Quantum Computing 3/31/2022

Lecture 17: Qubitization and Quantum Signal Processing

Instructor: Dieter van Melkebeek Scribe: Mingrui Liu, Ilay Raz

Last lecture, we took a detour and looked into input and output mechanisms for linear algebra routines that give exponential speedup compared to classical algorithms. We stated some caveats for these results, and talked about block encodings of matrices as a convenient way to describe large matrices with small quantum circuits. We will be using that today as well, when we pick up our discussion from where we left off in our discussion of Hamiltonian simulation using the additional ingredient of quantum signal processing.

We begin today by talking about the problem of eigenvalue transformation, which we saw in the lecture on Hamiltonian simulation as evaluating matrix functions on block encodings of Hermitian matrices. Then, to introduce quantum signal processing, we must first describe the process of qubitization which can be viewed as an advanced (reverse) version of phase kickback that we first saw in lecture 8. Finally, we state the main result that allows quantum signal processing as well as mention some applications of this paradigm and a further extension, namely singular value transformations.

1 Eigenvalue Transformation

The problem of eigenvalue transformation takes as input a function f and a block encoding of a n-qubit Hermitian matrix M, namely an $(\ell+n)$-qubit unitary $A = \begin{bmatrix} M & * \\ * & * \end{bmatrix}$. It then outputs a block encoding of $f(M)/c$ where $f(M)$ is the matrix with the same eigenvectors as M, only with eigenvalues mapped by $f(x)$, and c is as small as possible.

Note that the unitarity of the input and output block encodings implies that $\|M\|_2 \leq 1$ and $\|f(M)/c\|_2 \leq 1$. Thus, we can only hope to achieve $c = 1$ if $|f(x)| \leq 1$ for all $x \in [-1, 1]$. The latter condition is satisfied when f is a Chebyshev polynomial or when $f(x) = \exp(ixt)$ for $t \in \mathbb{R}$, as in the setting of Hamiltonian simulation. Indeed, in both of those settings we came up with efficient algorithms that achieved $c = 1$ and c very close to 1, respectively.

Here is an overview of the approaches that we have seen and will see for solving (instantiations of) the eigenvalue transformation problem:

1. Our first attempt at matrix inversion in lecture 15 used eigenvalue estimation and then quantum rejection sampling on $f(x) = 1/x$. This was not very efficient because eigenvalue estimation is not very accurate, as a bits of accuracy require an exponentially large number, 2^a, applications of the underlying unitary. Also, quantum rejection sampling can be slow if the probability of success is low, which was the case when the condition number was high.

2. We improved this by using quantum walks and then linear combination of unitaries. This approach relied on Chebyshev polynomials which approximated the function well; for matrix inversion we we able to reduce the dependency on the accuracy from exponential to polynomial. In Hamiltonian simulation the approach led to an algorithm that takes $O(t \log(1/\epsilon))$ time and $O(\log(t) + \log\log(1/\epsilon))$ extra ancillas.

3. Using quantum signal processing we will obtain an algorithm for Hamiltonian simulation that runs in time $O(t + \log(1/\epsilon))$ and only needs $O(1)$ extra ancillas.

2 Recap of Prior Approach

We begin with a recap of the quantum walks approach mentioned above, because our process will be quite similar.

The first step used quantum walks to realize the Chebyshev polynomials. In order to do so, we needed to first transform the block encoding A into another block encoding $\tilde{S}$ for M satisfying $\tilde{S}^2 = I$, and could do this using only a single extra ancilla qubit, so $\ell + 1$ ancillas in total. Then a quantum walk for d steps effectuates $(\tilde{S}R)^d$ and block encodes $T_d(M)$, where R is the reflection over $|0^{\ell+1}*\rangle$, the subspace that fixes the first $\ell + 1$ coordinates to be 0.

Then, we used linear combination of unitaries to approximate $f(M)$ by using the Chebyshev expansion of degree d of $f(x)$ To be precise, we aim to approximate $f(x)$ by $g(x) = \sum_{k=0}^{d} c_k T_k(x)$. To get a block encoding of $g(M)/c$ for $c \doteq \sum_{k=0}^{d} |c_k|$ we calculate

$$(C^* \otimes I)V(C \otimes I)$$

where C is a unitary mapping $|0^{\log d}\rangle$ to $\sum_{k=0}^{d} \sqrt{c_k/c}\,|k\rangle$, and V a unitary "multiplexer" mapping $|k\rangle\,|\psi\rangle$ to $|k\rangle\,(\tilde{S}R)^k\,|\psi\rangle$. This process requires $O(d)$ applications of $\tilde{S}R$, one of C, one of C^*, and $\log d$ extra ancillas. We can view this as the following circuit for $U \doteq \tilde{S}R$:

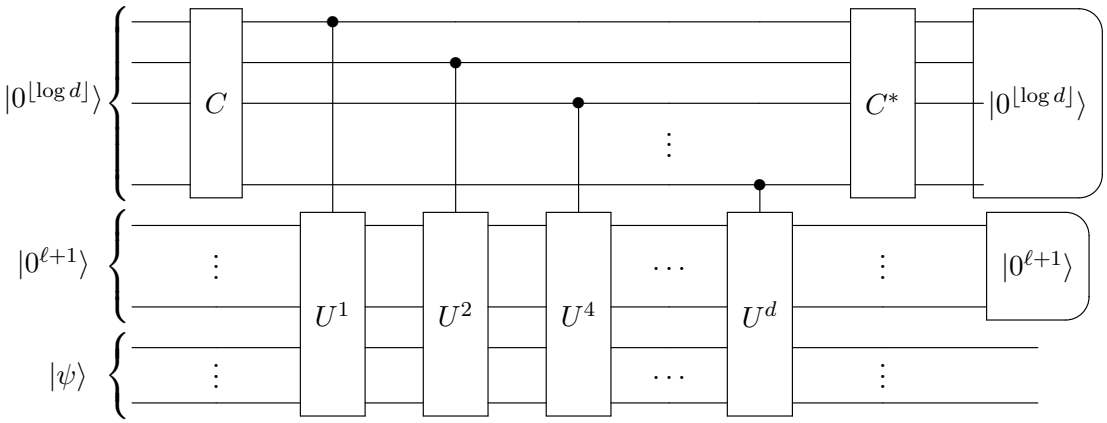

Figure 1: A circuit implementation of prior approach

Note that the output of the last register is $g(M)\,|\psi\rangle$ when the postselection operations (the curved elements in the upper and middle right) succeed. In order to achieve the improvement that we stated, we will avoid the use of LCU as it induces extra ancillas and the additional postselection on the top of Figure 1. This might have a low success probability due to a large value of c. For Hamiltonian simulation, we saw that $d = \Omega(t + \log(1/\epsilon))$ was sufficient to get a close enough approximation, and the value of c converged to e^t, leading to a low probability of success except for small t.

Note that if $g(x)$ happens to be $T_d(x)$, we don't need LCU – the native random walk approach already does the job. Instead of combining random walks of different lengths up to d as happens in the LCU approach, we'll modify the random walk process of length d by interjecting additional operations between every two successive steps of the random walk, where the additional operations are time dependent but do not make use of the blackbox. It turns out that unitary diagonal operations on a single ancilla qubit are sufficient to achieve our goal.

3 Qubitization

Similar to Grover's iterate, the random walk iterate $\tilde{S}R$ is qubitized in the sense that its effect on the entire system can be analyzed through one-qubit subsystems. This qubitization can be viewed as an advanced version of phase kickback in reverse. Phase kickback allows us to interpret an action on a single ancilla qubit as an operation on the actual system. Qubitization allows us to interpret the operation on the system as an action on a single qubit.

As we will see next, qubitization of the quantum walk follows from the fact that $\tilde{S}$ is self-inverse. For that reason, the process we saw earlier to transform a given block encoding A of M into a self-inverse block encoding $\tilde{S}$ of M, is often referred to as qubitization.

Self-inverse vs qubitized block encoding Consider an eigenvector $|\psi\rangle$ of M with corresponding eigenvalue $x \in [-1, 1]$: $M|\psi\rangle = x|\psi\rangle$. This means that the block encoding $\tilde{S}$ acts on $|0^{\ell+1}\rangle|\psi\rangle$ by acting as $M|\psi\rangle = x|\psi\rangle$ on one corner, and then some other perpendicular part. In other words we have for some $|\xi\rangle \perp |0^{\ell+1}*\rangle$

$$\tilde{S}|0^{\ell+1}\rangle|\psi\rangle = x|0^{\ell+1}\rangle|\psi\rangle + \sqrt{1-x^2}|\xi\rangle, \tag{1}$$

where the amplitude $\sqrt{1-x^2}$ comes from the fact that $\tilde{S}$ is unitary so the squared amplitudes must sum up to 1. Note that the state $|\xi\rangle$ is not well-defined if $|x| = 1$. In that case, $|0^{\ell+1}\rangle|\psi\rangle$ is an eigenvector of $\tilde{S}$, and the action is simple to analyze. We assume $|x| < 1$ from now on.

Multiplying both sides of (1) by $\tilde{S}$ from the left and using the fact that $\tilde{S}^2 = I$, we obtain

$$|0^{\ell+1}\rangle|\psi\rangle = x\tilde{S}|0^{\ell+1}\rangle|\psi\rangle + \sqrt{1-x^2}\tilde{S}|\xi\rangle,$$

which together with (1) shows that $\tilde{S}|\xi\rangle$ can be expressed as a linear combination of $|0^{\ell+1}\rangle|\psi\rangle$ and $|\xi\rangle$. It follows that the plane $\mathrm{span}(|0^{\ell+1}\rangle|\psi\rangle, |\xi\rangle)$ is invariant under $\tilde{S}$. Thus, the condition $\tilde{S}^2 = I$ implies that the action of $\tilde{S}$ on $|0^{\ell+1}*\rangle$ is qubitized. A more careful analysis of the argument establishes that qubitization of the block encoding $\tilde{S}$ of M is equivalent to the top left corner of $\tilde{S}^2$ of the same dimension as M being the identity I.

As $\tilde{S}$ is a self-inverse unitary, it is a reflection overall, and acts as a reflection in the invariant plane $\mathrm{span}(|0^{\ell+1}\rangle|\psi\rangle, |\xi\rangle)$. Working out the above calculations shows that the reflection is equivalent to the matrix

$$W(x) \doteq \begin{bmatrix} x & \sqrt{1-x^2} \\ \sqrt{1-x^2} & -x \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix}, \tag{2}$$

where we wrote $x \in [-1, 1]$ as $x \doteq \cos(\theta)$, and $\sqrt{1-x^2} = \sin(\theta)$. The plane $\mathrm{span}(|0^{\ell+1}\rangle|\psi\rangle, |\xi\rangle)$ is also invariant under the overall reflection R: $|0^{\ell+1}\rangle|\psi\rangle$ is an eigenvector with eigenvalue 1, and $|\xi\rangle$

an eigenvector with eigenvalue -1. In Figure 2, the effect of R is a reflection about the horizontal axis, equivalent to the matrix

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

The combined effect of $\tilde{S}R$ is a rotation over θ counter-clockwise, equivalent to the matrix

$$W(x)\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}.$$

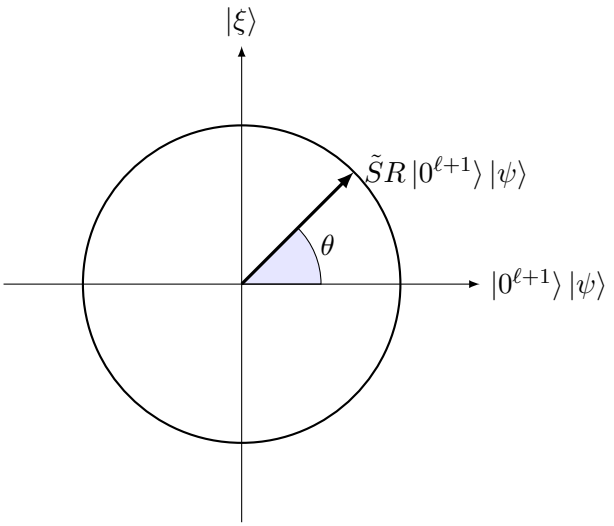

Figure 2: The effect of $\tilde{S}$ and $\tilde{S}R$ on $|0^{\ell+1}\rangle\,|\psi\rangle$

The above perspective clarifies a couple of points that we made earlier:

- The tight connection between Grover and quantum walks. In particular, both are qubitized, and in a very similar way.
- The fact that d steps of the quantum walk yield a block encoding of $T_d(M)$. Indeed, d rotations over θ are equivalent to a rotation over $d\theta$, and $T_d(x) = T_d(\cos\theta) \doteq \cos(d\theta)$.

Implementation To facilitate the transition to quantum signal processing, we introduce in the qubitized quantum walk circuit an actual qubit on which the reflection R acts, and make the action of R on that qubit explicit. We initialize the new ancilla qubit to $|0\rangle$, and implement each reflection R as in Figure 3. The top qubit is the new ancilla, the unfilled dots represent that the NOT is conditioned on the $\ell+1$ existing ancilla qubits being in state $|0\rangle$ rather $|1\rangle$ (as would be denoted with the usual filled dot), and G denotes the one-qubit gate

$$G = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{3}$$

The first controlled NOT splits off those basis states on the $\ell+1+n$ existing qubits whose first $\ell+1$ qubits are in state $|0^{\ell+1}\rangle$, and tags them with a value of $|1\rangle$ for the new ancilla, whereas the

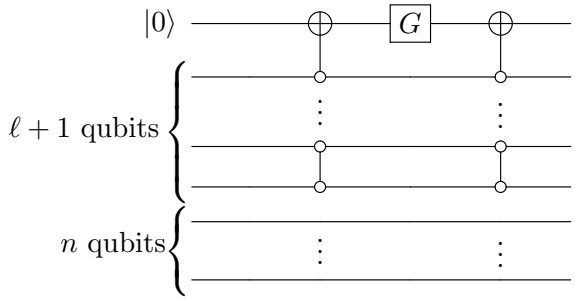

Figure 3: A circuit implementing the reflection R

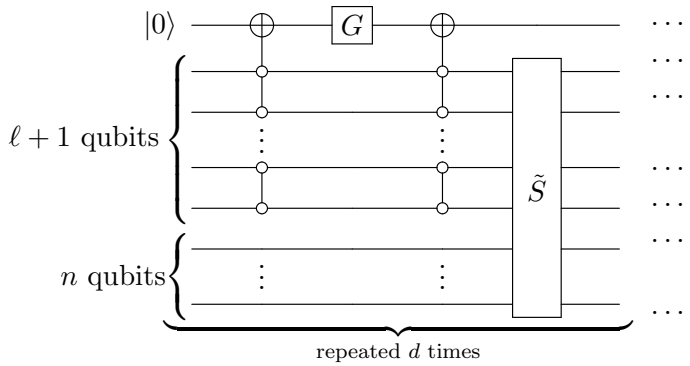

Figure 4: A circuit implementing $(\tilde{S}R)^d$ via qubitization

others remain tagged with $|0\rangle$. The gate G add a phase flip to those tagged with $|0\rangle$, and leaves the others untouched. The last controlled NOT undoes the tagging, which ensures the new ancilla qubit returns to state $|0\rangle$. The net effect is that those basis states on the $\ell + 1 + n$ existing qubits whose first $\ell + 1$ qubits are in a state other than $|0^{\ell+1}\rangle$, have their amplitudes multiplied by -1, while the new ancilla qubit remains in state $|0\rangle$. Thus, R is properly implemented while the new ancilla acts merely as a catalyst.

By incorporating $\tilde{S}$, we can use this circuit to implement $(\tilde{S}R)^d$, as shown in figure 4.

4 Quantum Signal Processing

Quantum signal processing is a technique that involves varying the gate G applied to the first qubit in Figure 4. Let us denote by G_k the gate G that is applied in the k-th step. On input of a state $|0^{\ell+2}\rangle |\psi\rangle$ with $M |\psi\rangle = x |\psi\rangle$, the circuit outputs

$$|0\rangle V \begin{bmatrix} |0^{\ell+1}\rangle |\psi\rangle \\ |\xi\rangle \end{bmatrix},$$

where $|\xi\rangle$ is defined by (1), and

$$V \doteq W(x)G_dW(x)G_{d-1}\dots G_2W(x)G_1, \tag{4}$$

with $W(x)$ given by (2). If the top left corner of the (2×2)-matrix V equals $g(x)$ for every $x \in [-1, 1]$, we know that the circuit represents a block encoding of $g(M)$.

There is an elegant characterization of exactly which (2×2)-matrices V can be realized when the gates G_k are exponentials of the Pauli operator Z and an additional such gate is allowed at the end. Recall that

$$Z \doteq \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad \text{and} \quad e^{i\phi Z} = \begin{bmatrix} e^{i\phi} & 0 \\ 0 & e^{-i\phi} \end{bmatrix}.$$

Lemma 1. *Fix $d \in \mathbb{N}$. There exist $\phi_k \in \mathbb{R}$ for $k \in [d+1]$ such that for all $x \in [-1, 1]$*

$$e^{i\phi_{d+1}Z} W(x) e^{i\phi_d Z} \dots e^{i\phi_2 Z} W(x) e^{i\phi_1 Z} = \begin{bmatrix} P(x) & Q(x)\sqrt{1-x^2} \\ (-1)^{d-1} Q^*(x)\sqrt{1-x^2} & (-1)^d P^*(x) \end{bmatrix} \tag{5}$$

iff $P(x), Q(x) \in \mathbb{C}[x]$ are such that:

(a) $\deg(P) \leq d$ and $\deg(Q) \leq d-1$,

(b) P has parity $d \bmod 2$ and Q the opposite parity, and

(c) $|P(x)|^2 + (1-x^2)|Q(x)|^2 = 1$ for all $x \in [-1, 1]$.

Moreover, the phases ϕ_k can be found to within ϵ in time $\mathrm{poly}(d, \log \frac{1}{\epsilon})$.

In the first requirement for $d = 0$, $\deg(Q) = -1$ means that $Q \equiv 0$. In the second requirement, we say that a function $f(x)$ has parity 0 if it is even, i.e., if $f(x) = f(-x)$ for every input x; we say that it has parity 1 if it is odd, i.e., $f(x) = -f(-x)$ for every input x. An even polynomial $P(x)$ is a polynomial that only contains even powers of x; an odd polynomial $P(x)$ is one that only contains odd powers of x. The third requirement simply expresses that the matrix on the right-hand side of (5) is unitary.

Proof. We give a proof by induction on d. The base case $d = 0$ follows because a polynomial P of degree 0 satisfies (c) with $Q \equiv 0$ iff P is of the form $P \equiv e^{i\phi_1}$ for some $\phi_1 \in \mathbb{R}$. For the induction step from $d-1$ to d for integers $d \geq 1$, we argue the two direction $\Rightarrow$ and $\Leftarrow$ separately.

For the direction $\Rightarrow$, by the induction hypothesis it suffices to prove that for every $\tilde{P}(x), \tilde{Q}(x) \in C[x]$ satisfying (a)-(c) for $d-1$, and for every $\phi_{d+1} \in \mathbb{R}$, there exist $P(x), Q(x) \in C[x]$ satisfying (a)-(c) for d such that

$$e^{i\phi_{d+1}Z} W(x) \begin{bmatrix} \tilde{P}(x) & \tilde{Q}(x)\sqrt{1-x^2} \\ (-1)^d \tilde{Q}^*(x)\sqrt{1-x^2} & (-1)^{d-1} \tilde{P}^*(x) \end{bmatrix} = \begin{bmatrix} P(x) & Q(x)\sqrt{1-x^2} \\ (-1)^{d-1} Q^*(x)\sqrt{1-x^2} & (-1)^d P^*(x) \end{bmatrix}. \tag{6}$$

The (2×2)-matrix equation (6) is equivalent to four equations over $\mathbb{C}$. The equations for the two top entries can be written as follows:

$$\begin{aligned} e^{-i\phi_{d+1}} P(x) &= x\tilde{P}(x) + (-1)^d (1-x^2)\tilde{Q}^*(x) \\ e^{-i\phi_{d+1}} Q(x) &= (-1)^{d-1} \tilde{P}^*(x) + x\tilde{Q}(x) \end{aligned} \tag{7}$$

The equations corresponding to the bottom elements are the complex conjugates of the equations in (7). Thus, (7) is equivalent to (6).

The equations (7) show that (a)-(b) carry over from $\tilde{P}(x)$ and $\tilde{Q}(x)$ for $d-1$ to $P(x)$ and $Q(x)$ for d. That (c) carries over follows from the fact that the matrices $e^{i\phi_{d+1}Z}$ and $W(x)$ are unitary.

For the direction $\Leftarrow$, by the induction hypothesis it suffices to prove that for every $P(x), Q(x) \in C[x]$ satisfying (a)-(c) for d, there exist $\phi_{d+1} \in \mathbb{R}$ and $\tilde{P}(x), \tilde{Q}(x) \in C[x]$ satisfying (a)-(c) for $d-1$ such that (6) holds. We have shown that (6) is equivalent to (7), which can be rewritten as

$$\begin{aligned} \tilde{P}(x) &= e^{-i\phi_{d+1}} xP(x) + (-1)^{d-1} e^{i\phi_{d+1}} (1-x^2) Q^*(x) \\ \tilde{Q}(x) &= (-1)^d e^{i\phi_{d+1}} P^*(x) + e^{-i\phi_{d+1}} xQ(x) \end{aligned} \tag{8}$$

Property (c) carries over for the same reason as in the direction $\Rightarrow$. The equations (8) show that (b) carries over for every choice of $\phi_{d+1} \in \mathbb{R}$, and that (a) carries over with d replaced by $d-1$ modulo one issue: If $\deg(P) = d$ or $\deg(Q) = d-1$, then $\tilde{P}$ may be of degree $d+1$ instead of at most $d-1$, and $\tilde{Q}$ may be of degree d instead of at most $d-2$. However, (8) shows that the coefficient of degree $d+1$ of $\tilde{P}$ as well as the coefficient of degree d of $(-1)^d Q$ are given by

$$e^{-i\phi_{d+1}} p_d + (-1)^d e^{i\phi_{d+1}} q_{d_1}, \tag{9}$$

where p_d denotes the coefficient of degree d of P, and q_{d-1} the coefficient of degree $d-1$ of Q. Property (c) of P and Q can be written as

$$P(x)P^*(x) + (1-x^2)Q(x)Q^*(x) = 1 \text{ for all } x \in [-1,1]. \tag{10}$$

The left-hand side of (10) is a polynomial of degree at most $2d$ with coefficient of degree $2d$ equal to $p_d p_d^* - q_{d-1} q_{d-1}^* = |p_d|^2 - |q_{d-1}|^2$. Since the right-hand side of (10) is the constant polynomial 1, and $d \geq 1$, the coefficient of degree $2d$ of the left-hand side must vanish, which happens iff $|p_d| = |q_{d-1}|$. In that case, we can choose $\phi_{d+1} \in \mathbb{R}$ such that (9) vanishes, namely any value in case $p_d = q_{d-1} = 0$, and a value such that $e^{2i\phi_{d+1}} = (-1)^{d-1} p_d / q_{d-1}$ otherwise. The latter equation allows us to compute the phase ϕ_{d+1} and by (8) the coefficients of $\tilde{P}$ and $\tilde{Q}$ up to an absolute error of at most ϵ in time polynomial in d and $\log(1/\epsilon)$ given the coefficients of P and Q, which suffices to achieve the efficiency stated in the theorem. □

To connect Lemma 1 to (4) for the case where the gates G_k are diagonal, note that every unitary diagonal G can be written as

$$G = \begin{bmatrix} e^{i\alpha} & 0 \\ 0 & e^{i\beta} \end{bmatrix} = e^{i(\alpha+\beta)/2} \begin{bmatrix} e^{i(\alpha-\beta)/2} & 0 \\ 0 & e^{-i(\alpha-\beta)/2} \end{bmatrix} = e^{i(\alpha+\beta)/2} \cdot e^{i\phi Z}$$

for $\alpha, \beta \in \mathbb{R}$ and $\phi \doteq (\alpha-\beta)/2$. It follows that up to a global phase and an extra diagonal unitary G_{d+1}, the expressions on the right-hand side of (4) are equivalent to those on the left-hand side of (5) in Lemma 1. Noting that the lemma provides a necessary and sufficient condition, it fully characterizes the types of operations we can express with quantum signal processing and states that, for operations which can be expressed in this way, the phases ϕ_k can be approximated efficiently.

From the perspective of block encoding, we only care about the top left corner of the (2×2)-matrix (5), i.e., about $P(x)$. The question is for which polynomials $P(x) \in C[x]$ we can (efficiently) find a polynomial $Q(x) \in C[x]$ such that conditions (a)-(c) in Lemma 1 are met. The conditions imply the following necessary properties on P:

(a') $\deg(P) \leq d$.

(b') P has parity d mod 2.

(c') $|P(x)| \leq 1$ for all$x \in [-1, 1]$.

In case P has real coefficients, these conditions turn out to be essentially sufficient, as well.

Lemma 2. *Fix $d \in \mathbb{N}$, and let $P(x) \in \mathbb{R}[x]$ be a polynomial satisfying conditions (a')-(c'). There exist $\phi_k \in \mathbb{R}$ for $k \in [d+1]$ such that for all $x \in [-1, 1]$ the top left corner of*

$$e^{i\phi_{d+1}Z} W(x) e^{i\phi_d Z} \dots e^{i\phi_2 Z} W(x) e^{i\phi_1 Z} \tag{11}$$

has $P(x)$ as its real part. Moreover, the phases ϕ_k can be found to within ϵ in time poly$(d, \log \frac{1}{\epsilon})$ and the time needed to find the roots of a real polynomial of degree at most $2d$ to within poly(ϵ).

Proof. Consider the multiset S of roots of the real polynomial $M(x) = 1 - |P(x)|^2 = 1 - P(x)^2$. Since $M(x)$ is an even real polynomial, for any root $s \in S$ we must have $-s \in S$ and $s^* \in S$. Define the following subsets of S:

$$\begin{aligned} S_0 &= \{s \in S : s = 0\} \\ S_{(0,1)} &= \{s \in S : s \in (0, 1)\} \\ S_{[1,\infty)} &= \{s \in S : s \in [1, \infty)\} \\ S_I &= \{s \in S : \Re(s) = 0 \text{ and } \Im(s) \neq 0\} \\ S_C &= \{s \in S : \Re(s) \neq 0 \text{ and } \Im(s) \neq 0\} \end{aligned} \tag{12}$$

For some scaling factor $k \in \mathbb{R}_+$ we have

$$M(x) = k^2 x^{|S_0|} \prod_{s \in S_{(0,1)}} (x^2 - s^2) \prod_{s \in S_{[1,\infty)}} (s^2 - x^2) \prod_{s \in S_I} (x^2 + |s|^2) \prod_{(a+bi)=s \in S_C} (x^4 + 2x^2(b^2 - a^2) + (a^2 + b^2)^2) \tag{13}$$

Rearrange the terms to

$$\begin{aligned} s^2 - x^2 &= (s^2 - 1)x^2 + s^2(1 - x^2) = |\sqrt{s^2 - 1}x + is\sqrt{1 - x^2}|^2 = |R_s(x)|^2 \\ x^2 + |s|^2 &= (|s|^2 + 1)x^2 + |s|^2(1 - x^2) = |\sqrt{|s|^2 + 1}x + i|s|\sqrt{1 - x^2}|^2 = |P_s(x)|^2 \\ x^4 + 2x^2(b^2 - a^2) + (a^2 + b^2)^2 &= |(cx^2 - a^2 - b^2) + i\sqrt{c^2 - 1}x\sqrt{1 - x^2}|^2 = |Q_{(a,b)}(x)|^2 \end{aligned} \tag{14}$$

where $c = a^2 + b^2 + \sqrt{2(a^2 + 1)b^2 + (a^2 - 1)^2 + b^4}$. Note that $c \in [1, \infty)$ as c is real and $c \geq a^2 + |a^1 - 1| \geq 1$.

Define

$$W(x) = kx^{|S_0|/2} \prod_{s \in S_{(0,1)}} \sqrt{x^2 - s^2} \prod_{s \in S_{[1,\infty)}} R_s(x) \prod_{s \in S_I} P_s(x) \prod_{(a+bi)=s \in S_C} Q_{(a,b)}(x) \tag{15}$$

Note that $x^{|S_0|/2} \prod_{s \in S_{(0,1)}} \sqrt{x^2 - s^2}$ is a polynomial since every root in S_0 and $S_{(0,1)}$ has even multiplicity as $M(x) \geq 0$ for $x \in (-1, 1)$. Also note that $W(x)$ is a product of expressions of the form $B'(x) + i\sqrt{1 - x^2}C'(x)$ where B' and C' are polynomials with real coefficients of opposite parities, thus $W(x)$ can also be written in a similar form as $W(x) = B(x) + i\sqrt{1 - x^2}C(x)$.

Now observe that $M(x) = |W(x)|^2 = B(x)^2 + (1-x^2)C(x)^2$. If $B(x)$ has the same parity as $P(x)$, we set $\tilde{B}(x) = B(x)$ and $\tilde{C}(x) = C(x)$. If $B(x)$ does not have the same parity as $P(x)$, we know $\deg(M) \leq 2d-2$. Then we can use $M(x) = |W(x)(x + i\sqrt{1-x^2})|^2 = \tilde{B}(x) + (1-x^2)\tilde{C}(x)^2$ where $\tilde{B}(x)$ and $\tilde{C}(x)$ are polynomials with real coeffcients; this way, $\tilde{B}(x)$ has again the same parity as $P(x)$.

Define $\tilde{P}(x) \doteq P(x) + i\tilde{B}(x)$, and $\tilde{Q}(x) \doteq \tilde{C}(x)$. We have constructed $\tilde{P}(x), \tilde{Q}(x) \in \mathbb{C}[x]$ that satisfy the conditions (a)-(c) in Lemma 1, and such that the real part of $\tilde{P}(x)$ equals $P(x)$. □

Putting everything together, we can perform eigenvalue transformation by any polynomial $P(x) \in \mathbb{C}[x]$ with a circuit similar to the one in Figure 4 under very mild condition on $P(x)$.

Theorem 3. *Let $P(x) \in \mathbb{C}[x]$ be a polynomial of degree d that satisfies one of the following conditions:*

- *$P(x) \in \mathbb{R}(x)$, has parity $d \bmod 2$, and satisfies $|P(x)| \leq 1$ for all $x \in [-1,1]$.*
- *$P(x) \in \mathbb{R}(x)$ or has parity $d \bmod 2$, and satisfies $|P(x)| \leq 1/2$ for all $x \in [-1,1]$.*
- *$|P(x)| \leq 1/4$ for all $x \in [-1,1]$.*

Given a block encoding A for a Hermitian matrix M with ℓ ancillas, we can compute a block encoding for $P(M)$ with $\ell + O(1)$ ancillas that involves d applications of A and A^*, one controlled application of A, and $O(\ell d)$ other quantum gates. The block encoding can be computed to within an absolute error of ϵ in time $\mathrm{poly}(d, \log \frac{1}{\epsilon})$ and the time needed to find the roots of a real polynomial of degree at most $2d$ to within $\mathrm{poly}(\epsilon)$.

Proof. We start with the first bullet. By qubitization, Lemma 1 and Lemma 2 we can compute phases ϕ_k for $k \in [d+1]$ such that the circuit in Figure 4 corresponding to (4) with $G_k = e^{i\phi_k X}$ is a block encoding of $e^{-i\phi_{d+1}}\tilde{P}(M)$, where $P(M) = \frac{1}{2}(\tilde{P}(M) + \tilde{P}(M)^*)$. The complex conjugate transpose of this block encoding for $\tilde{P}(M)$ gives us a block encoding for $e^{i\phi_{d+1}}\tilde{P}(M)^*$. Using the LCU method, we can combine the two block encodings into one for $P(M)$ of the stated complexity.

For the last bullet, we separate the even/odd and real/imaginary parts of $P(x)$:

$$\begin{aligned}
P^{(even)}(x) &= P(x) + P(-x) \\
P^{(odd)}(x) &= P(x) - P(-x) \\
P^{(even)}_{\Re}(x) &= P^{(even)}(x) + P^{*(even)} \\
P^{(even)}_{\Im}(x) &= (P^{(even)}(x) - P^{*(even)}(x))/i \\
P^{(odd)}_{\Re}(x) &= P^{(odd)}(x) + P^{*(odd)}(x) \\
P^{(odd)}_{\Im}(x) &= (P^{(odd)}(x) - P^{*(odd)}(x))/i
\end{aligned}$$

Each of the last four polynomials satisfy the conditions of first bullet. As

$$P(x) = \frac{1}{4}(P^{(even)}_{\Re}(x) + P^{(odd)}_{\Re}(x) + iP^{(even)}_{\Im}(x) + iP^{(odd)}_{\Im}(x)),$$

we can combine the block encodings corresponding to each of the four parts using the LCU method into one for $P(M)$ of the stated complexity.

A similar argument established the middle bullet, where we only need to consider two parts. □

Application to Hamiltonian simulation There exists a black-box algorithm that takes a block encoding of a Hermitian M with ℓ ancilla qubits, $t \in [0, \infty)$, and $\epsilon \in (0, \infty)$, and produces a block encoding of a matrix Q such that $||Q - \exp(iHt)||_2 \leq \epsilon$, using $q = O(t + \log(\frac{1}{\epsilon}))$ controlled applications of the black-box and its inverse, $O(\ell q)$ other quantum gates, and $\ell + O(1)$ ancilla qubits.

5 Conclusion

Quantum signal processing represents a powerful tool for eigenvalue transformation of Hermitian matrices. The question of applicability boils down to how easily the transformation function $f(x)$ can be approximated by low-degree polynomial $P(x)$. The sufficient condition for P is that $|P(x)| \leq 1/4$ for every $x \in [-1, 1]$. The approach extends to any matrix M that has a full orthonormal basis of eigenvectors. Such matrices are known as normal, and are characterized by the equation $MM^* = M^*M$. They include Hermitian matrices, unitary matrices, and many more, but not all matrices. The approach can be further extended to quantum singular value transformation [GSLW19], which applies to all matrices M.

This technique of eigen (or singular) value transformation captures many of the techniques and speedups we have discussed. There is a strong connection between these techniques and Grover's algorithm, amplitude amplification, and quantum walks, which also act by performing rotations expressed by two reflections across some states in a plane. In particular, the application of Chebyshev polynomials and fast forwarding of random walks follows directly from this framework. In addition to those techniques, which can achieve a quadratic speedup compared to classical results, the quantum signal processing framework can also be used to achieve exponential improvements by expressing the exponential functions used in Hamiltonian simulation and the inversion used in solving systems of linear equations. The functions used to solve the recommender system can also be expressed in this framework.

Among the techniques not captured by this framework are those obtained by phase estimation, such as factoring integers and solving the discrete log and hidden subgroup problems; in these cases, applying phase estimation is efficient because running the underlying unitary the large number of required times ($O(\frac{1}{\epsilon})$ for error bound ϵ) can be made efficient (e.g., by using iterated squaring to perform modular exponentiation efficiently in Shor's algorithm).

References

[GSLW19] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 193–204, New York, NY, USA, 2019. Association for Computing Machinery.