investigations suggest that

- (1) The two smallest eigenvalues are a proxy for the quality of the retrieval of phases. That is, the smaller the smallest eigenvalue, the smaller the value of the two residuals Eqs. (2) and (3) above.
- (2) A large gap opening between the two smallest eigenvalues suggests a larger overlap between the smallest eigenvector and the true assignment of phases. This being motivated by the fact that in the ideal noiseless case the solution is unique.

Typically, the smallest eigenvalue of $\tilde{A}_s^\dagger \tilde{A}_s$ drops rapidly as a function of s at a given value s^* [at $s = 105$ in Fig. 2(a)], which our numerical investigations suggests is an indicator of an accurate retrieval of phases around that value of s . Note that both features mentioned above are present in the $R = 10$ scenario, leading to a good reconstruction, but not in the $R = 1$ scenario, which does not provide an accurate reconstruction, as one can see in Fig. 2(b) (showing the associated estimate of the spectrum $|F_1|$ at $s = 105$).

More generally, our investigations suggest that the noise resilience of the vectorial phase retrieval algorithm is improved by taking $R > 1$. We note that increasing R increases linearly the amount of classical data to be processed and the number of quantum circuit shots required, but leads to a cubic scaling of the running time of the classical postprocessing algorithm (eigenvalue problem). The trade-off between resources cost and quality of solution will therefore play a role in the final choice of R . In addition, it is important to cleverly pick the secondary quantum states in a general setting, so that the circuit generating them and their superposition with the target state is a shallow circuit, so that it does not significantly increase the depth of the quantum circuit needed for the implementation.

B. Two-dimensional phase-retrieval

It is well-known in the phase-retrieval literature that two dimensional problems are more resilient to nontrivial ambiguities of the solution [35,36]. The nontrivial ambiguities in the 1D problem arise from the fact that the z -transform of the signal is a univariate reducible polynomial. In contrast, reducible multivariate polynomials have measure zero, which implies that the phase

retrieval problem in the multidimensional case typically has a unique solution, unless disguising a single variable polynomial as a multivariable one. Nonetheless, this result strongly suggests that two-dimensional phase-retrieval should work in practice, an intuition confirmed by a broad range of applications in optical imaging and image processing that demonstrate its practical value [37,38].

We embed our problem—which is naturally one dimensional—into a larger 2D problem so that we can exploit the benefits of two-dimensional phase-retrieval. The full details of this method can be found in Appendix C, but in summary we introduce an additional dummy Hamiltonian H_D and a second independent time variable z . A new signal $f[j, l]$ can be defined from samples of

$$f(t, z) = \langle \psi | e^{itH} e^{izH_D} | \psi \rangle, \quad (6)$$

at the discrete set of times $t_j = j \Delta t$ and $z_l = l \Delta z$, where $\Delta t = T/N$ and $\Delta z = T'/M$, where for simplicity we choose $T' = T$ [39]. This let us define a matrix $f[j, l]$ whose entries can roughly be thought of as samples of this function at these discrete times, up to a multiplication by a windowing function and a specific ordering of these elements to ensure the spectrum being real (see Appendix C for details). In this work we apply one of the oldest and most widely used phase retrieval algorithms to this new 2D problem, the HIO proposed by Fienup [20] in 1982. HIO is known to perform well when $F[k, m]$ —the DFT of $f[j, l]$ —is real and positive. This is because this knowledge can be used to drive the HIO algorithm. These constraints can be imposed on the problem very naturally, as we detail in Appendix C:

- (1) Positivity can be ensured by choosing a H_D which commutes with H , and by multiplying the time-series by a filter that has a positive discrete Fourier transform. For simplicity in our implementation we

choose a triangular window and multiply by this to form a discrete array $f[j, l]$.

- (2) $F[k, l]$ is real if we can define $f[j, l]$ so that it satisfies $f[j, l] = f^*[N-j, M-l]$, as any signal that satisfies this condition has a real DFT. We achieve this by exploiting the fact that $f(t, z) = f^*(-t, -z)$ and including samples at both (t_j, z_l) and $(-t_j, -z_l)$ when defining $f[j, l]$ (see Sec. C).

We can then incorporate this additional knowledge about the solution into the phase retrieval algorithm. HIO also makes it relatively simple to add additional constraints exploiting symmetries of the problems or partial information we may have on the solution, as we show below for a specific example. In this work we have only explored the simplest windowing function which ensures the positivity constraint is satisfied. There is an extensive signal processing literature on filtering [40] and the optimum choice of windowing function remains an open question.

As schematized in Fig. 3, Fienup's HIO [20] involves transforming back and forth between the Fourier (left) and object domain (right) interspersed by projections that guarantee the satisfaction of the time-series values $|f[j, l]|$ in the object domain and approximately guarantee the $F[k, m]$ constraints in the Fourier domain. The algorithm is non-linear and of an iterative nature, where the constraint on the spectrum defines a new element F^{i+1} with the update rule shown at the left side of Fig. 3, where the parameter $0 \leq \beta \leq 1$ is carefully selected to optimize the performance. We will now summarize the details of a specific instance of this setup for the Fermi-Hubbard model.

1. Implementation

We test this algorithm numerically using a 2D spin Fermi-Hubbard model, with $U = 4$ and $\tau = 1$. For the dummy Hamiltonian H_D we use the total particle number

```

1: procedure HYBRID( $|f|, \beta, L, F^1$ )
2:   for  $i = 1, 2, \dots, L$  do
3:      $f^i = \mathcal{DFT}^{-1}(F^i) = |f^i| e^{i \arg(f^i)}$ 
4:      $\tilde{f}^i = |f| e^{i \arg(f^i)}$ 
5:      $\tilde{F}^i = \text{real}(\mathcal{DFT}(\tilde{f}^i))$ 
6:     if  $\tilde{F}^i[k, m] \leq 0$  then
7:        $F^{i+1}[k, m] = F^i[k, m] - \beta \tilde{F}^i[k, m]$ 
8:     else
9:        $F^{i+1}[k, m] = \tilde{F}^i[k, m]$ 
10:    end if
11:  end for
12: end procedure

```

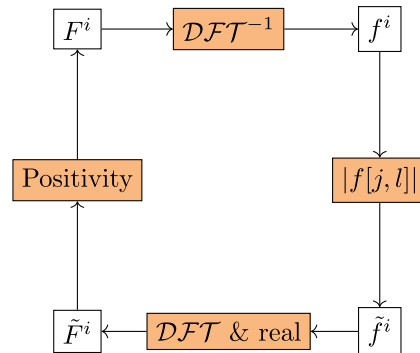


FIG. 3. Pseudocode (left) and schematic (right) for Fienup's hybrid input-output algorithm (HIO) in four steps. The i -th iteration starts with a candidate spectrum F^i . Step (1) transforms it into a time series f^i candidate via an inverse DFT; Step (2) generates a new $\tilde{f}^i$ that has same phase as f^i and satisfies $|\tilde{f}[j, l]| = |f[j, l]|$; Step (3) transforms it back to the Fourier domain and takes the real part; Step (4) implements the update rule in the schematic to impose a relaxed version of positivity of the spectrum.

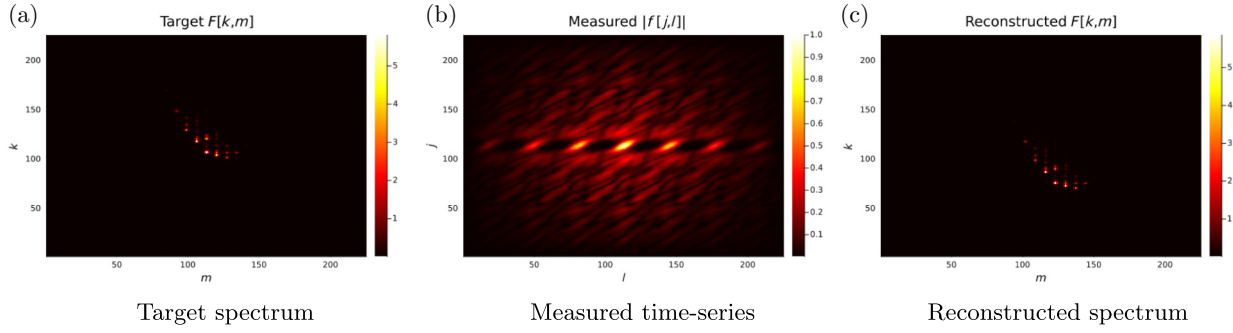


FIG. 4. These three plots show an example of 2D phase retrieval for a 2×2 Fermi-Hubbard Hamiltonian with $N \times M = 225 \times 225$. Here $T = 112$ and we take $|\psi\rangle$ to be a uniform superposition over computational basis states. H has been normalized so that its eigenvalues lie between 0 and π . We have chosen the H_D as the total number operator. This example excludes sampling noise. (a) The target 2D discretized spectrum $F[k, m]$ is shown here as a heatmap; (b) The absolute values $|f[j, l]|$ are shown here, these would be measured using only time-evolution and are the input to the hybrid input-output algorithm (HIO) phase-retrieval algorithm; (c) Here we show the result of using HIO on (b), we recover this reconstruction. Observe that the recovered spectrum contains a *trivial* ambiguity, in the form of a shift.

operator, which under the Jordan-Wigner encoding is simply a sum of single-qubit Pauli-Z operators. We use 2×2 spin Fermi-Hubbard model and set $N = M = 225$, time $T = 112$ and choose $|\psi\rangle$ to be the uniform superposition over computational basis states. H has been normalized so that its eigenvalues lie between 0 and π . Additionally we incorporate knowledge of the phases of $f[0, l]$ as driving constraints, as they can be computed efficiently classically for our specific input state. These points correspond to evolution under H_D only, which can be performed with local gates in any potential experiment.

The first step is to use the HIO algorithm to reconstruct a 2D image, summarized in Fig. 4. The left side of Fig. 4 shows the target 2D-spectrum $F[k, m]$, which we obtain from a direct numerical calculation of the time-series $f[j, l]$ including its phase. In the center we show the absolute values of $|f[j, l]|$, these would be measured using only time-evolution as sketched in Fig. 1(b), and form the input to the HIO algorithm. Observe that we also see the effect of the 2D triangular windowing function, the magnitude of $|f[j, l]|$ decreases as we approach the edge of the plot [41]. Finally to the right we show the reconstruction of $F[k, m]$ using the HIO algorithm with $\beta = 0.9$ and $L = 5000$. We can observe a trivial ambiguity in this reconstruction, the final spectrum is translated slightly. Nonetheless, the reconstruction is correct up to trivial ambiguities which we correct for when plotting.

Finally, we recover the 1D information following this image reconstruction. To do this we apply a DFT to the recovered signal $f[j, 0]$, which corresponds to the time-series of H without the dummy Hamiltonian. The result of this is shown in Fig. 5. We also model the effect of sample noise on this algorithm and find that for a modest number of samples—1000 per calculation of $|f[j, l]|$ —the algorithm can still reconstruct the location of the peaks.

C. Comparative Overview of Proposed Methods

We are now ready to compare the performance of these two approaches in the context of spectral estimation of the Fermi-Hubbard model. We choose a 3×3 instance of the (spinful) Fermi-Hubbard model (with $|U/\tau| = 4$) and set up the following task. The target spectrum is the ideal solution obtained from the DFT of the ideal time series $f[j] = \langle \psi | e^{ijH\Delta t} | \psi \rangle$ for time step $\Delta t = 0.12$ and maximum time evolution of $T = 15$, where the time series is obtained using brute-force calculation of $e^{ijH\Delta t}$ (for varying j), with the input state being

$$|\psi\rangle = \frac{1}{\sqrt{3}}(|010101010\ 010101010\rangle + |110101100\ 010100110\rangle + |011101010\ 000110111\rangle), \quad (7)$$

where the first (last) 9 bits are associated with the occupation of the spin-up (spin-down) modes. This state was chosen as it is suitable for our implementations of the vectorial phase retrieval algorithm and the two-dimensional phase retrieval algorithm. It does not provide an unfair advantage to either adaption and—importantly—it is not an eigenstate of H_D .

To get as close as possible to a fair comparison of both techniques we allocate a budget of $N_S = 5 \times 10^9$ circuit runs to our simulation of each phase retrieval technique. This directly limits the number of circuit runs required for each technique to retrieve each $|f[j]| = |\langle \psi | e^{ijH\Delta t} | \psi \rangle|$ which translates into shot noise on each value, which can later affect the performance of the phase-retrieval reconstruction. More concretely, we model the effect of shot noise induced by the finite number of samples during the statistical estimation of the time-series by sampling from a

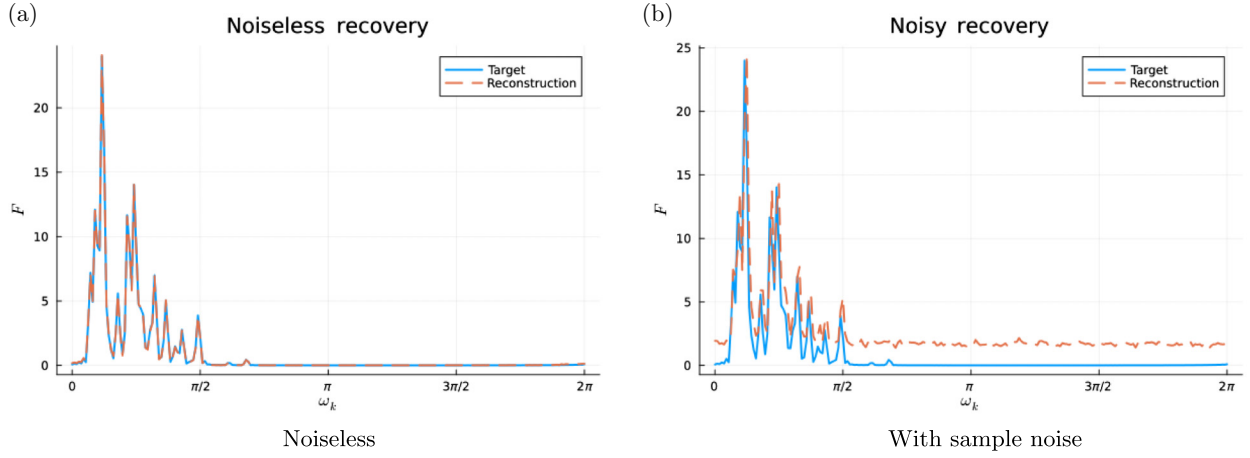


FIG. 5. This figure shows the recovered 1D spectrum following the image reconstruction in Fig. 4. (a) This shows $F[k] := \mathcal{DFT}[f[j, 0]]$ where f is the inverse discrete Fourier transform of the reconstructed spectrum shown in [Fig. 4(c)]. This is in the noiseless scenario. Note that the ideal solution includes the triangular window by definition. (a) This figure shows the effect of sample noise on this same problem. In this case we use only 1000 samples per time-series point and still recover the peaks but with a vertical offset in the reconstructed spectrum.

binomial distribution with success probability $|f[j]|^2$ (see Appendix A 3).

We chose to set up a comparison in this fashion as both methods have a variety of parameters which can be altered to ensure quality performance in the presence of sample noise. By fixing only the total number of runs we can compare the performance of both methods on the same footing. For example, with vectorial phase-retrieval there is now a trade-off between the number of additional secondary states R permitted and the total number of samples used for obtain an estimate of each $|f[j]|$. The same is true for the two-dimensional phase-retrieval

implementation and the parameter M . The optimum setting for each implementation remains an open question and depends on the particular problem under consideration. In this work we tried to optimize the choices of parameters for both approaches trying to retrieve a spectrum as close to the target one as possible.

The phase-recovery is followed by a DFT to reconstruct the Fermi-Hubbard Hamiltonian on a lattice described above, depicted in Fig. 6 for comparison.

For the 2D implementation we took $M = 25$ and used 1.6×10^6 samples per $f[j, l]$. In conjunction with $\beta = 0.9$ and $L = 5000$ iterations the HIO algorithm produces the

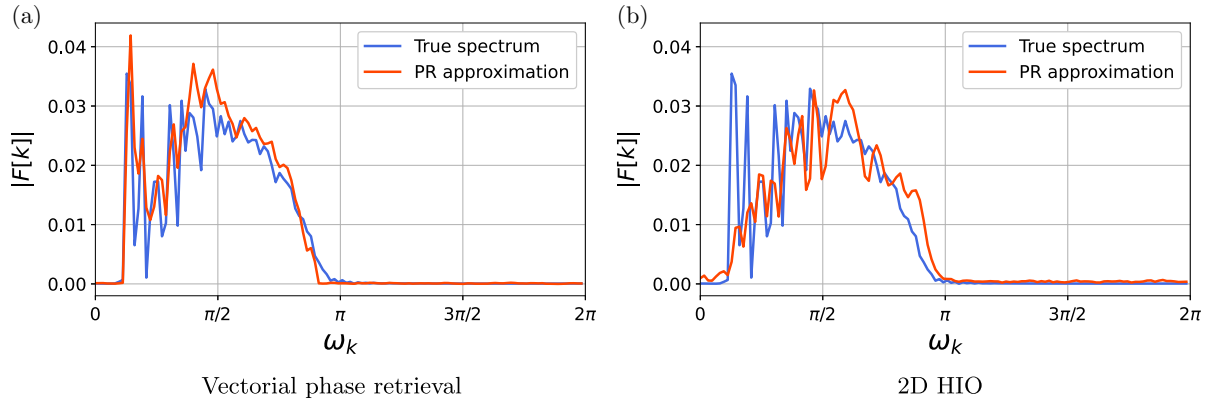


FIG. 6. This figure compares the two techniques on the same problem with the same sample budget of 5×10^9 . We target the spectrum given the input state $|\psi\rangle$ in Eq. (7) for the Fermi-Hubbard Hamiltonian H on a lattice of size 3×3 with on-site interaction strength $U = 4$ and hopping strength $\tau = 1$, with time step $\Delta t = 0.12$ and maximum time evolution of $T = 15$. In both plots the ideal solution is obtained from $f[j] = \langle \psi | e^{ijH\Delta t} | \psi \rangle$ and shown in blue; (a) Here we show the estimated spectrum (red) for vectorial phase-retrieval with $R = 40$ and 6×10^5 shots per point in time j per signal (we have taken s , the support size estimate, to be 49); (b) Here we show the estimate spectrum (red) for two-dimensional phase-retrieval with $M = 25$, $\beta = 0.9$ and $L = 5000$ and 1.6×10^6 shots per $f[j, l]$.

spectrum shown in Fig. 6. For the vectorial phase retrieval implementation, we used $R = 40$ secondary states, and 6×10^5 samples per signal per point in time j . Let us stress again that the *total* number of circuit runs $N_S = 5 \times 10^9$ is taken to be equal for both methods, to ensure a fair comparison. We observe that the 2D HIO method gives a poorer reconstruction in comparison to VPR. In particular, the HIO algorithm struggles to recover the amplitudes of the peaks. One can also check that the 1-norm (normalized by N) of the vector with entries $|F_{\text{reconstructed}}[k]| - |F_{\text{exact}}[k]|$ is 0.0018 for VPR and 0.0028 for 2DPR, confirming the intuition from a visual inspection of Fig. 6. Another observation is that the peaks recovered using 2D HIO appear slightly broadened (which generally leads to a lower resolution on the recovered spectrum), due to the windowing function we use to enforce the positivity constraint we use to drive HIO.

The accurate retrieval of the spectrum using the vectorial phase retrieval algorithm can be attributed to the large number of secondary states employed in the algorithm, leading improved noise resilience (as is outlined in Sec. II A). Since the vectorial phase retrieval algorithm performs better as the support of the spectrum becomes well-defined, there might be instances of spectra with less well-defined support for which vectorial phase retrieval performs worse. We remark that when HIO is observed to perform well in the noiseless scenario—such as when the peaks are well defined as in Fig. 5(a)—we observe that the corresponding noisy spectrum can typically be reconstructed well with a comparatively low number of samples as in Fig. 5(b). For the problem instance explored in Fig. 6, HIO does not recover the spectrum as well as VPR in the noiseless setting and this gap in performance persists in the noisy setting as shown in Fig. 6. We stress that the poorer performance of two-dimensional phase-retrieval in Fig. 6 is not necessarily a fundamental fact and one could potentially adapt more advanced modern two-dimensional phase retrieval methods to QPE problem while also exploiting other commuting Hamiltonians H_D that could potentially lead to better results.

The restriction to an 18-mode Fermi-Hubbard instance is inherent to our numerical modeling demonstration, but we expect that the phase-retrieval algorithms can be applied to system sizes beyond classically tractable ones and can be applied to experimental data from actual quantum hardware. The two phase-retrieval techniques are presented in full detail in Appendixes B and C. Finally, we believe further work could generalize the current framework to more general Loschmidt echo quantities, but also linear response of quantum systems, operator-resolved density of states or Green functions.

III. CONCLUSION AND DISCUSSION

We have demonstrated how to design phase-retrieval algorithms for the problem of statistical phase estimation.

These algorithms allow for the reconstruction of the phases of the (time evolution) time series of a quantum state $|\psi\rangle$ with respect to a given Hamiltonian H , while only having access to their absolute values. To achieve this seemingly impossible goal, one needs to address a more general problem involving the absolute values of a larger set of time series, resulting from increasing the set of input states or enlarging the range of time-evolutions available to us. To reach our goal we have adapted two well-known phase retrieval algorithms, vectorial phase retrieval and two-dimensional phase retrieval, which we demonstrate offer good performance in retrieving the spectrum of a Fermi-Hubbard model.

In the vectorial phase-retrieval scenario, we obtain absolute values of time series corresponding to time evolution of multiple input states, but—more importantly—time evolution of their superpositions. We also showed that despite QPE being a one-dimensional signal processing problem, one can exploit the benefits of two-dimensional phase-retrieval by adding a quench evolution governed by a dummy Hamiltonian H_D commuting with H . Despite the two techniques being rather different in design and post-processing algorithms, there are some analogies in their adaptation to QPE. One could see the time-evolution with a dummy Hamiltonian H_D as an alternative way of generating a nontrivial family of time series in addition to the one of the target input state $|\psi\rangle$, in a similar fashion as vectorial phase retrieval exploits a family of multiple input states and their superpositions. Our methods require some tailoring to the Hamiltonian at hand, such as a choice for the target and the secondary states for VPR such that the spectra have (approximately) well-defined support, or the choice of a dummy Hamiltonian for 2DPR. However, we have no reason to believe that our techniques only work for the Fermi-Hubbard model, and have merely chosen it because of its practical relevance.

We demonstrated that the performance of the algorithms is resilient to shot noise affecting the estimation of the time-series resulting from the limited number of accessible circuit runs. This is sufficient to analyse the nontrivial recovery performance of phase-retrieval. Further investigation of the subtle impact of circuit imperfections on the comparison between statistical phase-estimation and phase-retrieval is left to future work. Providing theoretical guarantees and asymptotic scaling of the different resource cost in realistic scenarios is an extremely complex task for the algorithms proposed here, specially for HIO. Nonetheless, it would be interesting to obtain heuristic asymptotic scalings, but this will need to use time-series resulting from time-evolutions that can be scaled numerically on classical computers, like free-fermions or 1D TFIM.

The fine tuning of the different phase-retrieval techniques to a specific spectrum of a given quantum state $|\psi\rangle$ and Hamiltonian H currently needs a handcrafted approach. It would be interesting to design techniques that can

automatize the procedure for arbitrary states and Hamiltonians. We note that despite the Hybrid Input-Output algorithm used in two dimensional phase-retrieval working well in-practice, it is highly probable that relaxing the problem to a convex optimization would bring additional benefits, from convergence guarantees, to better running times, scaling and potentially better resilience to noise. We leave this question open for future work. Even further, one could dream of having a unified framework that given a quantum state and Hamiltonian suggest the better phase-retrieval algorithm, ideally base on convex optimization tools that are known to be most resilient to noise.

We stress that phase retrieval algorithms recover the phases of the state-evolution time series, of which one learns just the absolute values in the scenario of control-free QPE. In some portions of this work, for convenience, we compare the discrete Fourier transform of the retrieved time series to that of the true time series to judge the accuracy of the phase retrieval. However, we note that the discrete Fourier transform is not necessarily the best way to recover the spectrum of a Hamiltonian from its time-series. Typically, classical postprocessing techniques like the one developed in Ref. [5] can be used to obtain a more accurate and useful description of the spectrum.

We observe that although our phase retrieval algorithms can achieve nontrivial reductions in quantum circuit complexity in many cases, whether this is useful in practice requires consideration of a series of trade-off. Firstly, there is a subtle cost comparison to do between the additional cost of the controlled time-evolution in statistical phase-estimation and the additional GHZ-like state input state required for vectorial phase-retrieval or the additional time-dynamics of the dummy Hamiltonian in 2D phase-retrieval. In the particular case of vectorial phase-retrieval vs statistical phase-estimation on 2D architectures, both required to build a GHZ-resource state, the difference is that the second one needs to be as large as the system, where the first only a fraction of it, which implies a saving of resources. An interesting question we leave open is whether the GHZ resource required for vectorial phase-retrieval could be made of constant cost, not scaling with system size N . Secondly, there is a trade-off between circuit complexity and number of shots required, since our algorithms generally increase the latter. It is an interesting question whether the increased number of shots required can be significantly reduced.

There is a wealth of advanced techniques in the classical phase retrieval literature that could be explored and evaluated to improve the techniques that we present in this work further. We believe further improvement in circuit depth, number of shots and resilience to error and recovery can be achieved by further adapting more advanced ideas in signal processing to quantum information theory.

ACKNOWLEDGMENTS

This work was supported by Innovate UK (Grant No. 44167). A.M. acknowledges funding from the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (Grant Agreement No. 817581). We thank an anonymous referee for useful comments on the relation between target and secondary states in vectorial phase retrieval.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available upon publication because it is not technically feasible and/or the cost of preparing, depositing, and hosting the data would be prohibitive within the terms of this research project. The data are available from the authors upon reasonable request.

APPENDIX A: TECHNICAL PRELIMINARIES

In this appendix we first introduce basics concepts of statistical phase-estimation and signal processing that will be of particular interest to readers not familiar with phase-estimation and its signal-processing material. Secondly, we give details on the Fermi-Hubbard Hamiltonian that we use as a benchmark and few technicalities related to our simulations.

1. Definitions

In what follows we adopt the following continuous Fourier transform convention

$$\begin{aligned} F(\omega) &= \mathcal{FT}[f] := \int_{-\infty}^{\infty} f(t) e^{-i\omega t} dt \\ f(t) &= \mathcal{FT}^{-1}[F] := \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega, \end{aligned} \quad (\text{A1})$$

and the discrete Fourier Transform definition as

$$\begin{aligned} F[k] &= \mathcal{DFT}[f] := \sum_{j=0}^{N-1} f[j] e^{-i2\pi \frac{kj}{N}} \\ f[j] &= \mathcal{DFT}^{-1}[F] := \frac{1}{N} \sum_{k=0}^{N-1} F[k] e^{i2\pi \frac{kj}{N}}. \end{aligned} \quad (\text{A2})$$

For a spectrum $F(\omega)$ to be a real function it needs to satisfy $F(\omega) = F^*(\omega)$ for all ω . It is easy to check from the definitions above that this is only possible if

$$F(\omega) - F^*(\omega) = \int [f(t) - f^*(-t)] e^{i\omega t} dt = 0, \quad (\text{A3})$$

which holds if and only if $f(t) = f^*(-t)$.

Let us consider an Hamiltonian $H = \sum_i E_i \Pi_i$, where E_i is the i th eigenvalue of the Hamiltonian and Π_i is the projector into its eigenspace. In the case of a rank 1 projector we have $\Pi_i = |E_i\rangle \langle E_i|$. Its time evolution reads $e^{iHt} = \sum_i e^{iE_i t} \Pi_i$ and $\text{Tr}[\rho \Pi_i] = p_i$ is the amplitude of spectrum of ρ at energy E_i , where the full spectrum of ρ can be written in a compact form as $F(\omega) = 2\pi \sum_i p_i \delta(\omega - E_i)$, where the positivity of p_i implies that of the spectrum $F(\omega)$. The Hadamard test in Fig. 1(a) used in statistical phase-estimation allows the computation of the complex amplitude

$$f(t) = \text{Tr}[\rho e^{iHt}] \quad (\text{A4})$$

for different times t , i.e., time series. It is easy to see that $f(t) = \text{Tr}[\rho e^{iHt}] = f^*(-t)$, due to ρ being hermitian and the cyclic property of the trace, which proves that the spectrum is real. We will now show that it is also positive in the ideal case.

a. Positivity of the spectrum

Applying the Fourier transform we obtain

$$\mathcal{FT}[f(t)] = \int \text{Tr}[\rho \sum_i e^{iE_i t} \Pi_i] e^{-i\omega t} dt,$$

which using linearity of the trace and some rearrangements leads to

$$\mathcal{FT}[f(t)] = \sum_i \text{Tr}[\rho \Pi_i] \int e^{-i(\omega - E_i)t} dt,$$

which using the definition of the amplitudes of the spectrum and $\int e^{-i(\omega - E_i)t} dt = 2\pi \delta(\omega - E_i)$ leads to

$$\mathcal{FT}[f(t)] = 2\pi \sum_i p_i \delta(\omega - E_i) = F(\omega).$$

In what follows we assume the Hamiltonian eigenvalues E_i satisfy the condition $E_i \in [0, 2\pi]$. Otherwise, assuming the Hamiltonian maximum and minimum eigenvalue energies, or providing a decent upper and lower bounds, one can always rescale the Hamiltonian accordingly.

2. Basics concepts on signal processing and Fourier transforms

a. Time-window and resolution

In practice continuous signals $f(t)$ can only be probed inside a finite time window $[-T/2, T/2]$. The simplest

windowing $\hat{f}(t) = f(t) \Pi_T(t)$ results from the multiplication of the original signal $f(t)$ and a time-window function,

$$\Pi_T(t) = \begin{cases} 1, & |t| \leq T/2, \\ 0, & |t| > T/2. \end{cases} \quad (\text{A5})$$

Using the convolution theorem, relating the Fourier transform of a product to the convolution of its respective Fourier transforms ($\mathcal{FT}[A \cdot B] = \mathcal{FT}[A] * \mathcal{FT}[B]$), together with the Fourier transforms of the step function,

$$\mathcal{FT}[\Pi_T(t)](\omega) = \frac{2 \sin(\frac{T\omega}{2})}{\omega} = T \text{sinc}(T\omega/2), \quad (\text{A6})$$

where we define $\mathcal{FT}[f(t)](\omega) = F(\omega)$ leads to

$$\mathcal{FT}[\hat{f}(t)](\omega) = F(\omega) * T \text{sinc}(T\omega/2). \quad (\text{A7})$$

We will use the notation $\hat{F}(\omega)$ for the convolution of $= F[\omega] * \text{sinc}(T\omega)$, resulting from the broadening of the spectral peaks due to the finite-size time window.

a. *Resolution.* The convolution $\hat{F}[\omega] = F[\omega] * \text{sinc}(T\omega/2)$ generally leads to a broadening of the spectral peaks of $F(\omega)$. The main lobe width of $\text{sinc}(T\omega/2)$ being $4\pi/T$, it is easy to see that a time window of size T can generally only resolve energy gaps of order $O(1/T)$. Therefore to have full resolution of the spectrum one needs to do time-evolution up to T inverse proportional to the smallest gap between eigenvalues in the spectrum.

The choice of T gives an upper-bound on how good the quality is of the recovered spectrum, i.e., even under ideal reconstruction conditions one can at best retrieve $= F[\omega] * \text{sinc}(T\omega)$. For Hamiltonians with gaps scaling beyond $1/\text{poly}(n)$, it is impossible to reconstruct the exact spectrum and retrieving the coarser version $\hat{F}[\omega]$ will be the only option.

b. *Alternative time-windows.* Remark that despite the ideal spectrum $F(\omega)$ being positive the convolution with a sinc function, resulting from the finite-size time window, may lead to a spectrum that has some slightly negativities. In our 2D phase-retrieval algorithm this could be an issue for the convergence of our algorithm. We will therefore use a different time window that will ensure the target spectrum in our reconstruction being all positive. The simplest way to achieve this is via a triangle time-window, resulting from the convolution of two rectangular time-windows $\Pi_T(t)$, which has a squared sinc function as Fourier transform. Unfortunately, this doubles the width of the main lobe of its Fourier spectrum further widening the recovered spectrum. There is a whole literature on optimized filters that would allow to limit the widening of the spectrum

while still guaranteeing its positivity, we leave that for future work.

b. Sampling and aliasing

In practice continuous signals $f(t)$ can only be probed inside a finite time window $[-T/2, T/2]$ and for a finite number N of time values. We will assume these samples are evenly spaced with time Δt . Note that in this discussion we always choose an odd number of samples N and set the spacing to be $\Delta t = T/N$. We have odd time values as $t = 0$ is also included and we are probing up to time $t_{N-1} = (N-1)/N \times T/2$. The discretized sampling over a finite range is modeled via the multiplication of the initial signal $\hat{f}(t)$ by a Dirac comb,

$$\text{III}_{\Delta t}(t) = \sum_{j \in \mathbb{Z}} \delta(t - j \Delta t), \quad (\text{A8})$$

leading to the new discretized and finite signal

$$g(t) = f(t) \Pi_T(t) \text{III}_{\Delta t}(t). \quad (\text{A9})$$

Using the convolution theorem together with the Fourier transforms of the Dirac comb

$$\begin{aligned} \mathcal{FT}[\text{III}_{\Delta t}(t)](\omega) &= \frac{2\pi}{\Delta t} \text{III}_{2\pi/\Delta t}(\omega) \\ &= \frac{2\pi}{\Delta t} \sum_{k \in \mathbb{Z}} \delta\left(\omega - k \frac{2\pi}{\Delta t}\right), \end{aligned} \quad (\text{A10})$$

together with Eq. (A7), we obtain

$$\begin{aligned} G(\omega) &:= \mathcal{FT}[g(t)](\omega) \\ &= 2\pi \frac{T}{\Delta t} \sum_{k \in \mathbb{Z}} F(\omega) * \text{sinc}(T\omega/2) * \delta\left(\omega - k \frac{2\pi}{\Delta t}\right). \end{aligned} \quad (\text{A11})$$

One can simplify the previous equation using the properties of delta functions and their convolutions and using the definition $\hat{F}(\omega) = F(\omega) * T \text{sinc}(T\omega/2)$ to get

$$G(\omega) = \frac{2\pi}{\Delta t} \sum_{k \in \mathbb{Z}} \hat{F}\left(\omega - k \frac{2\pi}{\Delta t}\right). \quad (\text{A12})$$

a. Aliasing. Due to the discrete-time nature of the sampling, the spectrum $\mathcal{FT}[g(t)](\omega)$ becomes periodic function of period $2\pi/\Delta t$. The celebrated Nyquist-Shannon theorem states that sampling at twice the bandwidth of the signal guarantees an unique recovery of the continuous function. If one samples at a lower rate, the weights of higher frequency lines are shifted to the lower part of the spectrum, an effect called aliasing. In an ideal scenario

(and unrealistic) scenario of an infinite time-window and the spectrum of H having largest absolute value frequency $\omega_{\max} = \max\{|\omega|\}$, it would be sufficient to guarantee the condition $2\Delta t \cdot \omega_{\max} < 1$ to have perfect reconstruction and zero aliasing. In practice, depending on the choice of time-window and the finite value of T we always observe some amount of aliasing, but a good choice of sampling rate, time-window shape and having access to large enough T will help make aliasing small or even negligible.

c. Connecting sampled time-series and discretized spectrum via a DFT

The algorithms we use exploit discretized version of time-series and spectrum connected via a DFT. In what follows we review this connection where the discretized spectrum can be understood from a periodic extension of the time-series.

We choose as set of discrete frequencies $\omega_k = 2\pi k/T$ where $k = 0, 1, \dots, N-1$ and define a length N vector $F[k] := \mathcal{FT}[g(t)](\omega_k) = G(\omega_k)$, whose entries are samples of the continuous time Fourier transform of $g(t) = f(t) \Pi_T(t) \text{III}_{\Delta t}(t)$ at points ω_k . We begin by writing the Fourier transform of our sampled function as

$$F[k] := \mathcal{FT}[g(t)](\omega_k) \quad (\text{A13})$$

$$= \int_{-\infty}^{\infty} f(t) \Pi_T(t) \text{III}_{\Delta t}(t) e^{-i\omega_k t} dt, \quad (\text{A14})$$

where using the definition of the Dirac comb in Eq. (A8) and $\int f(t) \delta(x-a) dt = f(a)$ we obtain

$$F[k] = \sum_{j=-\infty}^{\infty} f(j \Delta t) \cdot \Pi(j \Delta t) e^{-i\omega_k j \Delta t} \quad (\text{A15})$$

which using the discretization notation $\hat{f}[j] := f(j \Delta t) \Pi(j \Delta t)$ and the fact that $\omega_k j \Delta t = 2\pi j k / N$ leads to

$$F[k] = \sum_{j=-\infty}^{\infty} \hat{f}[j] e^{-2i\pi k j / N}. \quad (\text{A16})$$

The connection between the DFT and samples of the continuous functions at ω_k and $t_j = j \Delta t$ is almost complete. We have obtained the conventional DFT frequencies, however, we have an infinite sum in the above expression.

We deal with this as follows. One can replace the sum over integers by the following double sum:

$$F[k] = \sum_{j' \in \mathbb{Z}} \sum_{j=0}^{N-1} \hat{f}[j - j'N] e^{-i(2\pi kj/N + 2\pi j'k)} \quad (\text{A17})$$

$$= \sum_{j=0}^{N-1} \left(\sum_{j' \in \mathbb{Z}} \hat{f}[j - j'N] \right) e^{-i2\pi kj/N}, \quad (\text{A18})$$

then using the definition

$$f[j] := \sum_{j' \in \mathbb{Z}} \hat{f}[j - j'N] \quad (\text{A19})$$

leads to

$$F[k] = \sum_{j=0}^{N-1} f[j] e^{-i2\pi kj/N}. \quad (\text{A20})$$

This shows that $F[k]$ is actually the DFT of $f[j]$, which is simply an N periodic version of the windowed samples $\hat{f}[j]$.

Notice that $f[j]$ is a periodic sequence where the values within the window are repeated with period N . The previous exposition provides an interpretation of the discretized time-series linked to a discrete spectrum via the DFT, in a similar fashion as occurred for their continuous counterparts. Provided we properly choose T and Δt to avoid aliasing and limit leakage, we can safely work on a discrete setting using the DFT to move between the sampled time-series and a discretized version of the spectrum.

3. Fermi-Hubbard simulation and technicalities of our numerical experiments

In this paper, we benchmark our phase-retrieval algorithms for performing statistical QPE on spectrum recovery of the Fermi-Hubbard model.

a. Fermi-Hubbard. We will be interested in instances of the Fermi-Hubbard model over a square lattice graph G , where V is the set of vertices and E the edges, where each lattice site, i.e., vertex, has two spin modes $\sigma \in \{\uparrow, \downarrow\}$. The Hamiltonian reads

$$H = -\tau \sum_{\langle i,j \rangle \in E, \sigma} \left(a_{i\sigma}^\dagger a_{j\sigma} + a_{j\sigma}^\dagger a_{i\sigma} \right) + U \sum_{v \in V} n_{v\uparrow} n_{v\downarrow},$$

where the first term of the Hamiltonian consists of hopping terms among modes of same spin while the second sum corresponds to interactions between particles of opposite spin at the same lattice site. This model is of great

interest, since despite its simple form it demonstrates interesting phenomena found in more complex materials and molecular Hamiltonians [31, 42]. Throughout this work we will take $\tau = 1$ and $U = 4$, which is an intermediate coupling regime where the model exhibits nontrivial behavior [31]. All simulations in this work are performed on the qubit level, by applying a Jordan-Wigner transformation to the fermionic model.

b. Time-evolution simulation. In order to compute time-series of the form $|\langle \psi | e^{iHt} | \phi \rangle|^2$ (forming the input to our phase retrieval algorithms) and time-series $\langle \psi | e^{iHt} | \phi \rangle$ (which we use to compare to our phase-retrieved solution), one needs to simulate the time-evolution of the Fermi-Hubbard model e^{iHt} . We will be using numerical simulations to obtain the time-series, by brute-force calculation of e^{iHt_j} with $t_j = j \Delta t$. Then, for some states $|\psi\rangle$ and $|\phi\rangle$, we simply evaluate the expressions $f[j] = \langle \psi | e^{iHt_j} | \phi \rangle$ and $|f[j]|$. Due to the current limitation on near-term quantum computers, we will be interested to explore states that have shallow preparation circuits, as detailed in Appendixes B and C.

c. Sampling noise. A crucial aspect of the practical implementation of phase-retrieval algorithms is their resilience to noise. We model the sampling noise on time series $|\langle \phi | \exp(ij \Delta t H) | \psi \rangle|^2$ (for some $|\psi\rangle$ and $|\phi\rangle$) by sampling from a binomial random variable. In an experimental setting, one can estimate $|\langle 0^n | U_\phi^\dagger \exp(ij \Delta t H) U_\psi | 0^n \rangle|^2$ (where U_ψ and U_ϕ are state preparation unitaries s.t. $|\psi\rangle = U_\psi | 0 \rangle$ and $|\phi\rangle = U_\phi | 0 \rangle$) by acting with $U_\psi^\dagger \exp(ij \Delta t H) U_\psi$ on $|0^n\rangle$ and simply performing standard-basis projective measurements on each of the n qubits. The estimate is then given by the fraction of the number of times that output $|0\rangle^{\otimes n}$ is obtained and the total number of circuit executions M . Therefore, we model the sampling noise by estimating $|\langle \phi | \exp(ij \Delta t H) | \psi \rangle|^2$ by the number of successes M_0 (divided by M) obtained in a binomial random variable realization with success probability $|\langle \phi | \exp(ij \Delta t H) | \psi \rangle|^2$ and M trials.

d. Trivial ambiguities. We call operations that leave the absolute value of the time series unchanged trivial ambiguities. Throughout this paper, we ignore these trivial ambiguities and therefore match the recovered signal to the true time-series $f \in \mathbb{C}^N$ up to: (1) a global phase $e^{i\phi} f$; (2) linear phase shifts $\{e^{-i2\pi jm/N} f[j]\}$; (3) complex conjugation f^* , or combinations thereof. These respectively induce the following operations on the DFT of f : (1) multiplication by a global phase; (2) translation by m ; (2 combined with 3) reflection and complex conjugation. Note that a shift in the energies, provided that the support of the spectrum is bounded, can be ensured to not lead to confusion

in the ordering of the recovered peaks. The shape of the spectrum will be correct up to a constant shift. The final ambiguity can reflect the spectrum, an effect that can be mitigated if handled with proper care.

APPENDIX B: VECTORIAL PHASE RETRIEVAL

In this section we give a detailed presentation of vectorial phase retrieval (v-PR) (see Ref. [19]) and its adaptation to the quantum scenario, which we discussed briefly in Sec. II A. This problem corresponds to a version of the one-dimensional phase retrieval problem, where one assumes access to particular additional input in the form of interference measurements.

1. Vectorial phase retrieval with a single interference signal

Let us first discuss under which circumstances the (single-interference-signal) v-PR problem has a unique solution. This understanding of the uniqueness of the v-PR problem was used in Ref. [19] to develop a method for obtaining estimates of the phases of the time series $\mathbf{f} \in \mathbb{C}^N$.

In the v-PR framework, the (nontrivial) ambiguity of the one-dimensional phase retrieval problem is reduced by not only obtaining absolute value measurements $\{|f_1[j]|\}$ of a time series $\mathbf{f}_1 \in \mathbb{C}^N$, but also measurements $\{|f_2[j]|\}$ of some other time series $\mathbf{f}_2 \in \mathbb{C}^N$, and—crucially—of the absolute values of the time series corresponding to sums of $\mathbf{f}_1$ and $\mathbf{f}_2$. It was shown in Ref. [43] that under some relatively mild conditions on the time series $\mathbf{f}_1$ and $\mathbf{f}_2$, and assuming *exact* access to the aforementioned absolute value measurements, the resulting phase retrieval problem has a unique solution. The problem of *finding* this unique solution efficiently was addressed in Ref. [19].

Let us define the vectorial phase retrieval problem more exactly. Afterward, we will discuss under which circumstances the problem has a unique solution, and how one can in practice obtain an accurate estimate of the full time series using the vectorial phase retrieval algorithm [19].

Problem B1. [Vectorial phase retrieval (v-PR)]. Given measurements of the absolute values

$$\begin{aligned} |f_1[j]|^2, |f_2[j]|^2, |f_3[j]|^2 &:= |f_1[j] + f_2[j]|^2, \\ |f_4[j]|^2 &:= |f_1[j] + if_2[j]|^2, \end{aligned} \quad (\text{B1})$$

at times $j = 0, 1, \dots, N-1$, determine

$$\begin{aligned} F_1[k] &:= \sum_{j=0}^{N-1} f_1[j] \exp[-i2\pi jk/N], \\ F_2[k] &:= \sum_{j=0}^{N-1} f_2[j] \exp[-i2\pi jk/N], \end{aligned} \quad (\text{B2})$$

at $k = 0, 1, \dots, N-1$.

In the setting of control-free QPE, the input signals take the following form:

$$\begin{aligned} f_1[j] &= \langle \Phi | \exp(i\Delta t H_j) | \Phi \rangle, \\ f_2[j] &= \langle \Phi | \exp(i\Delta t H_j) | \psi \rangle, \\ f_3[j] &= \langle \Phi | \exp(i\Delta t H_j) (|\Phi\rangle + |\psi\rangle), \\ f_4[j] &= \langle \Phi | \exp(i\Delta t H_j) (|\Phi\rangle + i|\psi\rangle), \end{aligned} \quad (\text{B3})$$

for our states $|\Phi\rangle$ of interest and a secondary state $|\psi\rangle$. We will come back to which choices of $|\psi\rangle$ are appropriate, but we note that preparing their superposition ideally should not require execution of very deep circuits.

a. The noiseless scenario

Let us first address the setting in which one has access to the *noiseless* absolute value measurements as input to the v-PR problem. At the core of the vectorial phase retrieval algorithm is the understanding of when Problem B1 has a unique solution. To illustrate when uniqueness is guaranteed, let us introduce the following notions.

The DFTs $\mathbf{F}_1 \in \mathbb{C}^N$ and $\mathbf{F}_2 \in \mathbb{C}^N$ of the time-series $\mathbf{f}_1$ and $\mathbf{f}_2$ are spectrally independent if $P_1(z) := \sum_{k=0}^{N-1} F_1[k] z^k$ (with $z \in \mathbb{C}$) and $P_2(z) := \sum_{k=0}^{N-1} F_2[k] z^k$ (with $z \in \mathbb{C}$) have no common roots in the complex plane.

The DFTs $\mathbf{F}_1 \in \mathbb{C}^N$ and $\mathbf{F}_2 \in \mathbb{C}^N$ of the time-series $\mathbf{f}_1$ and $\mathbf{f}_2$ have well-defined support if they are such that there exist $k_{\min} < k_{\max} \in \{0, 1, \dots, N-1\}$ s.t. $k_{\max} - k_{\min} = \sigma \leq \lfloor N/2 \rfloor$ and $|F_1[k]|, |F_2[k]| \neq 0$ for $k_{\min} \leq k \leq k_{\max}$ and $|F_1[k]| = |F_2[k]| = 0$ for $k < k_{\min}$ and $k > k_{\max}$. In other words, they have well-defined support if they are nonzero inside of some interval $(k_{\min}, \dots, k_{\max})$ (of length $\sigma \leq \lfloor N/2 \rfloor$), and zero outside of that interval.

Especially the latter notion will play a central role in the vectorial phase retrieval algorithm, and in particular in our application of the algorithm to spectral estimation of Hamiltonians. The uniqueness of the solution of the vectorial phase retrieval problem is related to these notions in the following way.

The vectorial phase retrieval problem has a unique solution if and only if $\mathbf{F}_1$ and $\mathbf{F}_2$ are spectrally independent and have well-defined support [19].

See Ref. [19] for a proof of this statement. Note that Problem B1 has trivial ambiguities. The trivial ambiguities correspond in general to all signals related to $(\mathbf{f}_1, \mathbf{f}_2)$ by

$$\begin{aligned} &\{\exp(i\alpha) F_1[k], \exp(i\alpha) F_2[k]\}_{k=0}^{N-1} \quad (\text{with } \alpha \in \mathbb{R}), \\ &\{F_1[k - k_0], F_2[k - k_0]\}_{k=0}^{N-1} \quad (\text{with } k_0 \in \mathbb{Z}), \\ &\{F_1^*[-k], F_2^*[-k]\}_{k=0}^{N-1}, \end{aligned} \quad (\text{B4})$$

or combinations thereof. Solutions to the vectorial phase retrieval problem will only ever be unique up to these trivial ambiguities.

The vectorial phase retrieval algorithm (presented in Ref. [19]) is based on the fact that the unique solution to Problem B1 can be obtained by solving a convex optimization problem in the ideal noiseless scenario. The optimization procedure optimizes over assignments of phases to the time series measurements $\{|f_1[j]|\}$ and $\{|f_2[j]|\}$ such that they are consistent with the interference measurements and such that $\mathbf{F}_1$ and $\mathbf{F}_2$ have the right support size.

The *input* to the v-PR algorithm consists of $|f_1[j]|^2$, $|f_2[j]|^2$, $|f_3[j]|^2$, and $|f_4[j]|^2$ (at $j = 0, 1, \dots, N-1$) and of an estimate s of the support size σ of $\mathbf{F}_1$ and $\mathbf{F}_2$. Its *output* is given by a vector $\mathbf{y} \in \mathbb{C}^{2N}$, whose entries are the estimates of the phases of $f_1[j]$ and $f_2[j]$ at $j = 0, 1, \dots, N-1$. The algorithm is an optimization procedure that minimizes a cost function over possible assignments of phases $\mathbf{y} \in \mathbb{C}^{2N}$. Let us denote the vector containing the correct phases of $\mathbf{f}_1$ and $\mathbf{f}_2$ at $j = 0, 1, \dots, N-1$ by $\mathbf{x}$. The cost function has two components; one component ensures that the assignment of phases is consistent with the interference measurements, and the other component ensures that $\mathbf{F}_1$ and $\mathbf{F}_2$ are indeed zero outside of the interval of size s (which is an estimate of σ). These components are denoted by $Q_{\text{support}}^{(s)}(\mathbf{y})$ and $Q_{\text{interference}}(\mathbf{y})$, respectively. Note that due to the trivial ambiguities, minimization of $|F_1[k]|$ and $|F_2[k]|$ outside of their supported interval can be realized by simply minimizing $|F_1[k]|$ and $|F_2[k]|$ at $k = s, s+1, \dots, N-1$ without loss of generality.

The expression for the support component of the cost function is as follows:

$$\begin{aligned} Q_{\text{support}}^{(s)}(\mathbf{y}) &:= \sum_{k=s}^{N-1} |F_1[k]|^2 + \sum_{k=s}^{N-1} |F_2[k]|^2 \\ &= \sum_{k=s}^{N-1} \left| \sum_{j=0}^{N-1} |f_1[j]| y_j \exp[-i2\pi jk/N] \right|^2 \\ &\quad + \sum_{k=s}^{N-1} \left| \sum_{j=0}^{N-1} |f_2[j]| y_{N+j} \exp[-i2\pi jk/N] \right|^2. \end{aligned} \quad (\text{B5})$$

The expression for the interference component of the cost function is a slightly more involved. Because of the following relations derived from the definitions of $\mathbf{f}_3$ and $\mathbf{f}_4$ and involving the correct phases $\mathbf{x}$:

$$\begin{aligned} |f_3[j]|^2 &= |f_1[j] + f_2[j]|^2 = |f_1[j]|^2 + |f_2[j]|^2 \\ &\quad + 2|f_1[j]| |f_2[j]| \operatorname{Re}(x_j x_{N+j}^*), \\ |f_4[j]|^2 &= |f_1[j] + if_2[j]|^2 = |f_1[j]|^2 + |f_2[j]|^2 \\ &\quad + 2|f_1[j]| |f_2[j]| \operatorname{Im}(x_j x_{N+j}^*), \end{aligned} \quad (\text{B6})$$

we have that

$$\begin{aligned} x_j &= x_{N+j} G[j], \\ G[j] &:= \frac{|f_3[j]|^2 + i|f_4[j]|^2 - (1+i)(|f_1[j]|^2 + |f_2[j]|^2)}{2|f_1[j]| |f_2[j]|}. \end{aligned} \quad (\text{B7})$$

We note that $|G[j]| = 1$ for $j = 0, 1, \dots, N-1$. The second component of the cost function is expressed as follows and is equal to zero if $\mathbf{y} = \mathbf{x}$:

$$Q_{\text{interference}}(\mathbf{y}) := \sum_{j=0}^{N-1} |y_j - y_{N+j} G[j]|^2. \quad (\text{B8})$$

The total cost function is defined simply as the sum of the support component and the interference component. We note that this cost function is non-negative by definition,

$$Q^{(s)}(\mathbf{y}) := Q_{\text{support}}^{(s)}(\mathbf{y}) + Q_{\text{interference}}(\mathbf{y}). \quad (\text{B9})$$

Importantly, it can be expressed as

$$Q^{(s)}(\mathbf{y}) := \mathbf{y}^\dagger A_s^\dagger A_s \mathbf{y}, \quad (\text{B10})$$

where $A_s \in \mathbb{C}^{(2(N-s)+N) \times 2N}$ has entries

$$(A_s)_{p,q} = \begin{cases} |f_1[q]| \exp(i2\pi q(s+p)/N), & \text{if } q < N, p < N-s, \\ |f_2[q]| \exp(i2\pi(q-N)(s+p-(N-s))/N), & \text{if } N \leq q < 2N, (N-s) \leq p < 2(N-s), \\ 1, & \text{if } q < N, p = q + 2(N-s), \\ -G[q-N], & \text{if } N \leq q < 2N, p = q - N + 2(N-s), \\ 0, & \text{otherwise,} \end{cases} \quad (\text{B11})$$

with $p = 0, 1, \dots, (2(N-s)+N)-1$ and $q = 0, 1, \dots, 2N-1$.

If there exists an assignment of phases $\mathbf{y}$ such that $Q^{(s)}(\mathbf{y}) = 0$, then we are guaranteed that for that assignment $F_1[k] = F_2[k] = 0$ at $k = s, s+1, \dots, N-1$, and $y_j = y_{N+j} G[j]$ at $j = 0, 1, \dots, N-1$. The correct assignment of phases $\mathbf{x}$ is a minimizer of the cost function $Q^{(\sigma)}(\mathbf{y})$. In particular, since the correct assignment of phases attains zero cost on both cost functions (by definition), we have that

$$\phi_\sigma^{\text{noiseless}} := \min_{\substack{\mathbf{y} \in \mathbb{C}^{2N} \text{ s.t.} \\ |y_0| = |y_1| = \dots = |y_{N-1}| = 1}} Q^{(\sigma)}(\mathbf{y}) = Q^{(\sigma)}(\mathbf{x}) = 0, \quad (\text{B12})$$

where the optimization is over all vectors $\mathbf{y} \in \mathbb{C}^N$ whose entries are phase factors.

We are not, however, a priori guaranteed that $\mathbf{x}$ is the unique minimizer of $Q^{(\sigma)}(\mathbf{y})$, and therefore that solving Eq. (B12) provides the correct assignment of phases $\mathbf{x}$. However, it was shown in Ref. [19] that $Q^{(\sigma)}(\mathbf{y}) = 0$ for any vector $\mathbf{y} \in \mathbb{C}^{2N}$ (with nonzero 2-norm) iff $\mathbf{y}$ is the exact assignment $\mathbf{x}$. From this fact we conclude that the correct assignment of phases $\mathbf{x}$ can also be obtained by solving the following relaxation of the nonconvex optimization problem in Eq. (B12),

$$\begin{aligned} \chi_\sigma^{\text{noiseless}} &:= \min_{\substack{\mathbf{y} \in \mathbb{C}^{2N} \text{ s.t.} \\ \|\mathbf{y}\|_2 = \sqrt{2N}}} Q^{(\sigma)}(\mathbf{y}) = (R+1)N \lambda_{\min}(A_\sigma^\dagger A_\sigma) \\ &= Q^{(\sigma)}(\mathbf{x}) = 0, \end{aligned} \quad (\text{B13})$$

where the optimization is over *all* vectors with a 2-norm equal to $\sqrt{2N}$. In other words, in the noiseless scenario, the correct assignment of phases $\mathbf{x}$ can be obtained by determining the smallest eigenvector of $A_\sigma^\dagger A_\sigma$.

b. The noisy scenario

In practice, one only obtains the noisy versions of the absolute value measurements $|\tilde{f}_1[j]|^2$, $|\tilde{f}_2[j]|^2$,

$|\tilde{f}_3[j]|^2$, and $|\tilde{f}_4[j]|^2$, which we respectively denote by $|\tilde{f}_1[j]|^2$, $|\tilde{f}_2[j]|^2$, $|\tilde{f}_3[j]|^2$ and $|\tilde{f}_4[j]|^2$. We denote the correct vector of phases of $\tilde{\mathbf{f}}_1$ and $\tilde{\mathbf{f}}_2$ by $\tilde{\mathbf{x}} \in \mathbb{C}^{2N}$. Let us now define the cost function in terms of these noisy absolute value measurements as follows:

$$\tilde{Q}^{(s)}(\mathbf{y}) := \tilde{Q}_{\text{support}}^{(s)}(\mathbf{y}) + \tilde{Q}_{\text{interference}}(\mathbf{y}), \quad (\text{B14})$$

with

$$\begin{aligned} \tilde{Q}_{\text{support}}^{(s)}(\mathbf{y}) &:= \sum_{k=s}^{N-1} \left| \frac{1}{N} \sum_{j=0}^{N-1} |\tilde{f}_1[j]| y_j \exp[-i2\pi jk/n] \right|^2 \\ &\quad + \sum_{k=s}^{N-1} \left| \frac{1}{N} \sum_{j=0}^{N-1} |\tilde{f}_2[j]| y_{N+j} \exp[-i2\pi jk/n] \right|^2 \end{aligned} \quad (\text{B15})$$

and

$$\begin{aligned} \tilde{Q}_{\text{interference}}(\mathbf{y}) &:= \sum_{j=0}^{N-1} |y_j - y_{N+j} \tilde{G}[j] / |\tilde{G}[j]||^2, \\ \tilde{G}[j] &:= \frac{|\tilde{f}_3[j]|^2 + i|\tilde{f}_4[j]|^2 - (1+i)(|\tilde{f}_1[j]|^2 + |\tilde{f}_2[j]|^2)}{2|\tilde{f}_1[j]| |\tilde{f}_2[j]|}, \end{aligned} \quad (\text{B16})$$

where we stress that $\tilde{G}[j]$ does not necessarily have unit absolute value.

Again, the cost function can be expressed as

$$Q^{(s)}(\mathbf{y}) := \mathbf{y}^\dagger \tilde{A}_s^\dagger \tilde{A}_s \mathbf{y}, \quad (\text{B17})$$

where $\tilde{A}_s \in \mathbb{C}^{(2(N-s)+N) \times 2N}$ has entries

$$(\tilde{A}_s)_{p,q} = \begin{cases} |\tilde{f}_1[q]| \exp(i2\pi q(s+p)/N), & \text{if } q < N, p < N-s, \\ |\tilde{f}_2[q]| \exp(i2\pi(q-N)(s+p-(N-s))/N), & \text{if } N \leq q < 2N, (N-s) \leq p < 2(N-s), \\ 1, & \text{if } q < N, p = q + 2(N-s), \\ -\tilde{G}[q-N] / |\tilde{G}[q-N]|, & \text{if } N \leq q < 2N, p = q - N + 2(N-s), \\ 0, & \text{otherwise,} \end{cases} \quad (\text{B18})$$

with $p = 0, 1, \dots, (2(N-s)+N)-1$ and $q = 0, 1, \dots, 2N-1$.

In the noisy setting, there is no guarantee that there is an assignment of phases $\mathbf{y}$ for which $\tilde{Q}^{(s)}(\mathbf{y}) = 0$, but the optimal assignment of phases could be found by solving the nonconvex optimization problem,

$$\phi_s^{\text{noisy}} := \min_{\substack{\mathbf{y} \in \mathbb{C}^{2N} \text{ s.t.} \\ |y_0| = |y_1| = \dots = |y_{N-1}| = 1}} \tilde{Q}^{(s)}(\mathbf{y}), \quad (\text{B19})$$

at $s = \sigma$. In general, we now have that $\phi_s^{\text{noisy}} > 0$. The correct assignment $\tilde{\mathbf{x}}$ is not guaranteed to be the unique minimizer of $\tilde{Q}^{(\sigma)}(\mathbf{y})$. However, for relatively small noise magnitudes, any assignment of phases for which $\tilde{Q}^{(\sigma)}(\mathbf{y})$ is minimized tends to be close to the correct assignment $\tilde{\mathbf{x}}$. Like in the noiseless scenario, we will consider the following relaxation of the problem in Eq. (B19) to an eigenvalue problem,

$$\chi_\sigma^{\text{noisy}} := \min_{\substack{\mathbf{y} \in \mathbb{C}^{2N} \text{ s.t.} \\ \|\mathbf{y}\|_2 = \sqrt{2N}}} \tilde{Q}^{(\sigma)}(\mathbf{y}) = 2N \lambda_{\min}(\tilde{A}_\sigma^\dagger \tilde{A}_\sigma), \quad (\text{B20})$$

where we note that again, generally, $\chi_\sigma^{\text{noisy}} > 0$. For relatively small noise magnitudes, the optimal assignment obtained by solving Eq. (B20) (which we denote by $\mathbf{y}_{\min}$) will be close to the correct assignment $\tilde{\mathbf{x}}$.

Before discussing the details of the algorithm, let us consider a generalization that we introduce in this work. This generalization consists of the inclusion of *multiple* time series that are used for interference, as opposed to just a single one. As we shall demonstrate, this leads to improved performance of the algorithm in the presence of noise.

2. Vectorial phase retrieval using multiple interference signals

In this work, we consider the scenario in which one employs *multiple* signals $\mathbf{f}_2^{(r)} \in \mathbb{C}^N$ (with $r = 1, 2, \dots, R$) with which the target time series $\mathbf{f}_1$ interferes. As we will demonstrate in Sec. B3, we find that the performance of vectorial phase retrieval improves when multiple interference signals are employed in the noisy scenario. This multi-interference vectorial phase retrieval problem is formulated as follows:

Problem B2. [Multi-interference vectorial phase retrieval] Given measurements of the absolute values

$$|f_1[j]|^2, \left\{ |f_2^{(r)}[j]|^2, |f_3^{(r)}[j]|^2 := |f_1[j] + f_2^{(r)}[j]|^2, \right. \\ \left. |f_4^{(r)}[j]|^2 := |f_1[j] + if_2^{(r)}[j]|^2 \right\}_{r=1}^R, \quad (\text{B21})$$

at times $j = 0, 1, \dots, N-1$, determine

$$F_1[k] := \sum_{j=0}^{N-1} f_1[j] \exp[-i2\pi jk/N], \\ \left\{ F_2^{(r)}[k] := \sum_{j=0}^{N-1} f_2^{(r)}[j] \exp[-i2\pi jk/N] \right\}_{r=1}^R, \quad (\text{B22})$$

at $k = 0, 1, \dots, N-1$.

The trivial ambiguities of Problem B2 are equivalent to those of Problem B1, involving *all* $(R+1)$ signals $\mathbf{F}_1, \mathbf{F}_2^{(1)}, \dots, \mathbf{F}_2^{(R)}$ (instead of just $\mathbf{F}_1, \mathbf{F}_2$).

We will discuss how the multi-interference v-PR algorithm can be implemented to (approximately) solve Problem B2. But let us first discuss how the control-free QPE routine fits into the multi-interference v-PR framework. We take

$$f_1[j] = \langle \Phi | \exp(iH\Delta t j) | \Phi \rangle \text{ and} \\ \left\{ f_2^{(r)}[j] = \langle \Phi | \exp(iH\Delta t j) | \psi_r \rangle \right\}_{r=1}^R, \quad (\text{B23})$$

such that

$$\left\{ f_3^{(r)}[j] = f_1[j] + f_2^{(r)}[j] \right. \\ = \langle \Phi | \exp(iH\Delta t j) (|\Phi\rangle + |\psi_r\rangle) \text{ and} \\ \left. f_4^{(r)}[j] = f_1[j] + if_2^{(r)}[j] \right. \\ = \langle \Phi | \exp(iH\Delta t j) (|\Phi\rangle + i|\psi_r\rangle) \left. \right\}_{r=1}^R. \quad (\text{B24})$$

Hence by preparing states $|\Phi\rangle$ and $\{|\psi_r\rangle, 1/\sqrt{2}(|\Phi\rangle + |\psi_r\rangle), 1/\sqrt{2}(|\Phi\rangle + i|\psi_r\rangle)\}_{r=1}^R$, then time evolving *each* of the states under Hamiltonian H for a time $\Delta t j$, and measuring the overlap with $|\Phi\rangle$, we can estimate $|f_1[j]|^2$ and $\{|f_2^{(r)}[j]|^2, |f_3^{(r)}[j]|^2, |f_4^{(r)}[j]|^2\}_{r=1}^R$ at $j = 0, 1, \dots, N-1$. Of course, we do not want the states $|\Phi\rangle$ and $\{|\psi_r\rangle\}_{r=1}^R$ to be such that preparing superpositions of $|\Phi\rangle$ with any $|\psi_r\rangle$ requires applying circuits of very large depth.

As discussed before, one only ever obtains noisy versions of the absolute value measurements $|f_1[j]|^2, \{|f_2^{(r)}[j]|^2, |f_3^{(r)}[j]|^2, |f_4^{(r)}[j]|^2\}_{r=1}^R$ in practice, which we denote by $|\tilde{f}_1[j]|^2, \{|\tilde{f}_2^{(r)}[j]|^2, |\tilde{f}_3^{(r)}[j]|^2, |\tilde{f}_4^{(r)}[j]|^2\}_{r=1}^R$. The multi-interference v-PR algorithm takes as *input* these noisy absolute value measurements, and an estimate s of σ (with σ denoting the support size of $\mathbf{F}_1, \mathbf{F}_2^{(1)}, \dots, \mathbf{F}_2^{(R)}$). The *output* of the algorithm is a vector $\mathbf{y} \in \mathbb{C}^{(R+1)N}$ whose entries are estimates of the phases of $f_1[j], f_2^{(1)}[j], f_2^{(2)}[j], \dots, f_2^{(R)}[j]$ at $j = 0, 1, \dots, N-1$. The algorithm again consists of the minimization of a cost function over all these assignments of phases to the absolute value measurements $\{|\tilde{f}_1[j]|^2\}, \{|\tilde{f}_2^{(1)}[j]|^2\}, \dots, \{|\tilde{f}_2^{(R)}[j]|^2\}$. We denote by $\tilde{\mathbf{x}} \in \mathbb{C}^{(R+1)N}$ the correct phases of $f_1[j], f_2^{(1)}[j], f_2^{(2)}[j], \dots, f_2^{(R)}[j]$ at $j = 0, 1, \dots, N-1$.

The cost function again consists of a support component $\tilde{Q}_{\text{support}}^{(s)}(\mathbf{y})$ and an interference component $\tilde{Q}_{\text{interference}}(\mathbf{y})$.

The support component of the cost function can be written as

$$\begin{aligned} \tilde{Q}_{\text{support}}^{(s)}(\mathbf{y}) := & \sum_{k=s}^{N-1} \left| \sum_{j=0}^{N-1} |\tilde{f}_1[j]| y_j \exp[-i2\pi jk/n] \right|^2 \\ & + \sum_{r=1}^R \left[\sum_{k=s}^{N-1} \left| \sum_{j=0}^{N-1} |\tilde{f}_2^{(r)}[j]| y_j^{(r)} \exp[-i2\pi jk/n] \right|^2 \right], \end{aligned} \quad (\text{B25})$$

which is equivalent to the support component defined in the single-interference-signal framework, except that now

it constrains the support of $R + 1$ signals. The interference component $\tilde{Q}_{\text{interference}}(\mathbf{y})$, which is again an extension of the single-interference-signal interference component to $R + 1$ signals, can be defined as follows:

$$\tilde{Q}_{\text{interference}}(\mathbf{y}) := \sum_{r=1}^R \left[\sum_{j=0}^{N-1} \left| y_j - y_j^{(r)} \tilde{G}_r[j] / |\tilde{G}_r[j]| \right|^2 \right], \quad (\text{B26})$$

where the quantities $\tilde{G}_r[j]$ are defined at $j = 0, 1, \dots, N - 1$ for $r = 1, 2, \dots, R$ as

$$\tilde{G}_r[j] := \frac{|\tilde{f}_3^{(r)}[j]|^2 + i|\tilde{f}_4^{(r)}[j]|^2 - (1 + i) \left(|\tilde{f}_1[j]|^2 + |\tilde{f}_2^{(r)}[j]|^2 \right)}{2|\tilde{f}_1[j]| |\tilde{f}_2^{(r)}[j]|}. \quad (\text{B27})$$

The total cost function

$$\tilde{Q}^{(s)}(\mathbf{y}) := \tilde{Q}_{\text{support}}^{(s)}(\mathbf{y}) + \tilde{Q}_{\text{interference}}(\mathbf{y}) \quad (\text{B28})$$

can be expressed as

$$\tilde{Q}^{(s)}(\mathbf{y}) := \mathbf{y}^\dagger \tilde{A}_s^\dagger \tilde{A}_s \mathbf{y}, \quad (\text{B29})$$

where $\tilde{A}_{s,R} \in \mathbb{C}^{((R+1)(N-s)+RN) \times (R+1)N}$ has entries

$$(A_s)_{p,q} = \begin{cases} |\tilde{f}_1[q]| \exp(i2\pi q(s+p)/N), & \text{if } q < N, p < N - s, \\ |\tilde{f}_2[q]| \exp(i2\pi(q - N)(s+p - (N - s))/N), & \text{if } N \leq q < 2N, (N - s) \leq p < 2(N - s), \\ \vdots & \\ |\tilde{f}_{R+1}[q]| \exp(i2\pi(q - RN)(s+p - R(N - s))/N), & \text{if } RN \leq q < (R + 1)N, R(N - s) \leq p < (R + 1)(N - s), \\ 1, & \text{if } q < N, p = q + (R + 1)(N - s), \\ -\tilde{G}_2[q - N] / |\tilde{G}_2[q - N]|, & \text{if } N \leq q < 2N, p = q - N + (R + 1)(N - s), \\ 1, & \text{if } q < N, p = q + (R + 1)(N - s) + N, \\ -\tilde{G}_3[q - 2N] / |\tilde{G}_3[q - 2N]|, & \text{if } 2N \leq q < 3N, p = q - N + (R + 1)(N - s), \\ \vdots & \\ 1, & \text{if } q < N, p = q + (R + 1)(N - s) + (R - 1)N, \\ -\tilde{G}_{R+1}[q - RN] / |\tilde{G}_{R+1}[q - RN]|, & \text{if } RN \leq q < (R + 1)N, p = q - N + (R + 1)(N - s), \\ 0, & \text{otherwise,} \end{cases} \quad (\text{B30})$$

with $p = 0, 1, \dots, ((R + 1)(N - s) + RN) - 1$ and $q = 0, 1, \dots, (R + 1)N - 1$.

The optimal assignment of phases could be found by solving the nonconvex optimization problem

$$\Phi_\sigma^{\text{noisy}} := \min_{\substack{\mathbf{y} \in \mathbb{C}^{(R+1)N} \text{ s.t.} \\ |y_0| = |y_1| = \dots = |y_{N-1}^{(R)}| = 1}} \tilde{Q}^{(\sigma)}(\mathbf{y}). \quad (\text{B31})$$

In general, we have that $\Phi_\sigma^{\text{noisy}} > 0$. The correct assignment $\tilde{\mathbf{x}}$ is again not guaranteed to be a (unique) minimizer of $\tilde{Q}^{(\sigma)}(\mathbf{y})$. However, for relatively small noise magnitudes, any assignment of phases for which $\tilde{Q}^{(\sigma)}(\mathbf{y})$ is minimized tends to be close to the correct assignment $\tilde{\mathbf{x}}$.

We again relax the problem in Eq. (B31) to an eigenvalue problem as follows:

$$X_\sigma^{\text{noisy}} := \min_{\substack{\mathbf{y} \in \mathbb{C}^{(R+1)N} \text{ s.t.} \\ \|\mathbf{y}\|_2 = \sqrt{(R+1)N}}} Q^{(\sigma)}(\mathbf{y}) = (R+1)N \lambda_{\min} \left(\tilde{A}_\sigma^\dagger \tilde{A}_\sigma \right), \quad (\text{B32})$$

where we note that again, generally, $X_\sigma^{\text{noisy}} > 0$. As we shall demonstrate in Appendix B3, for relatively small noise magnitudes, the optimal assignment $\mathbf{y}_{\min}$ obtained by solving Eq. (B32) will be close to the correct assignment $\tilde{\mathbf{x}}$. Moreover, we will show that for a given noise magnitude, the assignment of phases obtained by solving Eq. (B32) will typically become a better approximation for $\tilde{\mathbf{x}}$ as R increases.

We note that the optimal assignment $\mathbf{y}_{\min}$ obtained by solving Eq. (B32) is not necessarily an assignment of phases. Although we ensure that $\|\mathbf{y}_{\min}\|_2 = \sqrt{(R+1)N}$, each entry can have absolute value different from one. One could consider rounding $\mathbf{y}_{\min}$ to an assignment of phases by dividing each entry by its absolute value. We do not perform this rounding as we find it does not *necessarily* improve the approximation to $\tilde{\mathbf{x}}$ in the scenarios that we consider.

In Appendix B3, we will demonstrate numerically how the v-PR algorithm can be used to perform spectral estimation for instances of the FH Hamiltonian, and discuss its performance in various settings.

3. Vectorial phase retrieval algorithm for spectral estimation of Hamiltonians

In this section, we numerically investigate the performance of the vectorial phase retrieval algorithm in the context of spectral estimation for Hamiltonians. The signals encountered in this context do not perfectly fit within the framework of v-PR. In particular, due to the fact that the (absolute values of) the time series are only measured for a finite measurement time, the signals $\mathbf{F}_1, \mathbf{F}_2^{(1)}, \dots, \mathbf{F}_2^{(R)}$ do not necessarily have well-defined support. Typically, $|F_1[k]|, |F_2^{(1)}[k]|, \dots, |F_2^{(R)}[k]|$ decay away from the supported eigenvalues, rather than being equal to zero exactly outside of some frequency interval. The more the signals suffer from this effect, the more ill-defined their support becomes. As a result, a good performance of the vectorial phase retrieval algorithm cannot be guaranteed without further considerations.

To set the stage, we first provide an example for which the v-PR input data is obtained from time evolution of some state under the FH Hamiltonian, where we have artificially ensured that the input data fits exactly in the v-PR framework. Namely, we have altered the input data in such a way that $\mathbf{F}_1, \mathbf{F}_2^{(1)}$ (we take $R = 1$ here) are exactly equal to zero outside of an interval of size $\sigma = 25$. Note that to do this, one generally needs access to the full time series and not just their absolute values (so this cannot be done in practice). The data we have used here is noiseless data. Based on the discussion in Sec. B1a, we expect the matrix $A_s^\dagger A_s$ to have a *single* zero eigenvalue at $s = \sigma$, and its smallest eigenvector to correspond to the exact assignment of phases. Figure 7 depicts the spectrum of $A_s^\dagger A_s$ as a function of s . Indeed, its smallest eigenvalue is nonzero for $s < \sigma$ and (numerically) zero at $s = \sigma$. For $s > \sigma$, there are several zero-eigenvalue eigenvectors, with the correct solution lying in the span of those eigenvectors. In the

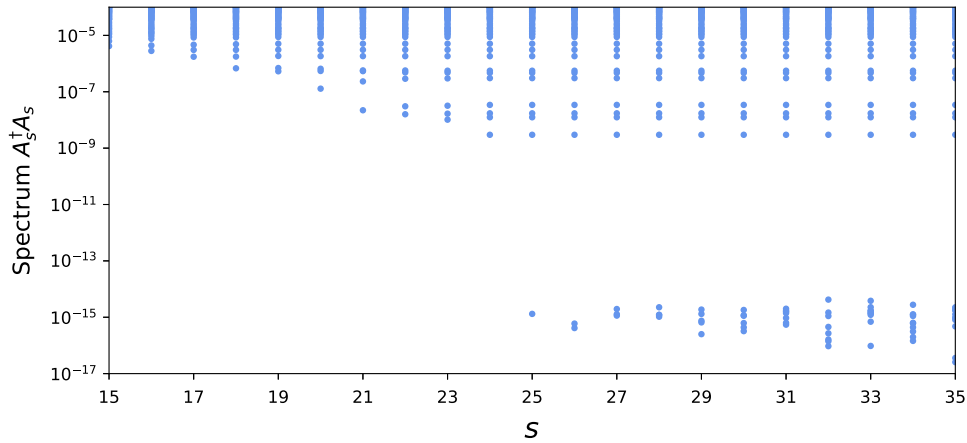


FIG. 7. Spectrum of $A_s^\dagger A_s$ for $R = 1$ in the noiseless setting, as a function of s . The time series amplitudes used to construct $A_s^\dagger A_s$ correspond to evolution under a 1×5 (i.e., 10-mode) Fermi-Hubbard Hamiltonian for which $|\tau/U| = 1/4$. We have taken $N = 300$. The signals f_1 and f_2 were artificially processed to ensure that they have well-defined support (here, $\sigma = 25$).

noisy setting, there will generally be *no* zero-eigenvalue solutions. However, for sufficiently small noise magnitudes, there will still be a drop in the value of the smallest eigenvalue of $A_s^\dagger A_s$ around σ . Our investigations suggest that:

- (1) The two smallest eigenvalues are a proxy for the quality of the retrieval of phases. That is, the smaller the smallest eigenvalue, the smaller the value of the two residuals Eqs. (2) and (3) above.
- (2) A large gap opening between the two smallest eigenvalues suggests a larger overlap between the

smallest eigenvector and the true assignment of phases. This being motivated by the fact that in the ideal noiseless case the solution is unique.

In the remainder of this section, we will discuss the performance of vectorial phase retrieval for input data obtained from time evolution under FH Hamiltonians. The procedure that is implemented is given as a pseudocode below. As discussed, this input data does not perfectly fit into the v-PR framework. Nevertheless, the v-PR algorithm performs well in performing phase retrieval in the scenarios that we consider.

```

1: procedure v-PR(  $|\tilde{f}_1[j]|\rangle, |\tilde{f}_2^{(r)}[j]\rangle, |\tilde{f}_3^{(r)}[j]\rangle, |\tilde{f}_4^{(r)}[j]\rangle$  for  $r = 1, 2, \dots, R$ , at  $j = 0, 1, \dots, N - 1$ .)
2:   for  $s = 0, 1, \dots, N-1$  do
3:     Evaluate  $G_r[j]$  for  $r = 1, 2, \dots, R$ , at  $j = 0, 1, \dots, N - 1$ .
4:     Obtain  $\tilde{A}_s^\dagger \tilde{A}_s \in \mathbb{C}^{(R+1)N \times (R+1)N}$ .
5:     Evaluate the two smallest eigenvalues of  $\tilde{A}_s^\dagger \tilde{A}_s$  (with the smallest denoted by  $\lambda_{\min}(\tilde{A}_s^\dagger \tilde{A}_s)$ ).
6:   end for
7:   Obtain optimal  $s^*$  (heuristically).
8:   Evaluate  $\mathbf{y}_{\min} \in \mathbb{C}^{(R+1)N}$  s.t.  $\mathbf{y}_{\min}^\dagger \tilde{A}_{s^*}^\dagger \tilde{A}_{s^*} \mathbf{y}_{\min} = (R+1)N \lambda_{\min}(\tilde{A}_{s^*}^\dagger \tilde{A}_{s^*})$ .
9:   Output  $\{ |f_1[j]| (y_{\min})_j \}_{j=0}^{N-1}$ .
10: end procedure

```

a. Interplay between R and the noise resilience of the vectorial phase retrieval algorithm

Figure 8 illustrates the difference in noise resilience of the v-PR algorithm between the $R = 1$ and the $R = 10$ scenarios. This difference materializes as follows: In the noisy scenario, there is a sharp drop in the smallest eigenvalue of $\tilde{A}_s^\dagger \tilde{A}_s$ as a function of s for $R = 10$, which is absent for $R = 1$. This can be seen in Fig. 8(c). Our numerical investigations suggest that this sharp drop is a signature of an accurate retrieval of phases at that value of s . As can be seen in Fig. 8(d), indeed the estimate of $\mathbf{F}_1$ for $R = 10$ is more accurate than for $R = 1$. Figure 8(b) suggests that choosing larger R does not influence the accuracy significantly in the noiseless scenario.

b. Performance of the vectorial phase retrieval algorithm as a function of sample size

For completeness, we include a plot of the l^2 -norm error of the recovered spectrum $\mathbf{F}_1$ in Fig. 9. Let us note that the error floor appearing around 10^6 samples seems to be an artifact of the “choice” for measuring the error using the l^2 norm: As can be seen in Fig. 8(d), the peaks in the spectrum are accurately recovered for $R = 10$ at these noise magnitudes, but the recovery of the tails is poorer.

c. Performance of the vectorial phase retrieval algorithm for shallower time evolutions

The fact that $\mathbf{F}_1, \mathbf{F}_2^{(1)}, \dots, \mathbf{F}_2^{(R)}$ do not have well-defined support is the most prominent reason that the v-PR input data in the current setting does not fit perfectly into the framework of v-PR. The support of these signals typically becomes more ill-defined (i.e., $\mathbf{F}_1, \mathbf{F}_2^{(1)}, \dots, \mathbf{F}_2^{(R)}$ decay more slowly away from the supported frequencies) as the total time evolution becomes smaller (due to spectral leakage). Since the circuit depths are limited in our near-term scenario, it is of interest to see how the v-PR algorithm fares for shallower time evolutions.

Figure 11(a) depicts the smallest two eigenvalues $A_s^\dagger A_s$ as a function of s . Clearly, in the noiseless setting, $\lambda_{\min}(A_s^\dagger A_s)$ decays to zero comparatively slowly, which is a result the signals having ill-defined support. We find that the quality of the approximation $|F_{1,\text{approx}}|$ to $|F_1|$ is highest at the value of s at which $\lambda_{\min}(A_s^\dagger A_s)$ first starts to decay, i.e., $s = 29$ for Fig. 10. Figure 11(b) depicts $|F_{1,\text{approx}}|$ and in the noiseless scenario, and for $N_{\text{samples}} = 10^6$. Remarkably, although the approximation of $|F_{1,\text{approx}}|$ to $|F_1|$ is worse (even in the noiseless scenario) than in Fig. 8, the approximation retains its resilience against the addition of sampling noise.

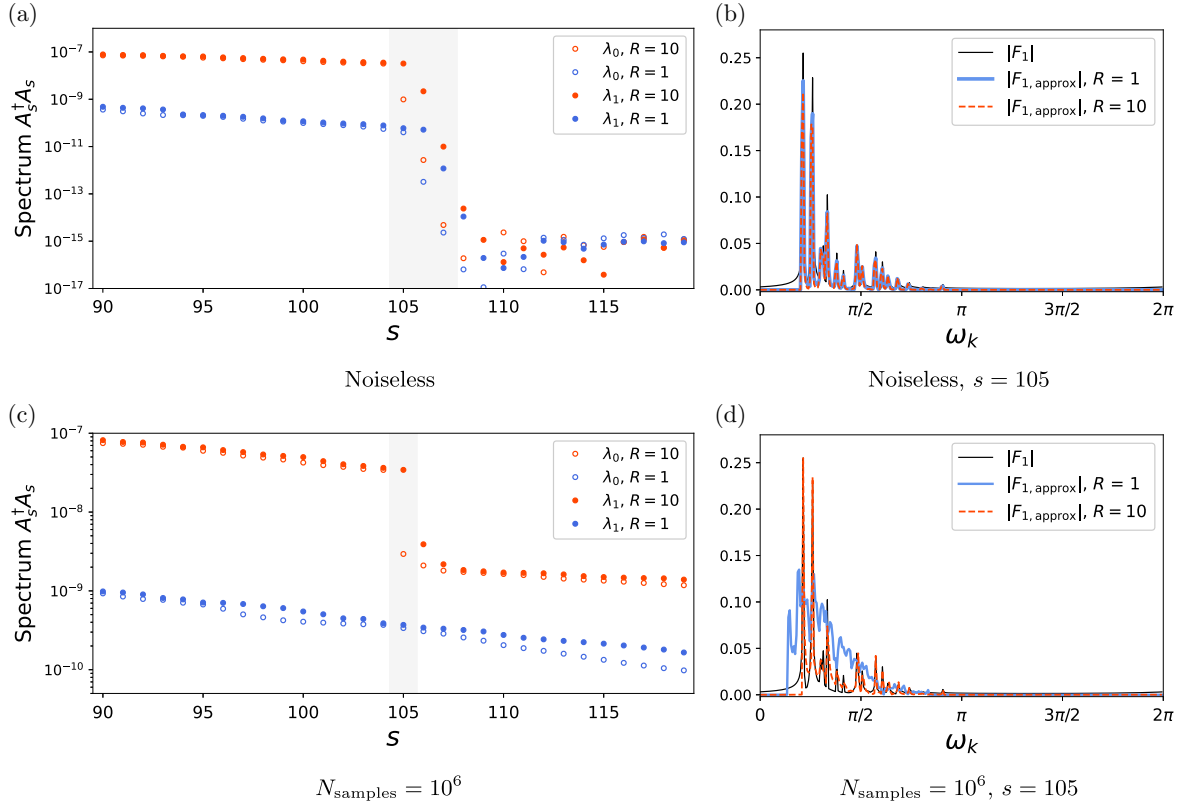


FIG. 8. The smallest two eigenvalues of $A_s^\dagger A_s$ for $R=1$ and $R=10$ in the noiseless setting [see (a)], and for $N_{\text{samples}} = 10^6$ [see (c)]. The eigenvalues are given as a function of s . Approximations to $|F_1|$ at fixed $s=105$ for $R=1$ and $R=10$ are depicted in the noiseless setting in (b) and for $N_{\text{samples}} = 10^6$ in (d). The time series amplitudes used to construct $A_s^\dagger A_s$ correspond to evolution under a 1×5 (i.e., 10-mode) Fermi-Hubbard Hamiltonian for which $|\tau/U| = 1/4$.

d. To round or not to round

At the end of Sec. B 2, we briefly commented on whether it is beneficial to round the smallest eigenvector $\mathbf{y}_{\min}$ (with 2-norm $\sqrt{(R+1)N}$) of the matrix $\tilde{A}_s^\dagger \tilde{A}_s$ to a

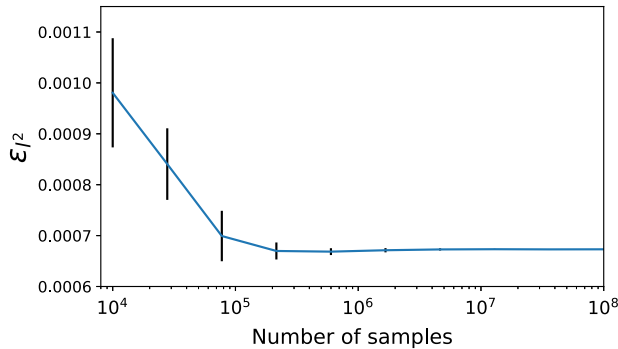


FIG. 9. Spectrum error as a function of number of samples used to estimate each $|f[j]|$. Here, we consider the $R=10$ instance from Fig. 8 and we define the spectrum error as $\epsilon_2 = \frac{1}{N} \sqrt{\sum_k |F[k] - F_{\text{true}}[k]|^2}$. At each fixed number of samples we average over 10 instances of the v-PR algorithm and include error bars indicating one standard deviation.

vector whose entries are phase factors, by dividing each entry of $\mathbf{y}_{\min}$ by its absolute value. Our numerical investigations suggest that using the entries of this rounded vector to reconstruct the time series does not *necessarily* lead to a more accurate reconstruction. In terms of the reconstructed spectra F [see Fig. 11 below, corresponding to the scenario in Fig. 6(a)], we note the following differences. The unrounded spectrum recovers the peak locations relatively well. The rounded spectrum performs better at recovering the absolute values $|F|$. However, the latter also seems to recover some peaks for frequencies at which F is not actually supported. Whether rounding is beneficial thus depends on which features one wishes to recover. To quantify the error in the particular case of Fig. 11, we note that the 1-norm (normalized by N) of the vector with entries $|F_{\text{reconstructed}}[k]| - |F_{\text{exact}}[k]|$ is 0.0016 without rounding and 0.0013 with rounding.

APPENDIX C: 2D PHASE RETRIEVAL

In what follows, we will review the 2D phase retrieval problem, the HIO for solving this problem and then our main contribution. We introduce a procedure for embedding the phase retrieval problem relevant to QPE into a

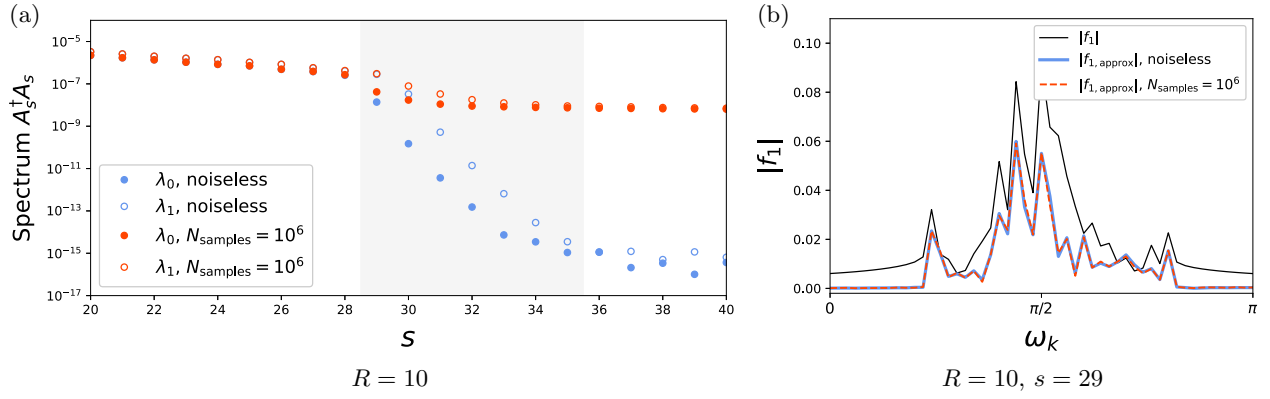


FIG. 10. The smallest two eigenvalues of $A_s^\dagger A_s$ for $R = 10$ in the noiseless and noisy settings [see (a)]. The eigenvalues are given as a function of s . Approximations to $|F_1|$ at fixed $s = 29$ for $R = 10$ are depicted in the noiseless setting and for $N_{\text{samples}} = 10^6$ in (b). The time series amplitudes used to construct $A_s^\dagger A_s$ correspond to evolution under a 1×5 (i.e., 10-mode) Fermi-Hubbard Hamiltonian for which $|\tau/U| = 1/4$, with a relatively small total time evolution T . This small total time evolution leads to ill-defined support of $|F_1|$. Note that the input state $|\Phi\rangle$ is chosen differently than in Fig. 8.

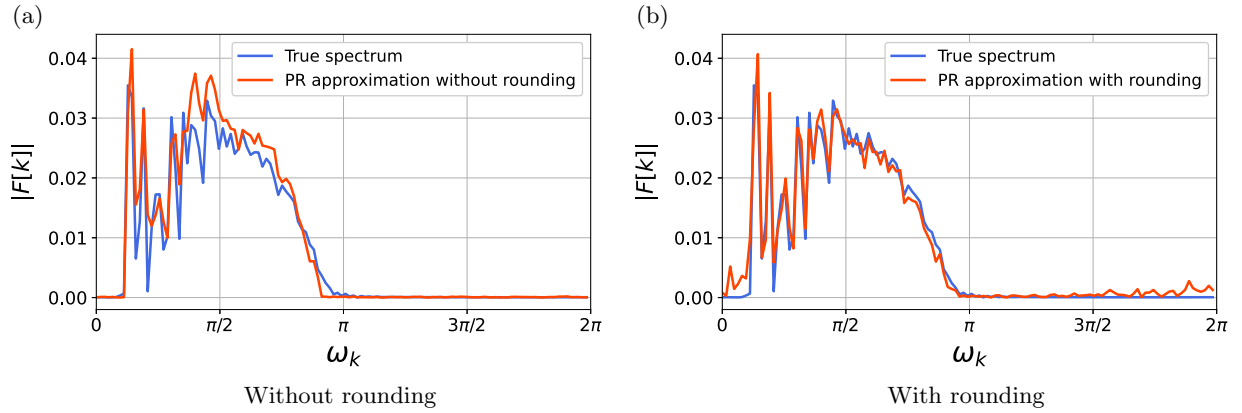


FIG. 11. Comparison between the unrounded (a) and rounded (b) solutions obtained through vectorial phase retrieval. The scenario used to obtain the figures is the same as that of Fig. 6.

2D time-series which we show can be guaranteed to be both real and positive. Finally, we demonstrate numerically on a Fermi-Hubbard model example that the HIO algorithm succeeds to retrieve time-series of enough quality to recover a good quality spectrum, showing that the algorithm is resilient to sample noise. In order to keep the analysis simple we just implement a recovery relying on a DFT of the time-series. However, we note that the discrete Fourier transform is not necessarily the most accurate reconstruction of the spectrum of a Hamiltonian. Typically, classical postprocessing techniques like the one developed in Ref. [5] can be used to obtain a more useful description of the spectrum.

1. The 2D Phase Retrieval Problem

In two-dimensional phase retrieval, one now attempts to solve the following (potentially ambiguous) problem.

Problem C1. [2D phase retrieval] Given measurements of the *absolute values* [44] of a signal $f[j, l]$ determine the discrete Fourier transform $F[k, m]$ of that signal [see Eq. (C1)], up to the trivial ambiguities of global phases, shifts and conjugate reflections. We have the following conventions

$$F[k, m] := \sum_{j, l=0}^{N-1, M-1} f[j, l] e^{-i2\pi(\frac{kj}{N} + \frac{ml}{M})}$$

$$f[j, l] := \frac{1}{NM} \sum_{k, m=0}^{N-1, M-1} F[k, m] e^{i2\pi(\frac{kj}{N} + \frac{ml}{M})}. \quad (\text{C1})$$

Unlike the one-dimensional problem, phase retrieval in two or more dimensions almost always has a unique solution [36]. This is the motivation behind studying a 2D version of the problem, and in demonstrating its applicability to the problem of QPE. The uniqueness properties

```

1: procedure HYBRID( $|f|$ ,  $\beta$ ,  $L$ ,  $F^1$ )
2:   for  $i = 1, 2, \dots, L$  do
3:      $f^i = \mathcal{DFT}^{-1}(F^i) = |f^i|e^{i\arg(f^i)}$ 
4:      $\tilde{f}^i = |f|e^{i\arg(f^i)}$ 
5:      $\tilde{F}^i = \text{real}(\mathcal{DFT}(\tilde{f}^i))$ 
6:     if  $\tilde{F}^i[k, m] \leq 0$  then
7:        $F^{i+1}[k, m] = F^i[k, m] - \beta \tilde{F}^i[k, m]$ 
8:     else
9:        $F^{i+1}[k, m] = \tilde{F}^i[k, m]$ 
10:    end if
11:  end for
12: end procedure

```

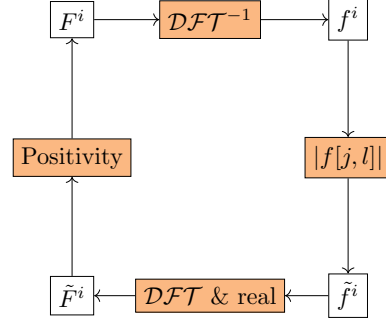


FIG. 12. (a) Schematic for Fienup's hybrid input-output algorithm (HIO) four steps. The i -th iteration starts with a candidate spectrum F^i : Step (1) transform it into a time series f^i candidate via an inverse DFT; Step (2) generates a new $\tilde{f}^i$ that has same phase as f^i and satisfies $|\tilde{f}[j, l]| = |f[j, l]|$; Step (3) transforms it back to the Fourier domain and takes the real part; Step (4) implements the update rule in the schematic to impose a relaxed version of positivity of the spectrum.

of a signal can be studied through setting up an associated polynomial, as discussed throughout the phase retrieval literature, [36]. Whether these polynomials are irreducible or not determines whether the problem has nontrivial ambiguities. Almost all 2D multivariate polynomials are irreducible, a fact which means almost all 2D signals can be recovered from their Fourier magnitudes [36]. This is our motivation for demonstrating one possible way of embedding QPE into a 2D phase retrieval problem. We will demonstrate one possible version of this, though many others can be imagined.

2. The Hybrid Input-Output Algorithm

There are many algorithms which can be used to solve the problem of phase retrieval [45,46]. We leave the exploration of these approaches to future work and instead simply demonstrate the effectiveness using one of the basic algorithms in widespread use, Fienup's hybrid input-output algorithm or HIO algorithm [20].

The HIO algorithm is a generalization of the earlier Gerchberg-Saxton algorithm. Figure 12 summarizes the algorithm which involves transforming back and forth between the Fourier and object domain and imposing constraints on each domain at every iteration. The signal constraints being the measured magnitudes, while the Fourier domain constraints are more flexible. Sometimes knowledge of the support of F may be used, or knowledge that $F[k, m]$ is real and positive. We will use these latter two constraints from now on.

Let F^i denote the estimate for the true DFT F at step i in the algorithm. An iteration of the algorithm can be summarized as follows:

- (1) The inverse discrete Fourier transform of F^i is taken to obtain $f^i = \mathcal{DFT}^{-1}(F^i) = |f^i|e^{i\arg(f^i)}$.
- (2) The absolute values $|f^i|$ are replaced with the true and known absolute values $|f|$. This defines

$\tilde{f}^i = |f|e^{i\arg(f^i)}$, a function which satisfies the absolute values constraints.

- (3) Next one takes the DFT of this time series, and enforces the constraint that the DFT is **real**, to obtain $\tilde{F}^i = \text{real}(\mathcal{DFT}(\tilde{f}^i))$.
- (4) Next the **positivity** constraint is used. One defines each element of F^{i+1} with the following update rule,

$$F^{i+1}[k, m] = \begin{cases} \tilde{F}^i[k, m], & \tilde{F}^i[k, m] \geq 0, \\ F^i[k, m] - \beta \tilde{F}^i[k, m], & \text{else.} \end{cases} \quad (C2)$$

This retains the elements which are positive and replaces the remaining entries by a relaxed version of the positivity constraint with parameter $0 \leq \beta \leq 1$.

The algorithm then loops for L iterations, hopefully converging to the true $F[k]$ or a good approximation of it.

3. Adapting 2D Quantum Signals for the HIO Algorithm

a. Extension to 2D

In this section we first explain how to embed spectral QPE, which is a 1D problem, into a 2D signal processing problem and then show that we can add constraints to this signal to ensure that the target F for the HIO algorithm is real and positive. For the purposes of QPE we are interested in 1D signals of the form $f(t) = \langle \psi | e^{itH} | \psi \rangle$, where H is the Hamiltonian being studied. To create a 2D problem we introduce a secondary time-evolution over a dummy Hamiltonian H_D with virtual time parameter z , defining the new continuous 2D signal,

$$f(t, z) := \langle \psi | e^{itH} e^{izH_D} | \psi \rangle. \quad (C3)$$

It is important to choose H_D such that $|\psi\rangle$ is not an eigenstate of it, or otherwise the problem remains of a 1D nature. This is one of the simplest ways we can imagine creating a 2D signal.

b. Continuous to Discrete

First we adapt some key concepts in Sec. A on continuous-time signals, their Fourier transforms and their discretizations to the two-dimensional setting. We establish these facts only so that we can show that the discrete Fourier transform F can be guaranteed to be real and positive. This is key to ensuring the reliability of the HIO algorithm.

Now the time signal windowed and sampled time-series $g(t, z)$ and its Fourier transform $G(\omega, \eta)$ now read

$$g(t, z) := f(t, z) \Pi_T(t, z) \text{III}_{\Delta t}(t, z), \quad (\text{C4})$$

$$G(\omega, \eta) := \mathcal{FT}[g(t, z)](\omega, \eta). \quad (\text{C5})$$

The two dimensional Dirac comb with $\Delta t = T/N$ which represents sampling in the continuous picture, and two dimensional windowing function are

$$\text{III}_{\Delta t}(t, z) = \sum_{j, l=-\infty}^{\infty} \delta(t - j \Delta t) \delta(z - l \Delta t), \quad (\text{C6})$$

$$\Pi_T(t, z) = \Lambda_T(t) \cdot \Lambda_T(z), \quad (\text{C7})$$

where $\Lambda_T(t)$ and $\Lambda_T(z)$ have support for times $t, z \in [-T/2, T/2]$ as before. In the above review and for the remainder of this analysis we take $M = N$ for simplicity, though what follows is readily generalized to $M \neq N$.

As established in Sec. A, the discrete and finite matrix $F[k, m]$ is related to the continuous time Fourier transform $G(\omega, \eta) = \mathcal{FT}[g(t, z)](\omega, \eta)$ at the discrete frequencies $\omega_k = 2\pi k/T$ and $\eta_m = 2\pi m/T$ via

$$F[k, m] := G(\omega_k, \eta_m), \quad (\text{C8})$$

with $k = 0, 1, \dots, N-1$ and $m = 0, 1, \dots, M-1$. Similarly $f[j, l]$ is related to this underlying continuous picture via

$$\hat{f}[j, l] := f(j \Delta t, l \Delta t) \Pi_T(j \Delta t, l \Delta t), \quad (\text{C9})$$

$$f[j, l] := \sum_{j', l' \in \mathbb{Z}} \hat{f}[j - j'N, l - l'M], \quad (\text{C10})$$

where the $\hat{f}[j, l]$ is defined for all $j, l \in \mathbb{Z}$ and $f[j, l]$ is defined for $j = 0, 1, \dots, N-1$ and $l = 0, 1, \dots, M-1$. In summary, we have two underlying continuous functions related by a continuous two-dimensional Fourier transform $G(\omega, \eta) = \mathcal{FT}[g(t, z)](\omega, \eta)$ and two matrices related by a two-dimensional discrete time Fourier transform $F[k, m] = \mathcal{DFT}[f[j, l]]$ and we can relate these two pictures via Eqs. (C8) and (C9).

c. Target DFT is Real and Positive

Now we can use the relationship between the continuous and discrete pictures to show that we can define a 2D phase retrieval problem where the DFT we wish to reconstruct is both real and positive. Knowing this, justifies using these facts as constraints in the HIO algorithm to get reliable performance.

a. Positivity constraint. First we can show that $F[k, m]$ will be positive when we choose H_D to commute with H and as a consequence of choosing a triangular windowing function

$$\Pi_T(t, z) = \Lambda_T(t) \cdot \Lambda_T(z), \quad (\text{C11})$$

$$\Lambda_T(t) = \begin{cases} 1 - 2|t|/T & |t| \leq T/2, \\ 0 & \text{else.} \end{cases} \quad (\text{C12})$$

This follows from establishing that $G(\omega, \eta)$ is positive, which then ensures $F[k, m]$ is positive as it is related to this continuous function via Eq. (C8). One can write $G(\omega, \eta)$ as a convolution as

$$G(\omega, \eta) = \mathcal{FT}[f(t, z)] * \mathcal{FT}[\Pi_T(t, z)] * \mathcal{FT}[\text{III}_{\Delta t}(t, z)]. \quad (\text{C13})$$

We will evaluate each term in this convolution in turn. First we have

$$\mathcal{FT}[f(t, z)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(t, z) e^{-i\omega t} e^{-i\eta z} dt dz \quad (\text{C14})$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \langle \psi | e^{itH + izH_D} | \psi \rangle e^{-i\omega t} e^{-i\eta z} dt dz. \quad (\text{C15})$$

Now we use the fact that $[H, H_D] = 0$ to write both Hamiltonians in their shared eigenbasis as $H = \sum_i E_i \Pi_i$ and $H_D = \sum_i E_i^D \Pi_i$, where $\Pi_i = |E_i\rangle \langle E_i|$ is the projector onto this shared eigenbasis. This gives

$$\begin{aligned} \mathcal{FT}[f(t, z)] &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \langle \psi | \Pi_i | \psi \rangle e^{itE_i + izE_i^D} e^{-i\omega t} e^{-i\eta z} dt dz \quad (\text{C16}) \\ &= \sum_i |\langle \psi | E_i \rangle|^2 \int_{-\infty}^{\infty} e^{it(E_i - \omega)} dt \int_{-\infty}^{\infty} e^{iz(E_i^D - \eta)} dz \end{aligned}$$

$$(\text{C17})$$

$$= 4\pi^2 \sum_i |\langle \psi | E_i \rangle|^2 \delta(\omega - E_i) \delta(\eta - E_i^D). \quad (\text{C18})$$

This function is positive for all ω and η . This is why we have introduced a commuting dummy Hamiltonian. If H

and H_D do not commute is possible to have negative values in this step.

The second term to evaluate is the Fourier transform of the windowing function. This is

$$\mathcal{FT}[\Pi_T(t, z)] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Lambda_T(t) \Lambda_T(z) e^{-i\omega t} e^{-i\eta z} dt dz \quad (\text{C19})$$

$$= \frac{8 \sin^2\left(\frac{T\omega}{4}\right)}{T\omega^2} \frac{8 \sin^2\left(\frac{T\eta}{4}\right)}{T\eta^2}, \quad (\text{C20})$$

which we can see is positive for all ω and η . Note that one could have used any windowing function with a positive Fourier transform and better choices than a triangular windows deserve further investigation. Finally, as discussed in Appendix A (the Fourier transform of a Dirac comb is another Dirac comb), the effect of $\mathcal{FT}[\text{III}_{\Delta t}(t, z)]$ on Eq. (C13), we can establish that $G(\omega, \eta)$ is a positive function and therefore that $F[k, m] = G(\omega_k, \eta_m)$ is always positive. This allows us to use this as a constraint to drive the convergence of the HIO algorithm.

b. Real constraint. We can show that F is real very simply, thus making this a second suitable driving constraint. In order for F to be real the 2D signal must satisfy

$$f[j, l] = f^*[N - j, M - l], \quad (\text{C21})$$

modulo N and M and where $j = 0, 1, \dots, N - 1$ and $l = 0, 1, \dots, M - 1$. This is a generic fact about the DFT and can be found in any textbook and easily verified. Any signal which satisfies this condition will have a real discrete Fourier transform.

We can now show that our signal satisfies this condition using the fact that

$$\hat{f}[j, l] = \hat{f}^*[-j, -l], \quad (\text{C22})$$

which follows from the definition of $\hat{f}[j, l] = f(t_j, z_l) \Pi_T(t_j, z_l)$ and the fact that the continuous functions satisfy $f(t, z) = f^*(-t, -z)$ and $\Pi_T(t, z) = \Pi_T^*(-t, -z)$.

We can combine this with the definition of $f[j, l]$ to show that it satisfies Eq. (C21). Writing this out explicitly we see that

$$f^*[N - j, M - l] = \sum_{l', j' \in \mathbb{Z}} \hat{f}^*[N - j - Nj', M - l - Ml'] \quad (\text{C23})$$

$$= \sum_{l'', j'' \in \mathbb{Z}} \hat{f}^*[-(j - Nj''), -(l - Ml'')] \quad (\text{C24})$$

$$= \sum_{l'', j'' \in \mathbb{Z}} \hat{f}[j - Nj'', l - Ml''] \quad (\text{C25})$$

$$= f[j, l]. \quad (\text{C26})$$

Therefore $f[j, l]$ satisfies Eq. (C21) and therefore its discrete Fourier transform $F[k, m]$ is real. Therefore we can use this fact to drive the HIO algorithm.

4. Implementation and Numerical Results

We test the performance of this algorithm for instances of the Fermi-Hubbard model as explained in Appendix A. We take as target input state $|\psi\rangle$ the uniform superposition over computational basis states. As dummy Hamiltonian H_D we choose the total number operator

$$H_D = \sum_{i \in V, \sigma} a_{i\sigma}^\dagger a_{i\sigma}. \quad (\text{C27})$$

Under the Jordan-Wigner encoding this maps to

$$H_D \rightarrow \frac{1}{2} \sum_{i \in V, \sigma} (I_{i\sigma} - Z_{i,\sigma}). \quad (\text{C28})$$

This makes e^{itH_D} trivial to implement via a single layer of single-qubit rotation gates. When $|\psi\rangle$ is a superposition of computational basis states we then satisfy the requirement that $|\psi\rangle$ must not be eigenstate of H_D . For examples such as this, we could also think of implementing e^{itH_D} as performing a simple state preparation.

a. Image Reconstruction

First we demonstrate that the HIO algorithm works well given quantum signals set up according to the previous section. We test this numerically using a 2D spin Fermi-Hubbard model as H in Eq. (C3), with the hopping strength $\tau = 1$ and on-site interactions $U = 4$.

In addition to the constraints that the signal is real and positive, we also incorporate knowledge of the phases for $f[0, l]$. This information can be computed classically in any given experiment as these points correspond to evolution under H_D only, provided the initial state it acts upon is also easy to classically simulate.

Here we show a typical example of signal reconstruction in Fig. 13 for a 2×2 Fermi-Hubbard model. For this case we have set $N = M = 225$, time $T = 112$ and chosen $|\psi\rangle$ as the uniform superposition over computational basis states. H has been normalized so that its eigenvalues lie between 0 and π .

The left of Fig. 13 displays in a colormap the value of the ideal two-dimensional time-series $F[k, m]$ obtained numerically through brute force simulation. In the center of Fig. 13 we show the absolute values $|f[j, l]|$. This is

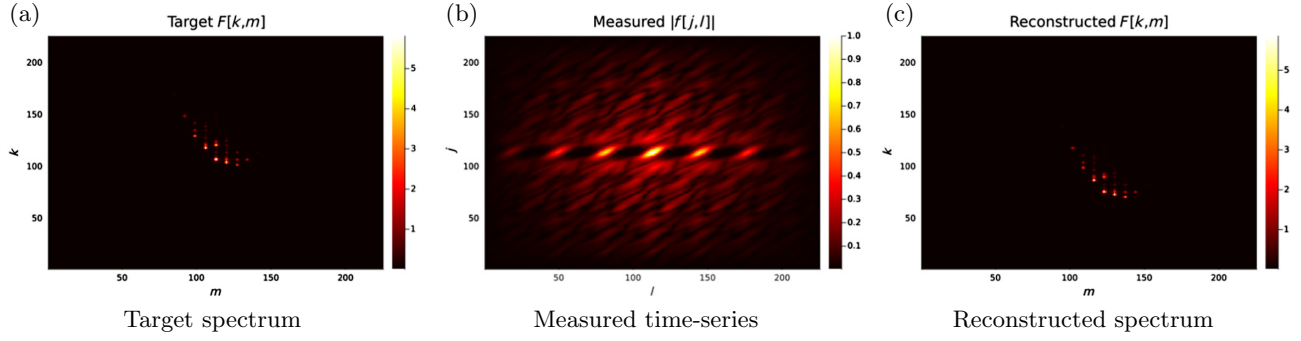


FIG. 13. An example of 2D phase retrieval for a 2×2 Fermi-Hubbard Hamiltonian with $N \times M = 225 \times 225$. We take $|\psi\rangle$ to be a uniform superposition over computational basis states. The true signal $f[j, l]$ is shown left, the absolute values $|F[k, m]|$ are shown in the center and finally the reconstructed $f[j, l]$ is shown to the right, where it has been recovered up to a trivial ambiguity corresponding to a translation.

the input to the HIO algorithm and is what would be measured via a quantum circuit implementing time evolution on the state $|\psi\rangle$. Observe that we also see the effect of the 2D triangular windowing function, so that the magnitude of $|f[j, l]|$ decreases toward the outer edges of the plot. Finally the reconstructed signal is shown on the right of Fig. 13. This was obtained with $\beta = 0.9$ and $L = 5000$ iterations of the HIO algorithm. The reconstruction is qualitatively good and in agreement with the true signal for the noiseless case, up to a translation, consistent with the trivial ambiguities discussed above and that can be compensated for.

b. Spectrum recovery

As mentioned earlier, the last step is to extract the desired information about the spectrum of H from the recovered signal by computing the DFT of the first column of the two dimensional time-series matrix, i.e., $F[k] := \mathcal{DFT}[f[j, 0]]$, corresponding to a zero virtual

time evolution $z = 0$. The result of this is shown in Fig. 14. On the left we reconstruct the spectrum from the ideal noiseless time-series, where in the right we included the effect of sampling noise, where we model a shot noise corresponding to 1000 samples (circuit shots) per required expectation value.

We compare the phase-retrieval recovery (orange) to the reference signal (blue) obtained with an ideal Hadamard-like quantum circuit and using statistical QPE with same parameters T and Δt , including the effect of the triangular window.

To obtain these plots we have shifted the recovered signal so that the trivial ambiguity is eliminated. This is to demonstrate the degree of overlap between the true and recovered signal up to this ambiguity. We see that in the noiseless case the spectrum is recovered well. When we introduce sampling noise the shape and location of the peaks in the spectrum are recovered, but there is a constant positive offset in the spectrum, that could potentially be filtered out without damaging the reconstructed spectrum too

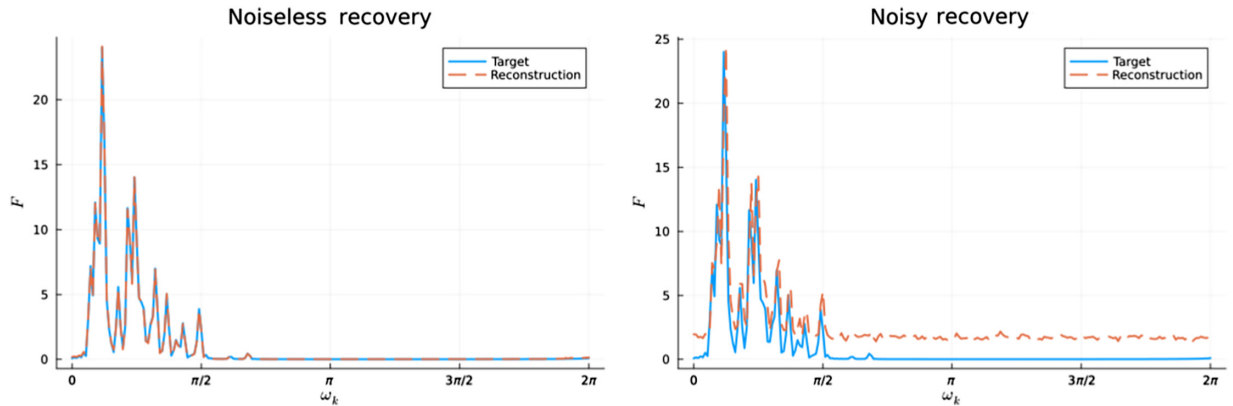


FIG. 14. The results of plotting $F[k] := \mathcal{DFT}[f[j, 0]]$. Here we show the result for the equal superposition over all computational basis states, with $T = 112$, $N = 225$ and $M = 225$ and H has been normalized so that its eigenvalues lie between 0 and π . Here the ideal solution includes the triangular window by definition. On the left we show the noiseless case and on the right we show the case for 1000 samples per time-series point. The true and recovered signals are shown and overlap almost exactly in the noiseless case.

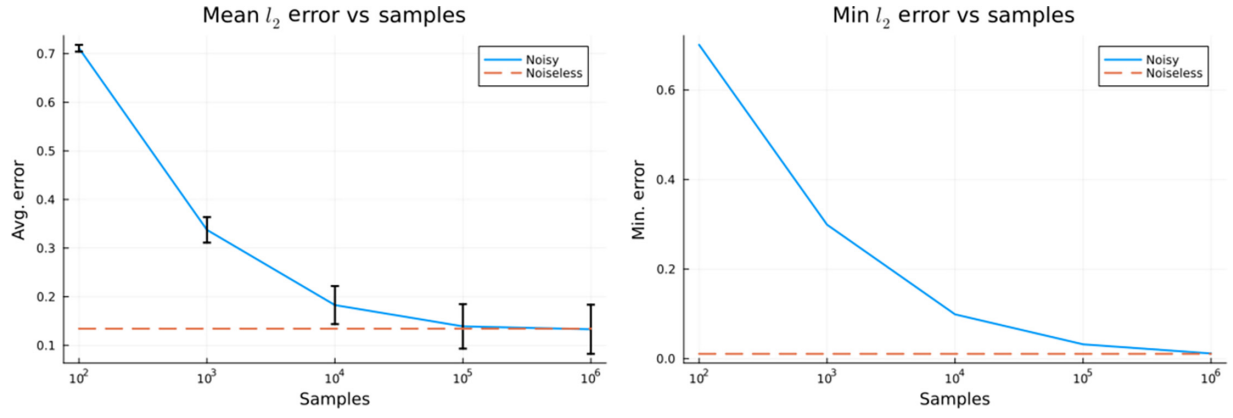


FIG. 15. For the 2×2 Fermi-Hubbard example. Here we show the average (left) and minimum (right) error $\epsilon_{l_2} = \frac{1}{N} \sqrt{\sum_k |F[k] - F_{\text{true}}[k]|^2}$ as a function of the number of samples used to estimate each $|f[j, l]|$. At each fixed number of samples we average over 100 random initializations of the hybrid input-output algorithm (HIO) algorithm with. Error bars indicate one standard deviation from this mean. We also show the error achievable average and minimum error in the absence of noise.

much. In Fig. 15 we show the scaling of the average and minimum of the error $\epsilon_{l_2} = 1/N \sqrt{\sum_k |F[k] - F_{\text{true}}[k]|^2}$ with number of samples used to estimate each $|f[j, l]|$. We average the error 100 random initialization for the HIO algorithm. We simulate using between 10^2 to 10^6 samples per estimation of each $|f[j, l]|$. We look at the minimum error as the HIO algorithm is sensitive the initialization state and can be restarted when stagnation occurs.

APPENDIX D: COMPARISON WITH THE STANDARD APPROACH

There are two key advantages of phase retrieval compared with the standard approach to statistical phase estimation: each qubit no longer needs to interact with an ancilla “control” qubit, and there is no increase in circuit complexity caused by the use of controlled unitary operations. One can consider many metrics for circuit complexity; here, we will focus on the number of CNOT gates required, and the quantum circuit depth.

To get a sense of how large the reduction in complexity can be, first observe that if the quantum hardware allows for all-to-all connectivity, any quantum circuit computing a unitary operator U can be converted into a quantum circuit for computing controlled- U by replacing every CNOT gate with a Toffoli gate, and every single-qubit unitary V with a controlled- V operator. A Toffoli gate can be implemented using six CNOT gates and controlled- V can be implemented using two CNOT gates for any single-qubit unitary V . So, if the original quantum circuit had c CNOT gates and s single-qubit gates, the controlled circuit will have at most $6c + 2s$ CNOT gates—at most a constant factor increase in complexity, if $c \geq s$ (for example), but still relatively substantial for near-term applications. However, the difference in complexity between the two situations can be greater if

there are restrictions on hardware connectivity, or if one considers circuit depth instead.

Here we approximately calculate the reduction in circuit complexity achieved by phase retrieval for several simple examples: the 1D Ising model with transverse field, both all-to-all and with matching hardware connectivity, and the spinless 2D Fermi-Hubbard model with square-lattice hardware connectivity. In each case, we consider the complexity of implementing k Trotter steps—i.e., k repetitions of an operator of the form $\prod_{j=1}^m e^{i\theta_j H_j}$ for a Hamiltonian of the form $H = \sum_j H_j$. Note that it is important to consider $k > 1$ steps because the cost of interacting with the control qubit can be amortized across multiple steps by preparing a GHZ state, as we will see below. This approach may add its own difficulties in terms of rendering the circuit more prone to decoherence, but we will ignore this issue for simplicity. We also stress that there is no guarantee that the implementations we describe here are optimal.

- (1) **1D Ising model with transverse field.** This Hamiltonian is defined as $H = \sum_{i=1}^{n-1} Z_i Z_{i+1} + \sum_{j=1}^n X_j$. A Trotter step can be implemented using $2(n-1)$ CNOT gates and in CNOT depth 4. To implement a controlled Trotter step, we need to implement controlled- $e^{i\theta ZZ}$ and controlled- $e^{i\theta X}$ gates. These have circuits using four and two CNOT gates, respectively. With all-to-all connectivity, then, each Trotter layer can be implemented using $4(n-1) + 2n = 6n - 4$ CNOT gates. Implementing these in a straightforward way would lead to a high-depth circuit, as each controlled unitary acts on the same control qubit. However, we can reduce the depth by constructing a GHZ state of n qubits, where each gate that needs to be applied in parallel uses a different qubit of this state. The CNOT depth of

TABLE II. The cost of implementing k Trotter layers for several Hamiltonians, an n qubits transverse field Ising model (TFIM) and an $n \times n$ spinless Fermi-Hubbard model. “Depth” is CNOT depth. Note that the algorithms achieving the minimal CNOT depth and CNOT count may be different. We consider the spinless Fermi-Hubbard model for simplicity, and these calculations are approximate.

Model	H/w	CNOTs (PR)	CNOTs (no PR)	Depth (PR)	Depth (no PR)
1D TFIM	All-to-all	$(2n - 2)k$	$(6n - 4)k$	$4k$	$2\lceil \log_2 n \rceil + 10k$
1D TFIM	1D	$(2n - 2)k$	$6(n - 1) + (6n - 4)k$	$4k$	$6\lceil n/2 \rceil + 10k$
2D FH	2D	$32kn(n - 1)/2$	$48kn(n - 1)/2 + 6\lceil (n - 1)^2/2 \rceil$	$32k$	$48k + 3(n - 2)$

producing such a state given all-to-all connectivity is at most $\lceil \log_2 n \rceil$. Following this, we can split the time-evolution steps into three groups (ZZ terms on (odd, even) qubits, ZZ terms on (even, odd) qubits, and X terms) and implement each group in parallel. At the end, the GHZ state is uncomputed, before measuring the control qubit as in the usual protocol. The total CNOT depth (with all-to-all connectivity) is then $2\lceil \log_2 n \rceil + 10k$ as each layer can be implemented in CNOT depth 10.

With 1D connectivity, the situation is more difficult as we need to create a distributed GHZ state. This can be done with a small extra cost in gate count, but a linear cost in depth. Now we include an ancilla qubit for controlled operations next to each qubit in the original 1D graph. Once these qubits have been prepared in a GHZ state, we can perform controlled operations across any pair of qubits in the original graph without the need for any swaps. The map $|000\rangle \mapsto |000\rangle$, $|100\rangle \mapsto |101\rangle$ can be carried out using three CNOT gates in 1d connectivity (CNOT₁₂, CNOT₂₃, CNOT₁₂), so the required state can be prepared using $3(n - 1)$ CNOT gates and in CNOT depth $3\lceil n/2 \rceil$. Once this has been prepared, the costs for the time-evolution part are the same as for all-to-all connectivity. The total CNOT cost for k Trotter steps, including the cost of uncomputing the GHZ state, is then $6(n - 1) + (6n - 4)k$, and the CNOT depth is $6\lceil n/2 \rceil + 10k$.

- (2) **2D Fermi-Hubbard model.** Consider an $n \times n$ Fermi-Hubbard model without spin on an 2D lattice. Consider even n only and use the compact fermionic encoding [47] and layout shown in Fig. 16. The full Hamiltonian is made of horizontal $\sum_{\langle i,j \rangle} (X_i X_j Y_{f(i,j)} + Y_i Y_j Y_{f(i,j)})/2$ and vertical $\sum_{\langle i,j \rangle} \pm (X_i X_j X_{f(i,j)} + Y_i Y_j X_{f(i,j)})/2$ interactions, where the sum is over pairs of horizontally or vertically adjacent sites on an $n \times n$ lattice and $f(i,j)$ are face qubits placed on this lattice. The Hamiltonian can be split into four sets of $n(n - 1)/2$ interactions. Within each group all interactions can be performed in parallel. Every interaction consists of two weight-3 (at most) Pauli interactions. Therefore each interaction can be done using 8 CNOT gates, meaning that a full Trotter step will have a depth of

32 CNOT gates and the total number of CNOT gates will be $32n(n - 1)/2$. This calculation is approximate and simply treats edge interactions identically to those in the bulk. When performing controlled evolution under this Hamiltonian one possible layout of control ancilla qubits is shown in Fig. 16. As in the case of the Ising model above, we consider creating a distributed GHZ state so that we can implement local control. This particular selection uses the remaining face qubits not used by the compact encoding. Now every “controlled” interaction can be performed by evolving under two weight-4 Pauli terms, where the interactions have been augmented so that they include a Pauli-Z on the local control ancilla. Therefore each interaction can be done using 12 CNOT gates, meaning that a full Trotter step will have a depth of 48 CNOT gates and the total number of CNOT gates will be $48n(n - 1)/2$. (This all uses the standard approach to performing a controlled rotation by a weight- k Pauli operation [48]) The additional cost of creating the distributed GHZ state depends on the number of control ancillas $\lceil (n - 1)^2/2 \rceil$ and their connectivity. The total count of CNOT gates will be $3 \times \lceil (n - 1)^2/2 \rceil$. The depth is approximately $3(n - 2)/2$ (for even lattices with

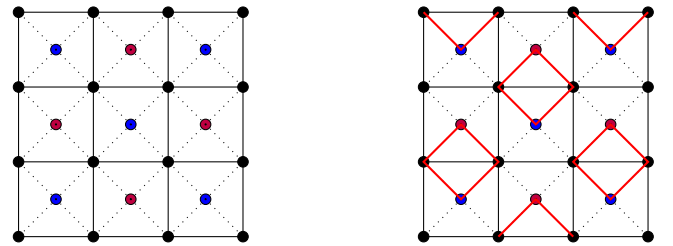


FIG. 16. A possible layout of the spinless Fermi-Hubbard model on a lattice with 2D connectivity. The connectivity is indicated by dashed lines while the Fermi-Hubbard lattice interactions are indicated by solid lines. The face qubits in purple are needed for the compact encoding. The face qubits in blue are used to implement geometrically local control. The diagram on the right indicates one of the four Trotter layer groupings of interactions which can be implemented in parallel, in this case horizontal hopping interactions. With the control and encoding qubits these interactions are at most Pauli weight-4.

$n > 2$). One can begin building the GHZ state from the central control qubit and move outward along the branches approximately in parallel. It takes 3 CNOT gates to spread the entanglement to the nearest control qubit, shown in blue. This is done in the same manner as described for the Ising model.

In all of the above costings, we do not include the cost of preparing (or uncomputing) the initial state $|\psi\rangle$ —which we assume to be simple—nor the cost of preparing a superposition of $|\psi\rangle$ with another state. In some situations, this may be fairly substantial, e.g., if this superposition is a GHZ state.

-
- [1] S. Duckett and B. Gilbert, *Foundations of Spectroscopy*, Oxford chemistry primers (Oxford University, Oxford, United Kingdom, 2000), ISBN 9780198503354, <https://books.google.co.uk/books?id=boG7ngEACAAJ>
 - [2] J. P. F. LeBlanc *et al.*, Solutions of the two-dimensional hubbard model: Benchmarks and results from a wide range of numerical algorithms, *Phys. Rev. X* **5**, 041041 (2015).
 - [3] Bela Bauer, Sergey Bravyi, Mario Motta, and Garnet Kin-Lic Chan, Quantum algorithms for quantum chemistry and quantum materials science, *Chem. Rev.* **120**, 12685 (2020), ISSN 1520-6890.
 - [4] R. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, Simulating physical phenomena by quantum networks, *Phys. Rev. A* **65**, 042323 (2002).
 - [5] Rolando D. Somma, Quantum eigenvalue estimation via time series analysis, *New J. Phys.* **21**, 123025 (2019).
 - [6] Thomas E. O'Brien, Brian Tarasinski, and Barbara M. Terhal, Quantum phase estimation of multiple eigenvalues for small-scale (noisy) experiments, *New J. Phys.* **21**, 023022 (2019).
 - [7] Lin Lin and Yu Tong, Heisenberg-limited ground-state energy estimation for early fault-tolerant quantum computers, *PRX Quantum* **3**, 010318 (2022).
 - [8] K. Wan, M. Berta, and E. T. Campbell, Randomized quantum algorithm for statistical phase estimation, *Phys. Rev. Lett.*, **129**, 030503 (2022).
 - [9] Guoming Wang, Daniel Stilck França, Ruizhe Zhang, Shuchen Zhu, and Peter D. Johnson, Quantum algorithm for ground state energy estimation using circuit depth with exponentially improved dependence on precision, *Quantum* **7**, 1167 (2023), ISSN 2521-327X.
 - [10] Nick S. Blunt, Laura Caune, Róbert Izsák, Earl T. Campbell, and Nicole Holzmann, Statistical phase estimation and error mitigation on a superconducting quantum processor, *PRX Quantum* **4**, 040341 (2023).
 - [11] T. E. O'Brien, S. Polla, N. C. Rubin, W. J. Huggins, S. McArdle, S. Boixo, J. R. McClean, and R. Babbush, Error mitigation via verified phase estimation, *PRX Quantum* **2**, 020317 (2021).
 - [12] A. Baroni, J. Carlson, R. Gupta, Andy C. Y. Li, G. N. Perdue, and A. Roggero, Nuclear two point correlation functions on a quantum computer, *Phys. Rev. D* **105**, 074503 (2022).
 - [13] Nishad Maskara, Stefan Ostermann, James Shee, Marcin Kalinowski, Abigail McClain Gomez, Rodrigo Araiza Bravo, Derek S. Wang, Anna I. Krylov, Norman Y. Yao, Martin Head-Gordon, Mikhail D. Lukin, and Susanne F. Yelin, Programmable simulations of molecules and materials with reconfigurable quantum processors (2023).
 - [14] B. Bauer, D. Wecker, A. J. Millis, M. B. Hastings, and M. Troyer, Hybrid quantum-classical approach to correlated materials, *Phys. Rev. X* **6**, 031045 (2016).
 - [15] Hans Hon Sang Chan, Richard Meister, Matthew L. Goh, and Bálint Koczor, Algorithmic shadow spectroscopy, (2024).
 - [16] Y. Yang, A. Christianen, M. C. Bañuls, D. S. Wild, and J. I. Cirac, Phase-sensitive quantum measurement without controlled operations, *Phys. Rev. Lett.* **132**, 220601 (2024).
 - [17] A. E. Russo, K. M. Rudinger, B. C. A. Morrison, and A. D. Baczewski, Evaluating energy differences on a quantum computer with robust phase estimation, *Phys. Rev. Lett.* **126**, 210501 (2021).
 - [18] Adriaan Walther, The question of phase retrieval in optics, *Optica Acta: Int. J. Opt.* **10**, 41 (1963).
 - [19] Oren Raz, Nirit Dudovich, and Boaz Nadler, Vectorial phase retrieval of 1-D signals, *IEEE Trans. Signal Process.* **61**, 1632 (2013).
 - [20] James Fienup, Phase retrieval algorithms: A comparison, *Appl. Opt.* **21**, 2758 (1982).
 - [21] William J. Huggins, Joonho Lee, Unpil Baek, Bryan O'Gorman, and K. Birgitta Whaley, A non-orthogonal variational quantum eigensolver, *New J. Phys.* **22**, 073009 (2020).
 - [22] S. Lu, M. C. Bañuls, and J. I. Cirac, Algorithms for quantum simulation at finite energies, *PRX Quantum* **2**, 020321 (2021).
 - [23] A. Schuckert, A. Bohrdt, E. Crane, and M. Knap, Probing finite-temperature observables in quantum simulators of spin systems with short-time dynamics, *Phys. Rev. B* **107**, L140410 (2023).
 - [24] K. Hémery, K. Ghanem, E. Crane, S. L. Campbell, J. M. Dreiling, C. Figgatt, C. Foltz, J. P. Gaebler, J. Johansen, M. Mills, S. A. Moses, J. M. Pino, A. Ransford, M. Rowe, P. Siegfried, R. P. Stutz, H. Dreyer, A. Schuckert, and R. Nigmatullin, Measuring the Loschmidt amplitude for finite-energy properties of the Fermi-Hubbard model on an ion-trap quantum computer, *PRX Quantum* **5**, 030323 (2024).
 - [25] In our experiments, the superposition used was essentially a GHZ state, but we do not believe this is a fundamental requirement.
 - [26] We use square brackets to stress that the function takes discrete input.
 - [27] A. V. Oppenheim, A. S. Willsky, and S. H. Nawab, *Signals & Systems*, Prentice-Hall signal processing series, (Prentice-Hall International, London, United Kingdom, 1997), ISBN 9780136511755, <https://books.google.co.uk/books?id=O9ZHSAAACAAJ>.
 - [28] Rick P. Millane, Phase retrieval in crystallography and optics, *J. Opt. Soc. Am. A-Optics Image Science and Vision* **7**, 394 (1990).
 - [29] J. Dainty and James Fienup, Phase retrieval and image reconstruction for astronomy, *Image Recovery: Theory Appl.* **13**, 01 (1987).

- [30] Joseph Vovrosh, Kiran E. Khosla, Sean Greenaway, Christopher Self, M. S. Kim, and Johannes Knolle, Simple mitigation of global depolarizing errors in quantum simulations, *Phys. Rev. E* **104**, 035309 (2021), ISSN 2470-0053.
- [31] Mingpu Qin, Thomas Schäfer, Sabine Andergassen, Philippe Corboz, and Emanuel Gull, The hubbard model: A computational perspective, *Annu. Rev. Condens. Matter Phys.* **13**, 275 (2022), ISSN 1947-5462.
- [32] Since we only work with relatively modest system sizes and total evolution times (since else we would have no interest in implementing *control-free* phase estimation), we note that the continuum limit behavior of these roots is not an immediate barrier for our method.
- [33] For $G[j]$ to be well-defined, the subsets of eigenstates of H with which $|\Phi\rangle$ and $|\psi\rangle$ have significant overlap should not be disjoint, as this would cause $|f_2[j]|$ to vanish.
- [34] Remark that the relaxation we use is slightly different to the one implemented in Ref. [19].
- [35] Yu.M. Bruck and L. G. S Sodin, On the ambiguity of the image reconstruction problem, *Opt. Commun.* **30**, 304 (1979), ISSN 0030-4018, <https://www.sciencedirect.com/science/article/pii/0030401879903584>.
- [36] Dani Kogan, Yonina C. Eldar, and Dan Oron, On the 2D phase retrieval problem, *IEEE Trans. Signal Process.* **65**, 1058 (2017), ISSN 1941-0476.
- [37] Tamir Bendory, Robert Beinert, and Yonina C. Eldar, *Fourier Phase Retrieval: Uniqueness and Algorithms* (Springer International Publishing, Cham, 2017), p. 55, ISBN 978-3-319-69802-1.
- [38] Eliyahu Osherovich, Numerical methods for phase retrieval, [arXiv:1203.4756](https://arxiv.org/abs/1203.4756). URL <https://api.semanticscholar.org/CorpusID:14566173>.
- [39] Note that $T = T'$ is not a fundamental requirement and other choices are possible and potentially beneficial.
- [40] Julius O. Smith, *Mathematics of the Discrete Fourier Transform (DFT)* (W3K Publishing, Palo Alto, United States, 2007), ISBN 978-0-9745607-4-8, <http://www.w3k.org/books/>.
- [41] We have plotted the zero frequency in the center.
- [42] Daniel P. Arovas, Erez Berg, Steven A. Kivelson, and Srinivas Raghu, The hubbard model, *Annu. Rev. Condens. Matter Phys.* **13**, 239 (2022), ISSN 1947-5462.
- [43] Robert Beinert, One-dimensional phase retrieval with additional interference intensity measurements, *Results Math.* **72**, 1 (2017).
- [44] The absolute value here means the absolute value of every element $f[j, l]$ of the $N \times M$ matrix f .
- [45] Emmanuel J. Candes, Thomas Strohmer, and Vladislav Voroninski, Phaselift: Exact and stable signal recovery from magnitude measurements via convex programming, [arXiv:1109.4499](https://arxiv.org/abs/1109.4499).
- [46] Kishore Jaganathan, Yonina C. Eldar, and Babak Hassibi, Phase retrieval: An overview of recent developments, [arXiv:1510.07713](https://arxiv.org/abs/1510.07713).
- [47] C. Derby, J. Klassen, J. Bausch, and T. Cubitt, Compact fermion to qubit mappings, *Phys. Rev. B* **104**, 035118 (2021).
- [48] Michael A. Nielsen and Isaac L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University, Cambridge, United Kingdom, 2010).