

# Hamiltonian Simulation by Qubitization

Guang Hao Low<sup>1</sup> and Isaac L. Chuang<sup>2</sup>

<sup>1</sup>Department of Physics, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA

<sup>2</sup>Department of Electrical Engineering and Computer Science, Department of Physics, Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA

July 12, 2019

We present the problem of approximating the time-evolution operator  $e^{-i\hat{H}t}$  to error  $\epsilon$ , where the Hamiltonian  $\hat{H} = (|G\rangle\langle G| \otimes \hat{\mathcal{I}})\hat{U}(|G\rangle\langle G| \otimes \hat{\mathcal{I}})$  is the projection of a unitary oracle  $\hat{U}$  onto the state  $|G\rangle$  created by another unitary oracle. Our algorithm solves this with a query complexity  $\mathcal{O}(t + \log(1/\epsilon))$  to both oracles that is optimal with respect to all parameters in both the asymptotic and non-asymptotic regime, and also with low overhead, using at most two additional ancilla qubits. This approach to Hamiltonian simulation subsumes important prior art considering Hamiltonians which are  $d$ -sparse or a linear combination of unitaries, leading to significant improvements in space and gate complexity, such as a quadratic speed-up for precision simulations. It also motivates useful new instances, such as where  $\hat{H}$  is a density matrix. A key technical result is ‘qubitization’, which uses the controlled version of these oracles to embed any  $\hat{H}$  in an invariant  $SU(2)$  subspace. A large class of operator functions of  $\hat{H}$  can then be computed with optimal query complexity, of which  $e^{-i\hat{H}t}$  is a special case.

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Overview of the Quantum Signal Processor</b>	<b>5</b>
<b>3</b>	<b>Explicit Encodings of <math>\hat{H} = ( G\rangle\langle G  \otimes \hat{\mathcal{I}})\hat{U}( G\rangle\langle G  \otimes \hat{\mathcal{I}})</math></b>	<b>7</b>
3.1	Linear Combination of Unitaries . . . . .	7
3.2	$d$ -Sparse Hamiltonians . . . . .	8
3.3	Purified Density Matrix . . . . .	9
<b>4</b>	<b>Qubitization in a Quantum Signal Processor</b>	<b>9</b>
<b>5</b>	<b>Operator Function Design on a Quantum Signal Processor</b>	<b>11</b>
5.1	Ancilla-Free Quantum Signal Processing . . . . .	12
5.2	Single-Ancilla Quantum Signal Processing . . . . .	12
<b>6</b>	<b>Application to Hamiltonian Simulation</b>	<b>15</b>
<b>7</b>	<b>Conclusion</b>	<b>16</b>
7.1	Developments after preprint release . . . . .	17
<b>8</b>	<b>Acknowledgments</b>	<b>17</b>
<b>A</b>	<b>Qubitization of Normal Operators</b>	<b>17</b>
A.1	Minimal example . . . . .	18
<b>B</b>	<b>Practical Details for Implementing Hamiltonian Simulation</b>	<b>20</b>

# 1 Introduction

Quantum computers were originally envisioned as machines for efficiently simulating quantum Hamiltonian dynamics. As Hamiltonian simulation is BQP-complete, the problem is believed to be intractable by classical computers, and remains a strong primary motivation. The first explicit quantum algorithms for Hamiltonian simulation were discovered by Lloyd [1] for local interactions, and then generalized by Aharonov and Ta-Shma [2] to sparse Hamiltonians. Celebrated achievements over the years [3–7] have each ignited a flurry of activity in diverse applications from quantum algorithms [8–11] to quantum chemistry [12–18]. In this dawning era of the small quantum computer [19, 20], the relevance and necessity of space and gate efficient procedures for practical Hamiltonian simulation has intensified.

The cost of simulating the time-evolution operator  $e^{-i\hat{H}t}$  depends on several factors: the number of system qubits  $n$ , evolution time  $t$ , target error  $\epsilon$ , and how information on the Hamiltonian  $\hat{H}$  is accessed by the quantum computer. This field has progressed rapidly following groundbreaking work in the fractional query model [21], which was the first to achieve query complexities that depend logarithmically on error. This was generalized by Berry, Childs, Cleve, Kothari, and Somma (BCCKS) [22] to the common case where  $\hat{H} = \sum_{j=1}^d \alpha_j \hat{U}_j$  is a linear combination of  $d$  unitaries<sup>1</sup> with an algorithm using  $\mathcal{O}\left(\frac{\log(d) \log(\tau/\epsilon)}{\log \log(\tau/\epsilon)}\right)$  ancilla qubits and only  $\mathcal{O}\left(\frac{\tau \log(\tau/\epsilon)}{\log \log(\tau/\epsilon)}\right)$  queries, where  $\tau \leftarrow \|\vec{\alpha}\|_1 t$  and  $\|\vec{\alpha}\|_1 = \sum_{j=1}^d |\alpha_j|$  is the usual  $L_1$  norm. Subsequently, an extension was made to  $d$ -sparse Hamiltonians [6] that have at most  $d$  non-zero elements per row, also using  $\mathcal{O}\left(\frac{\tau \log(\tau/\epsilon)}{\log \log(\tau/\epsilon)}\right)$  queries with  $\tau \leftarrow dt \|H\|_{\max}$ . A prominent open question in all these works was whether the *additive* lower bound  $\Omega\left(\tau + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)^2$  was achievable for any of these models.

Recently, we found for the  $d$ -sparse model [7], which is popular in algorithm design by quantum walks [23], a procedure achieving the optimal trade-off between all parameters up to constant factors, with query complexity  $\mathcal{O}(\tau + \log(1/\epsilon))^3$ . This strictly linear-time performance with additive complexity is a quadratic improvement over prior art for precision simulations when  $t = \Theta(\log(1/\epsilon))$ . Moreover, the number of ancilla qubits required is independent of  $\tau$  and  $\epsilon$ , which is another important practical improvement. The approach, based on Childs’ [4, 24, 25] extension of Szegedy’s quantum walk [26], required two quantum oracles: one accepting a row  $j$  and column  $k$  index to return the value of entry  $\hat{H}_{jk}$  to  $m$  bits of precision, and another accepting a  $j$  row and  $l$  sparsity index to return in-place the  $l^{\text{th}}$  non-zero entry in row  $j$ .

Unfortunately, the  $d$ -sparse model is less appealing in practical implementations for several reasons. First, it is exponentially slower than BCCKS when the  $\hat{U}_j$  are of high weight with sparsity  $\mathcal{O}(2^n)$ . Second, its black-box oracles can be challenging to realize. Avoiding the exponential blowup by exploiting sparsity requires that positions of non-zero elements are both few in number and efficiently row-computable, which is not always the case. Third, the Childs quantum walk requires a doubling of the  $n$  system qubits, which is not required by BCCKS. It is unclear whether our methodology could be applied to other formulations of Hamiltonian simulation, in contrast to alternatives that seem more flexible [27].

Ideally, the best features of these two algorithms could be combined, such as in Table 1. For example, given the decomposition  $\hat{H} = \sum_{j=1}^d \alpha_j \hat{U}_j$ , one would like the optimal additive complexity of sparse Hamiltonian simulation, but with the BCCKS oracles that are more straightforward to implement. Furthermore, one could wish for a constant ancilla overhead, of say  $\lceil \log_2(d) \rceil + 2$ , supe-

<sup>1</sup>A particularly interesting and general class of systems are  $k$ -local Hamiltonians with  $m$  terms. These are also  $m2^k$ -sparse, and can be expressed as a linear combination of  $\leq m4^k$  Pauli operators, each of which are unitary.

<sup>2</sup>A slightly tighter lower bound is  $\Omega(q)$  queries, where  $q = \min\{q \in \mathbb{Z}^+ : \epsilon < |\sin(|\tau|/q)|^q = \mathcal{O}((\tau/q)^q)\}$  [6, 21]. Note that  $q \in \mathcal{O}(\tau + \log(1/\epsilon)) \cap \Omega\left(\tau + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$ .

<sup>3</sup>In the respective limits  $\tau = \Theta(1)$  and  $\epsilon = \Theta(1)$ , the query complexity is  $\mathcal{O}\left(\frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$  and  $\mathcal{O}(\tau)$ . Importantly, this does not imply a complexity of  $\mathcal{O}\left(\tau + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$ . The precise form of the upper bound matches the lower bound with respect to all parameters up to constant factors and lies between  $\Omega\left(\tau + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}\right)$  and  $\mathcal{O}(\tau + \log(1/\epsilon))$

Algorithm	Model	Ancilla qubits	Query Complexity	Gates per Query $\mathcal{O}(\cdot)$	
			$\mathcal{O}(\cdot)$	Oracle	Primitive
LC [7]	Sparse	$n + \mathcal{O}(m)$	$dt + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}$	Varies	$n + \text{poly}(m)$
BCKKS [22]	$\sum_j \alpha_j \hat{U}_j$	$\mathcal{O}\left(\frac{\log(d) \log(\ \vec{\alpha}\ _1 t/\epsilon)}{\log \log(\ \vec{\alpha}\ _1 t/\epsilon)}\right)$	$\frac{\ \vec{\alpha}\ _1 t \log(\ \vec{\alpha}\ _1 t/\epsilon)}{\log \log(\ \vec{\alpha}\ _1 t/\epsilon)}$	$dC$	$\log(d)$
LMR [5]	Mixed $\rho$	$n + 1$	$t^2/\epsilon$	Varies	$\log(n)$
<b>Theorem 1</b>	$\langle G \hat{U} G\rangle$	$\lceil \log_2(d) \rceil + 2$	$t + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}$	Varies	$\log(d)$
<b>Corollary 16</b>	$\sum_j \alpha_j \hat{U}_j$	$\lceil \log_2(d) \rceil + 2$	$\ \vec{\alpha}\ _1 t + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}$	$dC$	$\log(d)$
<b>Corollary 17</b>	Purified $\rho$	$n + \lceil \log_2(d) \rceil + 2$	$t + \frac{\log(1/\epsilon)}{\log \log(1/\epsilon)}$	Varies	$\log(n)$

Table 1: Comparison of state-of-art with our new approaches (bottom three lines) for approximating  $e^{-i\hat{H}t}$  of  $\hat{H} \in \mathbb{C}^{2^n \times 2^n}$  with error  $\epsilon$ . The  $d$ -sparse simulation oracle describes entries of  $\hat{H}$  with maximum absolute value  $\|\hat{H}\|_{\max} = 1$  to  $m$  bits of precision. The BCKKS oracle provides the decomposition  $\hat{H} = \sum_{j=1}^d \alpha_j \hat{U}_j$ , and each  $\hat{U}_j$  is given a cost  $\mathcal{O}(C)$ . The LMR query complexity refers to samples of the density matrix  $\hat{\rho} = \hat{H}$ . This work generalizes the above with oracles  $\hat{G}|0\rangle = |G\rangle \in \mathbb{C}^d, \hat{U} \in \mathbb{C}^{2^n \times 2^n}$  such that  $\langle G|\hat{U}|G\rangle = \hat{H}$ , where  $\|\hat{H}\| \leq 1$ . A new model where the oracle that outputs the purification  $|\rho\rangle = \sum_{j=1}^d \alpha_j |j\rangle_a |\psi_j\rangle$ ,  $\text{Tr}_a[|\rho\rangle\langle\rho|] = \hat{\rho}$  is provided.

rior to either algorithm. These improvements would greatly enhance the potential of early practical applications of quantum computation.

We achieve precisely this optimistic fusion via an extremely general procedure, made possible by what we call ‘qubitization’, that subsumes both and motivates new formulations of Hamiltonian simulation, as captured by this main theorem:

**Theorem 1** (Optimal Hamiltonian simulation by Qubitization). *Let  $(\langle G|_a \otimes \hat{\mathcal{I}}_s) \hat{U} (|G\rangle_a \otimes \hat{\mathcal{I}}_s) = \hat{H} \in \mathbb{C}^{N \times N}$  be Hermitian for some unitary  $\hat{U} \in \mathbb{C}^{N^d \times N^d}$  and some state preparation unitary  $\hat{G}|0\rangle_a = |G\rangle_a \in \mathbb{C}^d$ . Then  $e^{-i\hat{H}t}$  can be simulated for time  $t$ , error  $\epsilon$  in spectral norm, and failure probability  $\mathcal{O}(\epsilon)$ , using at most  $\lceil \log_2(N) \rceil + \lceil \log_2(d) \rceil + 2$  qubits in total,  $\Theta(Q)$  queries to controlled- $\hat{G}$ , controlled- $\hat{U}$ , and their inverses, and  $\mathcal{O}(Q \log(d))$  additional two-qubit quantum gates where*

$$Q = \min \left\{ q \in \mathbb{Z}^+ : \epsilon \geq \frac{4|t|^q}{q!2^q} = \mathcal{O} \left( \left( \frac{e\tau}{2q} \right)^q \right) \right\} = \mathcal{O}(t + \log(1/\epsilon)). \quad (1)$$

The optimality of the procedure follows by using oracles  $\hat{G}$  and  $\hat{U}$  that implement the Childs quantum walk. Furthermore, the transparent Hamiltonian input model of [Theorem 1](#) significantly expedites the development of new useful formulations of Hamiltonian simulation. For instance, we easily obtain a new result for the scenario where  $\hat{H}$  is a density matrix  $\hat{\rho}$ . Whereas  $\hat{\rho}$  can be produced by discarding the ancilla of some output from a quantum circuit  $\hat{G}$ , we instead keep this ancilla, leading to an unconditional quadratic improvement in time scaling, and an exponential improvement in error scaling over the sample-based Lloyd, Mohseni, and Rebentrost (LMR) model [5, 28], as summarized in [Table 1](#). Though certainly a stronger model than was previously considered, the inputs to many quantum machine learning as well as quantum semidefinite programming [11] applications are actually of this more restricted form, and are thus enhanced.

In fact, Hamiltonian simulation is just one application of our main innovation: an approach we call the ‘quantum signal processor’, where the equation  $(\langle G|_a \otimes \hat{\mathcal{I}}_s) \hat{U} (|G\rangle_a \otimes \hat{\mathcal{I}}_s) = \hat{H}$  in [Theorem 1](#) is interpreted as a non-unitary *signal* operator  $\hat{H}$  encoded in a subspace of an oracle  $\hat{U}$  flagged by  $|G\rangle_a \otimes \hat{\mathcal{I}}_s$ . This general encoding defined in [Definition 1](#)<sup>4</sup> unveils a systematic approach toward exponential quantum speedups with generic inputs, and is central to our results:

<sup>4</sup>After the original preprint release of this manuscript, standard-form encoding has been replaced in the community by ‘block-encoding’ [29], which is a more informative description.

Problem	BCCKS [6]	$d$ -sparse [7]	Evolution by $\rho$	QPE	QLSP [9]	Gibbs [10]
$\hat{H}$	Hamiltonian	Hamiltonian	Density matrix	Unitary	Matrix	Hamiltonian
$\hat{U}$	Selects $\hat{U}_j$	Isometry $\hat{T}$	SWAP	Any	Any	Any
$\hat{G}$	$\hat{U}_j$ coefficients	Identity	Purified $\rho$	Any	Any	Any
Solution	$e^{-i\hat{H}t}$	$e^{-i\hat{H}t}$	$e^{-i\hat{\rho}t}$	Decision	$\hat{H}^{-1}$	$e^{-\beta\hat{H}}$

Table 2: List of six example problems (top row), solvable using the quantum signal processor approach to compute an operator function  $f[\cdot]$  of Hermitian  $\hat{H} = \langle G|\hat{U}|G\rangle$ . Through qubitization, the scope of inputs to the Quantum Linear Systems Problem (QLSP) and Gibbs Sampling (Gibbs) can be any  $\hat{H}$  of this form, either indirectly through Hamiltonian simulation, or directly through quantum signal processing. Quantum Phase Estimation (QPE) here decides whether eigenphases  $\theta$  of an implemented unitary satisfy some property e.g.  $f(\theta) \geq 1/2$ .

**Definition 1** (Standard-form). *A signal operator  $\hat{H}$  with spectral norm  $\|\hat{H}\| \leq 1$  is encoded in the standard-form if we may query a unitary oracle  $\hat{U} : \mathcal{H}_a \otimes \mathcal{H}_s \rightarrow \mathcal{H}_a \otimes \mathcal{H}_s$  and a unitary state preparation oracle  $\hat{G}|0\rangle_a = |G\rangle_a \in \mathcal{H}_a$  with the property  $(\langle G|_a \otimes \hat{\mathcal{I}}_s)\hat{U}(|G\rangle_a \otimes \hat{\mathcal{I}}_s) = \hat{H}$ . We also assume query access to the inverses and controlled versions of  $\hat{U}, \hat{G}$ .*

The inputs to many problems, highlighted in Table 2, can be of this form. We introduce ‘qubitization’: the essential first step that converts this description of  $\hat{H}$  into unitary evolution with properties that depend directly on  $\hat{H}$ . When  $\hat{H}$  is Hermitian, The oracles  $\hat{G}$  and  $\hat{U}$  are queried to obtain a Grover-like search parallelized over all eigenvalues  $\lambda$  of  $\hat{H}$  through the *iterate*, which is isomorphic to  $e^{-iY \otimes \cos^{-1}[\hat{H}]}$  in some subspace. Similar structures are of foundational importance to many quantum algorithms – the gap  $\Delta$  of eigenvalues  $\lambda = 1 - \Delta$  of  $\hat{H}$  is amplified to  $\cos^{-1}(\lambda) = \mathcal{O}(\sqrt{\Delta})$  in the phase of the iterate, which resembles spectral gap amplification [30], the quantization of stochastic matrices [31], as well as Szegedy’s [26] and Childs’ [24] quantum walk. The key difference lies in the generalized encoding of the signal operator through any  $\hat{G}$  and  $\hat{U}$  of Definition 1, instead of the more restrictive sparse Hamiltonian formulation.

Unlike Grover search, we do not seek to prepare some target state. Rather, we exploit a direct sum of Grover-like rotations that are each isomorphic to  $SU(2)$ , to engineer arbitrary target functions  $f(\lambda)$  of its overlap  $\lambda$ . The quantum signal processor exploits this structure to attack the often-considered problem of designing a quantum circuit  $\hat{Q}$  that queries  $\hat{G}$  and  $\hat{U}$  such that in the standard-form,  $(\langle G|_a \otimes \hat{\mathcal{I}}_s)\hat{Q}(|G\rangle_a \otimes \hat{\mathcal{I}}_s) = f[\hat{H}]$  for some target operator  $f[\cdot]$ . Though this is accomplished in prior art using the linear-combination-of-unitaries algorithm [25], that approach requires a case-by-case detailed analysis of  $f$  to obtain the  $L_1$  norm of its coefficients in a Taylor expansion, and has a success probability that decays with the inverse square of this norm.

Our quantum signal processor computes  $f[\hat{H}]$  with no such restrictions and with an optimal query complexity that exactly matches polynomial lower bounds for a large class of functions. We call this ‘quantum signal processing’, which generalizes our previous results [7] for  $d$ -sparse oracles to the standard-form and a larger class of functions. Thus generic improvements to all applications in Table 2 can be expected in query complexity, ancilla overhead, and scope of possible signal inputs. In particular, Theorem 1 follows directly from the query complexity of the choice  $f(\lambda) = e^{-i\lambda t}$ , which corresponds to applying  $-t \sin(\cdot)$  on eigenphases of the iterate.

In Section 2, we provide a detailed overview of the quantum signal processor, which uses the standard-form encoding of a matrix  $\hat{H}$  in Definition 1 as an input. Our claim to the generality of this encoding is justified in Section 3, where we map a variety of other common input models describing matrices to the standard-form. The next key result is qubitization, which converts a standard-form encoding to a Grover-like iterate, is proven in Section 4. By creatively applying this iterate, we show in Section 5 how a large class of polynomials of  $\hat{H}$  may be efficiently computed. Our final result on Hamiltonian simulation by qubitization is proven in Section 6 by a simple choice for this polynomial. Though we focus on qubitization of Hermitian matrices in the body of this paper, we also describe the extension to normal matrices in Appendix A.

## 2 Overview of the Quantum Signal Processor

Since coherent quantum computation is restricted to unitary operations, one commonly finds a situation such as in [Table 2](#), where post-selection is required to accomplish some desired quantum state transformation. Consider some arbitrary input system quantum state  $|\psi\rangle_s \in \mathcal{H}_s$  and suppose that it is transformed by some desired *signal operator*  $\hat{H}$  like  $|\psi\rangle_s \rightarrow \hat{H}|\psi\rangle_s$ . In order to realize this operation, which is non-unitary in general, it is necessary to embed  $\hat{H}$  in a larger Hilbert space.

This is embedding accomplished through some given unitary *signal oracle*  $\hat{U} : \mathcal{H}_a \otimes \mathcal{H}_s \rightarrow \mathcal{H}_a \otimes \mathcal{H}_s$  that acts jointly on  $\mathcal{H}_s$  and another Hilbert space  $\mathcal{H}_a$ . The signal oracle encodes  $\hat{H} = (\langle G|_a \otimes \hat{\mathcal{I}}_s) \hat{U} (|G\rangle_a \otimes \hat{\mathcal{I}}_s)$  in a subspace flagged by an ancilla *signal state*  $|G\rangle_a \in \mathcal{H}_a$ . This is the standard-form of [Definition 1](#), which applies  $\hat{H}$  as follows.

$$\hat{U}|G\rangle_a|\psi\rangle_s = |G\rangle_a\hat{H}|\psi\rangle_s + \sqrt{1 - \|\hat{H}|\psi\rangle_s\|^2}|G_\psi^\perp\rangle_{as}, \quad \hat{U} = \begin{pmatrix} \hat{H} & \\ & \cdot \end{pmatrix}, \quad (\langle G|_a \otimes \hat{\mathcal{I}}_s)|G_\psi^\perp\rangle_{as} = 0. \quad (2)$$

The signal state defines the measurement basis of the appended register  $\mathcal{H}_a$ , which naturally divides  $\hat{U}$  into two subspaces. First,  $\mathcal{H}_G = |G\rangle \otimes \mathcal{H}_s$  where the measurement succeeds with probability  $\|\hat{H}|\psi\rangle_s\|^2$  and  $\hat{U}|G\rangle_a|\psi\rangle_s$  is projected onto  $\frac{|G\rangle_a\hat{H}|\psi\rangle_s}{\|\hat{H}|\psi\rangle_s\|}$ . Second, the orthogonal complement  $\mathcal{H}_{G^\perp}$  where the measurement fails. As probabilities are bounded by 1, the signal operator  $\hat{H}$  must have spectral norm  $\|\hat{H}\| \leq 1$ . Whenever the context is clear, we drop the ancilla and system subscripts, and use  $|G\rangle_a \otimes \hat{\mathcal{I}}_s$  and  $|G\rangle$  interchangeably. We represent  $\hat{U}$  such that the top-left block is precisely  $\hat{H}$  and acts on an input state  $|G_\psi\rangle \equiv |G\rangle|\psi\rangle \in \mathcal{H}_G$ , whereas the undefined parts of  $\hat{U}$  transform  $|G_\psi\rangle$  into some orthogonal state  $|G_\psi^\perp\rangle \in \mathcal{H}_{G^\perp}$  of lesser interest.

In the following, we consider the case where  $\hat{H}$  is a Hermitian matrix. The case of normal  $\hat{H}$  is discussed in [Appendix A](#). Thus the action of  $\hat{U}$  on  $|G_\psi\rangle$  in [Eq. \(2\)](#) can be written more clearly in the eigenbasis of  $\hat{H}|\lambda\rangle = \lambda|\lambda\rangle$ . For each eigenstate of  $\hat{H}$ ,

$$\hat{U}|G\rangle|\lambda\rangle = \hat{U}|G_\lambda\rangle = \lambda|G_\lambda\rangle + \sqrt{1 - |\lambda|^2}|G_\lambda^\perp\rangle, \quad (3)$$

and we find it convenient to define the subspace  $\mathcal{H}_\lambda = \text{span}\{|G_\lambda\rangle, \hat{U}|G_\lambda\rangle\}$ . Note the trivial case  $\lambda = 1$  where  $\mathcal{H}_\lambda$  is one-dimensional, which we will ignore. We also emphasize that the action of  $\hat{U}$  on states orthogonal to  $|G_\lambda\rangle$  are generally not easily controlled by the user, and thus left undefined.

Given any scalar function  $f$ , our goal is to find an optimal quantum circuit that transforms any standard-form encoding of  $\hat{H}$  into a standard-form encoding of  $f[\hat{H}] \equiv \sum_\lambda f(\lambda)|\lambda\rangle\langle\lambda|$ . We accomplish this with the quantum signal processor.

**Definition 2** (Quantum Signal Processor). *A quantum signal processor solves the following.*

- Inputs:*
- A Hermitian matrix  $\hat{H}$  with bounded spectral norm  $\|\hat{H}\| \leq 1$ .
  - A function  $f : [-1, 1] \rightarrow \mathbb{D}$ , where  $\mathbb{D}$  is the complex unit disc, i.e.  $\forall x \in [-1, 1], |f(x)| \leq 1$ .
- Resources:*
- An encoding of  $\hat{H} = \langle G|\hat{U}|G\rangle$  by the oracles of [Definition 1](#).
  - A constant number of additional qubits.
  - Arbitrary single and two-qubit gates.
- Outputs:*
- A standard-form encoding of  $f[\hat{H}]$ , i.e. a unitary  $\hat{Q}$  where  $f[\hat{H}] = \langle 0|\hat{Q}|0\rangle$ .

In the simplest case, one might wish to apply  $\hat{H}$  multiple times to generate higher moments. For instance,  $\hat{H}^2$  would allow a direct estimate of variance. Unfortunately, the subspace  $\mathcal{H}_\lambda$  for each eigenstate  $|\lambda\rangle$  is not invariant under  $\hat{U}$  in general. As a result, repeated applications in this basis do not produce higher moments of  $\hat{H}$  due to leakage out of  $\mathcal{H}_\lambda$ . The structure of this leakage depends on the undefined components of  $\hat{U}$ , must be analyzed on a case-by-case basis, and thus is of limited utility.

Order can be restored to this undefined behavior by stemming the leakage. The simplest possibility that preserves the signal operator of Eq. (2) replaces  $\hat{U}$  with a unitary, the *iterate*  $\hat{W}$ , that also encodes  $\hat{H} = \langle G | \hat{W} | G \rangle$ , but for each eigenstate  $|\lambda\rangle$ , performs a rotation in  $SU(2)$  on disjoint two-dimensional subspaces  $\mathcal{H}_\lambda = \text{span}\{|G_\lambda\rangle, \hat{W}|G_\lambda\rangle\} = \text{span}\{|G_\lambda\rangle, |G_\lambda^\perp\rangle\}$ . This defines the state  $|G_\lambda^\perp\rangle$  through Gram-Schmidt orthogonalization, and a set of Pauli operators  $\hat{X}_\lambda, \hat{Y}_\lambda, \hat{Z}_\lambda$  for each subspace.

$$|G_\lambda^\perp\rangle = \frac{(\hat{W} - \lambda)|G_\lambda\rangle}{\sqrt{1 - |\lambda|^2}}, \quad \hat{X}_\lambda|G_\lambda\rangle = |G_\lambda^\perp\rangle, \quad \hat{Y}_\lambda|G_\lambda\rangle = i|G_\lambda^\perp\rangle, \quad \hat{Z}_\lambda|G_\lambda\rangle = |G_\lambda\rangle. \quad (4)$$

In this basis, the iterate for each eigenvalue of  $\hat{H}$  is defined to be exactly

$$\hat{W} = \left( \begin{array}{cc} \lambda & -\sqrt{1 - |\lambda|^2} \\ \sqrt{1 - |\lambda|^2} & \lambda \end{array} \right)_\lambda = \begin{array}{cc} \lambda|G_\lambda\rangle\langle G_\lambda| & -\sqrt{1 - |\lambda|^2}|G_\lambda\rangle\langle G_\lambda^\perp| \\ +\sqrt{1 - |\lambda|^2}|G_\lambda^\perp\rangle\langle G_\lambda| & +\lambda|G_\lambda^\perp\rangle\langle G_\lambda^\perp| \end{array}, \quad (5)$$

On a general input  $|G\rangle = \sum_\lambda a_\lambda |\lambda\rangle \in \mathcal{H}_G$ , the iterate is represented by

$$\hat{W} = \bigoplus_\lambda \left( \begin{array}{cc} \lambda & -\sqrt{1 - |\lambda|^2} \\ \sqrt{1 - |\lambda|^2} & \lambda \end{array} \right)_\lambda = \bigoplus_\lambda e^{-i\hat{Y}_\lambda \theta_\lambda}, \quad (6)$$

where  $\theta_\lambda$  is defined through the equality

$$\forall \lambda \in [-1, 1], \quad \lambda - i\sqrt{1 - |\lambda|^2} = \cos(\theta_\lambda) - i\sin(\theta_\lambda) \Rightarrow \theta_\lambda = \cos^{-1}(\lambda). \quad (7)$$

In the following, the iterate will always be applied to states spanned by the subspace  $\bigoplus_\lambda \mathcal{H}_\lambda$ . Thus its action on states outside this subspace need not be defined. The usefulness of this construct is evident. Due to its invariant subspace, multiple applications of the iterate result in highly structured behavior. However, implementing  $\hat{W}$  appears difficult in general. On one hand, the function  $\sqrt{1 - |\lambda|^2}$  must be computed for each eigenstate. In principle, this can be approximated using phase estimation with considerable overhead [32]. On the other hand, it is not at all clear whether a unitary with the form of  $\hat{W}$  can always be engineered from the standard-form encoding of  $\hat{H}$ . ‘Qubitization’ is our solution to constructing this iterate with minimal overhead.

**Theorem 2** (Qubitization). *Given a Hermitian matrix  $\hat{H} = \langle G | \hat{U} | G \rangle_a$  encoded in standard-form as described in Definition 1, the iterate  $\hat{W}$  of Eq. (6) can be constructed using at most one query each to  $\hat{G}$ , controlled- $\hat{U}$ , their inverses, at most one additional qubit, and  $\mathcal{O}(\log(\dim(\mathcal{H}_a)))$  quantum gates.*

Using the oracles  $\hat{G}$ ,  $\hat{U}$ , and arbitrary unitary operations on *only* the ancilla register, we also provide necessary and sufficient conditions in Lemma 8, for when  $\hat{W}$  can be implemented exactly using only one query to  $\hat{U}$ . As these conditions are somewhat restrictive, we then prove in Lemma 10 that qubitization is *unconditionally* possible by instead using the *controlled- $\hat{U}$*  oracle in a quantum circuit that generates the same  $\hat{H}$  and satisfies these conditions. We describe a similar construction for normal operators in Appendix. A.

Observe that  $\hat{W}^N$  efficiently produces Chebyshev polynomials  $T_N[\hat{H}]$  [9]. We call any function  $[\cdot]$  of the signal  $\hat{H}$  *target operators* when they occur in the top-left block and are thus automatically in standard-form. The fact that Chebyshev polynomials are the best polynomial basis for  $L_\infty$  function approximation on a finite interval [33] suggests that the any target operator  $f[\hat{H}] = A[\hat{H}] + iB[\hat{H}]$  could be approximated with a judicious choice of controls on the ancilla register. We present two implementations of the quantum signal processor described in Definition 2. The first does not require any additional qubits beyond that for qubitization, and realizes a broad class of target operators  $f$ .

**Theorem 3** (Ancilla-Free Quantum Signal Processing). *A quantum signal processor can be implemented with the following properties.*

- Inputs*
- The iterate  $\hat{W}$  obtained from the qubitization procedure of [Theorem 2](#) on a Hermitian matrix  $\hat{H} = \langle G|\hat{U}|G\rangle$  encoded by the oracles of [Definition 1](#).
  - Real polynomials  $A(\lambda), B(\lambda)$  of degree  $Q$  and equal parity satisfying all following.
    - $A(1) = 1$ ;
    - $\forall \lambda \in [-1, 1], A^2(\lambda) + B^2(\lambda) \leq 1$ ;
    - $\forall \lambda \geq 1, A^2(\lambda) + B^2(\lambda) \geq 1$ ;
    - $\forall Q$  even,  $\lambda \geq 0, A^2(i\lambda) + B^2(i\lambda) \geq 1$ .
- Output:*
- A standard-form encoding of  $A[\hat{H}] + iB[\hat{H}]$ .
- Cost*
- $Q$  queries to  $\hat{W}$ .
  - Zero additional qubits beyond that for  $\hat{W}$ .
  - $\mathcal{O}(Q \log(\dim(\mathcal{H}_a)))$  arbitrary single and two-qubit gates.

Note that the two polynomials in our solution of ancilla-free quantum signal processing are of the same parity. At this point, we assume that all  $\hat{U}$  have been qubitized and so use the iterate  $\hat{W}$  as our basic building block. A different basis set of functions with fewer restrictions on parity can be obtained by embedding  $\hat{W}$  into yet another  $SU(2)$  invariant subspace by adding an ancilla qubit. This leads to the second implementation of the quantum signal processor of [Definition 2](#).

**Theorem 4** (Single-Ancilla Quantum Signal Processing). *A quantum signal processor can be implemented with the following properties.*

- Inputs:*
- The iterate  $\hat{W}$  obtained from the qubitization procedure of [Theorem 2](#) on a Hermitian matrix  $\hat{H} = \langle G|\hat{U}|G\rangle$  encoded by the oracles of [Definition 1](#).
  - Real polynomials  $A(\lambda), C(\lambda)$  of degree  $Q/2$  and opposite parity satisfying all following.
    - $A(0) = 1$ ;
    - $\forall \lambda \in [-1, 1], A^2(\lambda) + C^2(\lambda) \leq 1$ ;
- Output:*
- A standard-form encoding of  $A[\hat{H}] + iC[\hat{H}]$ .
- Cost:*
- $Q$  queries to controlled- $\hat{W}$ .
  - One additional qubit beyond that for  $\hat{W}$ .
  - $\mathcal{O}(Q \log(\dim(\mathcal{H}_a)))$  arbitrary single and two-qubit gates.

These powerful tools for target operator processing, made possible by qubitization, are agnostic to the underlying oracles that describe the signal operator  $\hat{H}$ . In many instances, converting this description to the standard-form of [Definition 1](#) is straightforward and is indeed how our Hamiltonian simulation results for the varied input models of [Table 1](#) are proven.

### 3 Explicit Encodings of $\hat{H} = (\langle G| \otimes \hat{\mathcal{I}})\hat{U}(|G\rangle \otimes \hat{\mathcal{I}})$

We now justify our motivation for encoding matrices  $\hat{H}$  in the standard-form format of  $\hat{H} = \langle G|\hat{U}|G\rangle$  in [Definition 1](#). A number of common techniques for encoding matrix problems on quantum computers map naturally to the standard-form with minimal overhead. Thus taking the standard-form as our starting point is without any loss of generality. This is demonstrated by the following three explicit implementation of the oracles  $\hat{U}$  and  $\hat{G}|0\rangle = |G\rangle$  for Hamiltonians represented as a linear combination of unitaries,  $d$ -sparse Hamiltonians, and a new input model where Hamiltonians are represented by a purified density matrix.

#### 3.1 Linear Combination of Unitaries

One option for implementing the standard-form encoding is provided by the Linear-Combination-of-Unitaries algorithm (LCU) [3, 25], which underlies the BCKKS simulation algorithm. LCU is based

on the fact that any complex  $\hat{H}$  is a linear combination of some  $d$  unitary operators:

$$\hat{H} = \sum_{j=1}^d \alpha_j \hat{U}_j, \quad \|\hat{H}\| \leq \|\vec{\alpha}\|_1 = \sum_{j=1}^d |\alpha_j|, \quad (8)$$

where the upper bound on the spectral norm is  $\|\vec{\alpha}\|_1$ . Note that this bound depends on the choice of decomposition, but is tight for the some choice. Without loss of generality, all  $\alpha_j \geq 0$  by absorbing complex phases into  $\hat{U}_j$ . The algorithm assumes that the  $\alpha_j$  are provided as a list of  $d$  numbers, and each  $\hat{U}_j$  is provided as a quantum circuit composed of constant number  $\mathcal{O}(C)$  of primitive gates. With these inputs, the oracles

$$\hat{G} = \sum_{j=1}^d \sqrt{\frac{\alpha_j}{\|\vec{\alpha}\|_1}} |j\rangle\langle 0|_a + \dots, \quad \hat{U} = \sum_{j=1}^d |j\rangle\langle j|_a \otimes \hat{U}_j, \quad \langle G|\hat{U}|G\rangle = \frac{\hat{H}}{\|\vec{\alpha}\|_1} \quad (9)$$

can be constructed, where the ancilla state creation operator  $\hat{G}|0\rangle = |G\rangle$  is implemented with  $\mathcal{O}(d)$  primitive gates, and the selector  $\hat{U}$  is implemented with  $\mathcal{O}(dC)$  primitive gate. By direct expansion of  $\hat{U}\hat{G}|0\rangle_a|\psi\rangle_s$ , this leads exactly to Eq. (2). This proves the following.

**Lemma 5** (Standard-Form Encoding by a Linear Combination of Unitaries). *Let  $\hat{G}$  prepare the state  $|G\rangle_a = \sum_{j=1}^d \sqrt{\alpha_j/\|\vec{\alpha}\|_1} |j\rangle_a$  where  $\alpha_j \geq 0$ . Let  $\hat{U} = \sum_{j=1}^d |j\rangle\langle j|_a \otimes \hat{U}_j$ . These oracles encode the matrix  $\langle G|\hat{U}|G\rangle = \frac{1}{\|\vec{\alpha}\|_1} \sum_{j=1}^d \alpha_j \hat{U}_j$ .*

*Proof.* Consider the computation

$$\begin{aligned} (\langle G|_a \otimes \hat{\mathcal{I}}_s) \hat{U} (|G\rangle_a \otimes \hat{\mathcal{I}}_s) &= (\langle G|_a \otimes \hat{\mathcal{I}}_s) \left( \sum_{j=1}^d \sqrt{\frac{\alpha_j}{\|\vec{\alpha}\|_1}} |j\rangle_a \otimes \hat{U}_j \right) = \sum_{j,k=1}^d \frac{\sqrt{\alpha_j \alpha_k}}{\|\vec{\alpha}\|_1} \langle k|j\rangle \otimes \hat{U}_j \\ &= \frac{1}{\|\vec{\alpha}\|_1} \sum_{j=1}^d \alpha_j \hat{U}_j. \end{aligned} \quad (10)$$

□

Of course, the optimal decomposition that costs the fewest number of ancilla qubits and primitive gates may be difficult to find, and may not even fit naturally in this model, but LCU shows that implementing an encoding for any  $\hat{H}$  is possible in principle.

### 3.2 $d$ -Sparse Hamiltonians

The model of  $d$ -sparse Hamiltonians is another paradigm for specifying matrices to quantum computers, and is particularly common in the design of quantum algorithms by quantum walks [2]. Such Hamiltonians have at most  $d$  non-zero entries in any row, and information on their positions and matrix values  $\hat{H}_{jk}$  are accessed through the following two standard unitary oracles [4].

$$\hat{O}_H |j\rangle|k\rangle|z\rangle = |j\rangle|k\rangle|z \oplus \hat{H}_{jk}\rangle, \quad \hat{O}_F |j\rangle|l\rangle = |j\rangle|f(j,l)\rangle. \quad (11)$$

Observe that  $\hat{O}_H$  accepts a row  $j$  and column  $k$  index and returns  $\hat{H}_{jk}$  in some binary format. The other oracle  $\hat{O}_F$  accepts a row  $j$  index and a number  $l \in [d]$  to compute in-place the column index  $f(j,l)$  of the  $l^{\text{th}}$  non-zero element in row  $j$ . Given this input, we now encode  $\hat{H}$  in standard-form.

**Lemma 6** (Standard-Form Encoding of a  $d$ -Sparse Hamiltonian). *Let the oracles of Eq. (11) specify a  $d$ -sparse Hamiltonian  $\hat{H}$  with max-norm  $\|\hat{H}\|_{\max}$ . Then the oracles encoding  $\langle G|\hat{U}|G\rangle = \frac{\hat{H}}{d\|\hat{H}\|_{\max}}$  can be implemented using  $\mathcal{O}(1)$  queries.*



*Proof.* Let  $|G\rangle = |0\rangle_{a_1}|0\rangle_{a_2}|0\rangle_{a_3} \equiv |0\rangle_a$  be a computational basis state, and let  $F_j = \{f(j, l)\}_{l \in [d]}$  be the set of column indices to all non-zero elements in row  $j$ . Ref. [4] shows how each of the isometries

$$\begin{aligned}\hat{T}_1 &= \sum_j |\psi_j\rangle\langle 0|_a \langle j|_s, |\psi_j\rangle = \sum_{p \in F_j} \frac{|p\rangle_{a_3}}{\sqrt{d}} \left( \sqrt{\frac{H_{pj}}{\|\hat{H}\|_{\max}}} |0\rangle_{a_1} + \sqrt{1 - \frac{|H_{pj}|}{\|\hat{H}\|_{\max}}} |1\rangle_{a_1} \right) |0\rangle_{a_2} |j\rangle_s, \\ \hat{T}_2 &= \sum_k |\chi_k\rangle\langle 0|_a \langle k|_s, \langle \chi_k| = \sum_{p \in F_k} \frac{\langle p|_s}{\sqrt{d}} \left( \sqrt{\frac{H_{kp}}{\|\hat{H}\|_{\max}}} \langle 0|_{a_2} + \sqrt{1 - \frac{|H_{kp}|}{\|\hat{H}\|_{\max}}} \langle 1|_{a_2} \right) \langle 0|_{a_1} \langle k|_{a_3},\end{aligned}\quad (12)$$

can be implemented using using 2 queries to  $\hat{O}_H$  and 1 query to  $\hat{O}_F$ . By construction, the overlap of these states is  $\langle \chi_k | \psi_j \rangle = \frac{H_{kj}}{d\|\hat{H}\|_{\max}}$ . Now choose  $\hat{U} = \hat{T}_2^\dagger \hat{T}_1$ . By a direct computation,

$$\langle \langle G|_a \otimes \hat{I}_s \rangle \hat{U} (|G\rangle_a \otimes \hat{I}_s) = \sum_{k,j} |k\rangle_s \langle \chi_k | \psi_j \rangle \langle j|_s = \sum_{k,j} \frac{H_{kj}}{d\|\hat{H}\|_{\max}} |k\rangle \langle j|_s = \frac{\hat{H}}{d\|\hat{H}\|_{\max}}. \quad (13)$$

□

### 3.3 Purified Density Matrix

The simplicity of the standard-form encoding allows us to swiftly devise new input models for Hamiltonians. Here, we consider the case where a Hamiltonian  $\hat{H} = \text{Tr} [|G\rangle\langle G|]_{a_1}$  is a density matrix  $\hat{\rho}$  obtained by tracing out the ancilla register of a pure state prepared by some oracle  $\hat{G}$ . That is,

$$\hat{G}|0\rangle_a = |G\rangle_a = \sum_j \sqrt{\alpha_j} |j\rangle_{a_1} |\chi_j\rangle_{a_2}, \quad \hat{\rho} = \text{Tr} [|G\rangle\langle G|]_{a_1} = \sum_j \alpha_j |\chi_j\rangle\langle \chi_j|. \quad (14)$$

**Lemma 7** (Standard-Form Encoding of a Purified Density Matrix). *Let the oracle of Eq. (14) prepare a state  $\hat{G}|0\rangle_a = |G\rangle_a = \sum_j \sqrt{\alpha_j} |j\rangle_{a_1} |\chi_j\rangle_{a_2}$  that is a purification of the density matrix  $\hat{\rho} = \sum_j \alpha_j |\chi_j\rangle\langle \chi_j| = \text{Tr} [|G\rangle\langle G|]_{a_1}$ . Let  $\hat{U}$  be a unitary that swaps the register  $a_2$  with the system register  $s$ . These oracles encode the matrix  $\langle G | \hat{U} | G \rangle = \hat{\rho}$ .*

*Proof.* Let  $\{|\lambda\rangle\}$  be a complete basis on the system. By a direct computation,

$$\langle G | \hat{U} | G \rangle_a \sum_\lambda |\lambda\rangle \langle \lambda|_s = \sum_\lambda \sum_j \langle G | \sqrt{\alpha_j} |j\rangle_{a_1} |\lambda\rangle_{a_2} |\chi_j\rangle \langle \lambda|_s = \sum_\lambda \sum_j \alpha_j |\chi_j\rangle_s \langle \chi_j | \lambda \rangle \langle \lambda|_s = \hat{\rho}. \quad (15)$$

□

## 4 Qubitization in a Quantum Signal Processor

This section describes qubitization in detail: the process for creating the iterate  $\hat{W}$  given  $\hat{H}$  encoded in standard-form, and an essential component in a systematic procedure for implementing operator transformations of  $\hat{H}$ . In Lemma 8, we provide necessary and sufficient conditions on when  $\hat{W}$  can be constructed from the oracles  $\hat{G}, \hat{U}$ . Then in Lemma 10, we show that any  $\hat{G}, \hat{U}$  not satisfying these conditions can be efficiently transformed into a  $\hat{G}', \hat{U}'$  that do, and encode the same signal operator  $\hat{H} = \langle G' | \hat{U}' | G' \rangle$ . Together, these Lemmas in the remainder of this section complete the proof of Theorem 2 for the case of Hermitian  $\hat{H}$ .

For now, we assume that  $|G\rangle$  is known. This soon proves to be unnecessary and only oracle access to  $\hat{G}$  is required. Thus we must find a unitary  $\hat{S}'$ , acting only on the ancilla register such that the iterate  $\hat{W} = \hat{S}' \hat{U}$  of Eq. (6) is obtained. For the case of Hermitian  $\hat{H}|\lambda\rangle = \lambda|\lambda\rangle$ , we now determine

necessary and sufficient conditions on what  $\hat{S}'$  must be. As  $\hat{S}'$  is otherwise arbitrary, we use without loss of generality the ansatz of  $\hat{S}'$  being a product of a reflection about  $|G\rangle$  and another arbitrary unitary  $\hat{S}$  on the ancilla:

$$\hat{W} = ((2|G\rangle\langle G| - \hat{\mathcal{I}})_a \otimes \hat{\mathcal{I}}_s) \hat{S} \hat{U}, \quad |G_\lambda\rangle = |G\rangle|\lambda\rangle \Rightarrow |G_\lambda^\perp\rangle = \frac{\lambda|G_\lambda\rangle - \hat{S}\hat{U}|G_\lambda\rangle}{\sqrt{1-\lambda^2}}. \quad (16)$$

**Lemma 8** (Conditions on Qubitization). *For all signal oracles  $\hat{U}$  that implement the Hermitian signal operator  $\hat{H}$ , the unitary  $\hat{S}$  in Eq. (16) creates a unitary iterate  $\hat{W}$  with the same signal operator in the same basis, but in an  $SU(2)$  invariant subspace containing  $|G\rangle$  if and only if*

$$\langle G|_a \hat{S} \hat{U} |G\rangle_a = \hat{H} \text{ and } \langle G|_a \hat{S} \hat{U} \hat{S} \hat{U} |G\rangle_a = \hat{\mathcal{I}}. \quad (17)$$

*Proof.* In the forward direction, we assume Eq. (6), then compute and compare with Eq. (16):  $\lambda = \langle G_\lambda | \hat{W} | G_\lambda \rangle = \langle G_\lambda | \hat{S} \hat{U} | G_\lambda \rangle$ . By using this result repeatedly together with the fact that  $\hat{S} \hat{U}$  is unitary, Gram-Schmidt orthonormalization of  $\hat{W} | G_\lambda \rangle$  furnishes the state  $|G_\lambda^\perp\rangle = \frac{\lambda|G_\lambda\rangle - \hat{S}\hat{U}|G_\lambda\rangle}{\sqrt{1-\lambda^2}}$  which is orthogonal to  $|G_\lambda\rangle$ . By similarly computing and comparing  $-\sqrt{1-\lambda^2} = \langle G_\lambda | \hat{W} | G_\lambda^\perp \rangle = \frac{\lambda^2 - \langle G_\lambda | (\hat{S}\hat{U})^2 | G_\lambda \rangle}{\sqrt{1-\lambda^2}}$ , we obtain  $\langle G_\lambda | (\hat{S}\hat{U})^2 | G_\lambda \rangle = 1$ . As these must be true for all eigenvectors  $|\lambda\rangle$ , the conditions in Eq. (17) are necessary.

That these are also sufficient follows from assuming Eq. (17) and attempting to recover the components of Eq. (6) using the definitions of Eq. (16). By applying  $\langle G|_a \hat{S} \hat{U} |G\rangle_a = \hat{H}$ , we compute  $\hat{W} |G_\lambda\rangle = 2|G\rangle_a \langle G|_a \hat{S} \hat{U} |G_\lambda\rangle - \hat{S} \hat{U} |G_\lambda\rangle = 2\lambda|G_\lambda\rangle - \hat{S} \hat{U} |G_\lambda\rangle$ . In the basis of  $|G_\lambda\rangle$  and  $|G_\lambda^\perp\rangle$ ,  $\langle G_\lambda | \hat{W} | G_\lambda \rangle = 2\lambda - \langle G_\lambda | \hat{S} \hat{U} | G_\lambda \rangle = \lambda$  and  $\langle G_\lambda^\perp | \hat{W} | G_\lambda \rangle = \frac{\langle G_\lambda | \lambda - \langle G_\lambda | (\hat{S}\hat{U})^\dagger (2\lambda|G_\lambda\rangle - \hat{S}\hat{U}|G_\lambda\rangle)}{\sqrt{1-\lambda^2}} = \frac{2\lambda^2 - 2\lambda^2 - \lambda^2 + 1}{\sqrt{1-\lambda^2}} = \sqrt{1-\lambda^2}$ . A similar calculation for the remaining components requires  $\langle G|_a \hat{S} \hat{U} \hat{S} \hat{U} |G\rangle_a = \hat{\mathcal{I}}$  and reveals that  $\langle G_\lambda^\perp | \hat{W} | G_\lambda^\perp \rangle = \lambda$  and  $\langle G_\lambda | \hat{W} | G_\lambda^\perp \rangle = -\sqrt{1-\lambda^2}$ . As this must be true for all  $\lambda$ , we may indeed represent  $\hat{W} = \bigoplus_\lambda \begin{pmatrix} \lambda & -\sqrt{1-\lambda^2} \\ \sqrt{1-\lambda^2} & \lambda \end{pmatrix}_\lambda$ .  $\square$

In hindsight, these results are manifest. After all,  $\langle G | \hat{S} \hat{U} \hat{S} \hat{U} | G \rangle = \hat{\mathcal{I}}$  implies that  $\hat{S} \hat{U}$  is a reflection when controlled by input state  $|G\rangle$ , and it is well-known that a Grover iterate [34, 35] is the product of two reflection about start and target subspaces. Nevertheless, the sufficiency of these conditions highlights that this is the simplest method to extract controllable and predictable behavior out of  $\hat{U}$ . In particular, these conditions are automatically satisfied in the trivial case with  $\hat{S} = \hat{\mathcal{I}}_{as}$  when  $\hat{U}$  only has eigenvalues  $\pm 1$ , such as when it is a controlled-Pauli operator.

**Corollary 9** (Qubitization of Reflections). *For all signal oracles  $\hat{U}$  that satisfy  $\hat{U}^2 = \hat{\mathcal{I}}_{as}$  and implement the Hermitian signal operator  $\langle G|_a \hat{U} |G\rangle_a = \hat{H}$ , the unitary iterate  $\hat{W} = ((2|G\rangle\langle G| - \hat{\mathcal{I}})_a \otimes \hat{\mathcal{I}}_s) \hat{U}$  implements the same signal operator in the same basis, but in an  $SU(2)$  invariant subspace containing  $|G\rangle$ .*

*Proof.* Set  $\hat{S} = \hat{\mathcal{I}}_{as}$  in Lemma 8, and verify that Eq. (17) is satisfied.  $\square$

Unfortunately, a solution to Eq. (17) may not exist for more general  $\hat{U}$ . Lemma 8, amounts to choosing  $\hat{S}$  such that  $\hat{S} \hat{U} \hat{S}$  is the inverse  $\hat{U}^\dagger$  whilst preserving the signal operator  $\langle G | \hat{S} \hat{U} | G \rangle = \hat{H}$ . Given that  $\hat{S}$  only acts on the ancilla register, it is hard to see how this is always possible. Even if so,  $\hat{S}$  may be difficult to implement as it is an arbitrary unitary acting on a potentially large ancilla register. The solution is to construct a different quantum circuit  $\hat{U}'$  that contains  $\hat{U}$  but still implements the same signal operator, and crucially always has a extremely simple solution  $\hat{S}$ . We now show how this can be done in all cases using only 1 query to controlled- $\hat{U}$  and controlled- $\hat{U}^\dagger$ .

**Lemma 10** (Existence of Qubitization). *For all signal unitaries  $\hat{U}$  that implement the Hermitian signal operator  $\langle G | \hat{U} | G \rangle = \hat{H}$ , there exists a quantum circuit  $\hat{U}'$ , using one more qubit, that queries*

controlled- $\hat{U}$  and controlled- $\hat{U}^\dagger$  once to implements the same signal operator. Moreover,  $\hat{U}'$  satisfies the conditions Eq. (17)

*Proof.* We prove this by an explicit construction. Let the controlled- $\hat{U}$  operators be  $\hat{V}_1 = |0\rangle\langle 0| \otimes \hat{I} + |1\rangle\langle 1| \otimes \hat{U}^\dagger$ ,  $\hat{V}_2 = |0\rangle\langle 0| \otimes \hat{U} + |1\rangle\langle 1| \otimes \hat{I}$ . Thus the extra qubit states  $|0\rangle, |1\rangle$ , are flags that selects either  $\hat{U}_j$  or  $\hat{U}_j^\dagger$ . By multiplying,  $\hat{U}' = \hat{V}_1\hat{V}_2 = |0\rangle\langle 0| \otimes \hat{U} + |1\rangle\langle 1| \otimes \hat{U}^\dagger$ . Now consider the ancilla state  $|G'\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|G\rangle$ , and choose  $\hat{S} = (|0\rangle\langle 1| + |1\rangle\langle 0|) \otimes \hat{I}_{as}$ . It is easy to verify that the conditions of Eq. (17) is satisfied.

$$\langle G'|\hat{S}\hat{U}'|G'\rangle = \langle G'|\hat{U}'|G'\rangle = \frac{1}{2}(\hat{H} + \hat{H}^\dagger) = \hat{H}, \quad \langle G'|\hat{S}\hat{U}'\hat{S}\hat{U}'|G'\rangle = \langle G'|\hat{U}'^\dagger\hat{U}'|G'\rangle = \hat{I}, \quad (18)$$

where we have used the fact that  $\hat{S}|G'\rangle = |G'\rangle$  is an eigenstate, and that  $\hat{S}$  swaps the  $|0\rangle, |1\rangle$  ancilla states in  $\hat{U}'$ , thus transforming it into its inverse.  $\square$

Even if we are given  $\hat{U}$  for which there is no solution to Eq. (17), we can always apply Lemma 10 to construct a  $\hat{U}'$  that does with minimal overhead. Furthermore our proof uses no information about the detailed structure of  $|G'\rangle$ . Thus without loss of generality, we can assume that any  $\hat{G}, \hat{U}$  have already been qubitized.

## 5 Operator Function Design on a Quantum Signal Processor

The purpose of the quantum signal processor is to transform the signal  $\hat{H}$  into any desired target operator  $f[\hat{H}]$ . We present a systematic framework that furnishes the optimal complexity and a concrete procedure for almost any  $f$  and show how an exact connection is made between query complexity and the theory of best function approximations with polynomials [33, 36].

Qubitization in Section 4 is the essential first step that makes this endeavor plausible, as evidenced by the highly structured behavior of the iterate  $\hat{W}$  in Eq. (6), where for Hermitian  $\hat{H}$ , multiple applications elegantly generate Chebyshev polynomials  $T_L[\hat{H}]$  [9]. To go further, additional control parameters on  $\hat{W}$  are necessary, and in the following, we only consider Hermitian  $\hat{H}$ . Thus we introduce the *phased iterate*  $\hat{W}_\phi$  with the same invariant subspace as  $\hat{W}$ , and parameterized by  $\phi \in \mathbb{R}$ .

$$\hat{W}_\phi = \bigoplus_\lambda \begin{pmatrix} \lambda & -ie^{-i\phi}\sqrt{1-|\lambda|^2} \\ -ie^{i\phi}\sqrt{1-|\lambda|^2} & \lambda \end{pmatrix}_\lambda = \bigoplus_\lambda e^{-i\phi\lambda\theta_\lambda}, \quad (19)$$

where  $\hat{\phi}^\lambda = \cos(\phi)\hat{X}_\lambda + \sin(\phi)\hat{Y}_\lambda$ , and the eigenphase  $\theta_\lambda$  is defined similar to Eq. (7).

**Lemma 11.** *The phased iterate  $\hat{W}_\phi$  in Eq. (19) is equal to  $\hat{W}_\phi = \hat{Z}_{\phi+\pi/2}\hat{W}\hat{Z}_{-\phi-\pi/2}$ , where  $\hat{Z}_\phi = ((1+e^{-i\phi})|G\rangle\langle G| - \hat{I})$  is a partial reflection about  $|G\rangle$  by angle  $\phi \in \mathbb{R}$  and implements a relative phase between the  $|G_\lambda\rangle$  and  $|G_\lambda^\perp\rangle$  subspaces. In block form,*

$$\hat{Z}_\phi = \bigoplus_\lambda \begin{pmatrix} e^{-i\phi} & 0 \\ 0 & -1 \end{pmatrix}_\lambda. \quad (20)$$

*Proof.* Let us compute the phase applied to states  $|G_\lambda\rangle, |G_\lambda^\perp\rangle$ :  $\hat{Z}_\phi|G_\lambda\rangle = ((1+e^{-i\phi}) - 1)|G_\lambda\rangle = e^{-i\phi}|G_\lambda\rangle$  and  $\hat{Z}_\phi|G_\lambda^\perp\rangle = -|G_\lambda^\perp\rangle$ . As this true for all  $\lambda$ , Eq. (20) follows. Combining the representation of  $\hat{W}$  from Eq. (6) with this leads to Eq. (19),

$$\hat{W}_\phi = \hat{Z}_{\phi-\pi/2}\hat{W}\hat{Z}_{-\phi+\pi/2} = \bigoplus_\lambda \begin{pmatrix} -ie^{-i\phi} & 0 \\ 0 & -1 \end{pmatrix}_\lambda \begin{pmatrix} \lambda & -\sqrt{1-|\lambda|^2} \\ \sqrt{1-|\lambda|^2} & \lambda \end{pmatrix}_\lambda \begin{pmatrix} ie^{i\phi} & 0 \\ 0 & -1 \end{pmatrix}_\lambda.$$

$\square$

We provide algorithms with different overheads for large classes of transformations. ‘Ancilla-free quantum signal processing’ in [Section 5.1](#) implement target operators where the real and imaginary parts have the same parity with respect to  $\hat{H}$ . Opposite parity is obtained by ‘single-ancilla quantum signal processing’ in [Section 5.2](#).

## 5.1 Ancilla-Free Quantum Signal Processing

Consider a sequence of  $Q$  of phased iterates, where the angle  $\phi$  defining each may differ.

$$\begin{aligned}\hat{W}_{\vec{\phi}} &= \hat{W}_{\phi_Q} \cdots \hat{W}_{\phi_2} \hat{W}_{\phi_1} = \bigoplus_{\lambda} e^{-i\hat{\phi}_Q^{\lambda}\theta_{\lambda}} \cdots e^{-i\hat{\phi}_2^{\lambda}\theta_{\lambda}} e^{-i\hat{\phi}_1^{\lambda}\theta_{\lambda}}, \quad \hat{\phi}_j^{\lambda} = \cos(\phi_j)\hat{X}_{\lambda} + \sin(\phi_j)\hat{Y}_{\lambda}. \\ &= \bigoplus_{\lambda} \mathcal{A}(2\theta_{\lambda})\hat{\mathcal{I}}_{\lambda} + i\mathcal{B}(2\theta_{\lambda})\hat{\mathcal{Z}}_{\lambda} + i\mathcal{C}(2\theta_{\lambda})\hat{\mathcal{X}}_{\lambda} + i\mathcal{D}(2\theta_{\lambda})\hat{\mathcal{Y}}_{\lambda}.\end{aligned}\tag{21}$$

In each subspace  $\mathcal{H}_{\lambda}$ , this is a product of  $SU(2)$  rotations. As such, we may decompose this in the Pauli basis  $\hat{\mathcal{I}}_{\lambda}, \hat{\mathcal{X}}_{\lambda}, \hat{\mathcal{Y}}_{\lambda}, \hat{\mathcal{Z}}_{\lambda}$  with the real functions  $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$  as coefficients. As we can only prepare and measure in the basis of the state  $|G\rangle_a$ , consider the component  $\langle G|_a \hat{W}_{\vec{\phi}} |G\rangle_a = \sum_{\lambda} (\mathcal{A}(\theta_{\lambda}) + i\mathcal{B}(\theta_{\lambda}))|\lambda\rangle\langle\lambda| \equiv A[\hat{H}] + iB[\hat{H}]$ . Any choice of phases  $\vec{\phi} \in \mathbb{R}^Q$  generates sophisticated interference effects between elements of the sequence, leading to  $(\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D})$  with some non-trivial functional dependence on  $\hat{H}$ . Though the dependence of the output on  $\vec{\phi}$  seems hard to intuit, they nevertheless specify a *program* for computing functions of  $\hat{H}$ , similar to how a list of numbers might specify a polynomial.

To understand [Eq. \(21\)](#), it suffices to study the following sequence of single-qubit rotations.

$$\hat{U}_{\vec{\phi}} = e^{-i\hat{\phi}_Q\theta/2} \cdots e^{-i\hat{\phi}_2\theta/2} e^{-i\hat{\phi}_1\theta/2} = \mathcal{A}(\theta)\hat{\mathcal{I}} + i\mathcal{B}(\theta)\hat{\mathcal{Z}} + i\mathcal{C}(\theta)\hat{\mathcal{X}} + i\mathcal{D}(\theta)\hat{\mathcal{Y}},\tag{22}$$

These functions previously characterized in [Ref. \[37\]](#), and found the following.

**Lemma 12** (Achievable  $A, B$  – Thm. 2.3 of [Ref. \[37\]](#)). *For any integer  $Q > 0$ , a choice of functions  $\mathcal{A}(\theta) \equiv A(x), \mathcal{B}(\theta) \equiv B(x)$  in [Eq. \(22\)](#) is achievable by some  $\vec{\phi} \in \mathbb{R}^Q$  if and only if all the following are true:*

- (1)  $A(x)$  and  $B(x)$  are real parity- $(Q \bmod 2)$  polynomials in  $x = \cos(\theta/2)$  of degree at most  $Q$ ;
- (2)  $A(1) = 1$ ;
- (3)  $\forall x \in [-1, 1], A^2(x) + B^2(x) \leq 1$ ;
- (4)  $\forall x \geq 1, A^2(x) + B^2(x) \geq 1$ ;
- (5)  $\forall Q$  even,  $x \geq 0, A^2(ix) + B^2(ix) \geq 1$ .

Moreover,  $\vec{\phi} \in \mathbb{R}^Q$  can be computed in classical  $\mathcal{O}(\text{poly}(Q))$  time.

In other words, we may specify only the components  $\mathcal{A}, \mathcal{B}$ , independent of the others  $\mathcal{C}, \mathcal{D}$  that are of lesser interest. By mapping this result to [Eq. \(21\)](#), we complete the proof of [Theorem 3](#).

*Proof of Theorem 3.* The unitary operators in each subspace  $\mathcal{H}_{\lambda}$  of  $\hat{W}_{\vec{\phi}}$  are isomorphic to the product of single qubit rotations  $e^{-i\hat{\phi}_Q\theta/2} \cdots e^{-i\hat{\phi}_2\theta/2} e^{-i\hat{\phi}_1\theta/2}$ . We identify  $\theta/2 = \theta_{\lambda}$ , where  $\theta_{\lambda} = \cos^{-1}(\lambda)$  as defined in [Eq. \(7\)](#). Thus the result follows from [Lemma 12](#) by substituting  $x = \cos(\theta_{\lambda}) = \lambda$ .  $\square$

As polynomials form a complete basis on bounded real intervals, these results imply the query complexity of approximating any real function  $A[\hat{H}]$  with error  $\epsilon$  is exactly that of its best polynomial  $\epsilon$ -approximation satisfying the constraints of [Theorem 3](#), and similarly for the complex case.

## 5.2 Single-Ancilla Quantum Signal Processing

Using no ancilla, [Theorem 3](#) implements a target operator  $f[\hat{H}]$  where the real and complex parts are constrained to have the same parity. By introducing additional ancilla, other general classes of functions can be implemented, sometimes with looser constraints. In this section, we use a key observation from

Ref. [7]. Given any unitary  $\hat{V}$  with eigenstates  $\hat{V}|\lambda\rangle = e^{i\theta_\lambda}|\lambda\rangle$  and  $\hat{V}_0 = |+\rangle\langle +|_b \otimes \hat{\mathcal{I}}_s + |-\rangle\langle -|_b \otimes \hat{V}$  controlled by the single-qubit ancilla register  $b$  where  $\hat{X}_b|\pm\rangle_b = \pm|\pm\rangle_b$ , consider the sequence

$$\hat{V}_{\vec{\varphi}} = \prod_{k \text{ odd} \geq 1}^{Q/2} \hat{V}_{\varphi_{k+1}+\pi}^\dagger \hat{V}_{\varphi_k} = \hat{V}_{\varphi_Q+\pi}^\dagger \cdots \hat{V}_{\varphi_3} \hat{V}_{\varphi_2+\pi}^\dagger \hat{V}_{\varphi_1}, \quad \hat{V}_\varphi = (e^{-i\varphi\hat{Z}/2} \otimes \hat{\mathcal{I}}_s) \hat{V}_0 (e^{i\varphi\hat{Z}/2} \otimes \hat{\mathcal{I}}_s). \quad (23)$$

For each eigenstate  $|\lambda\rangle$ , we obtain a product of single qubit operators  $e^{-i\hat{\varphi}_Q\theta_\lambda/2} \cdots e^{-i\hat{\varphi}_1\theta_\lambda/2}$  similar to Eq. (21) but with a halved phased, and these only act on the ancilla  $b$ . Using the same reasoning as in Section 5.1, the choice of  $\vec{\varphi}$  determines the effective single-qubit ancilla operator

$$\hat{V}_{\vec{\varphi}} = \bigoplus_{\lambda} \left( \hat{\mathcal{I}}_b \mathcal{A}(\theta_\lambda) + i\hat{Z}_b \mathcal{B}(\theta_\lambda) + i\hat{X}_b \mathcal{C}(\theta_\lambda) + i\hat{Y}_b \mathcal{D}(\theta_\lambda) \right) \otimes |\lambda\rangle\langle \lambda|_s. \quad (24)$$

In the following, we focus on the functions  $\mathcal{A}, \mathcal{C}$ , which we also characterized fully in previous work.

**Lemma 13** (Achievable  $(\mathcal{A}, \mathcal{C})$  – Thm. 1 of Ref. [7]). *For any even integer  $Q > 0$ , a choice of functions  $\mathcal{A}(\theta), \mathcal{C}(\theta)$  in Eq. (22) is achievable by some  $\vec{\varphi} \in \mathbb{R}^Q$  if and only if all the following are true:*

- (1)  $\mathcal{A}(\theta) = \sum_{k=0}^{Q/2} a_k \cos(k\theta)$  is a real cosine Fourier series of degree at most  $Q/2$ ;
- (2)  $\mathcal{C}(\theta) = \sum_{k=1}^{Q/2} c_k \sin(k\theta)$  is a real sine Fourier series of degree at most  $Q/2$ ;
- (3)  $\mathcal{A}(0) = 1$ ;
- (4)  $\forall \theta \in \mathbb{R}, \mathcal{A}^2(\theta) + \mathcal{C}^2(\theta) \leq 1$ .

Moreover,  $\vec{\varphi} \in \mathbb{R}^Q$  can be computed in classical  $\mathcal{O}(\text{poly}(Q))$  time.

In some cases, one might be given target functions  $\mathcal{A}, \mathcal{C}$  that are only  $\epsilon$ -close to being achievable, for instance, if  $\mathcal{A}, \mathcal{C}$  are the output of some numerical procedure. This poses no fundamental difficulty, as we prove in the following, which generalizes slightly a similar statement in Ref. [7].

**Lemma 14** (Stability of Achievable  $(\mathcal{A}, \mathcal{C})$ ). *For any even integer  $Q > 0$ , let*

- (1)  $\tilde{\mathcal{A}}(\theta) = \sum_{k=0}^{Q/2} a_k \cos(k\theta)$  be a real cosine Fourier series of degree at most  $Q/2$ ;
- (2)  $\tilde{\mathcal{C}}(\theta) = \sum_{k=1}^{Q/2} c_k \sin(k\theta)$  be a real sine Fourier series of degree at most  $Q/2$ ;
- (3)  $\tilde{\mathcal{A}}(0) = 1 + \epsilon_1$ , where  $\epsilon_1 \leq 1$ ;
- (4)  $\forall \theta \in \mathbb{R}, \tilde{\mathcal{A}}^2(\theta) + \tilde{\mathcal{C}}^2(\theta) \leq 1 + \epsilon_2$ , where  $\epsilon_2 \in [0, 1]$ .

Then there exists Fourier series  $\mathcal{A}, \mathcal{C}$  that satisfy the conditions of Lemma 13, approximate

$$\max_{\theta \in \mathbb{R}} |(\mathcal{A} + i\mathcal{C}) - (\tilde{\mathcal{A}} + i\tilde{\mathcal{C}})| = \mathcal{O}(\sqrt{|\epsilon_1| + \epsilon_2}), \quad (25)$$

and are computable in classical  $\mathcal{O}(\text{poly}(Q))$  time.

*Proof.* First, we satisfy condition (4) of Lemma 13 by rescaling  $\mathcal{A}_1 = \frac{\tilde{\mathcal{A}}}{1+\epsilon_2}$ ,  $\mathcal{C} = \frac{\tilde{\mathcal{C}}}{1+\epsilon_2}$ .

Second, we use the polynomial sum-of-squares technique to compute real Fourier series  $\mathcal{B}_1 = \sum_{k=0}^{Q/2} b_k \cos(k\theta)$  and  $\mathcal{D}_1 = \sum_{k=1}^{Q/2} d_k \sin(k\theta)$  such that  $\mathcal{A}_1^2 + \mathcal{B}_1^2 + \mathcal{C}^2 + \mathcal{D}_1^2 = 1$ . Following [37, Section C], this can be done in classical  $\mathcal{O}(\text{poly}(Q))$  time. By construction,  $\mathcal{A}_1^2(0) + \mathcal{B}_1^2(0) = 1$  as  $\mathcal{C}$  and  $\mathcal{D}_1$  are odd functions. Let us define  $\sin(\delta) \equiv \mathcal{B}_1(0)$ . Hence,  $\cos(\delta) \equiv \mathcal{A}_1(0) = \frac{1+\epsilon_1}{1+\epsilon_2}$ .

Third, let  $\mathcal{A}(\theta) = \cos(\delta)\mathcal{A}_1(\theta) + \sin(\delta)\mathcal{B}_1(\theta)$ . By definition,  $\mathcal{A}(0) = \cos^2(\delta) + \sin^2(\delta) = 1$ . Thus  $\mathcal{A}, \mathcal{C}$  satisfy the conditions of Lemma 13.

Last, we bound the error of this approximation.

$$\begin{aligned}
& \max_{\theta \in \mathbb{R}} |(\mathcal{A} + i\mathcal{C}) - (\tilde{\mathcal{A}} + i\tilde{\mathcal{C}})| \\
& \leq \max_{\theta \in \mathbb{R}} |(\mathcal{A} + i\mathcal{C}) - (\mathcal{A}_1 + i\mathcal{C})| + \max_{\theta \in \mathbb{R}} |(\mathcal{A}_1 + i\mathcal{C}) - (\tilde{\mathcal{A}} + i\tilde{\mathcal{C}})| \quad \text{by a triangle inequality} \\
& \leq \max_{\theta \in \mathbb{R}} |(\cos(\delta) - 1)\mathcal{A}_1(\theta) + \sin(\delta)\mathcal{B}_1(\theta)| + \epsilon_2 \\
& \leq \sqrt{(\cos(\delta) - 1)^2 + \sin^2(\delta)} + \epsilon_2 \quad \text{using } \mathcal{A}_1^2(\theta) + \mathcal{B}_1^2(\theta) \leq 1 \\
& \leq \sqrt{2}\sqrt{1 - \cos(\delta)} + \epsilon_2 = \sqrt{2\frac{\epsilon_2 - \epsilon_1}{1 + \epsilon_2}} + \epsilon_2 \\
& = \mathcal{O}(\sqrt{|\epsilon_1| + \epsilon_2}). \tag{26}
\end{aligned}$$

□

We now prove [Theorem 4](#) by evaluating [Eq. \(24\)](#) using [Lemma 13](#) for the case where the arbitrary unitary  $\hat{V}$  is replaced with the more structured iterate  $e^{i\Phi}\hat{W}$ , and also where we have added a global phase  $\Phi \in \mathbb{R}$ .

*Proof of [Theorem 4](#).* Observe from [Eq. \(6\)](#) that  $e^{i\Phi}\hat{W}$  can be diagonalized to obtain its eigenvalues  $\theta_{\lambda\pm}$  and eigenvectors  $|G_{\lambda\pm}\rangle$ .

$$e^{i\Phi}\hat{W}|G_{\lambda\pm}\rangle = e^{i(\Phi + \theta_{\lambda\pm})}|G_{\lambda\pm}\rangle, \quad \theta_{\lambda\pm} = \mp \cos^{-1}(\lambda), \quad |G_{\lambda\pm}\rangle = \frac{|G_{\lambda}\rangle \pm i|G_{\lambda}^{\perp}\rangle}{\sqrt{2}}. \tag{27}$$

With this substitution  $\hat{V} \leftarrow e^{i\Phi}\hat{W}$ , [Eq. \(24\)](#) becomes

$$\hat{V}_{\varphi} = \bigoplus_{\lambda, \pm} \left( \hat{\mathcal{I}}_b \mathcal{A}(\Phi + \theta_{\lambda\pm}) + i\hat{\mathcal{X}}_b \mathcal{C}(\Phi + \theta_{\lambda\pm}) + \dots \right) \otimes |G_{\lambda\pm}\rangle \langle G_{\lambda\pm}|_{as}. \tag{28}$$

Similar to [Section 5.1](#), we are only allowed to prepare and measure in the subspace supported by signal state  $|G\rangle_a$ . Recall that  $\langle \langle G|_a \otimes \hat{\mathcal{I}}_s | G_{\lambda}^{\perp} \rangle \rangle = 0$ , so projecting the sequence [Eq. \(28\)](#) onto  $|+\rangle_b |G\rangle_a$  results in

$$\begin{aligned}
\langle G|_a \langle +|_b \hat{V}_{\varphi} | + \rangle_b |G\rangle_a &= \bigoplus_{\lambda, \pm} (\mathcal{A}(\Phi + \theta_{\lambda\pm}) + i\mathcal{C}(\Phi + \theta_{\lambda\pm})) \otimes \langle G|_a |G_{\lambda\pm}\rangle \langle G_{\lambda\pm}|_{as} |G\rangle_a \\
&= \bigoplus_{\lambda} \frac{\sum_{\pm} (\mathcal{A}(\Phi + \theta_{\lambda\pm}) + i\mathcal{C}(\Phi + \theta_{\lambda\pm}))}{2} \otimes |\lambda\rangle \langle \lambda|. \tag{29}
\end{aligned}$$

For this proof, it suffices to choose  $\Phi = \pi/2$ , and set  $a_k = 0$  for all odd  $k$ , and  $c_k = 0$  for all even  $k$ . We then evaluate each component of [Eq. \(29\)](#) using [Lemma 13](#) and the Chebyshev polynomials  $T_k(x) \equiv \cos(k \cos^{-1}(x))$ .

$$\begin{aligned}
\mathcal{A}(\pi/2 + \theta_{\lambda\pm}) &= \sum_{k \text{ even}}^{Q/2} a_k \cos(k(\pi/2 + \theta_{\lambda\pm})) = \sum_{k \text{ even}}^{Q/2} a_k i^k T_k(\lambda) = \sum_{k \text{ even}}^{Q/2} a'_k T_k(\lambda) = A(\lambda), \tag{30} \\
\mathcal{C}(\pi/2 + \theta_{\lambda\pm}) &= \sum_{k \text{ odd}}^{Q/2} c_k \sin(k(\pi/2 + \theta_{\lambda\pm})) = \sum_{k \text{ odd}}^{Q/2} c_k i^{k-1} T_k(\lambda) = \sum_{k \text{ odd}}^{Q/2} c'_k T_k(\lambda) = C(\lambda).
\end{aligned}$$

Thus Eq. (29) simplifies to

$$\langle G|_a \langle +|_b \hat{V}_{\mathcal{F}}|+\rangle_b |G\rangle_a = \bigoplus_{\lambda} (A(\lambda) + iC(\lambda)) \otimes |\lambda\rangle\langle\lambda|. \quad (31)$$

We now express the conditions of Lemma 13 in terms of polynomials. As  $T_k(-x) = (-1)^k T_k(x)$ , conditions (1) and (2) map to  $A(\lambda)$  being an even polynomial and  $C(\lambda)$  being an odd polynomial respectively. When  $\theta_{\lambda\pm} = -\pi/2$ , this implies that  $\lambda = \cos(\pm\pi/2) = 0$ . Hence condition (3) maps to  $A(\pi/2 + \theta_{\lambda\pm} = 0) = 1 \Rightarrow A(\lambda = 0) = 1$ . As  $\lambda \in [-1, 1]$  for all values of  $\theta_{\lambda\pm}$ , condition (4) maps to  $\forall \lambda \in [-1, 1], A^2(\lambda) + C^2(\lambda) \leq 1$ .  $\square$

Of course, other choices of  $\Phi$ , such as  $\Phi = 0, \pi/3, \pi/4, \dots$ , lead to different families of target functions. However, those are beyond the scope of this work.

## 6 Application to Hamiltonian Simulation

With these results for qubitization and operator function design with a quantum signal processor, the application to Hamiltonian simulation follows easily. We complete the proof Theorem 1, and then apply these results to obtain our claims of improvements in Table 1. Given any Hamiltonian  $\hat{H}$  encoded in the standard-form of Definition 1, Theorem 4 allows us to use  $2Q$  queries to encode exactly any function  $A[\hat{H}] + iC[\hat{H}]$  where  $A(\lambda)$  and  $C(\lambda)$  are bounded polynomials of opposite parity and degree  $Q$ . Hamiltonian simulation is accomplished by choosing a good degree  $Q$  polynomial approximation to  $A(\lambda) + iC(\lambda) = e^{-i\lambda t}$ .

*Proof of Theorem 1.* Similar to previous approaches [6, 7], we approximate  $e^{-i\lambda t}$  with the Jacobi-Anger expansion  $e^{i\cos(z)t} = \sum_{k=-\infty}^{\infty} i^k J_k(t) e^{ikz}$  [38], where  $J_k(t)$  are Bessel function of the first kind. By identifying  $\cos(z) = \lambda$ , suitable polynomials  $A, C$  are obtained by a truncation and rescaling of

$$e^{-i\lambda t} = J_0(t) + 2 \sum_{k \text{ even} > 0}^{\infty} (-1)^{k/2} J_k(t) T_k(\lambda) + i2 \sum_{k \text{ odd} > 0}^{\infty} (-1)^{(k-1)/2} J_k(t) T_k(\lambda). \quad (32)$$

The error from truncating this expansion for  $k > Q = q - 1$  is a sum of  $|J_k(t)|$  that was bounded in [6]:

$$\begin{aligned} \epsilon &= \max_{\lambda \in [-1, 1]} \left| 2 \sum_{k=q}^{\infty} (-1)^{\frac{k}{2}} J_k(t) T_k(\lambda) \right| \leq \sum_{k=q}^{\infty} 2 |J_k(t)| \leq \frac{4t^q}{2^q q!} = \mathcal{O}\left(\left(\frac{et}{q}\right)^q\right) \\ &\Rightarrow \log\left(\frac{1}{\epsilon}\right) = \mathcal{O}\left(q \log\left(\frac{2q}{et}\right)\right) \\ &\Rightarrow q = \mathcal{O}(t + \log(1/\epsilon)). \end{aligned} \quad (33)$$

The rapid converge by truncation arises as  $e^{-i\lambda t}$  is an entire analytic function [39].

With the choice

$$\tilde{A}(\lambda) = J_0(t) + 2 \sum_{k \text{ even} > 0}^Q (-1)^{k/2} J_k(t) T_k(\lambda), \quad \tilde{C}(\lambda) = 2 \sum_{k \text{ odd} > 0}^Q (-1)^{(k-1)/2} J_k(t) T_k(\lambda) \quad (34)$$

the error  $\max_{\lambda \in [-1, 1]} |\tilde{A}(\lambda) + i\tilde{C}(\lambda) - e^{-i\lambda t}| \leq \epsilon$ . Though this choice satisfies conditions (1-2) of Theorem 4, it is only  $\epsilon$ -close to satisfying conditions (3-4). Fortunately, this is not a fundamental problem following the stability analysis of Lemma 14. One can perturb  $\tilde{A}(\lambda), \tilde{C}(\lambda)$  to construct approximations  $A(\lambda) + iC(\lambda)$  that are achievable, with error  $\max_{\lambda \in [-1, 1]} |A(\lambda) + iC(\lambda) - e^{-i\lambda t}| = \mathcal{O}(\sqrt{\epsilon})$ . Thus the overall procedure constructs a unitary  $\hat{V}_{\mathcal{F}}$  that approximates the time-evolution operator with error

$\|\langle +|_b \langle G|_a \hat{V}_{\tilde{\varphi}}|G\rangle_a |+\rangle_b - e^{-i\hat{H}t}\| = \mathcal{O}(\sqrt{\epsilon})$ . The failure probability is  $\mathcal{O}(\epsilon)$ , and solving for  $Q$  furnishes stated the number of queries to  $\hat{W}$  and hence the oracles encoding  $\hat{H} = \langle G|\hat{U}|G\rangle$ .  $\square$

As we have already shown in [Section 3](#) how a variety of matrix input models such as linear-combination-of-unitaries,  $d$ -sparse Hamiltonians, and purified density matrices map to the standard-form encoding, their respective algorithms for simulating the time-evolution operators follow trivially.

**Corollary 15** (Hamiltonian Simulation of a Sparse Hermitian Matrix). *Given access to the oracle of [Eq. \(11\)](#) specifying a  $d$ -sparse Hamiltonian  $\hat{H}$ , time evolution by  $\hat{H}$  can be simulated for time  $t$  and error  $\epsilon$  with  $\mathcal{O}(dt\|\hat{H}\|_{\max} + \log(1/\epsilon))$  queries to  $\hat{O}_H, \hat{O}_F$ .*

*Proof.* Follows from combining [Theorem 1](#) with the standard-form encoding [Lemma 6](#).  $\square$

The query complexity in [Eq. \(33\)](#) for this sparse case exactly matches a lower bound based on simulating a Hamiltonian that solves PARITY with unbounded error [\[21\]](#), valid for all parameter values, and not just in asymptotic limits [\[6, 7\]](#). This completes the optimality proof of [Theorem 1](#).

The case where  $\hat{H}$  decomposes into a linear combination of unitaries is an immediate application:

**Corollary 16** (Hamiltonian Simulation of a Linear Combination of Unitaries). *Given access to the oracles of [Eq. \(9\)](#) specifying a Hamiltonian  $\hat{H} = \sum_{j=1}^d \alpha_j \hat{U}_j$ , time evolution by  $\hat{H}$  can be simulated for time  $t$  and error  $\epsilon$  with  $\mathcal{O}(\alpha t + \log(1/\epsilon))$  queries.*

*Proof.* Follows from combining [Theorem 1](#) with the standard-form encoding [Lemma 5](#).  $\square$

The intuitiveness of [Theorem 1](#) allows us to swiftly devise new models of Hamiltonian simulation.

**Corollary 17** (Hamiltonian Simulation of a Purified Density Matrix). *Given access to the oracle [Eq. \(14\)](#) specifying a Hamiltonian  $\hat{H} = \hat{\rho}$  that is a density matrix  $\hat{\rho}$ , time evolution by  $\hat{H}$  can be simulated for time  $t$  and error  $\epsilon$  with  $\mathcal{O}(t + \log(1/\epsilon))$  queries.*

*Proof.* Follows from combining [Theorem 1](#) with the standard-form encoding [Lemma 7](#).  $\square$

In all the above, the query complexity  $\mathcal{O}(\dots t + \log(1/\epsilon))$  is an upper bound on the more precise form of [Eq. \(33\)](#). The exact the tradeoff between  $\epsilon, t, Q$  in [Eq. \(33\)](#) is studied numerically in [Appendix B](#), together with example phases  $\tilde{\varphi}$  implementing  $\hat{V}_{\tilde{\varphi}}$  for the polynomials in [Eq. \(34\)](#).

## 7 Conclusion

Our general procedure for Hamiltonian simulation in [Theorem 1](#) extends the scope of possible useful formulations of Hamiltonian simulation. As seen in [Table 1](#), it encompasses any case where the Hamiltonian is embedded in a flagged subspace of the signal unitary. Given this, a simulation algorithm with query complexity optimal in all parameters, and also not just in asymptotic limits, is easily obtained with minimal overhead. While this procedure contains and significantly improves upon important models where the Hamiltonian is  $d$ -sparse or a linear combination of unitaries, its greater value lies in illuminating an intuitive and straightforward path to other as-yet undiscovered models of Hamiltonian simulation. In particular, our result for time-evolution by a purified density matrix is a quadratic improvement in time and an exponential improvement in error over the sample-based model – the proof of which consisted of just a few lines.

Many other exciting directions extend from this work. One example is how additional structural information about  $\hat{H}$  [\[40\]](#) may be exploited. This is illustrated by when the spectral norm of  $\|\hat{H}\|$  is significantly smaller than the sum of coefficients  $\|\vec{\alpha}\|_1$  of a particular linear combination of unitaries decomposition. If this decomposition were to be used, simulation would take time  $\mathcal{O}(\|\vec{\alpha}\|_1 t + \log(1/\epsilon))$  – a factor  $\|\vec{\alpha}\|_1/\|\hat{H}\|$  slowdown. Unfortunately, an  $\|\vec{\alpha}\|_1 = \mathcal{O}(\|\hat{H}\|)$  decomposition appears difficult to



find in general. Furthermore, it is easy to construct pathological  $\hat{H}$  with small norms, but nevertheless decompose by naive methods into components with large spectral norms [41]. Our approach offers a possible solution – one finds any encoding the Hamiltonian in the standard-form, rather than some specific decomposition into parts with properties dictated by the formulation. It remains an interesting challenge to identify the cases where, and determine how structural information may be incorporated.

These advances are special cases arising from our vision of the more general quantum signal processor. Through qubitization, structure is imposed onto any unitary process implementing some Hermitian signal operator. This structure allows for efficient processing of the signal by the techniques of quantum signal processing into some more desired form. As the query complexity of approximating any arbitrary target function of the signal exactly matches fundamental bounds lower in polynomial and Fourier approximation theory [33], we expect this to have numerous application in metrology [42] and other quantum algorithms [35].

## 7.1 Developments after preprint release

The preprint of this manuscript was released on October 2016. There have since been a number of developments building off, or relating to, the concepts of the standard-form encoding, qubitization, and quantum signal processing that could be of interest to the reader.

**Singular value transformations:** The standard-form encoding, qubitization, and quantum signal processing of Hermitian or normal matrices has been extended to perform polynomial transformations on the singular value of arbitrary matrices [43]. This generalization has proven to be surprisingly broad, and provides a unified framework for deriving and understanding many other quantum algorithms.

**Structured Hamiltonian simulation:** The Hamiltonian of many interesting systems possess additional structure that should be exploited to further reduce the cost of simulation. More refined simulation algorithms can exploit prior knowledge of geometric locality [44, 45], Hamiltonian induced one-norm [46], spectral norm [47], and separation of energy scales [48]. In particular, these algorithms are also optimal, up to sub-polynomial factors, with respect to their structural parameters.

**Numerical Quantum signal processing:** The numerical algorithm for decomposing target polynomials of degree  $Q$  into phases was originally claimed to be polynomial time [37], but only when counting arbitrary-precision arithmetical operations, and appeared to be ill-conditioned in case studies [49]. A more thorough analysis [50] proved that the algorithm had a runtime of  $\mathcal{O}(Q^3 \text{polylog}(Q/\epsilon))$  with finite-precision arithmetic, provided that some subtleties were carefully managed to control this ill-conditioned behavior.

## 8 Acknowledgments

G.H.Low and I.L.Chuang thank Robin Kothari, Yuan Su, and Andrew Childs for insightful discussions, the excellent reviews by anonymous referees, and acknowledge funding by the ARO Quantum Algorithms Program and NSF RQCC Project No.1111337.

## A Qubitization of Normal Operators

The results of Lemmas 8 and 10 for qubitization can be extended to normal operators. Let  $\langle G|_a \hat{U} |G\rangle_a = \hat{H}$  in Eq. (2) encode a normal matrix  $\hat{H}$ . It is well known that any normal matrix has a polar decomposition

$$\hat{H} = \hat{H}_U \cdot \hat{H}_H. \tag{35}$$

where  $\hat{H}_U$  is unitary,  $\hat{H}_H$  is positive-semidefinite,  $[\hat{H}_U, \hat{H}_H] = 0$  commute. Thus the eigenvalues are  $\hat{H}|\lambda\rangle = e^{i\theta_\lambda} \lambda|\lambda\rangle$ , where  $\lambda \geq 0$ ,  $\theta_\lambda \in \mathbb{R}$ , and

$$\hat{U}|G\rangle|\lambda\rangle = \hat{U}|G_\lambda\rangle = \lambda e^{i\theta_\lambda} |G_\lambda\rangle + \sqrt{1 - |\lambda|^2} |G_\lambda^\perp\rangle, \quad (36)$$

This reduces to a Hermitian operator when  $\hat{H}_U$  has eigenvalues  $\pm 1$ , and reduces to a unitary operator when all  $\hat{H}_H$  has eigenvalues 1. The trivial approach to qubitization in fact applies to any complex matrix. We simply use the construction of [Lemma 10](#) to implement the Hermitian signal operator  $\frac{1}{2}(\hat{H} + \hat{H}^\dagger)$ .

Another possibility uses two phased iterates in an alternating sequence on input state  $|G\rangle_a |\psi\rangle_s$ ,

$$\hat{W}_{\phi_\pm} = \hat{Z}_{\phi - \pi/2} (2|G\rangle\langle G| - \hat{I}) \hat{U}_\pm \hat{Z}_{-\phi + \pi/2}, \quad \hat{W}_{\bar{\phi}_\pm} = \hat{W}_{\bar{\phi}} = \hat{W}_{\phi_L} \cdots \hat{W}_{\phi_3} \hat{W}_{\phi_2} \hat{W}_{\phi_1}, \quad (37)$$

where  $\hat{U}_+ = \hat{U}$  and  $\hat{U}_- = \hat{U}^\dagger$ . We work through a minimal example in [Appendix A.1](#), from which it follows that for each eigenstate  $|\lambda\rangle$ , the iterates  $\hat{W}_{\phi_\pm}$  maps states in the subspace spanned by  $\{|G\rangle|\lambda\rangle, \hat{W}_{\pi/2, \pm} |G\rangle|\lambda\rangle\}$  to a state in the span of  $\{|G\rangle|\lambda\rangle, \hat{W}_{\pi/2, \mp} |G\rangle|\lambda\rangle\}$ . Let us define the following states by Gram-Schmidt orthogonalization

$$|G_\lambda^{\perp\pm}\rangle = \frac{(\lambda e^{i\theta_\lambda} - \hat{W}_{\pi/2, \pm}) |G_\lambda\rangle}{\sqrt{1 - |\lambda|^2}}. \quad (38)$$

Then these iterates have form

$$\hat{W}_{\phi_\pm} = \begin{array}{cc} \lambda e^{\pm i\theta_\lambda} |G_\lambda\rangle\langle G_\lambda| & -ie^{-i\phi} \sqrt{1 - \lambda^2} |G_\lambda\rangle\langle G_\lambda^{\perp\mp}| \\ -ie^{i\phi} \sqrt{1 - \lambda^2} |G_\lambda^{\perp\pm}\rangle\langle G_\lambda| & +\lambda e^{\mp i\theta_\lambda} |G_\lambda^{\perp\pm}\rangle\langle G_\lambda^{\perp\mp}| \end{array}. \quad (39)$$

The subspace spanned by these states is only invariant in general under the action of iterates with alternating sign  $\pm$ , such as the product  $\hat{W}_{\phi_i} \hat{W}_{\phi_j}$ . In other words, they are not invariant under any repeated application of an iterate of the same sign. With the understanding that we only consider alternating sequences, each  $\hat{W}_{\phi_\pm}$  has the representation

$$\hat{W}_{\phi_\pm} = \begin{pmatrix} e^{\pm i\theta_\lambda} \lambda & -ie^{-i\phi} \sqrt{1 - \lambda^2} \\ -ie^{i\phi} \sqrt{1 - \lambda^2} & e^{\mp i\theta_\lambda} \lambda \end{pmatrix}. \quad (40)$$

Note that when all eigenvalues  $\lambda$  are identical and  $\phi = \pi/2$ , this reduces to Oblivious amplitude amplification [4], and we recover Hermitian qubitization when all  $\theta_\lambda = 0$ . While this approach uses one less ancilla qubit than the construction of [Lemma 10](#), quantum signal processing can only be performed on controlled block of even length  $\hat{W}_{\bar{\phi}_\pm}$ , as only they have an invariant subspace. This limitation can be relevant in some cases, such as Hamiltonian simulation where quantum signal processing is applied to a single  $\hat{W}_\phi$ .

## A.1 Minimal example

A minimal example illustrating the phenomenon of normal qubitization is worked through in this section. For simplicity, let us drop the  $|\lambda\rangle$  state. Let us set the phase  $\phi = \pi/2$  in the  $\hat{Z}$  operator. Let  $|G\rangle \rightarrow |0\rangle$ . Let  $R = 2|0\rangle\langle 0| - 1$ . Let us map the subspace spanned by  $\{|G\rangle|\lambda\rangle, |G_\lambda^{\perp+}\rangle, |G_\lambda^{\perp-}\rangle\} \rightarrow \{|0\rangle, -|1\rangle, -|2\rangle\}$ . Then

$$\begin{aligned} U|0\rangle &= \lambda|0\rangle + \sqrt{1 - |\lambda|^2}|1\rangle, & U^\dagger|0\rangle &= \lambda^*|0\rangle + \sqrt{1 - |\lambda|^2}|2\rangle, \\ RU|0\rangle &= \lambda|0\rangle - \sqrt{1 - |\lambda|^2}|1\rangle, & RU^\dagger|0\rangle &= \lambda^*|0\rangle - \sqrt{1 - |\lambda|^2}|2\rangle. \end{aligned} \quad (41)$$

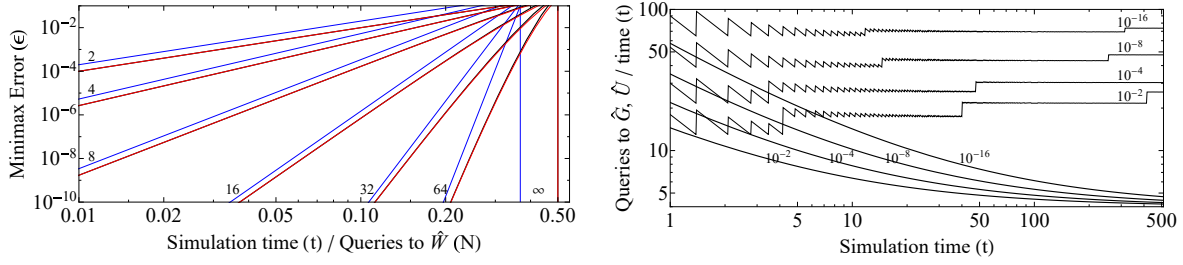


Figure 1: (Left) Approximation error  $\epsilon = \max_{\theta \in \mathbb{R}} |A[\theta] - iC[\theta] - e^{it \sin(\theta)}|$ . ( $A[\theta], C[\theta]$  are real Fourier series in  $(\cos(k\theta), \sin(k\theta))$ ,  $k = 0, \dots, Q/2$ , and  $\epsilon$  is plotted for the upper bound  $\frac{At^q}{2^{q!}}$  (blue), truncation  $\sum_{k=q}^{\infty} 2|J_k[t]|$  (black), and best possible [51] (red), for  $Q = 2, 4, 8, 16, 32, 64, \infty$  queries to the controlled iterate  $\hat{W}$ , where  $q = 1 + Q/2$ . (Right) Queries per unit of simulation time to unitary  $\hat{U}$  encoding  $\hat{H} = \langle G|\hat{U}|G \rangle$  at target error  $\|\langle +|\hat{V}_{\varphi}|+\rangle - e^{-i\hat{H}t}\| = 10^{-2, -4, -8, -16}$  for the BCCKS algorithm (thin), and this work using the truncated approximation to  $e^{it \sin(\theta)}$  (thick).

Note that  $\lambda \in \mathbb{C}$ . Note that  $\langle 1|2 \rangle \neq 0$  in general. Also note that  $\langle 0|URU^{\dagger}|0 \rangle = \langle 0|U^{\dagger}RU|0 \rangle = 2|\lambda|^2 - 1$  by an explicit calculation.

By Gram-Schmidt orthonormalization,

$$|1\rangle = \frac{(U - \lambda)|0\rangle}{\sqrt{1 - |\lambda|^2}}, \quad |2\rangle = \frac{(U^{\dagger} - \lambda^*)|0\rangle}{\sqrt{1 - |\lambda|^2}} \quad (42)$$

Promised that the first input state is  $|0\rangle$ , and we only consider sequences of the form  $(RU^{\dagger}RU)^k$  and  $RU(RU^{\dagger}RU)^k$  for integer  $k$ , we want to prove that any subsequent input to  $RU$  is in  $\text{span}\{|0\rangle, |2\rangle\}$ , and any input to  $RU^{\dagger}$  is in  $\text{span}\{|0\rangle, |1\rangle\}$ . Consider the projectors  $P_j = |0\rangle\langle 0| + |j\rangle\langle j|$ . Then proving the first case requires showing that  $\langle 1|URP_2RU^{\dagger}|1 \rangle = 1$  and for the second case,  $\langle 2|U^{\dagger}RP_1RU|2 \rangle = 1$ . In the first case,

$$\begin{aligned} \langle 1|URP_2RU^{\dagger}|1 \rangle &= \frac{|\langle 0|RU^{\dagger}(U - \lambda)|0 \rangle|^2}{1 - |\lambda|^2} + \frac{|\langle 0|(U - \lambda)RU^{\dagger}(U - \lambda)|0 \rangle|^2}{(1 - |\lambda|^2)^2} \\ &= \frac{|1 - |\lambda|^2|^2}{1 - |\lambda|^2} + \frac{|\lambda|\lambda|^2 - \lambda(2|\lambda|^2 - 1) - \lambda + \lambda|^2}{(1 - |\lambda|^2)^2} \\ &= \frac{|1 - \lambda^2|^2}{1 - \lambda^2} + \frac{\lambda^2|1 - \lambda^2|^2}{(1 - \lambda^2)^2} = 1. \end{aligned} \quad (43)$$

By an identical calculation, the second case is also true.

This means that  $RU$  maps any state in  $\text{span}\{|0\rangle, |2\rangle\}$  to a state in  $\text{span}\{|0\rangle, |1\rangle\}$ , and  $RU^{\dagger}$  maps any state in  $\text{span}\{|0\rangle, |1\rangle\}$  to a state in  $\text{span}\{|0\rangle, |2\rangle\}$ . Using the fact that  $\langle 0|RU|2 \rangle = \langle 2|U^{\dagger}|0 \rangle^* = \sqrt{1 - |\lambda|^2}$ , we know that the overlap  $|\langle 1|RU|2 \rangle| = |\lambda|$ . With a more careful evaluation,

$$\langle 1|RU|2 \rangle = \frac{\langle 0|(U^{\dagger} - \lambda^*)RU(U^{\dagger} - \lambda^*)|0 \rangle}{1 - |\lambda|^2} = \frac{\lambda^*|\lambda|^2 - \lambda^*(2|\lambda|^2 - 1) - \lambda^* + \lambda^*}{1 - |\lambda|^2} = \lambda^*. \quad (44)$$

This means that we can represent

$$\begin{aligned} RU &= (\lambda|0\rangle - \sqrt{1 - |\lambda|^2}|1\rangle)\langle 0| + (\sqrt{1 - |\lambda|^2}|0\rangle + \lambda^*|1\rangle)\langle 2| \\ &= \begin{matrix} \lambda|0\rangle\langle 0| & + \sqrt{1 - |\lambda|^2}|0\rangle\langle 2| \\ -\sqrt{1 - |\lambda|^2}|1\rangle\langle 0| & + \lambda^*|1\rangle\langle 2| \end{matrix}, \end{aligned} \quad (45)$$

and similarly for  $RU^{\dagger}$ .

$Q$	$\epsilon$	$t$	$\vec{\varphi} = (\varphi_1, \varphi_2, \dots, \varphi_Q)$ for $h(\theta) = t \sin(\theta)$
2	$10^{-2}$	0.0707	(-1.61, 1.67)
4	$10^{-2}$	0.311	(-1.03, 2.54, 1.36, -2.11)
8	$10^{-2}$	1.20	(-2.63, 1.66, -2.42, -1.79, 2.17, 1.43, -2.71),
16	$10^{-2}$	3.78	(0.23, 1.17, 3.07, -1.63, -1.78, 2.88, 1.71, 2.77, -1.95, 3.12, 1.97, -2.56, -2.87, 2.00, -2.29, 2.92, -0.51)
32	$10^{-2}$	10.1	(-2.94, 2.64, -2.32, 2.42, -2.86, -2.72, 2.4, -2.57, -2.96, 2.43, -2.53, -2.63, 2.33, 3.11, -2.17, 3.07, 2.24, -2.98, -1.99, -2.77, 2.22, 2.33, -2.62, -1.75, -2.15, 2.84, 1.68, 1.48, 2.46, -2.26, -0.92, -0.21)
2	$10^{-4}$	0.0070711	(-1.574, 1.5817)
4	$10^{-4}$	0.066948	(-0.6741, 2.5806, 0.7772, -2.4675)
8	$10^{-4}$	0.47498	(-0.0914, -1.2861, 2.1995, 0.995, -2.637, -1.5369, 1.9236, -3.0502)
16	$10^{-4}$	2.2164	(-0.5228, -2.9239, 1.3204, 2.6065, -1.73, -2.6191, 1.7225, 3.0121, -1.2286, -2.5276, 1.8058, -0.4605, 1.1563, -2.3492, 2.1623, -2.6188)
32	$10^{-4}$	7.3957	(1.3594, -2.7039, -1.5041, -1.0845, 2.4817, 2.5571, -3.0846, 1.0661, 2.6256, -2.0267, -2.0383, 3.0857, 1.9338, 2.6966, -2.1759, -2.562, 2.2839, 2.6493, -2.2281, -2.9435, 2.1726, -2.9934, -2.9327, 1.7167, -3.0853, -0.9383, -0.2802, -0.2376, -2.9556, 2.7875, -2.2875, 1.7822)

Table 3: Table of phases implementing target function  $h(\theta) = t \sin(\theta)$  in quantum signal processing. Errors quoted refer to  $\| \langle + | \hat{V}_{\vec{\varphi}} | + \rangle_b - e^{-i\hat{H}t} \| \leq \epsilon$ .

## B Practical Details for Implementing Hamiltonian Simulation

This appendix illustrates a specific application of the quantum signal processing approach to a signal unitary that encodes the Hamiltonian  $\hat{H}$  as a signal operator. In particular, a comparison of performance with the BCCKS approach is made. The details will be useful to readers interested in implementing our procedure on a quantum computer. These include plots for the exact error scaling Eq. (33) of the  $Q/2$  term Fourier approximation to  $e^{it \sin(\theta)}$  in Fig. 1(Left), and the number of required queries per unit of simulated time in Fig. 1(Right). Furthermore, a table of select phases  $\vec{\varphi}$  computed using the algorithm in [37] can be found in Table 3.

In the interests of a fair comparison, Fig. 1(Right) counts the number of queries to the signal operator  $\hat{U}$  in the  $\hat{H} = \langle G | \hat{U} | G \rangle$  encoding.  $\hat{U}$  is not assumed to be qubitized, thus incurring a factor 2 additional cost from querying  $\hat{U}$  and  $\hat{U}^\dagger$  each over the asymptotic limit of 2 queries per unit of simulation time in Fig. 1(Left). Similarly, the BCCKS algorithm [22] incurs a factor 3 additional cost from querying  $\hat{U}$  twice and  $\hat{U}^\dagger$  in their use of oblivious amplitude amplification. As BCCKS is known to be optimal in the regime  $t = \mathcal{O}(\frac{\log(1/\epsilon)}{\log \log(1/\epsilon)})$ , the improvement of our approach is most dramatic outside of it. In particular, the queries per unit time of BCCKS scales like  $\mathcal{O}(\frac{\log(t/\epsilon)}{\log \log(t/\epsilon)})$ , whereas our approach approaches 4 in the limit  $t \rightarrow \infty$ .

The number of queries to in the BCCKS model is  $3Kr$ , where  $K$  is the queries per segment  $e^{-i\hat{H} \log 2/r}$ ,  $r = \lceil t/\log(2) \rceil$  is the number of segments, and 3 is for oblivious amplitude amplification on each segment.  $K$  is chosen such that  $\sum_{k=K+1}^{\infty} \frac{\log^k 2}{k!} \leq \frac{\epsilon}{r}$ . An analysis of the procedure [22] shows that the error of its simulated evolution is a factor  $\mathcal{O}(1) \gtrsim 2$  larger than  $\epsilon$ . Thus Fig. 1 slightly overestimates BCCKS performance as we take  $\epsilon$  directly to be the error.

## References

- [1] S. Lloyd, “Universal Quantum Simulators,” *Science* **273**, 1073 (1996).
- [2] D. Aharonov and A. Ta-Shma, “Adiabatic quantum state generation and statistical zero knowl-

- edge,” in *Proceedings of the thirty-fifth ACM symposium on Theory of computing - STOC '03*, STOC '03 (ACM Press, New York, New York, USA, 2003) p. 20.
- [3] A. M. Childs and N. Wiebe, “Hamiltonian Simulation Using Linear Combinations of Unitary Operations,” *Quantum Information & Computation* **12**, 901 (2012).
  - [4] D. W. Berry and A. M. Childs, “Black-box Hamiltonian simulation and unitary implementation,” *Quantum Information & Computation* **12**, 29 (2012).
  - [5] S. Lloyd, M. Mohseni, and P. Rebentrost, “Quantum principal component analysis,” *Nature Physics* **10**, 631 (2014).
  - [6] D. W. Berry, A. M. Childs, and R. Kothari, “Hamiltonian Simulation with Nearly Optimal Dependence on all Parameters,” in *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, FOCS '15 (IEEE, Washington, DC, USA, 2015) pp. 792–809.
  - [7] G. H. Low and I. L. Chuang, “Optimal Hamiltonian Simulation by Quantum Signal Processing,” *Physical Review Letters* **118**, 010501 (2017).
  - [8] A. W. Harrow, A. Hassidim, and S. Lloyd, “Quantum Algorithm for Linear Systems of Equations,” *Physical Review Letters* **103**, 150502 (2009).
  - [9] A. M. Childs, R. Kothari, and R. D. Somma, “Quantum Algorithm for Systems of Linear Equations with Exponentially Improved Dependence on Precision,” *SIAM Journal on Computing* **46**, 1920 (2017).
  - [10] A. N. Chowdhury and R. D. Somma, “Quantum algorithms for Gibbs sampling and hitting-time estimation,” *Quantum Information & Computation* **17**, 41 (2017).
  - [11] F. G. Brandao and K. M. Svore, “Quantum Speed-Ups for Solving Semidefinite Programs,” *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, 415 (2017).
  - [12] M.-H. Yung, J. D. Whitfield, S. Boixo, D. G. Tempel, and A. Aspuru-Guzik, “Introduction to Quantum Algorithms for Physics and Chemistry,” in *Quantum Information and Computation for Chemistry* (John Wiley & Sons, Inc., 2014) pp. 67–106.
  - [13] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, and M. Troyer, “Gate-count estimates for performing quantum chemistry on small quantum computers,” *Physical Review A* **90**, 022305 (2014).
  - [14] D. Poulin, M. B. Hastings, D. Wecker, N. Wiebe, A. C. Doherty, and M. Troyer, “The Trotter step size required for accurate quantum simulation of quantum chemistry,” *Quantum Information & Computation* **15**, 361 (2015).
  - [15] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, “Elucidating reaction mechanisms on quantum computers,” *Proceedings of the National Academy of Sciences* **114**, 7555 (2017).
  - [16] R. Babbush, D. W. Berry, I. D. Kivlichan, A. Y. Wei, P. J. Love, and A. Aspuru-Guzik, “Exponentially more precise quantum simulation of fermions in second quantization,” *New Journal of Physics* **18**, 033032 (2016).
  - [17] I. D. Kivlichan, N. Wiebe, R. Babbush, and A. Aspuru-Guzik, “Bounding the costs of quantum simulation of many-body physics in real space,” *Journal of Physics A: Mathematical and Theoretical* **50**, 305301 (2017).
  - [18] P. J. J. O’Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, E. Jeffrey, E. Lucero, A. Megrant, J. Y. Mutus, M. Neeley, C. Neill, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, P. V. Coveney, P. J. Love, H. Neven, A. Aspuru-Guzik, and J. M. Martinis, “Scalable Quantum Simulation of Molecular Energies,” *Physical Review X* **6**, 031007 (2016).
  - [19] R. Barends, J. Kelly, A. Megrant, A. Veitia, D. Sank, E. Jeffrey, T. C. White, J. Mutus, A. G. Fowler, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, C. Neill, P. O’Malley, P. Roushan, A. Vainsencher, J. Wenner, A. N. Korotkov, A. N. Cleland, and J. M. Martinis, “Superconducting quantum circuits at the surface code threshold for fault tolerance,” *Nature* **508**, 500 (2014).
  - [20] S. Debnath, N. M. Linke, C. Figgatt, K. A. Landsman, K. Wright, and C. Monroe, “Demonstration of a small programmable quantum computer with atomic qubits,” *Nature* **536**, 63 (2016).

- [21] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, “Exponential improvement in precision for simulating sparse Hamiltonians,” in *Proceedings of the 46th Annual ACM Symposium on Theory of Computing - STOC '14*, STOC '14 (ACM Press, New York, New York, USA, 2014) pp. 283–292.
- [22] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, “Simulating Hamiltonian Dynamics with a Truncated Taylor Series,” *Physical Review Letters* **114**, 090502 (2015).
- [23] A. M. Childs, R. Cleve, E. Deotto, E. Farhi, S. Gutmann, and D. A. Spielman, “Exponential algorithmic speedup by a quantum walk,” in *Proceedings of the thirty-fifth ACM symposium on Theory of computing - STOC '03*, STOC '03 (ACM Press, New York, New York, USA, 2003) p. 59.
- [24] A. M. Childs, “On the Relationship Between Continuous- and Discrete-Time Quantum Walk,” *Communications in Mathematical Physics* **294**, 581 (2010).
- [25] R. Kothari, *Efficient algorithms in quantum query complexity*, Ph.D. thesis, University of Waterloo (2014).
- [26] M. Szegedy, “Spectra of Quantized Walks and a  $\sqrt{\delta\epsilon}$  rule,” arXiv preprint [quant-ph/0401053](https://arxiv.org/abs/quant-ph/0401053) (2004).
- [27] D. W. Berry and L. Novo, “Corrected Quantum Walk for Optimal Hamiltonian Simulation,” *Quantum Information & Computation* **16**, 1295 (2016).
- [28] S. Kimmel, C. Y.-Y. Lin, G. H. Low, M. Ozols, and T. J. Yoder, “Hamiltonian simulation with optimal sample complexity,” *npj Quantum Information* **3**, 13 (2017).
- [29] S. Chakraborty, A. Gilyén, and S. Jeffery, “The power of block-encoded matrix powers: improved regression techniques via faster Hamiltonian simulation,” arXiv preprint [arXiv:1804.01973](https://arxiv.org/abs/1804.01973) (2018).
- [30] R. D. Somma and S. Boixo, “Spectral Gap Amplification,” *SIAM Journal on Computing* **42**, 593 (2013).
- [31] M. Szegedy, “Quantum Speed-Up of Markov Chain Based Algorithms,” in *45th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '04 (IEEE, Washington, DC, USA, 2004) pp. 32–41.
- [32] A. Daskin and S. Kais, “An ancilla-based quantum simulation framework for non-unitary matrices,” *Quantum Information Processing* **16**, 33 (2017).
- [33] G. Meinardus, *Approximation of Functions: Theory and Numerical Methods*, Springer Tracts in Natural Philosophy, Vol. 13 (Springer Berlin Heidelberg, Berlin, Heidelberg, 1967).
- [34] L. K. Grover, “A fast quantum mechanical algorithm for database search,” *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing - STOC '96* STOC '96, 212 (1996).
- [35] T. J. Yoder, G. H. Low, and I. L. Chuang, “Fixed-Point Quantum Search with an Optimal Number of Queries,” *Physical Review Letters* **113**, 210501 (2014).
- [36] J. McClellan, T. Parks, and L. Rabiner, “A computer program for designing optimum FIR linear phase digital filters,” *IEEE Transactions on Audio and Electroacoustics* **21**, 506 (1973).
- [37] G. H. Low, T. J. Yoder, and I. L. Chuang, “Methodology of Resonant Equiangular Composite Quantum Gates,” *Physical Review X* **6**, 041067 (2016).
- [38] M. Abramowitz, I. A. Stegun, and Others, “Handbook of mathematical functions,” Applied mathematics series **55**, 62 (1966).
- [39] J. P. Boyd, “Rootfinding for a transcendental equation without a first guess: Polynomialization of Kepler’s equation through Chebyshev polynomial expansion of the sine,” *Applied Numerical Mathematics* **57**, 12 (2007).
- [40] A. M. Childs and R. Kothari, “Limitations on the Simulation of Non-sparse Hamiltonians,” *Quantum Information & Computation* **10**, 669 (2010).
- [41] R. D. Somma, “A Trotter-Suzuki approximation for Lie groups with applications to Hamiltonian simulation,” *Journal of Mathematical Physics* **57**, 062202 (2016).
- [42] G. H. Low, T. J. Yoder, and I. L. Chuang, “Quantum Imaging by Coherent Enhancement,” *Physical Review Letters* **114**, 100801 (2015).

- [43] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, “Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics,” in *Proceedings of the 51st Annual ACM Symposium on Theory of Computing - STOC '19* (ACM Press, New York, New York, USA, 2019) pp. 193–204.
- [44] J. Haah, M. Hastings, R. Kothari, and G. H. Low, “Quantum Algorithm for Simulating Real Time Evolution of Lattice Hamiltonians,” in *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, FOCS '18 (IEEE, Washington, DC, USA, 2018) pp. 350–360.
- [45] A. M. Childs and Y. Su, “Nearly optimal lattice simulation by product formulas,” *arXiv preprint arXiv:1901.00564* (2019).
- [46] G. H. Low and I. L. Chuang, “Hamiltonian Simulation by Uniform Spectral Amplification,” *arXiv preprint arXiv:1707.05391* (2017).
- [47] G. H. Low, “Hamiltonian simulation with nearly optimal dependence on spectral norm,” in *Proceedings of the 51st Annual ACM Symposium on Theory of Computing - STOC '19* (ACM Press, New York, New York, USA, 2019) pp. 491–502.
- [48] G. H. Low and N. Wiebe, “Hamiltonian Simulation in the Interaction Picture,” *arXiv preprint arXiv:1805.00675* (2018).
- [49] A. M. Childs, D. Maslov, Y. Nam, N. J. Ross, and Y. Su, “Toward the first quantum simulation with quantum speedup,” *Proceedings of the National Academy of Sciences* **115**, 9456 (2018).
- [50] J. Haah, “Product Decomposition of Periodic Functions in Quantum Signal Processing,” *arXiv preprint arXiv:1806.10236* (2018).
- [51] L. J. Karam and J. H. McClellan, “Chebyshev digital FIR filter design,” *Signal Processing* **76**, 17 (1999).