Efficient fully-coherent quantum signal processing algorithms for real-time dynamics simulation ⊘

John M. Martyn ✉ ⑩ ; Yuan Liu ⑩ ; Zachary E. Chin ⑩ ; Isaac L. Chuang ⑩

Check for updates

View Online   Export Citation   CrossMark

Articles You May Be Interested In

Students' mental models on the solubility and solubility product concept

AIP Conference Proceedings (May 2017)

POST-DETONATION ENERGY RELEASE FROM TNT-ALUMINUM EXPLOSIVES

AIP Conference Proceedings (December 2007)

Dynamical systems analysis as an additional tool to inform treatment outcomes: The case study of a quantitative systems pharmacology model of immuno-oncology

Chaos (February 2021)
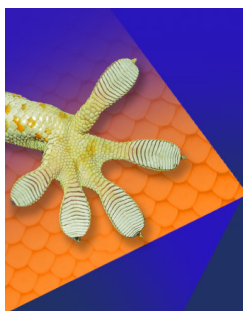
The Journal of Chemical Physics

Special Topic: Adhesion and Friction

Submit Today!

Efficient fully-coherent quantum signal processing algorithms for real-time dynamics simulation

View Online   Export Citation   CrossMark

John M. Martyn,[1,a)] [iD] Yuan Liu,[2] [iD] Zachary E. Chin,[3,4] [iD] and Isaac L. Chuang[2,3] [iD]

AFFILIATIONS

[1] Department of Physics, Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
[2] Department of Physics, Co-Design Center for Quantum Advantage, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
[3] Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
[4] Department of Chemistry, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA

[a)] Author to whom correspondence should be addressed: jmmartyn@mit.edu

ABSTRACT

Simulating the unitary dynamics of a quantum system is a fundamental problem of quantum mechanics, in which quantum computers are believed to have significant advantage over their classical counterparts. One prominent such instance is the simulation of electronic dynamics, which plays an essential role in chemical reactions, non-equilibrium dynamics, and material design. These systems are time-*dependent*, which requires that the corresponding simulation algorithm can be successfully concatenated with itself over different time intervals to reproduce the overall coherent quantum dynamics of the system. In this paper, we quantify such simulation algorithms by the property of being *fully-coherent*: the algorithm succeeds with arbitrarily high success probability $1 - \delta$ while only requiring a single copy of the initial state. We subsequently develop fully-coherent simulation algorithms based on quantum signal processing (QSP), including a novel algorithm that circumvents the use of amplitude amplification while also achieving a query complexity *additive* in time t, $\ln(1/\delta)$, and $\ln(1/\epsilon)$ for error tolerance ϵ: $\Theta(\|\mathcal{H}\||t| + \ln(1/\epsilon) + \ln(1/\delta))$. Furthermore, we numerically analyze these algorithms by applying them to the simulation of the spin dynamics of the Heisenberg model and the correlated electronic dynamics of an H_2 molecule. Since any electronic Hamiltonian can be mapped to a spin Hamiltonian, our algorithm can efficiently simulate time-dependent *ab initio* electronic dynamics in the circuit model of quantum computation. Accordingly, it is also our hope that the present work serves as a bridge between QSP-based quantum algorithms and chemical dynamics, stimulating a cross-fertilization between these exciting fields.

Published under an exclusive license by AIP Publishing. https://doi.org/10.1063/5.0124385

I. INTRODUCTION

Quantum computation owes its inception to the fundamental problem of Hamiltonian simulation,[1] wherein one aims to simulate the time evolution of a quantum system under a Hamiltonian $\mathcal{H}$ for a time t. Such a time evolution underpins nearly all dynamical processes as microscopic phenomena are governed by quantum mechanics. Efficient algorithms for Hamiltonian simulation are thus crucial for theoretical modeling of the physical world, including analyzing reaction mechanisms of complex chemical and biological systems,[2–5] probing non-equilibrium ultrafast dynamics,[6–8] understanding phases of strongly correlated condensed matter systems,[9,10] and simulating quantum field theories.[11,12]

Classical algorithms for simulating the time evolution of quantum systems have existed long before the inception of quantum computation. One major thrust of development comes from the need to model correlated electronic dynamics to high accuracy, where exciting developments have happened over the past few decades.[13] Besides the exact diagonalization (ED) method that scales exponentially with the system size, algorithms based on mean-field

theories[14–16] as well as their multi-configuration variants[17–19] are among the most computationally efficient methods, yet not applicable to strongly correlated electronic dynamics. More accurate classical methods for electronic dynamics include the time-dependent variants of high-level correlated electronic structure theories.[20–29] Many of these methods involve a truncation of the Hilbert space and have high-order polynomial scaling with the system size. In addition, real-time stochastic methods based on quantum Monte Carlo sampling[30–36] achieve a balance between computational cost and accuracy, yet their applicability is limited to short-time dynamics or small systems due to the notorious real-time phase problem on the sampling trajectories. In general, the phase problem cannot be resolved without truncating the sampling space and adding bias to the simulation results. A combination of the computationally efficient mean-field theories and more accurate high-level theories can be achieved in the embedding framework[37,38] by treating the most important parts of the system using correlated methods and the rest by mean-field theories.

These advancements in classical algorithm development for correlated electronic dynamics lean on the physical nature of classical computation. In particular, it is *necessary* to use an exponential amount of classical resources to represent the full dynamics of the entire Hilbert space of an interacting many-electron system, unless approximations are used. *Quantum algorithms*, on the other hand, hold the promise to overcome the difficulties encountered by classical algorithms by employing coherent superpositions and entanglement in quantum hardware.[39–41] While previous efforts leveraging quantum resources mostly focus on electronic structure problems,[39,42] in the present work, we, instead, are primarily interested in quantum dynamics, or equivalently Hamiltonian simulation.

Among a variety of efforts to subdue Hamiltonian simulation, prominent quantum algorithms include Trotterization and product formulas,[43–48] Taylor series truncation,[49–51] and quantum walks.[52–55] In recent years, however, Hamiltonian simulation algorithms have flourished with the advent of an algorithmic primitive known as *quantum signal processing* (QSP),[56,57] with QSP-based simulation algorithms touting a nearly optimal query complexity with respect to the simulation time t and error ϵ. From a high level, QSP provides a systematic method to apply a nearly arbitrary polynomial transformation to a quantum subsystem. Exploiting such flexibility, QSP-based simulation develops a polynomial transformation of the Hamiltonian that approximates the time evolution operator.

From an algorithmic viewpoint, a common theme among these Hamiltonian simulation algorithms, and quantum algorithms in general, is cascading together elementary subroutines to construct more powerful algorithms, whereas from an application point of view, the underlying electronic Hamiltonians that governs the dynamics are often time-dependent due to nuclei motion or interaction with external fields.[58–61] One common technique to deal with a time-dependent Hamiltonian is to trotterize[46,62,63] it into many time-independent pieces and then individually simulate each piece before coherently cascading them together. However, this cascading procedure is only viable if each time evolution subroutine succeeds with probability near unity, which indicates the importance of maintaining *coherence*, that is, retaining the correct wave function with arbitrarily high probability.

This is precisely the problem we study in this paper. We focus on *fully-coherent Hamiltonian simulation*, the goal of which is to accurately time evolve a state for a time t to within error ϵ with arbitrarily high success probability $1 - \delta$, provided only a single copy of the initial state. We also center our attention on *efficient* simulation algorithms, whose query complexities are polynomials in t, $\ln(1/\epsilon)$, and $\ln(1/\delta)$. Rigorous definitions for these terms are provided later in the text.

More concretely, we analyze three efficient fully-coherent Hamiltonian simulation algorithms, all of which are rooted in QSP. The first algorithm augments QSP-based simulation with conventional amplitude amplification to boost its success probability, which is the suggested remedy for the issue of post-selection in QSP algorithms.[64] This appends a multiplicative factor of $\ln(1/\delta)$ to the query complexity. The second algorithm integrates QSP-based simulation with the robust oblivious amplitude amplification protocol of Ref. 53, which contributes only an additive factor of $\ln(1/\delta)$.

In contrast, the third algorithm we present introduces a novel QSP technique: it first compresses the spectrum of the Hamiltonian with an affine transformation and subsequently applies QSP to it with a polynomial that approximates the time evolution operator only over the range of the compressed spectrum. By incorporating such a *pre-transformation* before applying QSP, this algorithm circumvents the need for amplitude amplification and also attains a query complexity additive in $\ln(1/\delta)$ instead of multiplicative: $\Theta(\|\mathcal{H}\||t| + \ln(1/\epsilon) + \ln(1/\delta))$. We further demonstrate that the probability of failure is dictated by the error of the complex exponential approximation such that $\delta = \Theta(\epsilon)$, which may be easily tuned by the choice of QSP polynomial. We dub this algorithm "coherent one-shot Hamiltonian simulation." In Table I, we summarize the query complexities of the fully-coherent simulation algorithms discussed in this work.

For completeness, we note that these algorithms can be seen as alternatives to the single-ancilla QSP algorithm of Ref. 57, which is a modified version of QSP that can also achieve fully-coherent simulation. However, whereas single-ancilla QSP is predicated on a set of constraints, which as we discuss later can inhibit its robustness to noise, here, we use new techniques such as the pre-transformation to achieve fully-coherent simulation. In addition, our algorithms are

TABLE I. Query complexities of the three efficient fully-coherent Hamiltonian simulation algorithms discussed in this paper. These include conventional QSP [denoted "QSP-LCU" in reference to the linear combination of unitaries (LCU) circuit used in its construction] augmented with amplification (AA); in particular, QSP-LCU + conventional AA and QSP-LCU + robust oblivious AA, both described in Sec. II. We also include our coherent one-shot simulation algorithm of Sec. III. In these expressions, t is the simulation time, α is an upper bound on the norm $\|\mathcal{H}\|$, ϵ is the error in the approximation of $e^{-i\mathcal{H}t}$, and δ is the probability of failure.

Simulation algorithm	Query complexity
QSP-LCU + conventional AA (Sec. II)	$\Theta\left(\ln\left(\frac{1}{\delta}\right)\left(\alpha|t| + \frac{\ln(1/\epsilon)}{\ln(e+\ln(1/\epsilon)/\alpha|t|)}\right)\right)$
QSP-LCU + robust oblivious AA (Sec. II)	$\Theta\left(\alpha|t| + \frac{\ln(1/\epsilon)}{\ln(e+\ln(1/\epsilon)/\alpha|t|)}\right)$
Coherent one-shot simulation (Sec. III)	$\Theta\left(\alpha|t| + \ln\left(\frac{1}{\epsilon}\right) + \ln\left(\frac{1}{\delta}\right)\right)$

constructed through recent generalizations of QSP known as the quantum eigenvalue transformation (QET) and the quantum singular value transformation (QSVT), which extend the framework of qubitization used in single-ancilla QSP and enable a more intuitive algorithmic development. We hence focus on the aforementioned three algorithms and briefly touch base with single-ancilla QSP later.

The rest of this paper is organized as follows: In Sec. II, we summarize conventional QSP-based Hamiltonian simulation as initially laid out in Ref. 64 and then integrate this algorithm with amplitude amplification (both conventional, and robust oblivious) to make it fully-coherent. In Sec. III, we develop coherent one-shot Hamiltonian simulation and then compare numerically the complexities of these algorithms as a function of simulation error and simulation time in Sec. IV. We thereafter use these algorithms in Sec. V to simulate the Heisenberg model with both time-independent and time-dependent external onsite fields, as well as electronic migration dynamics in a hydrogen molecule. Finally, Sec. VI concludes this work and discusses future research directions. In aggregate, we hope that this paper serves as an introduction to QSP-based quantum algorithms for the electronic dynamics community and spurs a cross-pollination between the two fields.

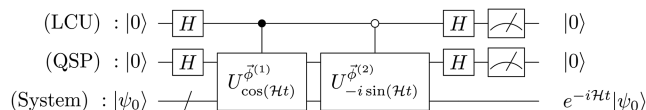## II. CONVENTIONAL QSP-BASED HAMILTONIAN SIMULATION

In order to establish the foundations of this work, we first review the conventional QSP-based Hamiltonian simulation algorithm based on a linear combination of untaries (LCU) circuit. Paralleling the original presentation in Ref. 64, we describe the procedure of this algorithm in Sec. II A, followed by a query complexity analysis in Sec. II B. For notational simplicity, we shall refer this approach as the "QSP-LCU" method. In Sec. II C, we discuss two methods for constructing fully-coherent versions of the QSP-LCU method via amplitude amplification, the first using conventional amplitude amplification and the second employing the robust oblivious amplitude amplification protocol of Ref. 49.

This section, as well as the rest of this paper, will assume familiarity with the basics of QSP, the quantum eigenvalue transformation (QET), and the quantum singular value transformation (QSVT). For a brief review of these fundamental concepts, see Appendix A.

A. Procedure

In the setup of this problem, we assume access to the Hamiltonian $\mathcal{H}$, of which we desire a unitary block encoding such that we may solve this problem with QSP techniques, in particular, the quantum eigenvalue transformation (QET). However, such a unitary block encoding is only realizable if $\|\mathcal{H}\| \leq 1$. In general, then, we instead determine an $\alpha \geq \|\mathcal{H}\|$ and construct a unitary block encoding of $\mathcal{H}/\alpha$. This requires some prior knowledge about $\mathcal{H}$, but fortunately such a block encoding can be achieved for a large class of Hamiltonians.[57,64] For simplicity, we will also assume that $\mathcal{H}/\alpha$ is encoded in the $|0\rangle\langle 0|$ matrix element of the unitary, but the following results may be easily adapted to other encodings as well.

With this rescaled block encoding, one can equivalently imagine that our goal is to simulate the time evolution of a system

FIG. 1. A quantum circuit, known as a linear combination of unitaries (LCU) circuit, that can apply the time evolution operator $\cos(\mathcal{H}t) - i\sin(\mathcal{H}t) = e^{-i\mathcal{H}t}$ to $|\psi_0\rangle$ in conventional QSP-based Hamiltonian simulation. Here, $U_{\cos(\mathcal{H}t)}^{\vec{\phi}^{(1)}}$ and $U_{-i\sin(\mathcal{H}t)}^{\vec{\phi}^{(2)}}$ are unitaries that block encode $\cos(\mathcal{H}t)$ and $-i\sin(\mathcal{H}t)$, respectively, in their $|+\rangle\langle+|$ matrix elements [hence, the Hadamards applied to the QSP qubit, which we have denoted by QSP] and may be constructed as QET sequences using phases $\phi^{\vec{(1)}}$ and $\phi^{\vec{(2)}}$, respectively, and a QSP qubit. The correct evolution of the input state is achieved only upon post-selection of both the LCU qubit [denoted LCU] and the QSP qubit in the $|0\rangle$ state. This occurs with a probability close to $\frac{1}{4}$, hence requiring amplitude amplification or repetition to increase the success probability.

under the rescaled Hamiltonian $\mathcal{H}/\alpha$ for an effective time $\tau = t\alpha$. This equivalence holds because the corresponding time evolution operators are identical: $e^{-i(\mathcal{H}/\alpha)(\alpha t)} = e^{-i\mathcal{H}t}$.

Hamiltonian simulation may then be straightforwardly achieved with QET. Naively, one may try to employ QET with a polynomial approximation to the function $e^{-ix\tau}$ (here, we view τ as a parameter representing effective time, not a variable), but because the complex exponential does not have definite parity, this function does not satisfy the constraints on $P(x)$ discussed in Appendix A, thus rendering this approach infeasible. Following Refs. 64 and 65, one can circumvent this issue by instead applying QET twice—once with an even polynomial approximation to $\cos(x\tau)$ and once with an odd polynomial approximation to $-i\sin(x\tau)$, both of which have definite parities. In addition, although these functions do not obey $|P(\pm 1)| = 1$, they may still be implemented by looking at the $|+\rangle\langle+|$ component of the QET sequence, as discussed in Appendix A 1. Then, using the linear combination of unitaries (LCU) circuit illustrated in Fig. 1, one can sum together the results of these two QET executions to obtain $\cos(\mathcal{H}t) - i\sin(\mathcal{H}t) = e^{-i\mathcal{H}t}$, as desired. We refer to this algorithm as conventional QSP-LCU Hamiltonian simulation.

How likely is this method to succeed and produce the correct time evolved state? As indicated in Fig. 1, this procedure only succeeds when the LCU qubit (used to achieve the linear combination of unitaries) and the QSP qubit (used to achieve the block encoding) are both measured in the state $|0\rangle$, so we are interested in the probability of accessing the $|00\rangle\langle00|$ matrix element. Supposing that our choice of QET polynomial allows us to construct an ϵ-approximation to $e^{-i\mathcal{H}t}$, we find that the $|00\rangle\langle00|$ matrix element of the entire unitary transformation illustrated in Fig. 1 is an $\epsilon/2$-approximation to $\frac{1}{2}e^{-i\mathcal{H}t}$, which we denote by $\frac{1}{2}(e^{-i\mathcal{H}t} \pm \epsilon)$. Accordingly, the probability of success is $p = \left\|\frac{1}{2}(e^{-i\mathcal{H}t} \pm \epsilon)\right\|^2$, i.e., $\frac{(1-\epsilon)^2}{4} \leq p \leq \frac{(1+\epsilon)^2}{4}$, which is close to $\frac{1}{4}$ and must be corrected to achieve arbitrarily high success probability.

B. Query complexity

Let us now look at the problem of constructing an appropriate QSP polynomial, from which the query complexity may be extracted as $\mathcal{O}(d)$, where d is the degree of the polynomial. We note that

in Ref. 64, Gilyén *et al.* approximated the functions $\cos(x\tau)$ and $\sin(x\tau)$ by polynomials using the Jacobi–Anger expansion,

$$\cos(x\tau) = J_0(\tau) + 2\sum_{k=1}^{\infty} (-1)^k J_{2k}(\tau) T_{2k}(x), \qquad (1)$$

$$\sin(x\tau) = 2\sum_{k=0}^{\infty} (-1)^k J_{2k+1}(\tau) T_{2k+1}(x), \qquad (2)$$

where $J_i(x)$ and $T_i(x)$ are the Bessel function and Chebyshev polynomial of order i, respectively. One can attain ϵ-approximations to $\cos(x\tau)$ and $\sin(x\tau)$ by truncating these infinite series at a sufficiently large index K. The necessary truncation index K may be determined by a function $r(\tau, \epsilon)$, which is defined implicitly as

$$\epsilon = \left(\frac{|\tau|}{r(\tau,\epsilon)}\right)^{r(\tau,\epsilon)} \text{ such that } r(\tau,\epsilon) \in (|\tau|, \infty). \qquad (3)$$

$r(\tau, \epsilon)$ may be solved for as $r(\tau, \epsilon) = |\tau| e^{W(\ln(1/\epsilon)/|\tau|)}$, where $W(x)$ is the Lambert-W function, and is proven to scale as

$$r(\tau,\epsilon) = \Theta\left(|\tau| + \frac{\ln(1/\epsilon)}{\ln(e + \ln(1/\epsilon)/|\tau|)}\right). \qquad (4)$$

Returning to the series, it is proven that truncating Eqs. (1) and (2) at $K(\tau, \epsilon) := \left\lfloor \frac{1}{2} r\left(\frac{e}{2}|\tau|, \frac{5}{4}\epsilon\right) \right\rfloor$ yields ϵ-approximations to $\cos(x\tau)$ and $\sin(x\tau)$, respectively, where $0 < \epsilon < 1/e$.[64] Because $T_i(x)$ is a polynomial of degree i with definite parity, these approximations are polynomials of degree $2K$ and $2K + 1$, respectively, with the correct even and odd parity. Let us denote these polynomials by $P_\epsilon^{\cos}(x; \tau)$ and $P_\epsilon^{\sin}(x; \tau)$.

Moreover, because cosine and sine are bounded in magnitude by 1, these ϵ-approximations only obey $|P_\epsilon^{\cos}(x;\tau)|, |P_\epsilon^{\sin}(x;\tau)| \le 1 + \epsilon$. However, a QSP polynomial must be bounded in magnitude by 1, which we may force by rescaling these polynomials by a factor of $\frac{1}{1+\epsilon}$, at the expense of increasing the error of these approximations to 2ϵ. This can be seen with the triangle inequality as

$$\left| \frac{1}{1+\epsilon} P_\epsilon^{\cos}(x;\tau) - \cos(x\tau) \right|$$
$$\le \frac{1}{1+\epsilon} \left(\left| P_\epsilon^{\cos}(x;\tau) - \cos(x\tau) \right| + |\epsilon \cos(x\tau)| \right)$$
$$\le \frac{1}{1+\epsilon} (\epsilon + \epsilon) \le 2\epsilon, \qquad (5)$$

and similarly for $P_\epsilon^{\sin}(x;\tau)$.

As we desire an ϵ-approximation to the complex exponential, we should use truncations of the Jacobi–Anger expansion that are $\epsilon/4$-approximate such that, when rescaled by $\frac{1}{1+\epsilon/4}$, they are $\epsilon/2$-approximations to $\cos(x\tau)$ and $\sin(x\tau)$. With this choice, the sum of these approximations, which is the approximation to $e^{-ix\tau}$, is ϵ-approximate by the triangle inequality. Therefore, recalling that our effective goal is to simulate the rescaled Hamiltonian $\mathcal{H}/\alpha$ for an effective time $\tau = \alpha t$, the polynomials of interest are $\frac{1}{1+\epsilon/4} P_{\epsilon/4}^{\cos}(x; \alpha t)$ and $\frac{1}{1+\epsilon/4} P_{\epsilon/4}^{\sin}(x; \alpha t)$.

Incorporating these conditions, we see that conventional QSP-LCU Hamiltonian simulation queries $\mathcal{H}$ a total number of times,

$$2K\left(\alpha t, \frac{\epsilon}{4}\right) + 2K\left(\alpha t, \frac{\epsilon}{4}\right) + 1$$
$$= 4 \cdot \left\lfloor \frac{1}{2} r\left(\frac{e}{2}\alpha|t|, \frac{5}{4}\frac{\epsilon}{4}\right) \right\rfloor + 1 =: N_{\mathcal{H}}^{\text{LCU}}(\epsilon; t, \alpha), \qquad (6)$$

where we have defined $N_{\mathcal{H}}^{\text{LCU}}(\epsilon; t, \alpha)$ to denote the sufficient number of queries to $\mathcal{H}$. Evidently, $N_{\mathcal{H}}^{\text{LCU}}(\epsilon; t, \alpha)$ scales asymptotically as

$$N_{\mathcal{H}}^{\text{LCU}}(\epsilon; t, \alpha) = \Theta\left(\alpha|t| + \frac{\ln(1/\epsilon)}{\ln(e + \ln(1/\epsilon)/(\alpha|t|))}\right). \qquad (7)$$

In comparing this query complexity with results quoted in the literature, α may be replaced with $\|\mathcal{H}\|$.

C. The quest for fully-coherent simulation

As we emphasized at lengths in the introduction, we desire our simulation algorithm to be efficient and fully-coherent. Precisely, we define *efficient* simulation algorithms as those whose query complexities are polynomial in t, $\ln(1/\epsilon)$, and $\ln(1/\delta)$:

Definition 1 (efficient Hamiltonian simulation). An efficient Hamiltonian simulation algorithm is an algorithm that queries the Hamiltonian a total number of times $\mathcal{O}(\text{poly}(\|\mathcal{H}\|\|t\|, \ln(1/\epsilon), \ln(1/\delta)))$.

Furthermore, fully-coherent algorithms perform accurate simulation while requiring only a single copy of the initial state.

Definition 2 (fully-coherent Hamiltonian simulation). A fully-coherent Hamiltonian simulation algorithm is an algorithm that, provided a single copy of an initial state $|\psi_0\rangle$, prepares a time evolved state $|\psi\rangle$ such that $\||\psi\rangle - e^{-i\mathcal{H}t}|\psi_0\rangle\| \le \epsilon$ for a generic time-independent Hamiltonian $\mathcal{H}$, with success probability at least $1 - \delta$, and for arbitrarily small ϵ and δ.

This definition of fully-coherent simulation disallows the use of repetition to boost the success probability, which necessarily requires multiple copies of the initial state and, moreover, is unsuitable for concatenation into larger algorithms. In addition, this definition applies to generic time-independent Hamiltonians, as we have placed no restrictions on its properties (sparsity, locality, etc.). The algorithms we present will, of course, assume access to the Hamiltonian $\mathcal{H}$ in some format (i.e., through a block encoding), but it is otherwise left unrestricted. Finally, we note that in the context of Hamiltonian simulation, we will often use "fully-coherent" and "coherent" interchangeably.

Let us now aim to make QSP-LCU Hamiltonian simulation fully-coherent as per Definition 2 by augmentation with amplitude amplification. While such an amplitude amplification procedure is implicitly performed in QSP-based algorithms,[64] its formal analysis is often omitted, thus disregarding the coherence of the algorithm. We first discuss how full-coherence may be achieved with conventional QSP-based amplitude amplification at the expense of a multiplicative factor of $\Theta(\ln(1/\delta))$ appended to the query complexity and subsequently how robust oblivious amplitude amplification may alternatively be employed to contribute only an additive term of $\Theta(\ln(1/\delta))$.

D. Achieving full coherence with conventional amplitude amplification

In this section, we discuss how to transform the incoherent simulation algorithm of Sec. II C into a fully-coherent algorithm by integration with amplitude amplification. We first describe the necessary amplitude amplification procedure in Sec. II D and then present a query complexity analysis in Sec. II D. We will refer to the algorithm presented in this section as the "QSP-LCU + amplitude amplification" method or "QSP-LCU+AA" for short.

Procedure. Recall that the output of conventional QSP-LCU Hamiltonian simulation is an $\epsilon/2$-approximation to $\frac{1}{2}e^{-i\mathcal{H}t}$, which we will denote by A. A has amplitude $\frac{1-\epsilon}{2} \leq \|A\| \leq \frac{1+\epsilon}{2}$, which we would like to increase to a value at least $\sqrt{1-\delta}$, such that the probability of failure (i.e., accessing the wrong block) is $\leq \delta$.

We will aim to achieve this amplification with a QSP-based procedure, noting that the QSP-LCU circuit of Fig. 1 is a unitary block encoding of A with projector $\Pi = |00\rangle\langle00|$. Our desired amplification is nontrivial to achieve using QET because the eigenvalues of A are necessarily complex, making it difficult to find a polynomial that will amplify just their magnitudes; for instance, the uniform spectral amplification polynomial of Ref. 66 applies only to real-valued inputs and will not suffice here. Such amplitude amplification necessitates the use of QSVT, a statement that we make rigorous with the following lemma:

Lemma 1 (eigenvalue magnitude transformation). Given a unitary block-encoding of $A = \sum_k \lambda_k |\lambda_k\rangle\langle\lambda_k|$, where the eigenvalues $\lambda_k = r_k e^{i\theta_k}$ may be complex, one can polynomially transform the magnitudes r_k while retaining the phases $e^{i\theta_k}$ by applying QSVT to the encoding with an odd degree polynomial $P(x)$. The output of this protocol is an encoding of a matrix $\tilde{A} = \sum_k P(r_k)e^{i\theta_k}|\lambda_k\rangle\langle\lambda_k|$, whose eigenvalues have the same phases as those of A, but magnitudes transformed by $P(x)$.

Proof. First, observe that A has singular value decomposition $A = \sum_k \sigma_k |w_k\rangle\langle v_k|$, where $\sigma_k = r_k$ and $|w_k\rangle\langle v_k| = e^{i\theta_k}|\lambda_k\rangle\langle\lambda_k|$. This expression makes it evident that the magnitudes of the eigenvalues are encoded in the singular values σ_k, while the phases are stored in the singular vector product $|w_k\rangle\langle v_k|$.

Therefore, to transform the magnitudes, one may apply to the encoding of A a singular value transformation with the desired polynomial. However, to ensure that the phases $e^{i\theta_k}$ are preserved by this transformation, we must employ QSVT with an odd degree polynomial such that the output will be $\sum_k P(\sigma_k)|w_k\rangle\langle v_k| = \sum_k P(r_k)e^{i\theta_k}|\lambda_k\rangle\langle\lambda_k| = \tilde{A}$ (see Appendix A 3), which transforms the magnitudes but retains the phases. On the other hand, an even degree polynomial would output $\sum_k P(\sigma_k)|v_k\rangle\langle v_k| = \sum_k P(r_k)|\lambda_k\rangle\langle\lambda_k|$, which does not retain the phases. $\square$

We may apply this lemma to our advantage. First, note that if $\mathcal{H}$ has eigenvalue decomposition $\mathcal{H} = \sum_k E_k |E_k\rangle\langle E_k|$, then A has singular value decomposition $A = \sum_k \sigma_k|w_k\rangle\langle v_k|$, where $\frac{1-\epsilon}{2} \leq \sigma_k \leq \frac{1+\epsilon}{2}$ and $\||w_k\rangle\langle v_k| - e^{-iE_k t}|E_k\rangle\langle E_k|\| \leq \epsilon$ (this last inequality comes from the worst case scenario when $\sigma_k = 1/2$ and all of the errors are due to incorrect phases in $|w_k\rangle\langle v_k|$). As we would like to amplify $\|A\|$ to a value at least $\sqrt{1-\delta}$, it will suffice to choose a polynomial $P(x)$ that maps inputs $\geq \frac{1-\epsilon}{2}$ to a value at least $1 - \delta/2 \geq \sqrt{1-\delta}$, which will

guarantee a probability of failure $\leq \delta$. Drawing inspiration from Refs. 64 and 65, we may select $P(x)$ to be a polynomial approximation to a sign function.

We denote the sign function by $\text{sign}(x)$, and it obeys

$$\text{sign}(x) = \begin{cases} -1, & x < 0, \\ 0, & x = 0, \\ 1, & x > 0. \end{cases} \quad (8)$$

In Appendix B, we illustrate a construction of a polynomial approximation to $\text{sign}(x)$ that is accurate away from the discontinuity at $x = 0$. This polynomial, which we denote by $P^{\text{sign}}_{\epsilon,\Delta}(x)$, obeys

$$\left| P^{\text{sign}}_{\epsilon,\Delta}(x) - \text{sign}(x) \right| \leq \epsilon \quad \text{for } x \in [-1, 1]\backslash[-\Delta/2, \Delta/2]. \quad (9)$$

That is, for x a distance at least $\Delta/2$ away from the discontinuity at $x = 0$, $P^{\text{sign}}_{\epsilon,\Delta}(x)$ provides an ϵ-approximation to the sign function. We prove in Appendix B that $P^{\text{sign}}_{\epsilon,\Delta}(x)$ can be constructed as a polynomial of odd degree $d = \gamma(\epsilon, \Delta)$, where $\gamma(\epsilon, \Delta)$ is a function explicitly defined in Eq. (B8) that scales as $\Theta\left(\frac{1}{\Delta}\ln\left(\frac{1}{\epsilon}\right)\right)$. As this degree is necessarily odd, Lemma 1 applies.

Therefore, to perform our sought-after amplitude amplification, it will suffice to choose the QSVT polynomial to be

$$P(x) = P^{\text{sign}}_{\delta/2,(1-\epsilon)}(x), \quad (10)$$

which will indeed map inputs $x > \frac{1-\epsilon}{2}$ to values $\geq 1 - \delta/2 \geq \sqrt{1-\delta}$, as desired. Applying this to A, we see that the output will be an ϵ-approximation to $e^{-i\mathcal{H}t}$ that is accessed with probability at least $1 - \delta$. We depict in Fig. 2 the quantum circuit that implements this fully-coherent simulation via amplitude amplification, highlighting how it applies QSVT to the unitary resulting from conventional QSP-LCU Hamiltonian simulation.

Query Complexity. As this procedure applies QSVT to A, which is already constructed as an eigenvalue transformation, we find that the number of queries to $\mathcal{H}$ is a product of the degrees of the two polynomials used in these transformations,

$$N^{\text{AA}}_{\mathcal{H}}(\epsilon, \delta; t, \alpha) = \gamma(\delta/2, 1 - \epsilon)N^{\text{LCU}}_{\mathcal{H}}(\epsilon; t, \alpha) \quad (11)$$

(the superscript "AA" standing for amplitude amplification). Using the asymptotic behavior of $\gamma(\delta/2, 1 - \epsilon)$ and $N^{\text{LCU}}_{\mathcal{H}}(\epsilon; t, \alpha)$, we find that $N^{\text{AA}}_{\mathcal{H}}(\epsilon, \delta; t, \alpha)$ scales as

$$N^{\text{AA}}_{\mathcal{H}}(\epsilon, \delta; t, \alpha) = \Theta\left(\ln\left(\frac{1}{\delta}\right)\left(\alpha|t| + \frac{\ln(1/\epsilon)}{\ln(e + \ln(1/\epsilon)/(\alpha|t|))}\right)\right). \quad (12)$$

Although efficient by Definition 1, this query complexity has a $\Theta\left(\ln\left(\frac{1}{\delta}\right)\right)$ multiplicative prefactor, which can be costly. This result evidences the following theorem:

Theorem 2 (QSP-LCU+AA Hamiltonian simulation). *Provided a block encoding of the Hamiltonian (possibly rescaled), the QSP-LCU+AA simulation algorithm achieves efficient fully-coherent Hamiltonian simulation with arbitrarily small ϵ and δ while querying (a block encoding of) the Hamiltonian a total number of times,*

$$\Theta\left(\ln\left(\frac{1}{\delta}\right)\left(\alpha|t| + \frac{\ln(1/\epsilon)}{\ln(e + \ln(1/\epsilon)/(\alpha|t|))}\right)\right). \quad (13)$$
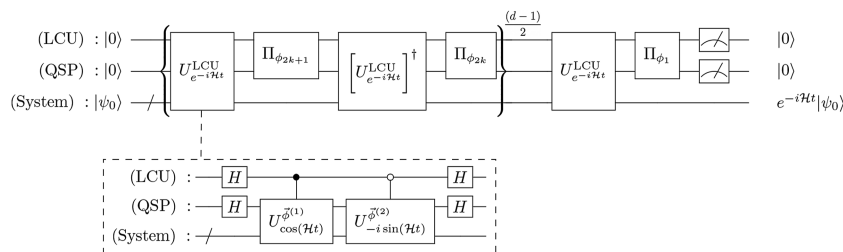
FIG. 2. A quantum circuit depicting the QSP-LCU+AA fully-coherent simulation algorithm. Here, our projectors are $\Pi = |00\rangle\langle 00| = \tilde{\Pi}$, and the corresponding projector-controlled phase shift is Π_{ϕ_k} where the phase ϕ_k is applied to the subspace $|00\rangle\langle 00|$. The gates within brackets are repeated $(d-1)/2$ times for $k = (d-1)/2, (d-1)/2 - 1, \ldots, 1$ with phases $\vec{\phi}$ to construct the appropriate QSVT sequence for amplitude amplification. This procedure succeeds if the two ancilla qubits (previously dubbed the LCU and QSP qubits) are measured in state $|00\rangle$, which occurs with high probability $\geq 1 - \delta$ after appropriate amplitude amplification. We also note that the unitary $U^{\text{LCU}}_{e^{-i\mathcal{H}t}}$ is defined by the circuit in the inset, which is identical to the circuit from Fig. 1 used in QSP-LCU simulation.

E. Achieving full coherence with robust oblivious amplitude amplification

An alternative approach to achieving full coherence is through robust oblivious amplitude amplification, which we achieve in the current section. We will refer to the resulting algorithm as the "QSP-LCU + robust oblivious amplitude amplification" method or "QSP-LCU+ROAA" for short.

Earlier, we established that the conventional QSP-LCU simulation algorithm only succeeds with probability $\frac{(1-\epsilon)^2}{4} \leq p \leq \frac{(1+\epsilon)^2}{4}$. If the success probability were known exactly (say $p = \frac{1}{4}$ precisely), one could use the amplification procedure of Ref. 67 to boost the success probability arbitrarily high while appending only an $\mathcal{O}(1)$ factor to the query complexity. However, as we only have upper and lower bounds on the probability of success here, one can alternatively employ the robust oblivious amplitude amplification protocol from Ref. 66 to boost the success probability, which also appends an $\mathcal{O}(1)$ factor to the query complexity. The resulting error scales as $\delta = \Theta(\epsilon)$, which renders the query complexity additive in $\ln(1/\delta)$ and, thus, on par with our coherent one-shot simulation algorithm of Sec. III.

Procedure. Robust oblivious amplitude amplification pertains to the following scenario: given a unitary W whose application to a state $|\psi\rangle$ produces a block encoding of $\tilde{U}|\psi\rangle$ with magnitude close to $1/2$ for some matrix $\tilde{U}$ close to a unitary U_r, we seek to construct an operator A whose application to $|\psi\rangle$ produces such a block encoding with magnitude close to 1. More precisely, given W and a projector $P = |0\rangle\langle 0| \otimes I$ such that

$$PW|0\rangle|\psi\rangle = \frac{1}{s}|0\rangle\tilde{U}|\psi\rangle, \tag{14}$$

where $|s - 2| = \mathcal{O}(\delta)$ and $\|\tilde{U} - U_r\| = \mathcal{O}(\delta)$, the robust oblivious amplitude amplification protocol constructs an operator A that satisfies

$$\|PA|0\rangle|\psi\rangle - |0\rangle U_r|\psi\rangle\| = \mathcal{O}(\delta). \tag{15}$$

In other words, A applies an approximation to the unitary U_r with error order δ. A is explicitly constructed as $A = -WRW^{\dagger}RW$, where $R = I - 2P$ is an ancilla reflection operator.

This construction is directly applicable to the amplitude amplification necessary for QSP-based simulation, the output of which is a QSP sequence that block encodes an ϵ-approximation to $e^{-i\mathcal{H}t}$ with amplitude ϵ-close to $\frac{1}{2}$. Therefore, by invoking this procedure with the QSP-LCU circuit as W and $P = |00\rangle\langle 00| \otimes I$, we may produce a block encoding of an $\mathcal{O}(\epsilon)$-approximation to $e^{-i\mathcal{H}t}$. We illustrate in Fig. 3 the quantum circuit that implements fully-coherent simulation via robust oblivious amplitude amplification.

Query Complexity. Mapping the QSP-LCU Hamiltonian simulation to the setting of robust oblivious amplitude amplification, we may take $U_r = e^{-i\mathcal{H}t}$, $s = 2$, and $\|\tilde{U} - U_r\| \leq \epsilon$, which corresponds to a block encoding of some operator $\frac{1}{2}(e^{-i\mathcal{H}t} \pm \epsilon)$, as desired. This dictates that

$$\|\tilde{U}\tilde{U}^{\dagger} - I\| \leq \|(U_r \pm \epsilon)(U_r^{\dagger} \pm \epsilon) - I\| \leq \delta\|U_r + U_r^{\dagger}\| + \epsilon^2 \leq 3\epsilon. \tag{16}$$

Reference 53 shows that $PA|0\rangle|\psi\rangle = |0\rangle\left(\frac{3}{s}\tilde{U} - \frac{4}{s^3}\tilde{U}\tilde{U}^{\dagger}\tilde{U}\right)|\psi\rangle$, which in our case lends itself to the inequality

$$\left\|\tilde{U}\left(\frac{3}{s} - \frac{4}{s^3}\tilde{U}^{\dagger}\tilde{U}\right) - U_r\right\| \leq \left\|\tilde{U}\left(\frac{3}{2} - \frac{1}{2}(I \pm 3\epsilon)\right) - U_r\right\|$$
$$\leq \frac{3}{2}\epsilon + \|\tilde{U} - U_r\| \leq \frac{5}{2}\epsilon. \tag{17}$$

Therefore, oblivious robust amplitude amplification applied to QSP-LCU Hamiltonian simulation achieves an error bounded by $\frac{5}{2}\epsilon$. This
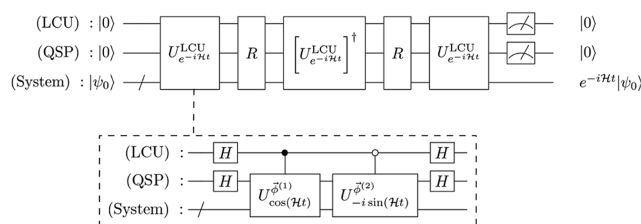
FIG. 3. A quantum circuit depicting the QSP-LCU+ROAA protocol, where $R = I - 2P$ (for projector $P = |00\rangle\langle 00| \otimes I$) is an ancilla operator as in Eq. (14). Like the circuit of Fig. 2, this procedure succeeds if the LCU and QSP qubits are measured in state $|00\rangle$, which occurs with high probability $\geq 1 - \delta$ after robust oblivious amplitude amplification.

implies that the correct block of the block encoding has magnitude $\geq 1 - \frac{5}{2}\epsilon$ and that the corresponding failure probability of this algorithm is bounded by $\delta \leq 1 - \left(1 - \frac{5}{2}\epsilon\right)^2 \leq 5\epsilon$.

If we are interested then in the query complexity of achieving fully-coherent simulation with a total error ϵ, we may take $\epsilon \mapsto \frac{2}{5}\epsilon$ in the equations of Sec. II B. Observing that the robust oblivious amplitude amplification operator A employs three instances of the QSP-LCU circuit W, the total query complexity is

$$N_{\mathcal{H}}^{\mathrm{ROAA}}(\epsilon; t, \alpha) = 3 N_{\mathcal{H}}^{\mathrm{LCU}}\left(\frac{2}{5}\epsilon; t, \alpha\right) \qquad (18)$$

("ROAA" for robust oblivious amplitude amplification). As above, the corresponding probability of failure is bounded as $\delta \leq 2\epsilon$. We thus have the following theorem:

Theorem 3 (QSP-LCU + ROAA Hamiltonian simulation). *Provided a block encoding of the Hamiltonian (possibly rescaled), the QSP-LCU+ROAA simulation algorithm achieves efficient fully-coherent Hamiltonian simulation with arbitrarily small ϵ and $\delta = 2\epsilon$ while querying (a block encoding of) the Hamiltonian a total number of times,*

$$\Theta\left(\alpha|t| + \frac{\ln(1/\epsilon)}{\ln(e + \ln(1/\epsilon)/(\alpha|t|))}\right). \qquad (19)$$

III. COHERENT ONE-SHOT HAMILTONIAN SIMULATION

In Sec. II, we presented a simple but non-optimal fully-coherent Hamiltonian simulation algorithm that acquired a multiplicative prefactor of $\ln(1/\delta)$ in its query complexity and subsequently showed how to remove this factor by using robust oblivious amplitude amplification[49] to achieve fully-coherent simulation.

In this section, we introduce our novel "coherent one-shot Hamiltonian simulation" algorithm that attains a query complexity *additive* in $\ln(1/\delta)$ rather than multiplicative and is thus also both fully-coherent and efficient. In contrast to the algorithms in Sec. II, our new construction guarantees the unitarity of the simulation algorithm by using a single QSP call to directly approximate the complex exponential function $e^{-ix\tau}$, without the need for any amplitude amplification technique. We will discuss the performance trade-off between our new coherent one-shot simulation algorithm and the two amplitude amplification based fully-coherent algorithms in detail in Sec. IV.

We outline the coherent one-shot algorithm in Sec. III A, where details for the block-encoding (Sec. III A 1), the construction of the complex exponential function (Sec. III A 2), as well as the success probability (Sec. III A 3) are presented. This is followed by a query complexity analysis in Sec. III B and a summary of the main result in Theorem 5.

A. Procedure

Coherent one-shot Hamiltonian simulation is achieved by taking a modified approach to QSP: we first apply a transformation to the Hamiltonian, realized via unitary gates and ancilla qubits, and subsequently apply QSP to the transformed Hamiltonian. We

call this first step a *pre-transformation*: it rescales the eigenvalues of the Hamiltonian, which enables us to circumvent the parity constraint on QSP polynomials and ultimately achieve a broader class of transformations. We also note that as the Hamiltonian is accessed via a block encoding, the output of the pre-transformation is really a block-encoding of the transformed Hamiltonian, possibly with a different projector. For the sake of simplicity, we will employ a linear pre-transformation in the development of coherent one-shot Hamiltonian simulation; however, the pre-transformation may be nonlinear, in general.

In more detail, our starting point is a block encoding of $\mathcal{H}/\alpha$ with eigenvalues in the range $[-1, 1]$. In the coherent one-shot algorithm, we first find a linear pre-transformation that rescales its eigenvalues into the range $(a, b) \subset [-1, 1]$ and then determine a QSP polynomial that is a good approximation to the complex exponential $e^{-ix\tau}$ over the range $x \in (a, b) \subset [-1, 1]$. Finally, by applying a quantum eigenvalue transformation (QET) to this encoding with our polynomial approximation to the complex exponential and an appropriately chosen effective time τ, we attain the time evolution operator up to a global phase. Note how the use of the pre-transformation shifts our focus to eigenvalues only in the reduced range $(a, b) \subset [-1, 1]$, which allows us to ignore the boundary conditions at $x = \pm 1$ imposed by QSP.

Below, we specialize to one particular implementation of this method, employing a linear pre-transformation and a polynomial that approximates the complex exponential $e^{-ix\tau}$ over a range of positive x: $x \in (a, b) \subset [0, 1]$. While this construction is sufficient for demonstrating the advantage of coherent one-shot simulation, it is by no means optimal as alternate constructions, including other pre-transformations, certainly exist.

In general, the use of a pre-transformation provides an alternative to the single-ancilla QSP algorithm of Ref. 57 for the construction of the complex exponential and even of more general functions of mixed parity. Whereas single-ancilla QSP imposes constraints on the QSP phases, which both complicates the optimization over the QSP phases and inhibits its robustness to noise, our method using a pre-transformation is less restricted. For example, employing PyQSP[68] to numerically estimate the complex exponential with $\tau = 4$ with a length $d = 14$ QSP sequence, single-ancilla QSP can be optimized to achieve an error 0.048 (i.e., the maximal difference from the desired function over the range of interest), while our method (with $a = 1/4, b = 3/4$) obtains an error 0.027. Furthermore, when the phases are subject to additive Gaussian noise of standard deviation 0.02, the error of single-ancilla QSP grows to 0.085 ± 0.029, while that of our method becomes only 0.067 ± 0.025.

1. Pre-transformation

With our encoding of $\mathcal{H}/\alpha$, we will employ a linear pre-transformation to block encode an operator whose spectrum is proportional to that of $\mathcal{H}/\alpha$ (up to an additive constant) but shrunken to be in the range $\left[\frac{1-\beta}{2}, \frac{1+\beta}{2}\right] \subset [0, 1]$ for some chosen $\beta < 1$.

To achieve this, we first block encode a rescaled Hamiltonian $\beta \mathcal{H}/\alpha$. Given the ability to block encode $\mathcal{H}/\alpha$, which is already scaled by a constant itself, it is not much more difficult to block encode $\beta \mathcal{H}/\alpha$, and hence, this step may not even be necessary. If this step is needed, however, it may formally be achieved by invoking Lemma 53 of Ref. 64 regarding products of block encoded matrices:
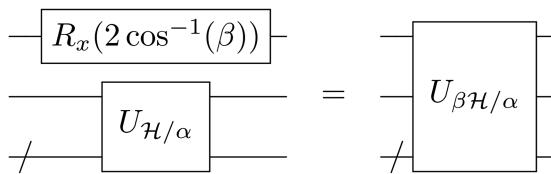
FIG. 4. A quantum circuit representation of the operator $U_{\beta\mathcal{H}/\alpha}$, which block encodes $\beta\mathcal{H}/\alpha$ in its $|00\rangle\langle00|$ matrix element. Here, we have assumed that $U_{\mathcal{H}/\alpha}$ encodes $\mathcal{H}/\alpha$ in its $|0\rangle\langle0|$ matrix element, as is conventional.

Lemma 4. If V_A is a block encoding of an n qubit operator A and if V_B is a block encoding of an n qubit operator B, then $(I_B \otimes V_A)(V_B \otimes I_A)$ is a block encoding of AB, where the identity operators in this expression act on each other's ancilla spaces.

This result, which is straightforwardly proven via direct computation, enables our construction. Suppose that $U_{\mathcal{H}/\alpha}$ is the block encoding of the n qubit operator $\mathcal{H}/\alpha$ and that $U_{\beta I}$ is a block encoding of βI_{2^n}. Then, using the formula of this lemma, we may construct a block encoding of $\mathcal{H}/\alpha \cdot \beta I_{2^n} = \beta\mathcal{H}/\alpha$, as desired. Moreover, we may easily construct $U_{\beta I}$ with an x-rotation applied to an ancilla qubit,

$$R_x(2\cos^{-1}(\beta)) \otimes I_{2^n} = \begin{bmatrix} \beta I_{2^n} & \cdot \\ \cdot & \cdot \end{bmatrix} = U_{\beta I}, \quad (20)$$

where $R_x(\theta)$ is the x-rotation through an angle θ. Equivalently, the block encoding of $\beta\mathcal{H}/\alpha$ is $U_{\beta\mathcal{H}/\alpha} = R_x(2\cos^{-1}(\beta)) \otimes U_{\mathcal{H}/\alpha}$, which we depict in Fig. 4. As QSP already employs a sequence of rotations to develop $U^{\vec{\phi}}$, this additional x-rotation is not particularly costly.

With this rescaled Hamiltonian, we may introduce an ancilla qubit and employ the circuit in Fig. 5 to obtain an encoding of $\frac{1}{2}(I + \beta\mathcal{H}/\alpha) =: \tilde{\mathcal{H}}$, which has eigenvalues in the range $\left[\frac{1-\beta}{2}, \frac{1+\beta}{2}\right] \subset [0, 1]$. This is precisely our desired linear pre-transformation: $\mathcal{H}/\alpha \mapsto \frac{1}{2}(I + \beta\mathcal{H}/\alpha)$. In total then, with the addition of two extra ancilla qubits, we are able to block encode the rescaled Hamiltonian $\tilde{\mathcal{H}} = \frac{1}{2}(I + \beta\mathcal{H}/\alpha)$, which has sufficiently bounded eigenvalues. As for accessing this operator, if the initial Hamiltonian $\mathcal{H}/\alpha$ were encoded in the $|0\rangle\langle0|$ matrix element of a unitary, as is conventional, then the procedure sketched here encodes our desired operator in the $|000\rangle\langle000|$ matrix element of a new unitary.
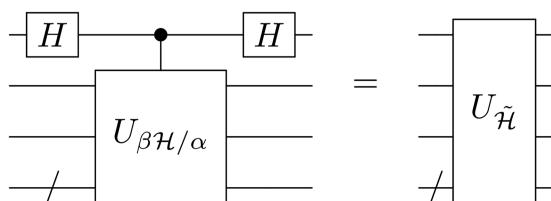
FIG. 5. A quantum circuit representation of the operator $U_{\tilde{\mathcal{H}}}$, which block encodes $\tilde{\mathcal{H}} = \frac{1}{2}(I + \beta\mathcal{H}/\alpha)$ in its $|000\rangle\langle000|$ matrix element.

2. Target polynomial

With an eye toward Hamiltonian simulation, we aim to approximate $e^{-ix\tau}$ as a polynomial, where x is the input variable and τ is a real parameter representing the effective time of simulation. Unfortunately, as we discussed in Sec. II, we cannot approximate $e^{-ix\tau}$ as a QSP polynomial $\forall x \in [-1, 1]$ because the complex exponential does not have definite parity. However, using the block encoding of $\tilde{\mathcal{H}}$ outlined above, we only need a polynomial that approximates $e^{-ix\tau}$ for $x \in \left[\frac{1-\beta}{2}, \frac{1+\beta}{2}\right] \subset [0, 1]$. We may design such a polynomial by estimating the *even extension of the complex exponential* (EECE),

$$\mathrm{EECE}(x; \tau) := \cos(\tau x) - i\sin(\tau x)\,\mathrm{sign}(x). \quad (21)$$

This function has definite parity and is of magnitude 1, so there exists a QSP polynomial $P(x) \approx \mathrm{EECE}(x; \tau)$, which can be accessed as the $|0\rangle\langle0|$ matrix element of a QSP sequence. In approximating this function as a polynomial, we may estimate the trigonometric functions with the truncated Jacobi–Anger expansions discussed in Sec. II and employ the polynomial approximation to the sign function explained in Appendix B.

Formally, it suffices to select the following as a polynomial approximation to the EECE:

$$P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau) := \frac{1}{1 + \epsilon/6} P_{\epsilon/6}^{\cos}(x; \tau) - i \frac{1}{1 + \epsilon/6} P_{\epsilon/6}^{\sin}(x; \tau) P_{\epsilon/3,\Delta}^{\mathrm{sign}}(x). \quad (22)$$

As $\frac{1}{1+\epsilon/6} P_{\epsilon/6}^{\cos}(x; \tau)$ and $\frac{1}{1+\epsilon/6} P_{\epsilon/6}^{\sin}(x; \tau)$ are $\epsilon/3$ approximations to $\cos(\tau x)$ and $\sin(\tau x)$, respectively, the error of this approximation for $|x| \geq \Delta/2$ is

$$\left\| \cos(\tau x) - \frac{1}{1 + \epsilon/6} P_{\epsilon/6}^{\cos}(x; \tau) \right\|$$
$$+ \left\| \sin(\tau x) - \frac{1}{1 + \epsilon/6} P_{\epsilon/6}^{\sin}(x; \tau) P_{\epsilon/3,\Delta}^{\mathrm{sign}}(x) \right\|$$
$$\leq \epsilon/3 + \left\| \sin(\tau x) - (\sin(\tau x) - \epsilon/3)(1 - \epsilon/3) \right\|$$
$$\leq \epsilon/3 + 2\epsilon/3 - (\epsilon/3)^2 \leq \epsilon, \quad (23)$$

as desired. Hence, $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$ ϵ-approximates the even extension of the complex exponential for $|x| \geq \Delta/2$. In addition, $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$ also has magnitude less than 1 by virtue of its constituent polynomials being bounded in magnitude, as is required by the third condition of the QSP theorem. We rigorously analyze the degree of $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$ in Sec. III B.

One concern with a QSP implementation of $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$ is that according to the third QSP condition, $|P(x \to \pm 1)| \to 1$, which $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$ only satisfies up to some error ϵ. Fortunately, this is not a severe problem for us because $x = \pm 1$ is outside our range of interest, that being $\left[\frac{1-\beta}{2}, \frac{1+\beta}{2}\right]$. Likewise, for reasonably small ϵ, there should exist a QSP polynomial that behaves as $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$ and also satisfies the third condition of the QSP theorem. In Sec. VI, we numerically verify that such a QSP polynomial can be constructed.

How might we employ this polynomial? Consider applying a QET to the aforementioned encoding of $\tilde{\mathcal{H}}$ with the target polynomial $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$. If we select a $\Delta/2 \leq (1 - \beta)/2$ such that $P_{\epsilon,\Delta}^{\mathrm{EECE}}(x; \tau)$ acts as the complex exponential on the eigenvalues of $\tilde{\mathcal{H}}$ and an

effective time $\tau = 2t\alpha/\beta$, the resulting operator of this transform will encode an ϵ-approximation to

$$e^{-i\frac{1}{2}(I+\beta H/\alpha)2t\alpha/\beta} = e^{-it\alpha/\beta}e^{-i\mathcal{H}t}, \tag{24}$$

which is the time evolution operator, up to a global phase. Hence, by using the polynomial $P_{\epsilon,(1-\beta)}^{\text{EECE}}(x;\ 2t\alpha/\beta)$, this procedure yields a unitary $U^{\vec{\phi}}$ and a projector Π (which, in the simple case mentioned earlier, is just $\Pi = |000\rangle\langle 000|$) such that $\|\Pi U^{\vec{\phi}}\Pi - e^{-i\mathcal{H}t}\| \leq \epsilon$, thus achieving Hamiltonian simulation.

3. Probability of success

As we mentioned above, we may employ our block encoding of $\tilde{\mathcal{H}}$ and polynomial approximation to the EECE to create a unitary that block encodes an ϵ-approximation to $e^{-i\mathcal{H}t}$. Because $\|e^{-i\mathcal{H}t}\| = 1$, the magnitude of this block is close to 1, which by unitarity implies that the other elements in its row/column have magnitude near zero and that the $e^{-i\mathcal{H}t}$ block is accessed with high probability. Specializing to the case where $\mathcal{H}/\alpha$ is encoded in the $|0\rangle\langle 0|$ matrix element of a unitary and the projector of interest is $\Pi = |000\rangle\langle 000|$, this means that the application of $U^{\vec{\phi}}$ to $|000\rangle|\psi_0\rangle$ outputs a good approximation $|000\rangle e^{-i\mathcal{H}t}|\psi_0\rangle$ with high probability. Therefore, this algorithm indeed performs Hamiltonian simulation with high success probability using only a single QET call, hence the aptly chosen name "coherent one-shot Hamiltonian simulation."

More precisely, as $\|\Pi U^{\vec{\phi}}\Pi - e^{-i\mathcal{H}t}\| \leq \epsilon$, the reverse triangle inequality indicates that the magnitude of this block is

$$\|e^{-ix\tau} + \epsilon\| \geq 1 - \epsilon. \tag{25}$$

Hence, unitarity dictates that the other elements in the column corresponding to the choice of projector have collective magnitude $\leq \sqrt{1-(1-\epsilon)^2}$. This implies that the probability of failure is in the worst case

$$\delta = 1 - (1-\epsilon)^2 \leq 2\epsilon \quad \Rightarrow \quad \delta = \Theta(\epsilon). \tag{26}$$

As evidenced by this expression, the probability of failure of this algorithm is controlled by the accuracy ϵ of the EECE approximation, which may easily be tuned by selecting an appropriate QSP polynomial. We note that a similar bound on the probability of failure was derived for the Hamiltonian simulation algorithm presented in Ref. 69, although their result applies to the simulation of a random Hamiltonian.

We graphically summarize the coherent one-shot algorithm in Fig. 6, in which we illustrate its quantum circuit. In this figure, we emphasize that this algorithm applies a QET to $\tilde{\mathcal{H}}$ and succeeds when the ancilla qubits are measured in the state $|000\rangle$.

B. Query complexity

Let us now investigate the query complexity of coherent one-shot Hamiltonian simulation, for which we will need to determine the degree of the polynomial approximation to the EECE. As we mentioned earlier, we should choose $\Delta/2 \leq (1-\beta)/2$ such that $P_{\epsilon,\Delta}^{\text{EECE}}(x;\tau)$ acts as the complex exponential on the eigenvalues of interest, and $\tau = 2t\alpha/\beta$ such that we simulate for the correct time. Appropriately, our polynomial of interest is $P_{\epsilon,(1-\beta)}^{\text{EECE}}(x;\ 2t\alpha/\beta)$,
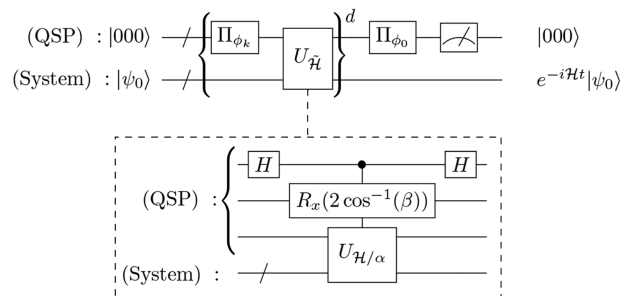
FIG. 6. The quantum circuit used to perform coherent one-shot simulation. Here, our projector is $\Pi = |000\rangle\langle 000|$ with corresponding projector-controlled phase shift being Π_{ϕ_k}. The gates within brackets are repeated d times for $k = d, d-1, \ldots, 1$ with phases $\vec{\phi}$ to construct the appropriate QET sequence for the polynomial $P_{\epsilon,(1-\beta)}^{\text{EECE}}(\tilde{\mathcal{H}}; 2t\alpha/\beta)$. This procedure outputs the correct time evolved state $e^{-i\mathcal{H}t}|\psi_0\rangle$ in the system register if the QSP qubits are measured in the state $|000\rangle$, which occurs with probability at least $1-2\epsilon$. We also note that the unitary $U_{\tilde{\mathcal{H}}}$ is defined in the inset as the block encoding of $\tilde{\mathcal{H}}$, which is attained by combining the circuits of Figs. 4 and 5.

which according to Eq. (22) has degree $d = \deg(P_{\epsilon/6}^{\sin}(x;\ 2t\alpha/\beta)) + \deg(P_{\epsilon/3,(1-\beta)}^{\text{sign}}(x))$.

Employing the bounds on the degrees of $P_{\epsilon/6}^{\sin}(x;\ 2t\alpha/\beta)$ and $P_{\epsilon/3,(1-\beta)}^{\text{sign}}(x)$, which are discussed in Sec. II B and Appendix B, respectively, we find that a sufficient degree to construct $P_{\epsilon,\Delta}^{\text{EECE}}(x;\ 2t\alpha/\beta)$ is

$$2K\left(\frac{2t\alpha}{\beta}, \frac{\epsilon}{6}\right) + 1 + \gamma\left(\frac{\epsilon}{3}, 1-\beta\right)$$

$$= 2 \cdot \left\lfloor \frac{1}{2} r\left(\frac{e\alpha|t|}{\beta}, \frac{5\epsilon}{24}\right) \right\rfloor + \gamma\left(\frac{\epsilon}{3}, 1-\beta\right) + 1$$

$$=: N_{\mathcal{H}}^{\text{OS}}(\epsilon, \delta = 2\epsilon, \beta; t, \alpha), \tag{27}$$

where $\gamma(\epsilon, \Delta)$ is a function explicitly defined in Eq. (B8). Because the transformed Hamiltonian to which QET is applied is linear in the Hamiltonian, this expression is the query complexity of our coherent one-shot Hamiltonian simulation, which we denote by $N_{\mathcal{H}}^{\text{OS}}(\epsilon, \delta = 2\epsilon, \beta; t, \alpha)$ in the last line of the above equation (with superscript "OS" for "one-shot"). This expression implicitly bounds the probability of failure as $\delta \leq 2\epsilon$ as per Sec. III A 3, which we note in the query complexity.

Using the appropriate asymptotic scalings, the $r(\cdot, \cdot)$ term in $N_{\mathcal{H}}^{\text{OS}}(\epsilon, \delta = 2\epsilon, \beta; t, \alpha)$ scales as $\sim 1/\beta$, whereas the $\gamma(\cdot, \cdot)$ term scales as $\sim 1/(1-\beta)$. As these terms are added together in $N_{\mathcal{H}}^{\text{OS}}(\epsilon, \delta = 2\epsilon, \beta; t, \alpha)$, there is a clear trade-off between choosing a large or small value for β, the optimal value of which depends on the details of the Hamiltonian (through α) and the desired error rate (through ϵ). Nonetheless, β may be chosen as some fixed constant (say, $\beta = 1/2$), and hence, the query complexity of coherent one-shot Hamiltonian simulation scales as

$$N_{\mathcal{H}}^{\text{OS}} = \Theta\left(\alpha|t| + \ln\left(\frac{1}{\epsilon}\right) + \ln\left(\frac{1}{\delta}\right)\right). \tag{28}$$

This certainly provides an improvement in the scaling with respect to δ over the QSP-LCU+AA coherent simulation algorithm. We have thus demonstrated a central result of this paper.

Theorem 5 (coherent one-shot Hamiltonian simulation). *Provided a block encoding of the Hamiltonian (possibly rescaled), the coherent one-shot Hamiltonian simulation algorithm achieves efficient fully-coherent Hamiltonian simulation with arbitrarily small ϵ and δ while querying (the block encoding of) the Hamiltonian a total number of times,*

$$\Theta\left(\alpha|t| + \ln\left(\frac{1}{\epsilon}\right) + \ln\left(\frac{1}{\delta}\right)\right). \tag{29}$$

IV. COMPARISON ON THE SCALING OF QUERY COMPLEXITY

Given the analytical bounds on query complexities of the QSP-LCU+AA and coherent one-shot simulation algorithms derived in Eqs. (11), (18), and (27), we now study the behavior of the query complexity with respect to simulation time and error. We compare and contrast the scaling of the complexity between different algorithms discussed above. We note that a similar analysis is presented in Ref. 70, wherein the query complexities of various Hamiltonian simulation algorithms are compared with each other and with empirical bounds, although coherence is not a focal point of their work.

Let us begin by investigating the scaling of the query complexity with time. With parameter choices $\alpha = 5$, $\beta = 0.5$, and fixed $\epsilon = 0.02$, we calculate $N_{\mathcal{H}}^{\text{AA}}$, $N_{\mathcal{H}}^{\text{ROAA}}$, and $N_{\mathcal{H}}^{\text{OS}}$ for a variety of ϵ using Eqs. (11) and (27), respectively. Because the coherent one-shot algorithm naturally has failure probability bounded above by 2ϵ, we set $\delta = 2\epsilon$ in the computation of $N_{\mathcal{H}}^{\text{AA}}$, which provides a fair comparison between the two algorithms. We plot these results in the left panel of Fig. 7, showcasing all of the query complexities in the main figure and zooming into $N_{\mathcal{H}}^{\text{OS}}$ and $N_{\mathcal{H}}^{\text{ROAA}}$ in the inset. As expected, these query complexities all scale linearly with time, with the QSP-LCU+AA method growing the most rapidly. The inset indicates that $N_{\mathcal{H}}^{\text{OS}}$ has a shallower slope than $N_{\mathcal{H}}^{\text{ROAA}}$, suggesting the superiority of the coherent one-shot algorithm for long time simulations.

Next, we look at the scaling of query complexity with error, ϵ. Again, with parameter choices $\beta = 0.5$, $\alpha = 5$, and fixed $t = 5$, we calculate $N_{\mathcal{H}}^{\text{AA}}$, $N_{\mathcal{H}}^{\text{OS}}$, and $N_{\mathcal{H}}^{\text{ROAA}}$ for a variety of errors ϵ and set $\delta = 2\epsilon$ to provide an accurate comparison between the algorithms. As before, we plot the results in the right panel of Fig. 7. We see that the QSP-LCU+AA algorithm attains the greatest query complexity with the most severe scaling. Moreover, the inset shows that $N_{\mathcal{H}}^{\text{OS}}$ exceeds $N_{\mathcal{H}}^{\text{ROAA}}$ for small ϵ indicating the utility of oblivious amplitude amplification for achieving highly accurate simulations.

In summary, both our coherent one-shot algorithm and the QSP-LCU+ROAA algorithms achieve significant speedup as compared to QSP-LCU+AA. However, there is a performance trade-off depending on the parameter regimes of simulation error ϵ and simulation time t as evident from the crossing in the insets of Fig. 7. In the case of $\delta = 2\epsilon$, our coherent one-shot algorithm out-performs QSP-LCU+ROAA for long simulation time with a moderate error tolerance.

V. APPLICATIONS AND SIMULATION RESULTS

One of the most promising applications of quantum algorithms is simulating the dynamics of interacting quantum many-body systems, which is known to be exponentially hard on classical computers due to particle–particle interactions. In this section, we employ the aforementioned fully-coherent simulation algorithms to simulate the Heisenberg model in both constant and time-dependent external fields (Sec. V A), as well as the electronic charge migration dynamics in a hydrogen molecule (Sec. V B). While our
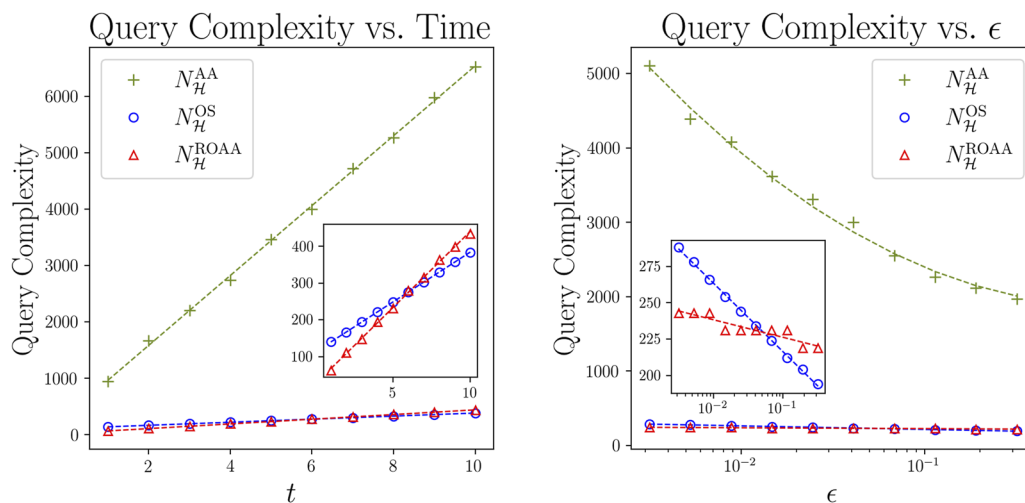
FIG. 7. Query complexity vs t and ϵ for the coherent Hamiltonian simulation algorithms presented in this paper ($N_{\mathcal{H}}^{\text{AA}}$ for QSP-LCU+AA, $N_{\mathcal{H}}^{\text{ROAA}}$ for robust oblivious amplitude amplification and $N_{\mathcal{H}}^{\text{OS}}$ for coherent one-shot simulation), as well as their lines of best fit. The insets zoom into $N_{\mathcal{H}}^{\text{OS}}$ and $N_{\mathcal{H}}^{\text{ROAA}}$ for a more clear comparison.

numerical simulations are limited to two spins in the Heisenberg model and minimal basis for the hydrogen molecule, they suffice to demonstrate the strengths of these fully-coherent algorithms. These algorithms can be readily applied to systems of larger size and to more complex molecules, provided an efficient block-encoding of the system Hamiltonian. We restrict our numerical simulations to the QSP-LCU+ROAA and the coherent one-shot algorithm, as these techniques have exhibited the greatest efficiencies in terms of query complexity.

A. The Heisenberg model

We use the QSP-LCU+ROAA and coherent one-shot Hamiltonian simulation algorithms to simulate a two-spin Heisenberg model with nearest neighbor interactions and an external magnetic field on each site. We discuss the full details of this model in Sec. V A 1. The simulation results are presented in Sec. V A 2, which further illustrate the advantages of the full-coherence property in Hamiltonian simulation.

1. The Hamiltonian

We focus on the one-dimensional Heisenberg spin chain with nearest–neighbor interaction, subjected to a site-specific time-dependent magnetic field along the z-direction, whose Hamiltonian is

$$\mathcal{H}(t) = \mathcal{H}_0(t) + \mathcal{H}_1, \tag{30}$$

$$\mathcal{H}_0(t) = \sum_{j=1}^{n} h_j(t)\sigma_j^z, \tag{31}$$

$$\mathcal{H}_1 = \sum_{j=1}^{n} \left(g_j^x \sigma_j^x \sigma_{j+1}^x + g_j^y \sigma_j^y \sigma_{j+1}^y + g_j^z \sigma_j^z \sigma_{j+1}^z \right), \tag{32}$$

where n is the total number of spins, $\{g_j^x, g_j^y, g_j^z\}$ are the interaction strengths along x, y, z directions between spins j and $j + 1$, and $h_j(t)$ is the on-site (time-dependent) magnetic field for spins $j = 1, \ldots, n$. At a high level, $\mathcal{H}_0(t)$ and $\mathcal{H}_1$ are the non-interacting term for each spin in the magnetic field and the pairwise nearest–neighbor interaction term, respectively.

We further choose a system size of $n = 2$. Although the simulation algorithms discussed in this paper apply equally well to larger systems as long as one is provided a block encoded Hamiltonian, simulations of larger systems quickly become difficult to emulate on classical hardware. The two spin Heisenberg Hamiltonian components $\mathcal{H}_0$ and $\mathcal{H}_1$ can be written explicitly in the computational basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ of σ_z eigenvectors as

$$\mathcal{H}_0 = \text{diag}\{h_+(t), h_-(t), -h_-(t), -h_+(t)\},$$

$$\mathcal{H}_1 = \begin{bmatrix} g_z & & & g_x - g_y \\ & -g_z & g_x + g_y & \\ & g_x + g_y & -g_z & \\ g_x - g_y & & & g_z \end{bmatrix}, \tag{33}$$

where $h_+(t) = h_1(t) + h_2(t)$, $h_-(t) = h_1(t) - h_2(t)$, and $|0\rangle$ and $|1\rangle$ represent spin up and down, respectively. It can be shown that the

commutator between two instances of the Hamiltonian evaluated at different times is

$$[\mathcal{H}(t_1), \mathcal{H}(t_2)]$$
$$= 2i\left((g_y\Delta h_2 - g_x\Delta h_1)\sigma_1^y\sigma_2^x + (g_y\Delta h_1 - g_x\Delta h_2)\sigma_1^x\sigma_2^y\right), \tag{34}$$

where $\Delta h_j = h_j(t_2) - h_j(t_1)$ for $j = 1, 2$. Therefore, in general, $[\mathcal{H}(t_1), \mathcal{H}(t_2)] \neq 0$ if $t_2 \neq t_1$, so instances of the Hamiltonian corresponding to different times do not necessarily commute.

2. Simulating the spin dynamics

To illustrate the advantages of full-coherence in Hamiltonian simulation, we use the QSP-LCU+ROAA and coherent one-shot methods to simulate the spin dynamics of the Heisenberg Hamiltonian given in Eq. (33) for both time-independent *and* time-dependent magnetic fields. The data and scripts used in our simulation are documented in Ref. 71.

After running both algorithms on a given initial state $|\psi_0\rangle$ with a fixed number of queries to the Hamiltonian block-encoding, we measure the expectation values of the first spin's z component, $\langle \sigma_1^z \rangle$ for various simulation time intervals and compare them to analytic results obtained by applying the exact time evolution operator to $|\psi_0\rangle$. In Subsections V A 2 a–V A 2 c, we provide details on the block-encoding used and then explore the numerical results for the time-independent and time-dependent Heisenberg Hamiltonian simulations.

a. Block-encoding. As a necessary preliminary to both algorithms, we first construct unitary matrices, which block-encode the total two-spin Hamiltonian. For the QSP-LCU+ROAA algorithm, we require a block encoding $U_{\mathcal{H}/\alpha}$ of $\mathcal{H}/\alpha$, for which it suffices to choose

$$U_{\mathcal{H}/\alpha} = \begin{bmatrix} \mathcal{H}/\alpha & \sqrt{I - \mathcal{H}^2/\alpha^2} \\ \sqrt{I - \mathcal{H}^2/\alpha^2} & -\mathcal{H}/\alpha \end{bmatrix}, \tag{35}$$

where $\mathcal{H} = \mathcal{H}_0 + \mathcal{H}_1$ given by Eq. (33) and the rescaling factor $\alpha \geq \|\mathcal{H}\|$ ensures that $\sqrt{I - \mathcal{H}^2/\alpha^2}$ is real. We illustrate a circuit based implementation of $U_{\mathcal{H}/\alpha}$ in Appendix C.

For the coherent one-shot algorithm, we employ the pre-transformation shown in the circuit of Fig. 6 to construct a block encoding $U_{\tilde{\mathcal{H}}}$ of $\tilde{\mathcal{H}} = \frac{1}{2}(I + \beta\mathcal{H}/\alpha)$. Then, the quantum circuits of Fig. 3 are used to simulate the time evolution with the QSP-LCU+ROAA method, while the circuit in Fig. 6 is employed to simulate via the coherent one-shot algorithm.

b. Time-independent simulation. We begin by using both of the efficient fully-coherent algorithms to simulate the spin dynamics of the aforementioned Heisenberg model with a constant external field, while making a fixed number of queries to the Hamiltonian.

The initial state is chosen to be $|\psi_0\rangle = |00\rangle$, indicating that both spins are initially spin-up. We then choose a value for the simulation time interval t and, using the Qiskit Aer quantum circuit simulator package,[72] apply the quantum circuits for each algorithm to a copy of the state $|\psi_0\rangle$ to obtain the time-evolved states $|\psi_{\text{ROAA}}\rangle = \left[e^{-i\mathcal{H}t}\right]_{\text{ROAA}}|\psi_0\rangle$ and $|\psi_{\text{OS}}\rangle = \left[e^{-i\mathcal{H}t}\right]_{\text{OS}}|\psi_0\rangle$, which correspond to the outputs of the QSP-LCU+ROAA and coherent one-shot algorithms for simulation time t, respectively. Finally, for each

time-evolved state $|\psi\rangle \in \{|\psi_{\text{ROAA}}\rangle, |\psi_{\text{OS}}\rangle\}$, we compute the expectation value of the first spin's z component with

$$\langle\sigma_1^z\rangle = \langle\psi|(\sigma^z \otimes I)|\psi\rangle. \tag{36}$$

This expectation value may be evaluated readily on a quantum computer by repeatedly preparing the state $|\psi\rangle$ and measuring the first qubit in the computational basis (which is the σ^z eigenbasis). Suppose a total of $n = n_0 + n_1$ such measurements are performed on n copies of $|\psi\rangle$ such that n_0 measurements observe the first qubit in the state $|0\rangle$ and n_1 measurements in the state $|1\rangle$. Then, the expectation value $\langle\sigma_1^z\rangle$ is approximately given by

$$\langle\sigma_1^z\rangle \approx \frac{1}{n}(n_0 - n_1). \tag{37}$$

In any practical implementation of these algorithms, the number of measurements n can be chosen large enough such that the measured $\langle\sigma_1^z\rangle$ converges satisfactorily to its true value. This entire process of preparing the algorithmically time-evolved state and inferring the expectation value $\langle\sigma_1^z\rangle$ is repeated for all simulation times t over the time domain of interest.

To provide a standard to benchmark the results of our QSP simulation algorithms, we determine the exact time-evolution operator by solving for the eigenvalues λ of the Hamiltonian (with associated eigenvectors $|\lambda\rangle$) and explicitly computing the exact time-evolution operator as

$$\left[e^{-i\mathcal{H}t}\right]_{\text{Exact}} = \sum_\lambda e^{-i\lambda t}|\lambda\rangle\langle\lambda|. \tag{38}$$

The exact values for $\langle\sigma_1^z\rangle$ can be calculated by substituting $|\psi\rangle = \left[e^{-i\mathcal{H}t}\right]_{\text{Exact}}|\psi_0\rangle$ into Eq. (36).

As an example, we first set $h_1(t) = h_2(t) = 0.5$ for all times t and $g_x = 1$, $g_y = g_z = 0$. The spin expectation $\langle\sigma_1^z\rangle$ and absolute error $|\langle\sigma_1^z\rangle - \langle\sigma_1^z\rangle_{\text{Exact}}|$ were measured using both algorithms over the time domain $t \in [0, 3.5]$.

The degrees of the QSP polynomials used for the two simulation schemes were chosen so that the number of queries made to the Hamiltonian block encoding in each algorithm would be nearly equivalent. For the QSP-LCU+ROAA algorithm, polynomials with degrees $d_{\cos} = 6$ and $d_{\sin} = 5$ were used to approximate $\cos(\mathcal{H}t)$ and $-i\sin(\mathcal{H}t)$, respectively. The total number of queries made to the Hamiltonian block encoding for the entire QSP-LCU+AA algorithm is thus $N_\mathcal{H}^{\text{ROAA}} = 3(d_{\cos} + d_{\sin}) = 33$. Note that $N_\mathcal{H}^{\text{ROAA}}$ is necessarily odd here because $d_{\sin}$ must both be odd, while $d_{\cos}$ must be even. On the other hand, for the coherent one-shot algorithm, the number of queries $N_\mathcal{H}^{\text{OS}}$ is necessarily even since the EECE is approximated with an even degree polynomial; subsequently, the EECE was approximated with degree $d_{\text{EECE}} = 32$ so that $N_\mathcal{H}^{\text{OS}} = 32$ was nearly equal to $N_\mathcal{H}^{\text{AA}}$.

The QSP phases corresponding to these polynomials were generated using the PyQSP optimization package with the scaling parameters $\alpha = 1.5$ and $\beta = 0.4$. Again, the parameter α was chosen to be slightly larger than the largest eigenvalue of $\mathcal{H}$ (which is equal to $\sqrt{2}$ in this case) so that all blocks of the Hamiltonian block encoding are real. Regarding the parameter β, recall that by Eq. (27) we expect an optimal β to exist for a given Hamiltonian and the desired error probability. However, the exact value of the optimal β also

depends on the choice of QSP polynomial and, therefore, may be difficult to calculate directly. Recognizing nonetheless that very long effective simulation times $\tau \propto 1/\beta$ must lead to increased error (and therefore decreased success probability), we selected a sufficiently large β to reduce the effective simulation time and boost the success probability. Note that β cannot be chosen too large, since the polynomial approximation to the EECE over the range $\left[\frac{1-\beta}{2}, \frac{1+\beta}{2}\right]$ will become inaccurate if this range is too wide.

The simulation results for the time-independent Heisenberg model are plotted in Fig. 8. While both algorithms very closely track the exact result for $t \leq 1.5$, the outputs of both algorithms begin to gradually accrue error at later time points. In this particular simulation, we observe that our novel coherent one-shot method introduces less error than QSP-LCU+ROAA at these later time points when each algorithm is provided (nearly) the same number of Hamiltonian queries; this observation is corroborated by the results of Fig. 7, which suggest that, in the large-time limit, one-shot should require fewer queries than QSP-LCU+ROAA to achieve a given simulation error.

Finally, the success probability $(1 - \delta)$ of each algorithm is determined by the probability that the desired block containing the $e^{-i\mathcal{H}t}$ matrix is accessed upon measurement of the circuit. The success probabilities for the QSP-LCU+ROAA and coherent one-shot algorithms were calculated to be 99.8% and 97.4% on average over the range of time $t \in [0, 3.5]$, respectively. These near-unity success probabilities indicate that the two methods indeed maintain a high degree of coherence.

c. Time-dependent simulation. We now explore the application of both efficient fully-coherent simulation algorithms to a Heisenberg model with a time-dependent magnetic field. In general, the time-evolution operator $\mathcal{U}_\mathcal{H}(t)$ of a time-dependent Hamiltonian $\mathcal{H}(t)$ corresponding to an evolution time t is given by

$$\mathcal{U}_\mathcal{H}(t) = \mathcal{T}\exp\left[-i\int_0^t \mathcal{H}(t')dt'\right], \tag{39}$$
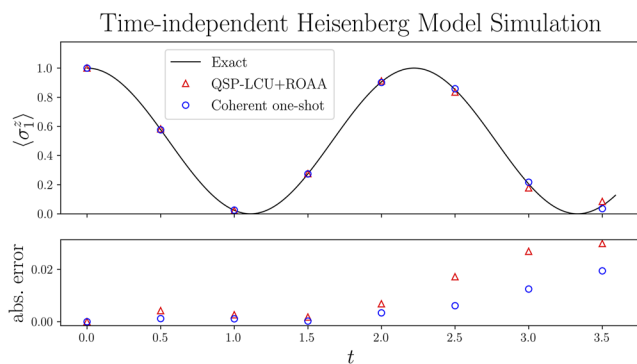
FIG. 8. Hamiltonian simulation for a two-spin Heisenberg model with a constant magnetic field $h_1 = h_2 = 0.5$ using both the conventional LCU algorithm with robust oblivious amplitude amplification (QSP-LCU+ROAA) and the coherent one-shot algorithm. The QSP-LCU+ROAA scheme made 33 queries to the Hamiltonian block-encoding for each simulation interval, while the one-shot scheme made 32. (Top) Expectation value of the first spin's z component at each simulation time interval. (Bottom) Absolute error between algorithm output and exact result at each simulation time interval.

where $\mathcal{T}$ is the time-ordering operator. This expression indicates that if the Hamiltonian does not commute with itself at different times, it does not suffice to naively integrate the Hamiltonian over the desired simulation period and exponentiate to obtain the time-evolution operator. Hence, time-dependent simulation cannot be performed with only a single call to either fully-coherent simulation algorithm, a by-product of the Hamiltonian's time-dependence and non-commutativity [as in Eq. (34)].

To circumvent this difficulty, we trotterize the full time-evolution operator into the product of many short time-evolution operators where the Hamiltonian is assumed to be constant over each short time period,

$$\mathcal{U}_{\mathcal{H}}(t) \approx \prod_{k=0}^{L-1} \exp[-i\mathcal{H}(k\Delta t)\Delta t]. \qquad (40)$$

Here, L is the number of discrete (Trotter) time steps used in the decomposition, and $\Delta t = t/L$ is the time step size. In the limit that $L \to \infty$ and $\Delta t \to 0$, the product approaches the exact time-evolution operator.

To implement Trotterization with our two algorithms, we employ each of the QSP-LCU+ROAA and coherent one-shot techniques to approximate the complex exponentials $\exp[-i\mathcal{H}(k\Delta t)\Delta t]$ for $k = 0, 1 \ldots, L-1$, resulting in a length L sequence for each algorithm. To simulate the time-evolution, the operators of each sequence are applied successively to a given input state $|\psi_0\rangle$ in accordance with Eq. (40), and the final output state is a close approximation to $\mathcal{U}_{\mathcal{H}}(t)|\psi_0\rangle$ for sufficiently small Δt.

Note that an implementation of Trotterization using an incoherent simulation algorithm (with a non-unit success probability) will require post-selection after each time slice. For the QSP-LCU simulation algorithm of Sec. II, which has probability of success close to $1/4$, this implies that the overall success probability decreases as $(1 - \delta) \sim 4^{-L}$, which approaches zero exponentially fast with an increasing number of Trotter steps L. This ruins the quantum simulation quickly and essentially forbids long-time Hamiltonian simulations. In contrast, we expect that our coherent Hamiltonian simulation protocol provides significant speed up in this case because the simulation of L time slices may be chained together coherently via simple multiplication without an exponential increase in the probability of failure.

Also note the fundamental difference between our strategies for simulating the time-independent and time-dependent cases: for the time-independent case, the output at each time t was the result of a single call to one of the fully-coherent simulation algorithms, and a new polynomial approximation to the complex exponential e^{-ixt} was generated for each simulation time t. In contrast, for the time-dependent case, the output at each time $t = k\Delta t$ is the result of k calls to one of our coherent simulation algorithms applied successively to the input state, and the *same* polynomial approximation to $e^{-ix\Delta t}$ is used for each call. The disadvantage of the latter strategy is that the total query complexity of the simulation acquires a multiplicative prefactor equal to the number of Trotter steps L, as is analyzed in the following.

The query complexity of the single call to coherent one-shot algorithm is given by Eq. (28) such that the query complexity of L calls is given by

$$N_{\mathcal{H}}^{\mathrm{OS,L}} = \Theta\left(L \cdot \left[\alpha|\Delta t| + \ln\left(\frac{1}{\epsilon'}\right) + \ln\left(\frac{1}{\delta'}\right)\right]\right), \qquad (41)$$

where ϵ' and δ' are the error and failure probability for each call of time Δt. To guarantee a final error of ϵ and final failure probability δ after L calls, it is sufficient to select $\epsilon' < \epsilon/L$ and $\delta' < \delta/L$, which upon substitution into Eq. (41) yields

$$N_{\mathcal{H}}^{\mathrm{OS,L}} = \Theta\left(\alpha|t| + L\ln\left(\frac{L}{\epsilon}\right) + L\ln\left(\frac{L}{\delta}\right)\right). \qquad (42)$$

It is easy to see that $N_{\mathcal{H}}^{\mathrm{OS,L}} \geq N_{\mathcal{H}}^{\mathrm{OS}}$ by comparison with Eq. (28), with equality at $L = 1$, as is expected.

As a benchmark for our simulations of the time-dependent model, we first calculate the exact time-evolution operator $\mathcal{U}_{\mathcal{H}}(t)$ over the desired domain of t. We begin with the time-dependent Schrödinger equation for the evolution of the state $|\psi(t)\rangle$,

$$i\frac{d}{dt}|\psi(t)\rangle = \mathcal{H}(t)|\psi(t)\rangle. \qquad (43)$$

Substituting $|\psi(t)\rangle = \mathcal{U}_{\mathcal{H}}(t)|\psi_0\rangle$ into the above equation and noting that $|\psi_0\rangle$ is constant, we obtain the following ordinary differential equation:

$$i\frac{d}{dt}\mathcal{U}_{\mathcal{H}}(t) = \mathcal{H}(t)\mathcal{U}_{\mathcal{H}}(t). \qquad (44)$$

Observe that solutions $\mathcal{U}_{\mathcal{H}}(t)$ of this equation will be of the form given in Eq. (39). The matrix version of Eq. (44) was solved numerically in Mathematica for finely spaced discrete times $t \in [0, 15]$, and the exact value for $\langle \sigma_1^z \rangle$ at a given time t was determined by applying the corresponding exact solution of $\mathcal{U}_{\mathcal{H}}(t)$ to $|\psi_0\rangle$ and substituting the resulting state into Eq. (36).

We simulated the evolution of the Heisenberg system for $t \in [0, 12]$ with the same initial state $|\psi_0\rangle = |00\rangle$ as in the time-independent case but instead with a linearly increasing onsite magnetic field $h_1(t) = h_2(t) = t/15$. Note for this simulation that $g_x = 1$, $g_y = g_z = 0$ were chosen again. A Trotter step size of $\Delta t = 0.5$ was used in the simulation, corresponding to a total number of Trotter steps $L = 24$. At each step, in the QSP-LCU+ROAA scheme, we used degrees $d_{\cos} = 2$ and $d_{\sin} = 3$ to approximate $\cos(\mathcal{H}\Delta t)$ and $\sin(\mathcal{H}\Delta t)$, respectively; in the coherent one-shot scheme, we used a $d = 14$ degree polynomial. Thus, the number of queries made to the Hamiltonian per step (or equivalently, per call to the simulation algorithm being used) was 15 for QSP-LCU+ROAA and 14 for coherent one-shot. The QSP phases corresponding to these polynomials were also solved using the PyQSP optimization package with the scaling parameters $\alpha = 2.5$ and $\beta = 0.25$; the QSP circuits for each algorithm were again modeled using the Qiskit Aer simulator. Similar to the time-independent case, the parameter α was chosen to be greater than the largest eigenvalue of $\mathcal{H}(t)$ (which is $\sqrt{5}$ in this example), while the parameter β was chosen to be sufficiently large such that a high success probability was still achieved for the one-shot scheme.

The simulation results for the time-dependent case are displayed in Fig. 9 with $\langle \sigma_1^z \rangle$ and its absolute error plotted at each Trotter step in the upper and lower panel, respectively. While the
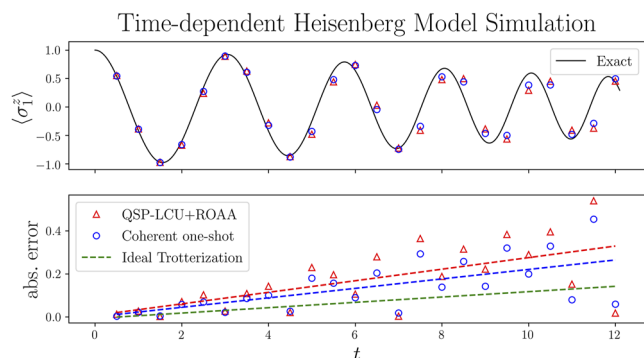
FIG. 9. Hamiltonian simulation for a two-spin 1D Heisenberg model with a linearly increasing magnetic field $h_1(t) = h_2(t) = t/15$ using both the conventional LCU algorithm with robust oblivious amplitude amplification (QSP-LCU+ROAA) and the coherent one-shot algorithm. The QSP-LCU+ROAA scheme made 15 queries to the Hamiltonian block-encoding for each Trotter step, while the one-shot scheme made 14. A Trotter step size of $\Delta t = 0.5$ is used. (Top) Expectation value of the first spin's z component $\langle \sigma_1^z \rangle$ at each simulation time interval. (Bottom) Absolute error between algorithm output and exact result at each simulation time interval. Trendlines (appropriately colored) are also included. The (non-QSP) error intrinsic to the Trotterization was calculated at each step, and a linear trendline for this ideal Trotterization error is shown in green.

outputs of both the QSP-LCU+ROAA and the coherent one-shot algorithms agree well with the exact results for the first few time steps, the absolute error increases roughly linearly in both cases as a function of the total simulation time t. This linear behavior in error vs time is expected as a direct result of the Trotterization because, in Eq. (40), ϵ accumulates linearly with t as $\epsilon \sim t^2/L \sim t\Delta t$ for fixed $\Delta t = t/L$. To determine this intrinsic Trotterization error, the ideal trotterized evolution operators were computed using the formula in Eq. (40) by numerically calculating $e^{-i\mathcal{H}\Delta t}$ for each individual Trotter step exactly. A linear trendline fitted to the error of this ideal Trotterization is also shown in Fig. 9, alongside similar trendlines for the total errors of the QSP-LCU+ROAA and one-shot results.

A lower slope of the error accumulation trendline reflects a higher quality approximation to the $e^{-i\mathcal{H}(k\Delta t)\Delta t}$ operators in each time step. Because the error trendline for our coherent one-shot algorithm has a lower slope than the trendline for the QSP-LCU+ROAA method, we conclude that for this particular simulation, our coherent one-shot algorithm introduces less error at each time step than QSP-LCU+ROAA. Relative to the ideal Trotterization error, the fitted error for one-shot at the end of the simulation period ($t = 12$) reached a value of 0.123, demonstrating roughly a factor of 1.5 improvement over the corresponding relative error 0.187 of the QSP-LCU+ROAA algorithm. Note further that by decreasing the step size Δt, the intrinsic Trotterization error can be arbitrarily minimized to achieve the desired simulation accuracy.

The total probabilities of success were also calculated after each Trotter step using the same techniques as in the time-independent case. Both the QSP-LCU+ROAA and coherent one-shot algorithms boasted near-unity success probabilities of 99.9% and 98.4% on average, respectively; both of these are expected to increase with the number of Hamiltonian queries. Such a high probability of success

after each step enables the sequential execution of many Trotter evolution steps without the need for classical repetition, which is a clear advantage not available to existing incoherent QSP-based simulation techniques.

B. Electronic dynamics of molecules

Electronic dynamics plays an important role in determining the chemical reactivity and response properties of molecules interacting with external field.[73] In this section, we demonstrate the ability of our algorithms to tackle these kind of applications by simulating the local charge oscillation dynamics of a H_2 molecule under a minimal basis. We choose H_2 to benchmark the performance of our algorithms because it is the simplest molecule with a correlated electronic structure. In principle, these algorithms can be extended to larger molecules beyond the capability of numerically exact classical algorithms for long-time electronic dynamics, given access to a fault-tolerant quantum computer.

We give numerical details of the electronic Hamiltonian of H_2 molecules in Sec. V B 1, followed by descriptions of the initial state and observables used in the simulation. In Sec. V B 3, expectation values of proper observables are presented as a function of simulation time, which reveals the local charge oscillation dynamics.

1. Electronic Hamiltonian of hydrogen molecule

We obtain the electronic Hamiltonian $\mathcal{H}$ of the H_2 molecule with an internuclear distance of 0.5 Å using a minimal STO-3G basis. To simulate the local charge oscillation dynamics, it is convenient to transform the molecular integrals from the canonical molecular orbital representation to localized orbitals. In this case, we use Lowdin's orthogonalized atomic orbitals.[74]

The Jordan–Wigner mapping[75] is then applied to map the second-quantized electronic Hamiltonian under the localized basis to qubits, resulting in a 4-qubit Hamiltonian with a dimension of 16×16. A unitary dilation (block-encoding) of this Hamiltonian is then formed using one additional ancilla qubit, in similar spirit to Eq. (35), giving a total of 5-qubit unitary with 32×32 dimensions. The Pauli sum representation of the H_2 Hamiltonian after Jordan–Wigner mapping is documented in Appendix D. Given this block-encoding, the rest of the simulation algorithm follows as in Sec. III.

2. Initial state and observables

To simulate the local charge oscillation, we start from a Slater determinant that occupies the two spin-orbitals on the hydrogen atom H_B and monitor the occupation number on atom H_A as the simulation time t grows (see the inset of Fig. 10). Since the two hydrogen atoms are completely equivalent, the system behaves much like a double-well potential, and the localized charge initially on H_B will migrate gradually to H_A and then back to H_B through tunneling. This process persists as no vibronic coupling is considered in our Hamiltonian. Note that, in practice, the electronic energy may gradually dissipate to vibrations of the H_2 molecule. We shall leave the case of vibronic simulation to future work.

In more concrete terms, assume that the four system qubits are in block-spin format where the first two are spin-up orbitals on hydrogen H_A and H_B and the last two are spin-down orbitals on H_A and H_B. As we have chosen the initial state to have one electron in
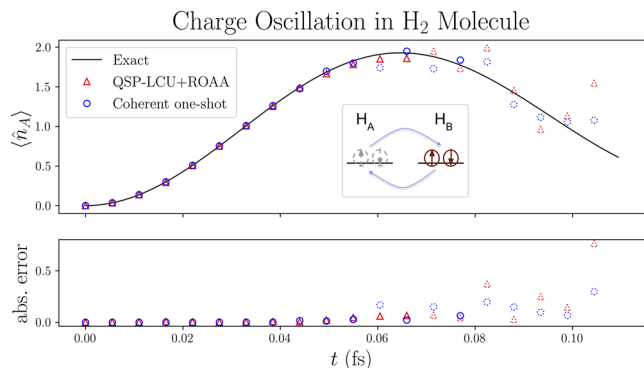
FIG. 10. Local charge oscillation dynamics in an H_2 molecule with an internuclear distance of 0.5 Å simulated using both the conventional LCU algorithm with robust oblivious amplitude amplification (QSP-LCU+ROAA) and the coherent one-shot algorithm. The time scale on the horizontal axis is given in femtoseconds. The QSP-LCU+ROAA scheme made 33 queries to the Hamiltonian block-encoding for each simulation interval, while the one-shot scheme made 32. The inset shows the arrangement of the localized orbitals and initial states. (Top) Expectation value of the total electron occupation number $\langle \hat{n}_A \rangle$ on atom H_A. (Bottom) Absolute error between the algorithm output and the exact result at each simulation time interval. Note for both plots that data points associated with success probability $1 - \delta > 0.85$ are presented with solid markers, while points with $1 - \delta \leq 0.85$ are instead presented with dotted markers.

each of the spin-up and spin-down orbitals on atom H_B, the initial state $|\psi\rangle_{\text{init}}$ is

$$|\psi\rangle_{\text{init}} = |0101\rangle, \qquad (45)$$

where the $|\cdot\rangle$ notation on the right-hand side represents a Slater determinant and the qubit state $|1\rangle$ ($|0\rangle$) stands for one (no) electron occupying the corresponding spin-orbital.

To quantify the charge dynamics, we monitor the expectation value of the total occupation number $\langle \hat{n}_A \rangle$ on hydrogen H_A, given by

$$\langle \hat{n}_A \rangle = \langle \psi | \left(a^\dagger_{A,\uparrow} a_{A,\uparrow} + a^\dagger_{A,\downarrow} a_{A,\downarrow} \right) | \psi \rangle, \qquad (46)$$

where $|\psi\rangle \in \{ [e^{-i\mathcal{H}t}]_{\text{ROAA}} |\psi_0\rangle, [e^{-i\mathcal{H}t}]_{\text{OS}} |\psi_0\rangle \}$ for increasing simulation time t. On quantum computers, the $\hat{n}_A$ operator is mapped to the following spin operators via the Jordan–Wigner mapping:

$$\hat{n}_A = \frac{I_{A,\uparrow} - Z_{A,\uparrow}}{2} + \frac{I_{A,\downarrow} - Z_{A,\downarrow}}{2}, \qquad (47)$$

where I and Z are the Pauli operators acting on the corresponding qubits.

3. Results

Figure 10 shows the time-dependence of $\langle \hat{n}_A \rangle$ after the initial state $|\psi\rangle_{\text{init}}$ is time-evolved using both the QSP-LCU+ROAA and coherent one-shot algorithms. To benchmark these numerical results, the values of $\langle \hat{n}_A \rangle$ corresponding to exact time-evolution [computed via Eq. (38)] are included in the plot as well. As in the earlier Heisenberg model simulations, the degrees of the QSP polynomials used in the two schemes were chosen so that the number of queries made to the Hamiltonian block encoding by each

algorithm would be nearly equivalent. Explicitly, polynomials with degrees $d_{\cos} = 6$ and $d_{\sin} = 5$ were used to approximate $\cos(\mathcal{H}t)$ and $-i \sin(\mathcal{H}t)$, respectively, in the QSP-LCU+ROAA scheme; a polynomial with degree $d_{\text{EECE}} = 32$ was used in the coherent one-shot scheme. Hence, QSP-LCU+ROAA employs 33 queries to the block-encoding per simulation interval, while one-shot employs 32. The QSP phases corresponding to these polynomials were obtained (as in the Heisenberg model simulations) using the PyQSP optimization package with the parameters $\alpha = 1$ and $\beta = 0.5$.

As anticipated, the total occupation number on atom H_A gradually increases from 0 to near 2 before charge is transferred back to atom H_B. This oscillation has a period of roughly 0.12 fs, which lies in the typical regime of attosecond electronic dynamics. For this particular simulation, our coherent one-shot algorithm produced results with lower error than those of QSP-LCU+ROAA: over the simulation intervals tested, one-shot resulted in an average population error of 0.064, compared to 0.094 for QSP-LCU+ROAA.

Note that for times $t < 0.06$ fs, both algorithms were found to succeed with reasonably high probability (namely, with success probabilities $1 - \delta > 0.85$). However, for some simulation intervals after $t = 0.06$ fs, the success probabilities of both algorithms dropped below this 0.85 threshold. These data points corresponding to lower success probability are distinguished with dotted markers in Fig. 10. Decreasing success probability is expected at long simulation times, since the QSP polynomial approximations to the functions $\cos(x\tau)$ and $\sin(x\tau)$ (in the case of QSP-LCU+ROAA) and $e^{-ix\tau}$ (in the case of coherent one-shot) deteriorate for large τ if the QSP polynomial degrees are held constant. If a higher success probability is desired at long simulation times, the success probability can be arbitrarily improved to satisfaction by increasing the degrees of the QSP polynomials used.

VI. DISCUSSION AND CONCLUSION

In this work, we have presented the concept of an *efficient fully-coherent* Hamiltonian simulation algorithm, which performs Hamiltonian simulation and retains the time evolved wave function with arbitrarily high success probability while also achieving a query complexity polynomial in the effective time $\|\mathcal{H}\| t$, the logarithm of the inverse error $\ln(1/\epsilon)$, and the logarithm of the inverse failure probability $\ln(1/\delta)$. We focused on developing said algorithms through QSP techniques, provided a block encoding of the Hamiltonian. We showed in Sec. II that conventional QSP-based simulation is not fully-coherent. However, it can be made fully-coherent by augmentation with amplitude amplification as in Sec. II D at the expense of appending a $\ln(1/\delta)$ multiplicative factor to the query complexity or by augmentation with robust oblivious amplitude amplification as in Sec. II E, which circumvents this factor.

We further introduced the concept of a pre-transformation in Sec. III and used it to develop our coherent one-shot Hamiltonian simulation algorithm, which achieves efficient fully-coherent simulation and attains a query complexity additive in $\ln(1/\delta)$. In Sec. IV, we compared the query complexities of the coherent algorithms, illustrating a trade-off between the coherent one-shot algorithm and QSP-based simulation augmented with robust oblivious amplitude amplification. Finally, the applications of our algorithms in Sec. V to the Heisenberg model and an H_2 molecule under a minimal basis

evidence these fully-coherent algorithms as powerful tools for the simulation of spin chains and correlated electronic dynamics.

As established in Sec. V, using a fixed number of queries to the Hamiltonian, the coherent one-shot algorithm is shown to efficiently simulate time evolution for both small and large values of times; thus, it is a highly effective tool for both time-independent Hamiltonian simulations with a single long-time algorithm call *and* trotterized time-dependent simulations with many short-time algorithm calls.

At the heart of coherent one-shot Hamiltonian simulation is the use of a pre-transformation to rescale the eigenvalues of the Hamiltonian. This is crucial as the complex exponential cannot be implemented as a QSP polynomial for arbitrary inputs but can be done over a restricted range as we analytically verified by the polynomial constructed in Sec. III. Looking toward extensions of this idea, we expect pre-transformations to generate a broader class of achievable transformations and also note that out framework of compressing the spectrum and then applying an even/odd extension of a function could certainly be generalized to functions other than the complex exponential. In addition, while we employed a linear pre-transformation here, it would be interesting to study the applicability of nonlinear pre-transformations. One way to construct such a pre-transformation would be via QSP itself, which suggests the power of concatenated QSP algorithms—algorithms that apply a sequence of QSP transformations.

Regarding improvements of the coherent one-shot algorithm, it would be highly desirable to attain a better analytic expression for a QSP polynomial that approximates $e^{-ix\tau}$ over a range of positive x. While the construction provided in Sec. III is sufficient, it is by no means optimal. Additionally, the query complexity $N_{\mathcal{H}}^{\mathrm{OS}}$ could likely be sharpened to better elucidate the trade-off between the coherent one-shot algorithm and the QSP-LCU+ROAA algorithm.

Moreover, one interesting application of the fully-coherent simulation algorithms discussed here is to time-dependent Hamiltonians, as discussed in Sec. V A 2. However, the lower-bound on errors in this case is limited by the Trotterization error, which suggests the utility of studying the coherent simulation of time-dependent Hamiltonians beyond Trotterization. For instance, it may be fruitful to combine a fully-coherent simulation algorithm with the time-dependent methods presented in Refs. 62 and 63. In addition, a fully-coherent algorithm could also be integrated with higher-order product formulas for the time evolution operator[46] or even invoked as a subroutine in specialized algorithms for local Hamiltonian simulation.[76]

We also note that efficient block-encoding of correlated electronic Hamiltonians of large systems is paramount to the success of QSP-based algorithms. We look forward to novel constructions of more efficient block-encodings for correlated electronic structure Hamiltonians beyond the LCU framework[77,78] and tensor-hyper contraction.[79]

We hope that this work will shed light on the importance of full-coherence in quantum algorithms. The fully-coherent Hamiltonian simulation algorithms presented here should be applicable as subroutines in larger quantum algorithms, where reliable passing of *quantum* data from one subroutine to another is required without measurement or post-selection. In this sense, the coherence of our Hamiltonian simulation algorithm, and hopefully many more to come, can facilitate integration with other quantum primitives and

give rise to further novel and powerful quantum algorithms. Moreover, in quantum dynamical processes of many physical systems, we expect the system itself passes quantum information coherently in some form from one part to the other. Because of this, we believe such capability to pass quantum data coherently from one subroutine to another in quantum simulation algorithms is crucial to model the quantum dynamics in a variety of chemical, biology, and condensed matter systems with ever increasing complexity. It is also our wish that this paper emphasizes the power and flexibility of QSP techniques for quantum simulation and beyond. As QSP has already revolutionized the design of quantum algorithms for problems of physical relevance, it is likely to also provide novel pathways to many applications of chemical relevance.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

John M. Martyn: Conceptualization (equal); Data curation (supporting); Formal analysis (lead); Methodology (lead); Writing – original draft (lead); Writing – review & editing (equal). **Yuan Liu**: Conceptualization (equal); Formal analysis (supporting); Methodology (equal); Writing – original draft (equal); Writing – review & editing (equal). **Zachary E. Chin**: Data curation (lead); Software (lead); Writing – original draft (equal); Writing – review & editing (equal). **Isaac L. Chuang**: Conceptualization (supporting); Funding acquisition (lead); Writing – review & editing (supporting).

DATA AVAILABILITY

The data that support the findings of this study are openly available in PyQSP package[68] and are available from the corresponding author upon reasonable request.

APPENDIX A: A FORAY INTO QSP, QET, AND QSVT

In this appendix, we review the technique of quantum signal processing, presenting the theorems spelled out in Refs. 56, 57, and 80. From a high level, QSP provides a systematic method to apply a nearly arbitrary polynomial transformation to a quantum subsystem. For a target polynomial of degree d, this is achieved using $\mathcal{O}(d)$ elementary quantum gates. In particular, one applies a specific sequence of $(d+1)$ SU(2) rotations to the subsystem of

interest, where each rotation is parameterized by an angle $\phi_k \in \mathbb{R}$. The QSP algorithm is parametric in that the polynomial transformation achieved is completely characterized by the choice of phase angles $\{\phi_k\}$.

Subsequently, we will discuss how QSP may be naturally generalized to apply a polynomial transformation to the eigenvalues of a matrix encoded in a block of a unitary matrix, in which case this procedure is known as a quantum eigenvalue transform (QET) and ultimately produces a polynomial transform of a matrix.[57,65,80] Similarly, this procedure may be further extended to the singular values of a possibly non-square matrix, in which case this is known as a quantum singular value transformation (QSVT), and the polynomial is applied to the singular values of the matrix.[64] This ability to polynomially transform the singular values of an arbitrary linear operator is incredibly powerful and has been shown to be a building block from which of nearly all known quantum algorithms may be derived.[64,65]

1. Overview of QSP

Quantum signal processing (QSP)[56,57,80] works by interleaving a signal rotation operator W and a signal processing rotation operator S. These operators are taken to be SU(2) rotations about different axes, where the signal rotation operator is a rotation through a fixed angle θ and the signal processing rotation operator is a rotation through a variable angle parameterized by a real number ϕ. Typically, W is taken to be an x-rotation and S a z-rotation, although other equivalent conventions exist.

Let us follow standard conventions and define the signal rotation operator as

$$W(x) = \begin{bmatrix} x & i\sqrt{1-x^2} \\ i\sqrt{1-x^2} & x \end{bmatrix}, \tag{A1}$$

which is an x-rotation through an angle $\theta = -2\cos^{-1} x$ ($x \in [-1,1]$). Similarly, define the signal processing rotation operator as

$$S(\phi) = e^{i\phi Z}, \tag{A2}$$

which is a z-rotation through an angle -2ϕ. Then, with a set of *QSP phases* $\vec{\phi} = (\phi_0, \phi_1, \ldots, \phi_d) \in \mathbb{R}^{d+1}$, one can construct the QSP sequence, $U^{\vec{\phi}}$, which is defined as the following interspersed sequence of W and S:

$$U^{\vec{\phi}} = S(\phi_0)\prod_{k=1}^{d} W(x)S(\phi_k) = e^{i\phi_0 Z}\prod_{k=1}^{d} W(x)e^{i\phi_k Z}. \tag{A3}$$

Curiously, the matrix elements of this operation are polynomials of x, parameterized by the QSP phases $\vec{\phi}$. In particular, the authors of Ref. 56 proved that

$$U^{\vec{\phi}} = e^{i\phi_0 Z}\prod_{k=1}^{d} W(a)e^{i\phi_k Z} = \begin{bmatrix} P(x) & iQ(x)\sqrt{1-x^2} \\ iQ^*(x)\sqrt{1-x^2} & P^*(x) \end{bmatrix}, \tag{A4}$$

where $P(x)$ and $Q(x)$ are polynomials that obey

1. $\deg(P) \leq d$, $\deg(Q) \leq d-1$,
2. $P(x)$ has parity $d \bmod 2$, $\quad$ (A5)
3. $|P(x)|^2 + (1-x^2)|Q(x)|^2 = 1$, $\forall\ x \in [-1,1]$.

This result is quite fascinating. It tells us that we can prepare polynomials of x by projecting into a block of $U^{\vec{\phi}}$. For instance, $\langle 0|U^{\vec{\phi}}|0\rangle = P(x)$, so $P(x)$ dictates the probability a particle in state $|0\rangle$ remains in its initial state upon application of $U^{\vec{\phi}}$.

The block encoding has thus far focused on the $|0\rangle\langle 0|$ matrix elements: $x = \langle 0|W(x)|0\rangle$ and $P(x) = \langle 0|U^{\vec{\phi}}|0\rangle$. While this choice seems natural, it is not necessary and is actually inhibitory. In particular, matrix elements of $U^{\vec{\phi}}$ in bases other than the computational basis may be projected out, in which case these matrix elements are linear combinations of $P(x)$ and $Q(x)\sqrt{1-x^2}$ (and their complex conjugates). For instance, we may project into the $\{|+\rangle, |-\rangle\}$ basis, wherein we find the useful result,

$$\langle +|U^{\vec{\phi}}|+\rangle = \text{Re}(P(x)) + i\,\text{Re}(Q(x))\sqrt{1-x^2}. \tag{A6}$$

This result is important because projecting into the $|+\rangle\langle +|$ matrix element allows us to construct a larger class of polynomials than those achieved by projection into the computational basis. Whereas $P(x)$ is required to obey $|P(x = \pm 1)| = 1$ as per the third condition in Eq. (A5), the above polynomial needs not obey this restriction. In particular, it can be shown that the projection into the $|+\rangle\langle +|$ matrix element can accurately approximate any real polynomial with parity $d \bmod 2$ such that $\deg(\text{Poly}) \leq d$, and $|\text{Poly}(a)| \leq 1\,\forall\ x \in [-1,1]$. This can be achieved by selecting an appropriate P whose real part approximates the desired function and a Q with a small real component. As we will see later, this additional polynomial freedom will be crucial to conventional QSP-based Hamiltonian simulation.

Implicit in this construction is that we can access the correct block of the QSP sequence, i.e., we can choose to access some matrix element such as $\langle 0|U^{\vec{\phi}}|0\rangle$. Accessing the correct block is fundamental to applications of QSP, as QSP-based algorithms will only be successful if a desired block is accessed. In such an application, we must project the final state into being in the desired block by performing a projective measurement. For instance, if our goal is to access the $|0\rangle\langle 0|$ block of the QSP sequence, we may measure the auxiliary qubit (used to achieve the block encoding) in the computational basis, which will isolate the desired polynomial if $|0\rangle$ is measured.

In general, however, this measurement will output the desired result with non-unit probability $p = |\langle 0|P(x)|0\rangle|^2$, which is dictated by the choice of polynomial. Nonetheless, the probability of accessing the correct block can be increased by using classical repetition or amplitude amplification. Although this procedure is implicitly performed in QSP-based algorithms, it has the downside that it requires multiple instance of QSP. As a result, if one desires a probability of success $\geq 1 - \delta$, then there is a δ-dependent multiplicative factor in the complexity of the QSP-based algorithm, and this factor is often ignored in statements of query complexity. For example, in the case of amplitude amplification, this factor is $\Theta\left(\ln\left(\frac{1}{\delta}\right)\right)$. In Sec. III,

we eliminated this multiplicative factor for QSP-based Hamiltonian simulation by presenting a fully-coherent simulation algorithm that succeeds with arbitrarily high probability and requires only a single QSP instantiation while retaining a near-optimal scaling of the query complexity.

2. Generalization to QET

The construction of QSP may be generalized to apply polynomial transformations to the eigenvalues of a matrix, a procedure that we call a *quantum eigenvalue transformation* (QET). To see this, note that QSP has the following interpretation: QSP begins with a matrix $W(x)$ that is effectively a block encoding of x,

$$W(x) = \begin{bmatrix} x & \cdot \\ \cdot & \cdot \end{bmatrix}. \tag{A7}$$

By interspersing this matrix with rotation operators parameterized by $\vec{\phi}$, we constructed an operator that block encodes a polynomial transformation of x, $P(x)$,

$$U^{\vec{\phi}} = \begin{bmatrix} P(x) & \cdot \\ \cdot & \cdot \end{bmatrix}. \tag{A8}$$

Hence, in terms of block encodings, this procedure performs the operation $x \mapsto P(x)$.

Paralleling this scenario, one may instead begin with a unitary block encoding U of a matrix $A = \sum_\lambda \lambda |\lambda\rangle\langle\lambda|$,

$$U = \begin{bmatrix} A & \cdot \\ \cdot & \cdot \end{bmatrix} \tag{A9}$$

(this is only possible if $\|A\| \leq 1$, which we assume here; if this condition is not met, one can rescale A by a constant to meet this condition and block encode this rescaled matrix.) More generally, we assume that A can be accessed with a projector Π as $A = \Pi U \Pi$.

In QET, one intersperses U with a rotation operator that acts as a z-rotation within each eigenspace of A. This operator, known as the projector-controlled phase shift, may be expressed as $\Pi_\phi := e^{i\phi(2\Pi - I)}$ for a rotation angle ϕ and can be straightforwardly constructed as per Fig. 11.[64]
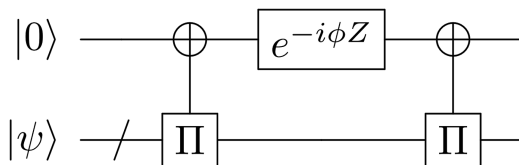
FIG. 11. The circuit used to realize the projector-controlled phase shift Π_ϕ (up to a global phase), as a product of projector-controlled-NOT gates and a z-rotation.

Paralleling Eq. (A4), one can then define the following QET sequence, which effectively performs QSP within each eigenspace and a polynomial transformation of A:

$$U^{\vec{\phi}} = \Pi_{\phi_0} \prod_{k=1}^{d} U \Pi_{\phi_k} \Rightarrow \Pi \ U^{\vec{\phi}} \Pi = \sum_\lambda P(\lambda) |\lambda\rangle\langle\lambda| = P(A). \tag{A10}$$

In terms of block encodings, QET maps a matrix $A \mapsto P(A)$, thus developing a polynomial transformation of the matrix. As in Appendix A 1, it is conventional that $\Pi = |0\rangle\langle0|$, in which case we may obtain $P(A)$ when we project into the $|0\rangle\langle0|$ matrix element of $U^{\vec{\phi}}$, which can be done with some finite probability by measuring the ancilla qubit used to achieve the encoding. Alternatively, we may also project into the $|+\rangle\langle+|$ component to achieve a larger class of polynomial transformations of A.

3. Generalization to QSVT

Generalizing even further, this construction may be adapted to apply a polynomial transformation to the singular values of an arbitrary linear operations (i.e., a possibly non-square matrix). Recall that we may write the singular value decomposition of a matrix A as $A = W\Sigma V^\dagger$, where W and V are unitaries and Σ contains the singular values (non-negative) along its diagonal. Denoting the singular values by $\{\sigma_i\}$, this may be re-expressed as

$$A = \sum_i \sigma_i |w_i\rangle\langle v_i|, \tag{A11}$$

where $|w_i\rangle$ and $|v_i\rangle$ are the columns of W and V respectively.

Suppose we have a unitary U that block encodes A as $\tilde{\Pi} U \Pi = A$, where $\tilde{\Pi}$ and Π are projectors. We may follow a procedure analogous to that of QET by interspersing this block encoding and its Hermitian conjugate (a block encoding of $A^\dagger$) with an operator that acts a rotation within the left and right singular vector spaces (where the left singular vector space is spanned by $\{|w_k\rangle\}$, and the right singular vector space is spanned by $\{|v_k\rangle\}$). As before, these are projector-controlled phase shifts, denoted by Π_ϕ and $\tilde{\Pi}_\phi$, respectively, and can be constructed analogous to Fig. 11.

We then define the following QSVT sequence, which performs QSP within each singular vector space and, thus, applies a polynomial transformation to the singular values. Its form differs from that of QSP and QET, as it must deal with the different left and right singular vector spaces,

$$U^{\vec{\phi}} = \begin{cases} \tilde{\Pi}_{\phi_1} U \left[\prod_{k=1}^{(d-1)/2} \Pi_{\phi_{2k}} U^\dagger \tilde{\Pi}_{\phi_{2k+1}} U \right], & d \text{ odd,} \\ \left[\prod_{k=1}^{d/2} \Pi_{\phi_{2k-1}} U^\dagger \tilde{\Pi}_{\phi_{2k}} U \right], & d \text{ even} \end{cases}$$

$$\Rightarrow \quad P^{(SV)}(A) = \begin{cases} \tilde{\Pi} U^{\vec{\phi}} \Pi, & d \text{ odd,} \\ \Pi U^{\vec{\phi}} \Pi, & d \text{ even,} \end{cases} \tag{A12}$$

where

$$P^{(SV)}(A) := \begin{cases} \sum_i P(\sigma_i) |w_i\rangle\langle v_i|, & P \text{ odd,} \\ \sum_i P(\sigma_i) |v_i\rangle\langle v_i|, & P \text{ even.} \end{cases} \tag{A13}$$

This protocol, known as the *quantum singular value transformation* (QSVT), is a powerful subroutine that has been demonstrated to

underlie nearly all quantum algorithms.[64,65] Note that $P^{(SV)}(A)$ differs for odd and even polynomials through the basis in which it is expressed. This arises because each application of A and $A^\dagger$ switches between the left and right singular vector spaces and vice versa, respectively, and the number of applications directly corresponds to the parity of the polynomial. This crucial distinction comes into play in Sec. II C.

Ultimately, QSP is at the crux of QET and QSVT. Hence, if we can develop a useful polynomial through QSP, say, one that accurately approximates a function of interest, then we may apply it to the eigenvalues of a matrix via QET or to the singular values via QSVT. A particularly useful such function is the complex exponential for Hamiltonian simulation, whose construction we discuss in Secs. II and III.

APPENDIX B: POLYNOMIAL APPROXIMATION OF THE SIGN FUNCTION

Here, we outline the construction of a polynomial approximation to the sign function. We closely follow the development of Refs. 81 and 82, wherein the full details and proofs of some statements employed here can be found.

To start the construction, we note that Ref. 81 proves that $\mathrm{erf}(kx) = \frac{2}{\sqrt{\pi}} \int_0^{kx} e^{-y^2}\, dy = \frac{2k}{\sqrt{\pi}} \int_0^x e^{-(ku)^2}\, du$ is an ϵ-approximation to the sign function for $|x| \ge \Delta/2$ if $k = \frac{\sqrt{2}}{\Delta}\sqrt{W(\frac{2}{\pi\epsilon^2})} < \frac{\sqrt{2}}{\Delta}\sqrt{\ln(\frac{2}{\pi\epsilon^2})}$ (for $\frac{2}{\pi\epsilon^2} > e$), where $W(x)$ is the Lambert W function. A polynomial approximation to the sign function may then be built by first constructing a polynomial approximation to $e^{-k^2 u^2}$, inserting it into this expression, and integrating as necessary.

The exponential function $e^{-a(x+1)}$ (which is bounded in magnitude by 1 for $x \in [-1, 1]$) may be approximated for $x \in [-1, 1]$ by truncating the following modified Jacobi–Anger expansion:

$$e^{-a(x+1)} = e^{-a}\left(I_0(a) + 2\sum_{j=1}^{\infty} I_j(a) T_j(-x)\right), \tag{B1}$$

where $I_j(x)$ is the modified Bessel function (of the first kind) of order j and $T_j(x)$ is the Chebyshev polynomial of order j. Reference 81 proves that one can attain a polynomial ϵ-approximation to $e^{-a(x+1)}$ by truncating this series at $j \ge \sqrt{2 \cdot \max(ae^2, \ln(2/\epsilon)) \cdot \ln(4/\epsilon)}$, which is thus a polynomial of degree at least $d_{\exp,a,\epsilon} := \sqrt{2 \cdot \max(ae^2, \ln(2/\epsilon)) \cdot \ln(4/\epsilon)}$. It will be convenient to express this as $\epsilon_{\exp,a,d} \le \epsilon$ for $d \ge d_{\exp,a,\epsilon}$, i.e., the error suffered is less than ϵ when the degree is greater than $d_{\exp,a,\epsilon}$.

Reference 81 further demonstrates how this construction may be used to estimate $e^{-(ku)^2}$ with a simple change of variables. Inserting the resulting polynomial into the expression $\mathrm{erf}(kx) = \frac{2k}{\sqrt{\pi}} \int_0^x e^{-(ku)^2}\, du$, we obtain a polynomial approximation to the $\mathrm{erf}(kx)$ that suffers error $\epsilon_{\mathrm{erf},k,d} \le \frac{4k}{\sqrt{\pi}d}\epsilon_{\exp,k^2/2,(d-1)/2}$.

Ultimately, we desire an $\epsilon/2$-approximation to $\mathrm{erf}(kx)$ with $k = \frac{\sqrt{2}}{\Delta}\sqrt{W(\frac{2}{\pi(\epsilon/2)^2})} = \frac{\sqrt{2}}{\Delta}\sqrt{W(\frac{8}{\pi\epsilon^2})}$ such that this function provides an ϵ-approximation to the sign function by the triangle inequality. Thus, it will suffice to choose

$$\frac{4k}{\sqrt{\pi}d}\epsilon_{\exp,k^2/2,(d-1)/2} \le \epsilon/2 \Rightarrow \epsilon_{\exp,k^2/2,(d-1)/2} \le \frac{\sqrt{\pi}}{8}\frac{\epsilon d}{k}, \tag{B2}$$

or equivalently,

$$(d-1)/2 \ge d_{\exp,k^2/2,\frac{\sqrt{\pi}}{8}\frac{\epsilon d}{k}}$$
$$= \sqrt{2 \cdot \max\left(\frac{e^2 k^2}{2},\ \ln\left(\frac{16k}{\sqrt{\pi}\epsilon d}\right)\right) \cdot \ln\left(\frac{32k}{\sqrt{\pi}\epsilon d}\right)}. \tag{B3}$$

If the first term is larger, then this bound requires

$$(d-1)/2 \ge ek\sqrt{\ln\left(\frac{32k}{\sqrt{\pi}\epsilon}\right) + \ln\left(\frac{1}{d}\right)}. \tag{B4}$$

Using the variable substitution $v = (d-1)/2$, squaring this equation, and using $\frac{1}{1+2v} < \frac{1}{2v}$, we find that it will suffice to choose a $v^2 \ge \frac{1}{2}e^2 k^2 W(\frac{512}{\pi e^2}\frac{1}{\epsilon^2})$, where $W(x)$ is the Lambert W function. Equivalently,

$$d \ge \frac{2e}{\Delta}\sqrt{W\left(\frac{8}{\pi\epsilon^2}\right) W\left(\frac{512}{e^2\pi}\frac{1}{\epsilon^2}\right)} + 1, \tag{B5}$$

where we have inserted the explicit expression for k.

On the other hand, if the second term in Eq. (B3) is larger, then the bound requires

$$(d-1)/2 \ge \sqrt{2 \cdot \ln\left(\frac{16k}{\sqrt{\pi}\epsilon d}\right)\left(\ln 2 + \ln\left(\frac{16k}{\sqrt{\pi}\epsilon d}\right)\right)}. \tag{B6}$$

Again setting $v = (d-1)/2$ and using the bound $\sqrt{x^2 + ax} \le x + a/2$, we find that this can be satisfied if

$$d \ge 2\sqrt{2}W\left(\frac{8\sqrt{2}}{\sqrt{\pi}\Delta\epsilon}\sqrt{W\left(\frac{8}{\pi\epsilon^2}\right)}\right) + 1. \tag{B7}$$

In general, then, it will suffice to truncate at degree

$$d = 2 \cdot \left\lceil \max\left(\frac{e}{\Delta}\sqrt{W\left(\frac{8}{\pi\epsilon^2}\right) W\left(\frac{512}{e^2\pi}\frac{1}{\epsilon^2}\right)},\right.\right.$$
$$\left.\left. \sqrt{2}W\left(\frac{8\sqrt{2}}{\sqrt{\pi}\Delta\epsilon}\sqrt{W\left(\frac{8}{\pi\epsilon^2}\right)}\right)\right)\right\rceil + 1 =: \gamma(\epsilon, \Delta), \tag{B8}$$

which is necessarily an odd integer.

Hence, in summary, an ϵ approximation to the sign function, valid for $|x| \ge \Delta/2$, can be constructed as polynomial of odd degree $\gamma(\epsilon, \Delta)$. Noting that $W(x) = \ln x - \ln(\ln x) + o(1) = \Theta(\ln(x))$ for large x, we see that $\gamma(\epsilon, \Delta)$ scales as $\gamma(\epsilon, \Delta) = \Theta\left(\frac{1}{\Delta}\ln\left(\frac{1}{\epsilon}\right)\right)$, corroborating the claims in the literature.[81]

APPENDIX C: QUANTUM CIRCUIT FOR UNITARY BLOCK-ENCODING

For completeness, the gate-level Hamiltonian block-encoding circuit for the two-spin Heisenberg model simulated in Sec. V A 2 is provided in Fig. 12. This circuit implements the unitary $U_{\mathcal{H}/\alpha}$
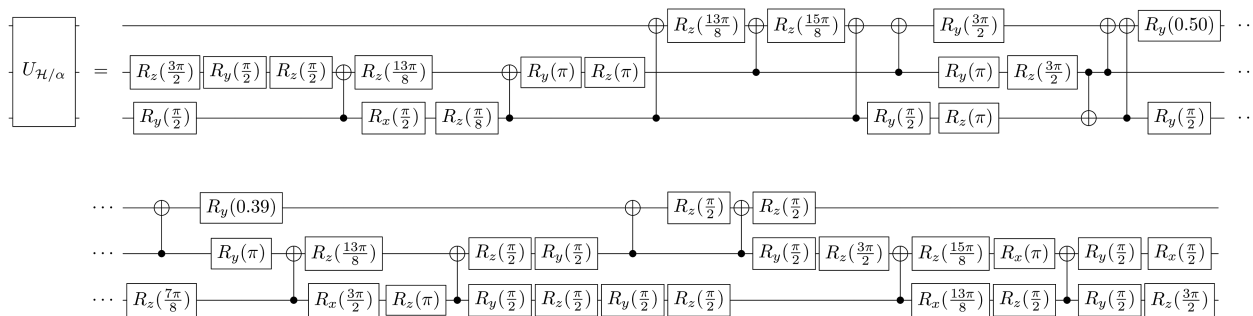
FIG. 12. Quantum circuit implementing the Hamiltonian block encoding $U_{\mathcal{H}/\alpha}$ of the two-qubit Heisenberg model simulated in Sec. V A 2. From top to bottom, the qubits in the circuit are ordered from most to least significant.

defined in Eq. (35) with $\alpha = 1.5$. Recall that this specific block-encoded Hamiltonian $\mathcal{H}$ is defined according to Eqs. (30)–(32) with the parameters $h_1(t) = h_2(t) = 0.5$ for all times t and $g_x = 1$, $g_y = g_z = 0$. This quantum circuit for $U_{\mathcal{H}/\alpha}$ was generated using the quantum Shannon decomposition method via the UniversalQCompiler software package in Mathematica.[83]

TABLE II. The Pauli sum representation of the H_2 electronic Hamiltonian at bond length of 0.5 Å under STO-3G basis.

Coefficients	Pauli operators
−0.678 523	IIII
−0.077 605	ZIII
−0.077 605	IZII
+0.134 592	ZZII
−0.077 605	IIZI
+0.222 157	ZIZI
+0.137 722	IZZI
−0.077 605	IIIZ
+0.137 722	ZIIZ
+0.222 157	IZIZ
+0.134 592	IIZZ
−0.291 540	XXII
−0.291 540	YYII
+0.001 571	XXZI
+0.001 571	YYZI
+0.001 571	XXIZ
+0.001 571	YYIZ
−0.291 540	IIXX
+0.001 571	ZIXX
+0.001 571	IZXX
−0.291 540	IIYY
+0.001 571	ZIYY
+0.001 571	IZYY
+0.003 129	XXXX
+0.003 129	YYXX
+0.003 129	XXYY
+0.003 129	YYYY

APPENDIX D: PAULI SUM REPRESENTATION OF ELECTRONIC HAMILTONIAN OF H_2

The Pauli sum representation of the H_2 electronic Hamiltonian is provided below in Table II.

REFERENCES

[1] R. P. Feynman, Int. J. Theor. Phys. **21**, 467 (1982).

[2] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, Proc. Natl. Acad. Sci. U. S. A. **114**, 7555 (2017).

[3] S. McArdle, S. Endo, A. Aspuru-Guzik, S. C. Benjamin, and X. Yuan, Rev. Mod. Phys. **92**, 015003 (2020).

[4] N. Lambert, Y.-N. Chen, Y.-C. Cheng, C.-M. Li, G.-Y. Chen, and F. Nori, Nat. Phys. **9**, 10 (2013).

[5] J. Cao, R. J. Cogdell, D. F. Coker, H.-G. Duan, J. Hauer, U. Kleinekathöfer, T. L. Jansen, T. Mančal, R. D. Miller, J. P. Ogilvie et al., Sci. Adv. **6**, eaaz4888 (2020).

[6] F. Krausz and M. Ivanov, Rev. Mod. Phys. **81**, 163 (2009).

[7] M. Maiuri, M. Garavelli, and G. Cerullo, J. Am. Chem. Soc. **142**, 3 (2020).

[8] L. Young, K. Ueda, M. Gühr, P. H. Bucksbaum, M. Simon, S. Mukamel, N. Rohringer, K. C. Prince, C. Masciovecchio, M. Meyer et al., J. Phys. B: At., Mol. Opt. Phys. **51**, 032003 (2018).

[9] A. Klein and D. Jaksch, Phys. Rev. A **73**, 053613 (2006).

[10] W. Hofstetter and T. Qin, J. Phys. B: At., Mol. Opt. Phys. **51**, 082001 (2018).

[11] J. Preskill, in The 36th Annual International Symposium on Lattice Field Theory. 22–28 July (2018), p. 24; arXiv:1811.10085 [hep-lat].

[12] W. A. de Jong, K. Lee, J. Mulligan, M. Płoskoń, F. Ringer, and X. Yao, Phys. Rev. D **106**, 054508 (2021); arXiv:2106.08394 [quant-ph].

[13] X. Li, N. Govind, C. Isborn, A. E. DePrince III, and K. Lopata, Chem. Rev. **120**, 9951 (2020).

[14] A. D. McLachlan and M. A. Ball, Rev. Mod. Phys. **36**, 844 (1964).

[15] P. Jorgensen, Annu. Rev. Phys. Chem. **26**, 359 (1975).

[16] E. Runge and E. K. U. Gross, Phys. Rev. Lett. **52**, 997 (1984).

[17] H.-D. Meyer, U. Manthe, and L. S. Cederbaum, Chem. Phys. Lett. **165**, 73 (1990).

[18] J. Zanghellini, M. Kitzler, T. Brabec, and A. Scrinzi, J. Phys. B: At., Mol. Opt. Phys. **37**, 763 (2004).

[19] E. Fromager, S. Knecht, and H. J. A. Jensen, J. Chem. Phys. **138**, 084101 (2013).

[20] L. Greenman, P. J. Ho, S. Pabst, E. Kamarchik, D. A. Mazziotti, and R. Santra, Phys. Rev. A **82**, 023406 (2010).

[21] N. Rohringer, A. Gordon, and R. Santra, Phys. Rev. A **74**, 043420 (2006).

[22] J. B. Schriber and F. A. Evangelista, J. Chem. Phys. **151**, 171102 (2019).

[23] C. Huber and T. Klamroth, J. Chem. Phys. **134**, 054113 (2011).

[24] A. F. White and G. K.-L. Chan, J. Chem. Theory Comput. **14**, 5690 (2018).

[25] A. F. White and G. K.-L. Chan, J. Chem. Theory Comput. **15**, 6137 (2019).

[26] P. Shushkov and T. F. Miller III, J. Chem. Phys. **151**, 134107 (2019).

[27] M. A. Cazalilla and J. B. Marston, Phys. Rev. Lett. **88**, 256403 (2002).

[28] S. R. White and A. E. Feiguin, Phys. Rev. Lett. **93**, 076401 (2004).

[29] A. J. Daley, C. Kollath, U. Schollwöck, and G. Vidal, J. Stat. Mech.: Theory Exp. **2004**, P04005.

[30] W. M. C. Foulkes, L. Mitas, R. J. Needs, and G. Rajagopal, Rev. Mod. Phys. **73**, 33 (2001).

[31] N. Makri and W. H. Miller, Chem. Phys. Lett. **139**, 10 (1987).

[32] J. D. Doll, R. D. Coalson, and D. L. Freeman, J. Chem. Phys. **87**, 1641 (1987).

[33] M. Schiró, Phys. Rev. B **81**, 085126 (2010).

[34] G. Cohen, E. Gull, D. R. Reichman, and A. J. Millis, Phys. Rev. Lett. **115**, 266802 (2015).

[35] M. Motta and S. Zhang, Wiley Interdisc. Rev.: Comput. Mol. Sci. **8**, e1364 (2018).

[36] M. S. Church and B. M. Rubenstein, J. Chem. Phys. **154**, 184103 (2021).

[37] J. K. Freericks, V. M. Turkowski, and V. Zlatić, Phys. Rev. Lett. **97**, 266408 (2006).

[38] J. S. Kretchmer and G. K.-L. Chan, J. Chem. Phys. **148**, 054108 (2018).

[39] Y. Cao, J. Romero, J. P. Olson, M. Degroote, P. D. Johnson, M. Kieferová, I. D. Kivlichan, T. Menke, B. Peropadre, N. P. D. Sawaya et al., Chem. Rev. **119**, 10856 (2019).

[40] I. D. Kivlichan, N. Wiebe, R. Babbush, and A. Aspuru-Guzik, J. Phys. A: Math. Theor. **50**, 305301 (2017).

[41] I. Kassal, S. P. Jordan, P. J. Love, M. Mohseni, and A. Aspuru-Guzik, Proc. Natl. Acad. Sci. U. S. A. **105**, 18681 (2008).

[42] S. Lee, J. Lee, H. Zhai, Y. Tong, A. M. Dalzell, A. Kumar, P. Helms, J. Gray, Z.-H. Cui, W. Liu et al., arXiv:2208.02199 (2022).

[43] S. Lloyd, Science **273**, 1073 (1996).

[44] M. Suzuki, J. Math. Phys. **32**, 400 (1991).

[45] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Commun. Math. Phys. **270**, 359–371 (2006).

[46] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe, and S. Zhu, Phys. Rev. X **11**, 011020 (2021).

[47] Y. Su, H.-Y. Huang, and E. T. Campbell, Quantum **5**, 495 (2021).

[48] M. C. Tran, Y. Su, D. Carney, and J. M. Taylor, PRX Quantum **2**, 010323 (2021).

[49] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Phys. Rev. Lett. **114**, 090502 (2015).

[50] L. Novo and D. Berry, Quantum Inf. Comput. **17**, 623 (2017).

[51] Q. Zhao and X. Yuan, Quantum **5**, 534 (2021); arXiv:2103.07988 [quant-ph].

[52] A. M. Childs, Phys. Rev. Lett. **102**, 180501 (2009).

[53] D. W. Berry, A. M. Childs, and R. Kothari, in *2015 IEEE 56th Annual Symposium on Foundations of Computer Science* (IEEE, 2015).

[54] D. W. Berry and A. M. Childs, Quantum Inf. Comput. **12**, 29 (2012).

[55] Z. Zhang, Q. Wang, and M. Ying, arXiv:2105.11889 [quant-ph] (2021).

[56] G. H. Low, T. J. Yoder, and I. L. Chuang, Phys. Rev. X **6**, 041067 (2016).

[57] G. H. Low and I. L. Chuang, Quantum **3**, 163 (2019).

[58] D. R. Weinberg, C. J. Gagliardi, J. F. Hull, C. F. Murphy, C. A. Kent, B. C. Westlake, A. Paul, D. H. Ess, D. G. McCafferty, and T. J. Meyer, Chem. Rev. **112**, 4016 (2012).

[59] K. Ramasesha, S. R. Leone, and D. M. Neumark, Annu. Rev. Phys. Chem. **67**, 41 (2016).

[60] F. Calegari, D. Ayuso, A. Trabattoni, L. Belshaw, S. De Camillis, S. Anumula, F. Frassetto, L. Poletto, A. Palacios, P. Decleva et al., Science **346**, 336 (2014).

[61] I. Rotter and J. P. Bird, Rep. Prog. Phys. **78**, 114001 (2015).

[62] Y.-H. Chen, A. Kalev, and I. Hen, PRX Quantum **2**, 030342 (2021).

[63] G. H. Low and N. Wiebe, arXiv:1805.00675 [quant-ph] (2018).

[64] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, in Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, 2019.

[65] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, PRX Quantum **2**, 040203 (2021); arXiv:2105.02859 [quant-ph].

[66] G. H. Low and I. L. Chuang, arXiv:1707.05391 [quant-ph] (2017).

[67] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum Comput. Inf. **305**, 53–74 (2002).

[68] I. Chuang, A. Tan, and J. M. Martyn, "PyQSP: Python quantum signal processing," https://github.com/ichuang/pyqsp, 2020.

[69] Y. Dong, K. B. Whaley, and L. Lin, arXiv:2108.03747 [quant-ph] (2021).

[70] A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, Proc. Natl. Acad. Sci. U. S. A. **115**, 9456 (2018).

[71] J. M. Martyn, Z. E. Chin, and Y. Liu, "Coherent Hamiltonian simulation," https://github.com/jmmartyn/Coherent_One_Shot, 2021.

[72] S. Anis et al., "Qiskit: An open-source framework for quantum computing," 2021.

[73] M. Nisoli, P. Decleva, F. Calegari, A. Palacios, and F. Martín, Chem. Rev. **117**, 10760 (2017).

[74] P. O. Löwdin, J. Chem. Phys. **18**, 365 (1950).

[75] P. Jordan and E. P. Wigner, *The Collected Works of Eugene Paul Wigner* (Springer, 1993), pp. 109–129.

[76] J. Haah, M. B. Hastings, R. Kothari, and G. H. Low, SIAM J. Comput. **0**, FOCS18 (2021).

[77] A. M. Childs, R. Kothari, and R. D. Somma, SIAM J. Comput. **46**, 1920–1950 (2017).

[78] L.-C. Wan, C.-H. Yu, S.-J. Pan, S.-J. Qin, F. Gao, and Q.-Y. Wen, Phys. Rev. A **104**, 062414 (2021).

[79] J. Lee, D. W. Berry, C. Gidney, W. J. Huggins, J. R. McClean, N. Wiebe, and R. Babbush, PRX Quantum **2**, 030305 (2021).

[80] G. H. Low and I. L. Chuang, Phys. Rev. Lett. **118**, 010501 (2017).

[81] G. H. Low, "Quantum signal processing by single-qubit dynamics," Ph.D. thesis, Massachusetts Institute of Technology, 2017.

[82] S. Sachdeva and N. K. Vishnoi, Found. Trends Theor. Comput. Sci. **9**, 125 (2014).

[83] R. Iten, O. Reardon-Smith, E. Malvetti, L. Mondada, G. Pauvert, E. Redmond, R. S. Kohli, and R. Colbeck, arXiv:1904.01072 [quant-ph] (2021).