Quantum linear system algorithm with optimal queries to initial state preparation

Guang Hao Low[1,2] and Yuan Su[1]

Abstract

The quantum linear system problem provides one of the most enticing sources of exponential quantum speedups, and its resolution underlies other interesting quantum algorithms for differential equations and eigenvalue processing. The goal is to produce a state proportional to the solution $A^{-1}|b\rangle$ of a linear system with accuracy ϵ, by querying an oracle O_A that block encodes the coefficient matrix and an oracle O_b that prepares the initial state.

We present a quantum linear system algorithm with query complexity $\mathbf{\Theta}\left(1/\sqrt{p}\right)$ to O_b that is optimal, and query complexity $\mathbf{O}\left(\kappa \log\left(1/p\right)\left(\log\log\left(1/p\right)+\log\left(1/\epsilon\right)\right)\right)$ to O_A that is nearly optimal in all parameters including the condition number $\kappa = \|A\|\|A^{-1}\|$ and success amplitude $\sqrt{p} = \|A^{-1}|b\rangle\|/\|A^{-1}\|$. In various applications to solving differential equations, preparing ground states of operators with real spectra, estimating and transforming eigenvalues of non-normal matrices, we can further improve the dependence on p to nearly match or outperform best previous results based on other methods. As κ can be arbitrarily larger than $1/\sqrt{p}$, our algorithm contrasts with recent results that have $\mathbf{O}\left(\kappa \log\left(1/\epsilon\right)\right)$ complexity to both oracles, which, while optimal in O_A, is highly suboptimal in O_b.

We achieve this by a new Variable Time Amplitude Amplification algorithm with Tunable thresholds (Tunable VTAA), which fully characterizes generic nested amplitude amplifications, eliminates redundant nestings, and is of independent interest. With an optimized schedule of thresholds, we prove that the complexity of Tunable VTAA scales with $\ell_{\frac{2}{3}}$-quasinorm of the input costs, improving over the ℓ_1-norm result of Ambainis and the more common ℓ_2-norm scaling. Specialized to the quantum linear system problem, we construct a discretized inverse state, for which a deterministic amplification schedule exists. This leads to a substantially simplified VTAA with an optimal initial state preparation cost, even when the solution norm is not known *a priori*.

We also introduce a *block preconditioning* scheme that can artificially boost $\sqrt{p}$ in generic situations, in contrast to previous negative preconditioning results focusing on reducing κ. This further reduces the cost of initial state preparation in linear-system-based differential equation solvers, ground state preparators and eigenvalue processors. Additionally, block preconditioning furnishes a particularly simple quantum linear system algorithm with optimal $\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\right)$ queries to O_A by using $|b\rangle$ itself as the preconditioner. It also realizes a block-encoded eigenvalue transformer with $\mathbf{O}(n)$ scaling in degree of the target polynomial, compared to the best existing result of $\mathbf{O}\left(n^{1.5}\right)$.

Most of the work was completed while the first author was affiliated with Microsoft.

[1]Azure Quantum, Microsoft, Redmond, WA 98052, USA

[2]Google Quantum AI, Venice, CA 90291, USA

Contents

1 Introduction

1.1 Quantum linear system algorithms

The problem of solving large systems of linear equations provides one of the most enticing sources of exponential speedups for quantum computers. Since the output of any quantum circuit can be simulated by inverting an appropriately chosen matrix, the quantum linear system problem is BQP-complete and captures the full potential of quantum computing—this was proved in the seminal work by Harrow, Hassidim and Lloyd [25], where the first quantum algorithm for sparse systems of linear equations was developed. As linear equations are ubiquitous in science, quantum linear system algorithms have found broad applications such as computing electromagnetic scattering [45], estimating electrical resistance of networks [59], solving differential equations [4, 8, 9, 20, 32], optimization, and more recently, processing eigenvalues of non-normal matrices [39]. While identifying end-to-end scientific applications of quantum linear system solvers with exponential speedups remains challenging, physically-motivated candidates such as computing Green's functions of quantum many-body systems [56] may have well-characterized complexities.

Quantum linear system solvers also provide a natural reduction for designing more advanced quantum algorithms. One class of these algorithms aim to efficiently apply functions $f(A)$ to high-dimensional matrices accessed by a quantum computer through block encoding oracles. When the input is Hermitian, the target function can be readily implemented by the so-called Quantum Singular Value Transformation (QSVT) [22], which since its conception has unified many existing quantum algorithms and uncovered new ones [41]. The general case where A is non-normal is relevant for various other applications such as solving differential equations and simulating transcorrelated quantum chemistry, but the problem appears to be considerably more difficult. One prior approach [20, 54–56] uses the Cauchy integral formula: $f(A) = \frac{1}{2\pi i}\int_{\mathcal{C}} \mathrm{d}z\ f(z)(zI - A)^{-1}$ for $\mathcal{C}$ a contour enclosing all eigenvalues of A. By discretizing the integral, $f(A)$ can be approximated by a linear combination of matrix inverses. Alternative method employs a rational matrix generating function to create a history state encoding a polynomial basis in quantum superposition, which can be post-processed to yield the desired $f(A)$. This technique was initially developed to create monomial basis for solving differential equations [9], but is more recently extended to handle Faber polynomials [39] for implementing more general functions over the complex plane. In any event, these advanced algorithms all rely on quantum linear system solvers as a subroutine and would directly benefit from any enhancements to the solvers.

With an eye towards these applications, we now take a closer look at the cost of quantum linear system algorithms. Here, the goal is to produce a state proportional to the solution $A^{-1}|b\rangle$ of a linear system, using oracular queries to the coefficient matrix A and the initial state $|b\rangle$. The query complexity of quantum linear system algorithms has gradually improved over a long series of work. The seminal work of Harrow, Hassidim and Lloyd showed that $\mathbf{O}\left(\mathrm{poly}\left(\kappa, \frac{1}{\epsilon}\right)\right)$ queries suffice to produce the solution state with accuracy ϵ [25], where

$$\kappa = \|A\| \left\|A^{-1}\right\| \tag{1}$$

is the spectral condition number of the target linear system. The scaling in κ was later improved to almost linear by Ambainis' Variable Time Amplitude Amplification (VTAA) algorithm [2]. Subsequently, the error scaling is improved exponentially to $\mathbf{O}\left(\mathrm{polylog}\left(\frac{1}{\epsilon}\right)\right)$ using a linear combination of quantum walks within VTAA [15]. Recent work achieves a similar scaling in κ and ϵ using alternative methods based on the adiabatic evolution and eigenstate filtering [5, 35, 53]. This ultimately culminated in an algorithm with $\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\right)$ scaling and constant success probability [18]—this was claimed to be optimal, citing an unpublished lower bound by Harrow and Kothari. Even

more recently, a simpler quantum algorithm with the same $\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\right)$ scaling was designed by performing kernel reflection on an augmented linear system [19].

There are two distinct query oracles involved in the quantum linear system problem: the first one is O_A which block encodes the matrix A to be inverted, and the second one is O_b which prepares the initial state $|b\rangle$ from a standard reference state, chosen to be the computational basis state $|0\rangle$ without loss of generality. The query complexities cited above are all the worst-case combined cost where the two oracles O_A and O_b are treated on equal footing. However, the cost to O_b can in fact be much lower than that to O_A, as we will now argue.

Consider the approach where we first construct a block encoding of $\frac{A^{-1}}{2\|A^{-1}\|}$ (we have used $\|A^{-1}\|$ in place of its upper bound for presentational purpose). This can be achieved using QSVT with accuracy ϵ by making $\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\right)$ queries to O_A [22, Corollary 69]. Applying it to the initial state then produces the unnormalized state $\frac{A^{-1}|b\rangle}{2\|A^{-1}\|}$ with accuracy ϵ and success probability $\frac{p_{\text{succ}}}{4}$, consuming one query to O_b, where

$$p_{\text{succ}} = \frac{\left\|A^{-1}|b\rangle\right\|^2}{\|A^{-1}\|^2}, \qquad 1 \leq \frac{1}{\sqrt{p_{\text{succ}}}} = \frac{\|A^{-1}\|}{\|A^{-1}|b\rangle\|} \leq \kappa. \tag{2}$$

This probability can be boosted close to unity with $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$ rounds of amplitude amplification, but the error of solution state is also amplified accordingly. So to achieve an accuracy ϵ in $\frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}$, it suffices to query the oracles a number of times scaling like

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(O_b) + \frac{\kappa}{\sqrt{p_{\text{succ}}}} \log\left(\frac{1}{\sqrt{p_{\text{succ}}}\epsilon}\right) \mathbf{Cost}\left(O_A\right)\right). \tag{3}$$

Thus the number of queries to O_b attains a strictly linear scaling with $\frac{1}{\sqrt{p_{\text{succ}}}}$, surpassing almost all previous results quoted in Table 1 and can be much smaller than κ in practice. In fact, this scaling is already achievable by the algorithm of [25] as is observed in [51]. Note also that the approach works even without a prior knowledge of p_{succ}, in which case we simply use the success probability lower bound $\alpha_{p_{\text{succ}}} \leq p_{\text{succ}}$. Of course, this makes $\mathbf{O}\left(\kappa^2 \log\left(\frac{\kappa}{\epsilon}\right)\right)$ queries to O_A when $\frac{1}{\sqrt{p_{\text{succ}}}} \approx \kappa$, suffering from a quadratic slow down in the worst case. Methods based on the adiabatic theorem and kernel reflection technique have the scaling

$$\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\epsilon}\right)\mathbf{Cost}(O_A)\right), \tag{4}$$

where the complexity of initial state preparation becomes significantly worse.

To date, the *one* and *only one* method offering a close-to-optimal complexity simultaneously to both O_b and O_A is based on VTAA. The state-of-the-art VTAA algorithm makes $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\log(\kappa)\right)$ queries to O_b, under the nontrivial assumption that the probabilities with which the algorithm potentially succeeds at intermediate stages are all estimated to a constant multiplicative accuracy [12]. Without this prior knowledge, the query complexity worsens to $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\log^3(\kappa)\log\log(\kappa)\right)$, and the algorithm becomes considerably more complicated (although this is a one-time cost to pay only when VTAA is compiled into a quantum circuit).

Suppose one were willing to accept the logarithmic slowdown of VTAA and its substantial setup cost—would that improve the complexity of initial state preparation for existing quantum differential equation solvers, eigenvalue estimators and transformers? In solving differential equations, the norm $\|A^{-1}\| \sim t$ scales close to linearly with the evolution time t, whereas $\|A^{-1}|b\rangle\| \sim \sqrt{t}$. So

Year	Algorithm	Primary innovation	Queries to O_b	Queries to O_A
2008	[25]	Phase estimation + Hamiltonian simulation	$\mathbf{O}(\frac{1}{\sqrt{p_{\text{succ}}}})$	$\mathbf{O}\left(\frac{\kappa}{\sqrt{p_{\text{succ}}}\epsilon^2}\right)$
2012	[2]	VTAA	$\mathbf{O}\left(\kappa \operatorname{polylog}\left(\frac{\kappa}{\epsilon}\right)\right)$	$\mathbf{O}\left(\frac{\kappa}{\epsilon^3}\log^3\left(\frac{\kappa}{\epsilon}\right)\right)\log^2\left(\frac{1}{\epsilon}\right)\right)$
2017	[15]	Linear combination of quantum walks	$\mathbf{O}(\frac{1}{\sqrt{p_{\text{succ}}}}\log\left(\frac{\kappa}{\epsilon}\right))$	$\mathbf{O}\left(\frac{\kappa}{\sqrt{p_{\text{succ}}}}\operatorname{polylog}\left(\frac{\kappa}{\epsilon}\right)\right)$
2017	[15]	Gapped phase estimation + Quantum walk + VTAA	$\mathbf{O}(\kappa \operatorname{polylog}\left(\frac{\kappa}{\epsilon}\right))$	$\mathbf{O}\left(\kappa\operatorname{polylog}\left(\frac{\kappa}{\epsilon}\right)\right)$
2018	[53]	Randomized abiabatic evolution	$\mathbf{O}\left(\frac{\kappa}{\epsilon}\log(\kappa)\right)$	$\mathbf{O}\left(\frac{\kappa}{\epsilon}\log(\kappa)\right)$
2018	[12]	Detailed analysis of VTAA	$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\log(\kappa)\right)$	$\mathbf{O}\left(\kappa\log(\kappa)\log^2\left(\frac{\kappa}{\epsilon}\right)\right)$
2019	[5]	Continuous time abiabatic evolution	$\mathbf{O}\left(\kappa\operatorname{polylog}\left(\frac{\kappa}{\epsilon}\right)\right)$	$\mathbf{O}\left(\kappa\operatorname{polylog}\left(\frac{\kappa}{\epsilon}\right)\right)$
2019	[35]	Eigenstate filtering	$\mathbf{O}\left(\kappa\log\left(\frac{\kappa}{\epsilon}\right)\right)$	$\mathbf{O}\left(\kappa\log\left(\frac{\kappa}{\epsilon}\right)\right)$
2022	[18]	Discrete time adiabatic evolution	$\mathbf{O}\left(\kappa\log\left(\frac{1}{\epsilon}\right)\right)$	$\Theta\left(\kappa\log\left(\frac{1}{\epsilon}\right)\right)$
2023	[13]	QSVT-based GPE + VTAA	$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\log(\kappa)\right)$	$\mathbf{O}\left(\kappa\log(\kappa)\log\left(\frac{\kappa}{\epsilon}\right)\right)$
2024	[19]	Kernel reflection	$\mathbf{O}\left(\kappa\log\left(\frac{1}{\epsilon}\right)\right)$	$\Theta\left(\kappa\log\left(\frac{1}{\epsilon}\right)\right)$
2024	This work	Tunable VTAA + Discretized inverse state	$\Theta\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$	$\mathbf{O}\left(\kappa\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\log\left(\frac{\log(1/\sqrt{p_{\text{succ}}})}{\epsilon}\right)\right)$
2024	This work	Block preconditioning	$\mathbf{O}\left(\kappa\log\left(\frac{1}{\epsilon}\right)\right)$	$\Theta\left(\kappa\log\left(\frac{1}{\epsilon}\right)\right)$

Table 1: Complexity comparison of the new and previous methods for the quantum linear system problem. Optimal query complexities are shaded dark gray. The present work achieves for the first time an optimal query complexity to initial state preparation among innovations shaded light gray with nearly optimal query complexity to both oracles. The block preconditioning technique further boosts p_{succ} close to 1 in applications to solving differential equations, preparing ground states of non-Hermitian operators and eigenvalue processing, and furnishes an extremely simple algorithm with an optimal coefficient block encoding complexity in the last row.

the dependence on inverse success amplitude scales like $\sim \sqrt{t}$, but there exist methods with cost independent of the evolution time [4, 20]. In estimating eigenvalues of non-normal matrices, the norm $\left\|A^{-1}\right\| \sim \frac{1}{\epsilon}$ scales linearly with the inverse accuracy, whereas $\left\|A^{-1}|b\rangle\right\| \sim \frac{1}{\sqrt{\epsilon}}$. So the dependence on inverse success amplitude scales like $\sim \frac{1}{\sqrt{\epsilon}}$, but again there exist methods with complexity independent of the inverse accuracy [60]. In all cases, one finds oneself in the intermediate regime where

$$1 \ll \frac{1}{\sqrt{p_{\text{succ}}}} = \frac{\left\|A^{-1}\right\|}{\left\|A^{-1}|b\rangle\right\|} \ll \kappa = \|A\|\left\|A^{-1}\right\|, \tag{5}$$

with a high query complexity of initial state preparation. This becomes problematic when the initial state itself is created by an expensive quantum subroutine, as could arise in natural applications to ground state preparation [10], differential equations, and eigenvalue processing.

1.2 Main result

We develop faster quantum linear system algorithms, with a focus on improving the query complexity of initial state preparation. Our main result includes the following.

(i) We develop a quantum linear system algorithm that makes $\Theta\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$ queries to O_b and $\mathbf{O}\left(\kappa\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\left(\log\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)+\log\left(\frac{1}{\epsilon}\right)\right)\right)$ queries to O_A. The strictly linear scaling with inverse success amplitude holds even when a multiplicative estimate of solution norm $\left\|A^{-1}|b\rangle\right\|$ is unavailable, where p_{succ} is replaced by its lower bound, in contrast to all previous approaches where finding such an estimate incurs polylogarithmic overhead. The optimal lower bound to O_b is proven through a reduction to Grover search.

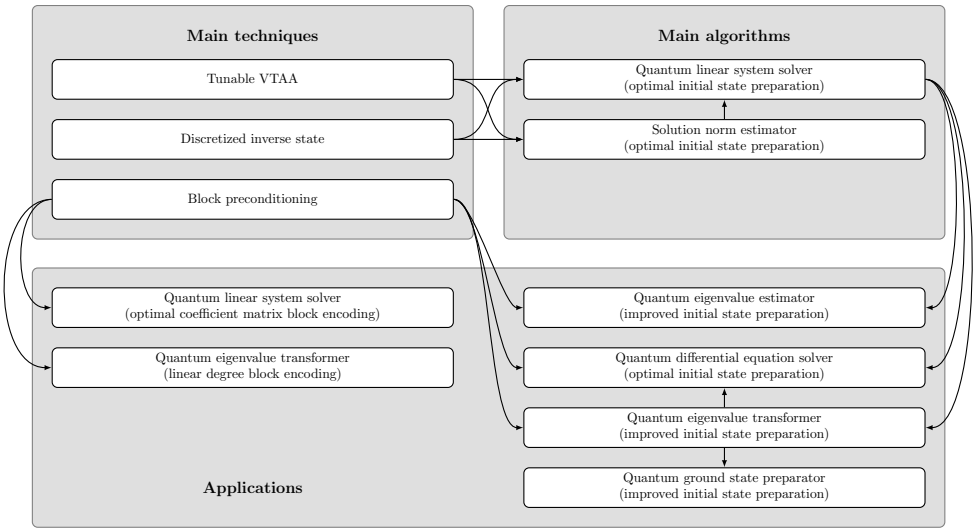


Figure 1: A diagrammatic illustration of the main result and its applications.

(ii) We show how one can artificially boost p_{succ} close to 1. We demonstrate that this leads to differential equation solvers, ground state preparators and eigenvalue processors with improved complexities of initial state preparation, matching or outperforming the state of the art.

Result (i) is realized by a substantially simplified VTAA that uses a deterministic amplification schedule, whose prerequisites are met by a discretized inverse state we design. Result (ii) is based on a simple block preconditioning technique that amplifies a subspace flagged by the initial ancilla state.

Additionally, we find that the block preconditioning technique can be more broadly applied to improve other complexity scalings of quantum linear system solvers. Specifically, by choosing initial state $|b\rangle$ itself as the preconditioner, we develop a simple quantum linear system algorithm that makes $\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\right)$ queries to both O_A and O_b. Our method appears to be conceptually even simpler than the recent kernel-reflection approach [19]. We also develop a block-encoded quantum eigenvalue transformation algorithm with $\mathbf{O}(n)$ scaling in degree of the target polynomial, by applying the preconditioning technique within the block-encoded version of quantum linear system algorithm, whereas the best previous result was $\mathbf{O}(n^{1.5})$. We illustrate our results in Figure 1 and tabulate them in Table 1 and Table 2 against previous results.

1.3 Tunable variable time amplitude amplification

Our first main technical contribution is the analysis of a generic version of VTAA in Section 3, with adjustable amplification thresholds that can be tuned/optimized for different input algorithms and initial states. We name this method *Tunable VTAA*.

Before discussing Tunable VTAA in detail, let us first review the VTAA framework introduced by Ambainis [2]. The goal here is to construct a quantum state by applying a sequence of quantum algorithms $A_1, \ldots, A_m$ on a starting state $|\psi_0\rangle$ incrementally, where the desired state resides in a subspace flagged by the *flag projection* $\overline{\Pi_b} = I - \Pi_b$. If we apply the quantum algorithms naively, then the state is prepared with probability

$$p_{\text{succ}} = \left\| \overline{\Pi_b} A_m \cdots A_1 |\psi_0\rangle \right\|^2 . \tag{6}$$

To prepare it with probability close to unity, we perform $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$ rounds of amplitude amplification, leading to a total cost of

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(|\psi_0\rangle\right)+\frac{1}{\sqrt{p_{\text{succ}}}}\sum_{j=1}^{m}\mathbf{Cost}\left(A_j\right)\right). \tag{7}$$

This is a worst-case scenario as all input algorithms $A_1,\ldots,A_m$ are treated on equal footing and get invoked the same number of times in the final stage. VTAA avoids this worst-case cost by performing amplitude amplification at intermediate stages. Formally, we have $m+1$ orthogonal projections Π_j $(j=0,\ldots,m)$, all commuting with Π_b and partially ordered according to their positive semidefinitess as $0=\Pi_0\leq\Pi_1\leq\cdots\leq\Pi_m=I$. These *clock projections* are used to represent that quantum algorithms can terminate early at or before intermediate stages, i.e., we have $A_j\Pi_{j-1}=\Pi_{j-1}$ for all $j=1,\ldots,m$. Then instead of deferring the amplification to the very end, we perform r_j rounds of amplitude amplification at stage j toward the potentially good subspace flagged by $\overline{\Pi_j\Pi_b}=I-\Pi_j\Pi_b$. This means $\widetilde{A}_j=\left(-\left(I-2A_j\widetilde{A}_{j-1}|\psi_0\rangle\langle\psi_0|\widetilde{A}_{j-1}^{\dagger}A_j^{\dagger}\right)\left(I-2\overline{\Pi_j\Pi_b}\right)\right)^{r_j}A_j\widetilde{A}_{j-1}$, where $\widetilde{A}_0=I$. We call $2r_j+1$ the *amplification schedules* or *amplification step numbers*.

To ensure that $\widetilde{A}_m$ produces the desired state with a sufficiently large probability, one needs to carefully choose r_j to avoid under/over amplification. In the work of [2] and many of its followups, the schedule is chosen as follows: set r_j to be the smallest nonnegative integer satisfying $(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\geq\frac{1}{3\sqrt{m}}$. Then using the state-of-the-art VTAA analysis, this gives an algorithm with query complexity

$$\mathbf{O}\left(\frac{\sqrt{m}}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(|\psi_0\rangle\right)+\frac{\sqrt{m}}{\sqrt{p_{\text{succ}}}}\sum_{j=1}^{m}\left\|\overline{\Pi_{j-1}\Pi_b}A_{j-1}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_j\right)\right). \tag{8}$$

Compared to the naive amplification cost Eq. (7), VTAA makes fewer queries to the input algorithms by balancing the summation of cost. Note the above discussion implicitly assumes that the amplitudes $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|$ are known a prior. This assumption is only for presentational purpose: asymptotic scaling of the complexity remains the same if we have a constant multiplicative estimate of the amplitudes instead. Otherwise, we need to perform amplitude estimation to compute each $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|$ to a constant multiplicative accuracy, incurring a substantial overhead—although this only needs to be done once during compilation.

In contrast, our Tunable VTAA uses the following schedule: set r_j to be the smallest nonnegative integer satisfying $(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\geq\frac{1}{3}\sqrt{\alpha_j}$. Here α_j are *amplification thresholds* satisfying the technical conditions $0\leq\alpha_j\leq 1$ and $\sum_{j=1}^{m}\alpha_j=\mathbf{O}(1)$. Obviously, Tunable VTAA offers more flexibility as α_j can now be tuned/optimized for different input algorithms and initial states, although it may not be immediately apparent how much more powerful it is over previous results. We prove the following.

(i) Tunable VTAA consists of nested amplitude amplifications with $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\leq\frac{1}{2r_j+1}$ and constant *loss factor* $\prod_{j=1}^{m}\frac{\left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\|}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}=\mathbf{\Omega}(1)$. Conversely, any nested amplitude amplifications with no overshoot $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\leq\frac{1}{3(2r_j+1)}$ and constant loss factor $\prod_{j=1}^{m}\frac{\left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\|}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}=\mathbf{\Omega}(1)$ is a Tunable VTAA.

(ii) Nontrivial amplifications $r_j \geq 1$ in Tunable VTAA happen only at l stages $1 \leq s_1 \leq \cdots \leq s_l \leq m$, where

$$l = \mathbf{O}\left(\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right). \tag{9}$$

Under the convention $s_{l+1} = m+1$ and $A_{m+1} = I$, Tunable VTAA has query complexity

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(A_{s_1}\cdots A_1|\psi_0\rangle) + \frac{1}{\sqrt{p_{\text{succ}}}}\sum_{v=2}^{l+1}\frac{1}{\sqrt{\alpha_{s_{v-1}}}}\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right), \tag{10}$$

(iii) Pre-merging trivial stages and using thresholds

$$\alpha_{s_{v-1}} \propto \left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}, \tag{11}$$

Tunable VTAA attains the $\ell_{\frac{2}{3}}$-*quasinorm* scaling

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(A_{s_1}\cdots A_1|\psi_0\rangle) + \frac{1}{\sqrt{p_{\text{succ}}}}\left(\sum_{v=2}^{l+1}\left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}\right)^{\frac{3}{2}}\right). \tag{12}$$

Let us interpret this result. Property (i) shows that Tunable VTAA is universal, in the sense that it captures the full power of a generic nested amplitude amplification algorithm. Stated differently, to improve the complexity of a nested amplification, it suffices to optimize over the threshold values. Such an exhaustive optimization can be computationally demanding, but one may find suboptimal thresholds that have analytic forms. In any event, it is this universality that motivates us to examine Tunable VTAA in greater detail. Property (ii) demonstrates that the number of nontrivial amplification stages is at most logarithmic in the inverse success amplitude. Thus in the regime where $\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right) \ll m$, majority of the input algorithms can be pre-merged and VTAA can be significantly simplified. Our query complexity then follows from a tightened analysis of VTAA with a $\sqrt{m}$ factor improvement, taking into account the fact that trivial stages can be pre-merged. To compile this into a quantum algorithm, one would naively estimate all $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|$ using amplitude estimation and compare with threshold values to determine the schedule, but this is not really necessary if the amplification thresholds are chosen analytically, as we will show for the quantum linear system problem. Property (iii) follows by minimizing the cost of Tunable VTAA under the constraints $0 \leq \alpha_{s_{v-1}} \leq 1$ and $\sum_{v=2}^{l+1}\alpha_{s_{v-1}} = \mathbf{O}(1)$. This $\ell_{\frac{2}{3}}$-quasinorm scaling is lower than the ℓ_1- or ℓ_2-norm result from previous work as

$$\begin{aligned}&\left(\sum_{v=2}^{l+1}\left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}\right)^{\frac{3}{2}}\\ &\leq \sqrt{l}\sum_{v=2}^{l+1}\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right).\end{aligned} \tag{13}$$

Again, to achieve this "ultimate performance" of VTAA with a quantum algorithm, one needs to estimate all potentially good probabilities $\left\|\overline{\Pi_{j-1}\Pi_b}A_{j-1}\cdots A_1|\psi_0\rangle\right\|$ with logarithmic overhead, so this is more advantageous in cases where the algorithm may be repeatedly invoked after compilation.

1.4 Discretized inverse state

For the quantum linear system problem, our goal is to produce a normalized version of the solution $A^{-1}|b\rangle$. Here, the coefficient matrix is block encoded as A/α_A with normalization factor $\alpha_A \geq \|A\|$, and a norm upper bound on its inverse $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ is known a prior. By considering the Hermitian dilation $|0\rangle\langle 1| \otimes A + |1\rangle\langle 0| \otimes A^\dagger$, we may without loss of generality assume that A itself is Hermitian and has the spectral decomposition $A = \sum_u \lambda_u |\phi_u\rangle\langle\phi_u|$, where $\frac{1}{\alpha_{A^{-1}}} \leq |\lambda_u| \leq \alpha_A$. Meanwhile, the initial state can be expanded in the eigenbasis as $|b\rangle = \sum_u \gamma_u |\phi_u\rangle$, so a direct application of QSVT produces an unnormalized state close to

$$\frac{A^{-1}}{2\alpha_{A^{-1}}}|b\rangle = \sum_u \frac{1}{2\alpha_{A^{-1}}\lambda_u}\gamma_u|\phi_u\rangle, \qquad \frac{\left\|A^{-1}|b\rangle\right\|}{\alpha_{A^{-1}}} = \frac{1}{\alpha_{A^{-1}}}\sqrt{\sum_u \left|\frac{\gamma_u}{\lambda_u}\right|^2} = \sqrt{p_{\text{succ}}}. \tag{14}$$

In prior art [15], the solution is prepared by enlarging the Hilbert space and constructing an intermediate state of the form

$$\begin{aligned} \psi_{\text{inv}} &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{2^{k+1}}, \frac{1}{2^k}\right)} \frac{1}{\alpha_{A^{-1}}\lambda_u} \left(\zeta_{k+1,u}|k\rangle + \zeta_{k,u}|k-1\rangle\right)\gamma_u|\phi_u\rangle, \\ \|\psi_{\text{inv}}\| &= \frac{1}{\alpha_{A^{-1}}}\sqrt{\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{2^{k+1}}, \frac{1}{2^k}\right)} \left(\left|\zeta_{k+1,u}\right|^2 + \left|\zeta_{k,u}\right|^2\right)\left|\frac{\gamma_u}{\lambda_u}\right|^2} = \sqrt{p_{\text{succ,inv}}}. \end{aligned} \tag{15}$$

Here, the ancilla register can be thought of as a "clock register" holding $m = \mathbf{Ceil}\left(\log_2\left(\alpha_A\alpha_{A^{-1}}\right)\right) = \mathbf{\Theta}(\log(\kappa))$ values, and $|\zeta_{k+1,u}|^2 + |\zeta_{k,u}|^2 \lessapprox 1$ are close to being normalized for all u, implying $p_{\text{succ,inv}} = \mathbf{\Theta}(p_{\text{succ}})$. This intermediate state can in turn be created using VTAA with a constant success probability. Specifically, one chooses the clock projections to be $\Pi_j = \sum_{k=0}^{j-1}|k\rangle\langle k|$ for $j = 1, \ldots, m$, satisfying $0 \leq \Pi_1 \leq \Pi_2 \leq \cdots \leq \Pi_m = I$. Within the VTAA stage k, one performs a procedure called Gapped Phase Estimation (GPE) to approximately check whether the eigenvalues are in $\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{2^k}, 1\right)$ with a certain confidence level, inverting the matrix if the condition is satisfied and terminating the current branch of computation afterwards. Finally, the clock register can be uncomputed by running GPE in reverse (without VTAA), resulting in the desired $A^{-1}|b\rangle$.

Our second main technical contribution, to be detailed in Section 4, is the replacement of above state by the *discretized inverse state*

$$\begin{aligned} \psi_{\text{d-inv}} &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{\rho^{k+1}}, \frac{1}{\rho^k}\right)} \left(\zeta_{k+1,u}\frac{\rho^{k+1}}{\rho^m}|k\rangle + \zeta_{k,u}\frac{\rho^k}{\rho^m}|k-1\rangle\right)\gamma_u|\phi_u\rangle, \\ \|\psi_{\text{d-inv}}\| &= \sqrt{\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{\rho^{k+1}}, \frac{1}{\rho^k}\right)} \left(\left|\zeta_{k+1,u}\frac{\rho^{k+1}}{\rho^m}\right|^2 + \left|\zeta_{k,u}\frac{\rho^k}{\rho^m}\right|^2\right)|\gamma_u|^2} = \sqrt{p_{\text{succ,d-inv}}}, \end{aligned} \tag{16}$$

where ρ is an odd integer which can be chosen as $\rho = 3$ without loss of generality and $m = \mathbf{Ceil}\left(\log_3\left(\alpha_A\alpha_{A^{-1}}\right)\right) = \mathbf{\Theta}(\log(\kappa))$. Note that we have switched from a base-2 partition of the eigenvalues to an odd-number base partition. This change plays the central role in achieving the optimal initial state preparation cost while substantially simplifying the structure of VTAA. We will return to this point momentarily. The second change is we replace the eigenvalue inversion

$\frac{1}{\lambda_u}$ by the discrete values $\frac{3^{k+1}}{3^m}$ and $\frac{3^k}{3^m}$, which can be introduced using only controlled single-qubit rotations. This modification frees the expensive matrix inversion from any nested amplitude amplifications, further simplifying the algorithm. Otherwise, we still perform GPE in each stage k with cost $\mathbf{O}\left(3^k \log\left(\frac{1}{\epsilon_{\text{gpe}}}\right)\right)$. Since the eigenvalues are discretized with constant multiplicative accuracy, we have $p_{\text{succ,d-inv}} = \mathbf{\Theta}(p_{\text{succ}})$.

We now explain how to achieve optimal initial state preparation cost with a simple VTAA schedule. We first recall from Property (ii) that nontrivial amplifications happen only at $\mathbf{O}\left(\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right)$ stages of VTAA, and majority of the algorithms can be pre-merged. In our case, the complexity 3^k increases exponentially in k, so intuitively one should pre-merge the initial stages with lower cost. We rigorize this idea by showing that the asymptotic complexity of VTAA remains unaffected after pre-merging the first $m-l$ stages, as long as $l = \mathbf{\Omega}\left(\log_3\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right)$. For the quantum linear system problem, the $\ell_{\frac{2}{3}}$-quasinorm from Property (iii) can be difficult to evaluate directly, so we relax it to the ℓ_2-norm, which is attained by setting the amplification thresholds $\alpha_k \propto 9^k \left\|\overline{\Pi_k \Pi_b} A_k \cdots A_1 |\psi_0\rangle\right\|^2$. These values sum up to $\sum_{k=m-l+1}^{m} 9^k \left\|\overline{\Pi_k \Pi_b} A_k \cdots A_1 |\psi_0\rangle\right\|^2 = \mathbf{O}\left(9^m p_{\text{succ}}\right)$, so we should choose $\alpha_k = \mathbf{\Theta}\left(\frac{9^k \left\|\overline{\Pi_k \Pi_b} A_k \cdots A_1 |\psi_0\rangle\right\|^2}{9^m p_{\text{succ}}}\right)$ to fulfill the requirement $\sum_k \alpha_k = \mathbf{O}(1)$. With the ℓ_2-norm cost, the Tunable VTAA makes $\mathbf{O}\left(l 3^m \log\left(\frac{1}{\epsilon_{\text{gpe}}}\right)\right)$ queries to the block encoding oracle O_A. This cost can be minimized by reducing l, but recall that we need l sufficiently large due to pre-merging, so we should choose $l = \mathbf{\Theta}\left(\log_3\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right)$. This implies $p_{\text{succ}} = \mathbf{\Theta}\left(9^{-l}\right)$ and hence the thresholds

$$\alpha_k = \begin{cases} \mathbf{\Theta}\left(9^{k-m+l} \left\|\overline{\Pi_k \Pi_b} A_k \cdots A_1 |\psi_0\rangle\right\|^2\right), & k = m-l+1, \ldots, m, \\ 0, & k = 1, \ldots, m-l. \end{cases} \tag{17}$$

Property (i) provides the theoretical guarantee that the above Tunable VTAA can be translated into a nested amplitude amplification. To realize this, one would naively perform amplitude estimation to determine all thresholds α_k to a constant multiplicative accuracy, which introduces polylogarithmic overhead and ruins the optimal scaling of initial state preparation. But this is unnecessary overkill. Suppose we have adjusted the constant factor so that $\alpha_k = c^2 9^{k-m+l} \left\|\overline{\Pi_k \Pi_b} A_k \cdots A_1 |\psi_0\rangle\right\|^2$ for some $c \gtrapprox 1$ (say $c = 1.001$). Then at the first nontrivial stage $k = m-l+1$, we would compare $\left\|\overline{\Pi_{m-l+1} \Pi_b} A_{m-l+1} \cdots A_1 |\psi_0\rangle\right\|$ with $\frac{1}{3}\sqrt{\alpha_{m-l+1}} = c \left\|\overline{\Pi_{m-l+1} \Pi_b} A_{m-l+1} \cdots A_1 |\psi_0\rangle\right\|$. We find that our current amplitude falls just short of the threshold, concluding that we need $2r_{m-l+1}+1 = 3$ steps of amplitude amplification. After this stage, we have amplified the amplitude by 3 up to a loss factor. At the next stage $k = m-l+2$, our current amplitude $\left\|\overline{\Pi_{m-l+2} \Pi_b} A_{m-l+2} \widetilde{A}_{m-l+1} |\psi_0\rangle\right\| \lessapprox 3 \left\|\overline{\Pi_{m-l+2} \Pi_b} A_{m-l+2} \cdots A_1 |\psi_0\rangle\right\|$ is slightly below $\frac{1}{3}\sqrt{\alpha_{m-l+2}} = c3 \left\|\overline{\Pi_{m-l+2} \Pi_b} A_{m-l+2} \cdots A_1 |\psi_0\rangle\right\|$, so we conclude once more we need $2r_{m-l+2}+1 = 3$ steps of amplitude amplification, which boosts the amplitude again by 3 up to a loss factor. Proceeding in this way, we obtain the following simple amplification schedule

$$2r_k + 1 = \begin{cases} 3, & k = m-l+1, \ldots, m, \\ 1, & k = 1, \ldots, m-l. \end{cases} \tag{18}$$

Note again that this *deterministic schedule* is derived entirely from the definition of Tunable VTAA, and the derivation requires *no* amplitude estimation. The amplification schedule Eq. (18) is reminiscent of another one used in the recent study of quantum search [3] (see also [14, 27, 46]). This is not a coincidence. For the quantum search algorithm of [3], the target cost function is the ℓ_2-norm of input cost, which grows exponentially in the stage number like $\sim 3^k$. Following an analysis

similar to above, one can show that Tunable VTAA reproduces the search algorithm developed in [3].

With the deterministic schedule, the GPE algorithm is invoked at most $m3^l$ times. To achieve accuracy ϵ, we choose $\epsilon_{\text{gpe}} = \frac{\epsilon}{m3^l}$, giving a VTAA algorithm with complexity $\mathbf{O}\big(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right) \log\left(\frac{\log(\kappa)}{\sqrt{p_{\text{succ}}}\epsilon}\right) \mathbf{Cost}(O_A)\big)$. By using a more complicated error schedule in the first $m - l$ stages and projecting error onto the potentially good subspaces, we further improve this result to

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right) \log\left(\frac{\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)}{\epsilon}\right) \mathbf{Cost}(O_A)\right). \tag{19}$$

This achieves an optimal cost of initial state preparation with a substantially simplified VTAA schedule Eq. (18). We summarize this algorithm in Theorem 2 of Section 5 and prove the optimality in Theorem 3.

Note that our above discussion assumes that we have a constant multiplicative approximation of $\sqrt{p_{\text{succ}}}$, or equivalently, the solution norm $\left\|A^{-1}|b\rangle\right\|$. If this information is not available, then we develop a solution norm estimation algorithm in Theorem 1 that estimates $\left\|A^{-1}|b\rangle\right\|$ to a constant multiplicative accuracy, by running VTAA with $l = 0, 1, 2, \ldots$ that terminates at $l = \mathbf{\Theta}\left(\log_3\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right)$ with high probability. This costs $\mathbf{O}\left(\frac{1}{\sqrt{\alpha_{p_{\text{succ}}}}}\right)$ queries to the initial state preparation oracle with $\alpha_{p_{\text{succ}}}$ a lower bound on the success probability, again achieving the strictly linear scaling.

1.5 Block preconditioning

As aforementioned, when applying the quantum linear system solvers, we are often in the intermediate regime where $1 \ll \frac{1}{\sqrt{p_{\text{succ}}}} = \frac{\alpha_{A^{-1}}}{\|A^{-1}|b\rangle\|} \ll \kappa = \alpha_A \alpha_{A^{-1}}$. Thus, the cost of querying O_b can be quite significant even with our optimal scaling algorithm. We now describe the third main contribution, a simple block preconditioning technique, that artifically boosts the amplitude of solution norm $\left\|A^{-1}|b\rangle\right\|$ and thereby reduces the complexity of initial state preparation. In fact, for solving differential equations and processing eigenvalues, the initial state query cost can even be reduced close to constant (in the evolution time and degree of the input polynomial). See Section 6 for further details.

Our block preconditioning technique targets at a subspace containing the initial state. Specifically, suppose that there is an orthogonal projection Π_{cond} whose image includes the initial state

$$\Pi_{\text{cond}}|b\rangle = |b\rangle. \tag{20}$$

Then we introduce the scaling operator

$$S = s\Pi_{\text{cond}} + (I - \Pi_{\text{cond}}), \qquad S^{-1} = \frac{1}{s}\Pi_{\text{cond}} + (I - \Pi_{\text{cond}}), \tag{21}$$

where $0 < s < 1$ is a scaling parameter that can be tuned for specific applications. Assuming Π_{cond} is accessible through an oracle reflecting along the image space $-\Pi_{\text{cond}} + (I - \Pi_{\text{cond}}) = I - 2\Pi_{\text{cond}}$, we can block encode S with normalization factor 1 through the linear combination $S = \frac{1-s}{2}(I - 2\Pi_{\text{cond}}) + \frac{1+s}{2}I$. This assumption is justified because in all our applications, we can efficiently implement unitaries V_{cond} such that $V_{\text{cond}}^\dagger (I - 2\Pi_{\text{cond}}) V_{\text{cond}}$ are reflections in the computational basis. Note it is also possible to make a weaker assumption that a block encoding of Π_{cond} is available, although this will increase the prefactor in the complexity of our algorithm.

Application	Improvement	Preconditioner	Previous scaling	New scaling
Linear system solver Theorem 4	Input matrix	Initial state	$\frac{\kappa}{\sqrt{p_{\text{succ}}}} \log\left(\frac{1}{\sqrt{p_{\text{succ}}}\epsilon}\right)$	$\kappa \log\left(\frac{1}{\epsilon}\right)$
Differential equation solver Theorem 5	Initial state	Ancilla state	$\frac{t}{\|e^{tA}\|b\rangle\|} \log\left(\frac{1}{\|e^{tA}\|b\rangle\|\epsilon}\right) \log\left(\frac{t}{\epsilon}\right)$	$\frac{1}{\|e^{tA}\|b\rangle\|}$
Eigenvalue estimator Theorem 6	Initial state	Ancilla state	$\frac{\kappa_S}{\epsilon}$	κ_S
Eigenvalue transformer Theorem 7	Initial state	Ancilla state	$\frac{\kappa_S^2 n \log(n)}{\left\|p\left(\frac{A}{\alpha_A}\right)\|\psi\rangle\right\|} \log\left(\frac{\kappa_S \log(n)}{\left\|p\left(\frac{A}{\alpha_A}\right)\|\psi\rangle\right\|\epsilon}\right)$	$\frac{\kappa_S \log(n)}{\left\|p\left(\frac{A}{\alpha_A}\right)\|\psi\rangle\right\|}$
Ground state preparator Theorem 8	Initial state	Ancilla state	$\frac{\kappa_S^2}{\|\gamma_0\|\delta_A} \log^2\left(\frac{\kappa_S}{\|\gamma_0\|\epsilon}\right)$	$\frac{\kappa_S}{\|\gamma_0\|} \log\left(\frac{1}{\delta_A} \log\left(\frac{\kappa_S}{\|\gamma_0\|\epsilon}\right)\right)$
Block-encoded eigenvalue transformer Theorem 9	Input matrix	Ancilla state	$\frac{n^{1.5}\kappa_S^2}{\left\|p\left(\frac{A}{\alpha_A}\right)\right\|} \log\left(\frac{\sqrt{n}\kappa_S}{\left\|p\left(\frac{A}{\alpha_A}\right)\right\|\epsilon}\right) \log\left(\frac{1}{\epsilon}\right)$	$\frac{n\kappa_S^2}{\left\|p\left(\frac{A}{\alpha_A}\right)\right\|} \log\left(\frac{\kappa_S}{\left\|p\left(\frac{A}{\alpha_A}\right)\right\|\epsilon}\right) \log\left(\frac{1}{\epsilon}\right)$

Table 2: Summary of applications of the block preconditioning technique. Using the initial ancilla state as the preconditioner, one can reduce the cost of initial state preparation in linear-system-based differential equation solvers, ground state preparators and eigenvalue processors, nearly matching or outperforming the state of the art. By choosing the initial state itself as the preconditioner and applying QSVT, one gets an extremely simple quantum linear system solver with optimal queries to the coefficient matrix. See the corresponding theorem statements for definitions of the parameters.

Now instead of $Ax = b$, we consider the preconditioned linear system $SAx = Sb$, which yields the original solution $\frac{(SA)^{-1}Sb}{\|(SA)^{-1}Sb\|} = \frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}$ when solved exactly. Here, the preconditioned coefficient matrix can be block encoded as $(SA)/\alpha_A$ with the same normalization as before, and the initial state $\frac{Sb}{\|Sb\|} = |b\rangle$ also remains the same. However, the solution norm now becomes

$$\left\|(SA)^{-1}|b\rangle\right\| = \left\|A^{-1}S^{-1}|b\rangle\right\| = \frac{1}{s}\left\|A^{-1}|b\rangle\right\|, \tag{22}$$

whereas the inverse matrix has a norm bounded by

$$\begin{aligned}\left\|(SA)^{-1}\right\| = \left\|A^{-1}S^{-1}\right\| &= \left\|\frac{1}{s}A^{-1}\Pi_{\text{cond}} + A^{-1}\left(I - \Pi_{\text{cond}}\right)\right\| \\ &\leq \sqrt{\frac{\left\|A^{-1}\Pi_{\text{cond}}\right\|^2}{s^2} + \left\|A^{-1}\left(I - \Pi_{\text{cond}}\right)\right\|^2} \leq \sqrt{\frac{\left\|A^{-1}\Pi_{\text{cond}}\right\|^2}{s^2\left\|A^{-1}\right\|^2} + 1}\left\|A^{-1}\right\|.\end{aligned} \tag{23}$$

So the condition number stays the same as long as $s = \mathbf{\Omega}\left(\frac{\left\|A^{-1}\Pi_{\text{cond}}\right\|}{\|A^{-1}\|}\right)$, but the solution norm will be artificially boosted by a factor of $\frac{1}{s}$ thanks to block preconditioning.

Preconditioning is a classical subject in numerical linear algebra, the goal of which is to transform linear system problems to improve their algorithmic solvability. Among the various preconditioning methods, the one closest to our work is the so-called matrix scaling (also known as matrix balancing or matrix equilibration); see the review article [28] and the references therein. Using that language, our result can be interpreted as follows. When partitioned with respect to the orthogonal decomposition $\mathbf{Im}\left(\Pi_{\text{cond}}\right) \oplus \mathbf{Im}\left(I - \Pi_{\text{cond}}\right)$, operators A^{-1} and S^{-1} take the form

$$A^{-1} = \begin{bmatrix} A^{-1}\big|_{\mathbf{Im}(\Pi_{\text{cond}})} & A^{-1}\big|_{\mathbf{Im}(I-\Pi_{\text{cond}})} \end{bmatrix}, \qquad S^{-1} = \begin{bmatrix} \frac{1}{s}\, I\big|_{\mathbf{Im}(\Pi_{\text{cond}})} & 0 \\ 0 & I\big|_{\mathbf{Im}(I-\Pi_{\text{cond}})} \end{bmatrix}. \tag{24}$$

Here, the block $A^{-1}\big|_{\mathbf{Im}(\Pi_{\text{cond}})}$ is of most interest to us, because it completely determines the action of A^{-1} on a subspace containing the initial state $|b\rangle$. The issue is that $\left\|A^{-1}\big|_{\mathbf{Im}(\Pi_{\text{cond}})}\right\| \ll \left\|A^{-1}\right\|$ can

often happen in practice, so we are inverting a coefficient matrix with a large condition number, while the initial state actually resides in a subspace with a relatively small condition number. To remedy this issue, we use the scaling matrix to boost the block $\frac{1}{s}\left\|A^{-1}\big|_{\mathbf{Im}(\Pi_{\text{cond}})}\right\| \approx \left\|A^{-1}\right\|$ without significantly increasing the overall condition number, leading to faster quantum linear system solvers; hence the name *block preconditioning*. It is worth mentioning that preconditioning has also been explored in previous quantum work such as [16, 49, 56], although their focus is on constructing preconditioners for linear systems with specific structures, as opposed to the generic linear system problem considered here.

As an immediate application, let us consider what if we choose the initial state itself as the preconditioner $\Pi_{\text{cond}} = |b\rangle\langle b|$. Then, the coefficient matrix

$$SA = \left(s|b\rangle\langle b| + (I - |b\rangle\langle b|)\right) A \tag{25}$$

can be block encoded with normalization factor α_A, using 1 query to O_A and 2 queries to O_b and its inverse. Given a constant multiplicative approximation of $\left\|A^{-1}|b\rangle\right\|$, we set

$$s = \mathbf{\Theta}\left(\frac{\left\|A^{-1}|b\rangle\right\|}{\alpha_{A^{-1}}}\right) = \mathbf{\Theta}\left(\sqrt{p_{\text{succ}}}\right). \tag{26}$$

This implies the scaling of solution norm

$$\left\|(SA)^{-1}|b\rangle\right\| = \frac{1}{s}\left\|A^{-1}|b\rangle\right\| = \mathbf{\Theta}\left(\alpha_{A^{-1}}\right), \tag{27}$$

and norm bound on the inverse coefficient matrix

$$\left\|(SA)^{-1}\right\| \leq \sqrt{\frac{\left\|A^{-1}|b\rangle\right\|^2}{s^2\left\|A^{-1}\right\|^2} + 1}\left\|A^{-1}\right\| = \mathbf{O}\left(\alpha_{A^{-1}}\right). \tag{28}$$

Therefore, by applying QSVT to the preconditioned problem, we obtain an extremely simple quantum linear system algorithm with complexity

$$\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\epsilon}\right)\mathbf{Cost}(O_A)\right). \tag{29}$$

The result appears to be conceptually even simpler than the recent kernel-reflection method, which instead solves a padded linear system over an expanded Hilbert space [19]. Of course, one needs to estimate the solution norm to a constant multiplicative accuracy before running either approach. This can be achieved using either the method of [19] which makes optimal queries to the coefficient matrix, or our Tunable VTAA which makes optimal queries to the initial state preparation and nearly optimal uses of the coefficient matrix.

Observe that the preconditioned coefficient matrix SA can be block encoded with no query overhead, if we choose the initial ancilla state as the preconditioner. We utilize this observation to reduce the cost of initial state preparation in differential equation solvers, ground state preparators and eigenvalue processors, nearly matching or outperforming the state of the art. This refutes the widely held beliefs that linear-system-based differential equation solvers and eigenvalue estimators necessarily require more queries to the initial state oracles than their alternatives [4, 20, 60]. Of independent interest, we also get a block-encoded eigenvalue transformation algorithm with $\mathbf{O}(n)$ scaling in degree of the target polynomial, whereas the best previous result was $\mathbf{O}(n^{1.5})$. See Table 2 for a summary of applications of the block preconditioning technique.

We summarize in Section 2 preliminaries required to understand our algorithms, and include in Section 7 a brief recap of the main result and a collection of questions for future work.

2 Preliminaries

This section contains preliminaries useful for understanding our quantum linear system algorithms and their applications. We provide common notation and terminology in Section 2.1 to be used throughout the paper. We then review known results on VTAA in Section 2.2. Unlike prior work, we introduce an axiomatic formulation of VTAA which is not bonded by any concrete circuit implementation and offers more flexibility. We refer the reader to [39, Section 2.4] for further preliminaries about the block encoding framework, within which our algorithms are developed.

2.1 Notation and terminology

We use lowercase Latin and Greek alphabets to denote scalars, vectors and functions. For instance, we often use ϵ for the accuracy of algorithm, κ for a known upper bound on the condition number, p_{succ} for success probability of the input algorithm of VTAA or the probability of directly inverting the input matrix with block encoding, m for the number of VTAA stages, and r for the number of amplification steps/schedules. We also represent known upper/lower bounds of quantities or amplification threshold values using α, writing $\alpha_A \geq \|A\|$ for a known upper bound on the spectral norm of input matrix A for block encoding, $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ for a known upper bound on the norm of A^{-1} with $\kappa = \alpha_A \alpha_{A^{-1}}$, $\alpha_{p_{\text{succ}}} \leq p_{\text{succ}}$ a known lower bound on the success probability, and α_j amplification thresholds associated with a Tunable VTAA. We employ standard notations for number sets, using $\mathbb{Z}$ for integers ($\mathbb{Z}_{\geq 0}$ for nonnegative integers), $\mathbb{R}$ for real numbers, and $\mathbb{C}$ for complex numbers.

We user uppercase Latin and Greek alphabets to represent matrices and operators. For example, we write A for the coefficient matrix of a linear system, O_A for its block encoding unitary, O_b for the unitary preparing the initial state, A_j ($j = 1, \ldots, m$) for the input algorithms of VTAA, Π for an orthogonal projection ($\overline{\Pi} = I - \Pi$ for its complement), S for the scaling operator used for preconditioning, and I, X, Y, Z for the identity and Pauli matrices. We use $\|A\|$ to denote the spectral norm of A, i.e., its largest singular value. For a matrix A and scalar $\alpha_A \geq 0$, we say A/α_A can be block encoded if there exists an isometry G and unitary U such that $\frac{A}{\alpha_A} = G^\dagger U G$. Such a *block encoding* is mathematically feasible if and only if $\alpha_A \geq \|A\|$, but additional normalization factors may be introduced when the block encoding is implemented by a quantum circuit.

We use boldface symbols to denote functions and operations having specific meanings. We write $\mathbf{Cost}(\cdot)$ for the (query) cost of implementing an operator, $\mathbf{Ceil}(\cdot)/\mathbf{Floor}(\cdot)$ for the nearest integer rounded up/down, $\mathbf{Ker}(\cdot)/\mathbf{Im}(\cdot)$ for the kernel/image of a linear operator, $\mathbf{P}(\cdot)$ for the probability of an event, and $\mathbf{T}_j(\cdot)/\mathbf{U}_j(\cdot)$ for Chebyshev polynomials of the first/second kind (writing its rescaled version as $\widetilde{\mathbf{T}}_j(\cdot)$ with $\widetilde{\mathbf{T}}_0 = \frac{1}{2}$). If Π is an orthogonal projection, $\mathbf{Im}(\Pi) \oplus \mathbf{Im}(\overline{\Pi})$ is an orthogonal decomposition of the underlying space and we may use it to bound the spectral norm of an operator as

$$\|A\| = \sqrt{\|AA^\dagger\|} \leq \sqrt{\|A\Pi A^\dagger\| + \left\|A\overline{\Pi}A^\dagger\right\|} = \sqrt{\|A\Pi\|^2 + \left\|A\overline{\Pi}\right\|^2}. \tag{30}$$

We write $\mathbf{O}(\cdot)$ to mean asymptotically less than, $\mathbf{\Omega}(\cdot)$ to mean asymptotically more than, and $\mathbf{\Theta}(\cdot)$ to represent quantities having the same asymptotic scaling. We let a summation be zero and a product by one if their lower limits exceed upper limits.

In analyzing the performance of Tunable VTAA, we will need to evaluate

$$\|\beta\|_p = \left(\sum_{j=1}^{n} |\beta_j|^p\right)^{\frac{1}{p}}, \qquad p \in (0, +\infty] \tag{31}$$

for vectors $\begin{bmatrix}\beta_1 & \cdots & \beta_n\end{bmatrix} \in \mathbb{C}^n$. When $1 \le p \le \infty$, $\|\cdot\|_p$ is the ℓ_p-norm and its properties are summarized in standard references such as [26, Chapter 5]. In particular, the case $p = 2$ corresponds to the Euclidean norm and we may drop the subscript since it is the same as spectral norm when we equate vectors with column matrices. However if $0 < p < 1$, $\|\cdot\|_p$ is no longer a vector norm and its properties may be less familiar. In this case, $\|\cdot\|_p$ is known as a *quasinorm* [17, 44], which satisfies the following defining properties.

(i) $\|\beta\|_p \ge 0$ for all $\beta \in \mathbb{C}^n$, with $\|\beta\|_p = 0$ if and only if $\beta = 0$.

(ii) $\|c\beta\|_p = |c| \, \|\beta\|_p$ for all $\beta \in \mathbb{C}^n$ and $c \in \mathbb{C}$.

(iii) $\|\beta + \gamma\|_p \le 2^{\frac{1}{p}-1} \left(\|\beta\|_p + \|\gamma\|_p \right)$ for all $\beta, \gamma \in \mathbb{C}^n$.

The factor $2^{\frac{1}{p}-1}$ in Property (iii) is the best one can get for a modified triangle inequality $\|\beta + \gamma\|_p \le c \left(\|\beta\|_p + \|\gamma\|_p \right)$ with some universal constant $c \ge 0$ for all $\beta, \gamma \in \mathbb{C}^n$. In fact, if β and γ are entry-wise nonnegative, it holds the reverse Minkowski's inequality $\|\beta + \gamma\|_p \ge \|\beta\|_p + \|\gamma\|_p$. Additionally, when $0 < p < q \le \infty$,

$$\|\beta\|_q \le \|\beta\|_p \le n^{\frac{1}{p}-\frac{1}{q}} \|\beta\|_q , \tag{32}$$

and $n^{\frac{1}{p}-\frac{1}{q}}$ is again the best constant one can hope for.

We use the Dirac notation $|\psi\rangle$ for a vector only when it is normalized with respect to the Euclidean norm $\||\psi\rangle\| = 1$. For general nonzero vectors ψ and ϕ, we define

$$\psi \propto \phi \quad \Leftrightarrow \quad \frac{\psi}{\|\psi\|} = \frac{\phi}{\|\phi\|} \quad \Leftrightarrow \quad \psi = c\phi \ \exists c > 0. \tag{33}$$

That is, ψ is proportional to ϕ if and only if they agree up to a positive rescaling, whereas $|\psi\rangle \propto |\phi\rangle \Leftrightarrow |\psi\rangle = |\phi\rangle$ holds for unit vectors. It can be verified that proportionality is reflexive, symmetric, and transitive. Moreover,

$$\psi \propto \phi \quad \Rightarrow \quad B\psi \propto B\phi \tag{34}$$

for an operator B whenever the composition makes sense.

Finally, we say $u > 0$ is a *c-multiplicative approximation* of $v > 0$ if

$$\frac{1}{c} \le \frac{u}{v} \le c \tag{35}$$

for some $c \ge 1$. It is apparent that this relation is reflexive, symmetric, and invariant under reciprocal: any $u > 0$ is a 1-multiplicative approximation of itself; if $u > 0$ is a c-multiplicative approximation of $v > 0$, then v is a c-multiplicative approximation of u, and $\frac{1}{u}$ is a c-multiplicative approximation of $\frac{1}{v}$. Moreover, if u_1, u_2 are c_1-, c_2-multiplicative approximations of v_1, v_2 respectively, their product $u_1 u_2$ is a $(c_1 c_2)$-multiplicative approximation of $v_1 v_2$. When analyzing VTAA, we often consider the case where c is constant. But there may also be scenarios where we want $1 - \epsilon \le \frac{u}{v} \le \frac{1}{1-\epsilon}$ for some small $0 < \epsilon < 1$. This then leads to a simple product rule:

$$1 - \epsilon_1 \le \frac{u_1}{v_1} \le \frac{1}{1-\epsilon_1}, \quad 1 - \epsilon_2 \le \frac{u_2}{v_2} \le \frac{1}{1-\epsilon_2} \quad \Rightarrow \quad 1 - (\epsilon_1 + \epsilon_2) \le \frac{u_1 u_2}{v_1 v_2} \le \frac{1}{1-(\epsilon_1+\epsilon_2)}. \tag{36}$$

By a possible rescaling of ϵ, we can also consider alternative definitions such as $\frac{1}{1+\epsilon} \le \frac{u}{v} \le 1 + \epsilon$ and $1 - \epsilon \le \frac{u}{v} \le 1 + \epsilon$, which relate more closely to additive approximations as

$$1 - \epsilon \le \frac{u}{v} \le 1 + \epsilon \quad \Leftrightarrow \quad |u - v| \le v\epsilon. \tag{37}$$

2.2 Variable time amplitude amplification

We now formally introduce variable time amplitude amplification, and review known results about this framework [2, 12], using an axiomatic formulation not bonded by its circuit implementation. We assume throughout this subsection that an underlying Hilbert space $\mathcal{H}$ has been fixed on which all operators act.

Definition 1 (Variable time algorithm and amplification)**.** *A variable time quantum algorithm is a* 3*-tuple* $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m\right)$ *satisfying the following axioms.*

(i) Π_j *are orthogonal projections partially ordered as* $0 = \Pi_0 \leq \Pi_1 \leq \cdots \leq \Pi_m = I$*.*

(ii) Π_b *is an orthogonal projection commuting with all* Π_j*:* $\Pi_b\Pi_j = \Pi_j\Pi_b$ *for* $j = 0, \ldots, m$*.*

(iii) A_j *are unitaries such that* $A_j\Pi_{j-1} = \Pi_{j-1}$ *for all* $j = 1, \ldots, m$*, and* $A_0 = I$*.*

A variable time amplification algorithm is a 5*-tuple* $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m, \{\widetilde{A}_j\}_{j=0}^m, |\psi_0\rangle\right)$ *that additionally satisfies*

(iv) $\widetilde{A}_j$ *are unitaries such that* $\frac{\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle}{\left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\|} = \frac{\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle}{\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}$ *for all* $j = 1, \ldots, m$*, and* $\widetilde{A}_0 = I$*.*

The above definition deserves a few remarks. We call Π_j $(j = 0, \ldots, m)$ the *clock projections*. As Π_j are Hermitian, they can be partially ordered according to the positive semidefinitess, and Axiom (i) requires that $\Pi_j \leq \Pi_{j+1}$ hold true for all $j = 0, \ldots, m-1$. In Appendix A.1, we present a number of equivalent characterizations of the partial ordering $\Pi_j \leq \Pi_{j+1}$, showing in particular that $\Pi_j = \Pi_{j+1}\Pi_j = \Pi_j\Pi_{j+1}$. As an immediate corollary, we have

$$\Pi_j = \Pi_k\Pi_j = \Pi_j\Pi_k, \qquad 0 \leq j \leq k \leq m \tag{38}$$

and $\Pi_k - \Pi_j$ are themselves orthogonal projections. In a variable time quantum algorithm, $\mathbf{Im}(\Pi_j)$ represents the part of space in which the algorithm stops running before or at stage j. The space $\mathbf{Im}(\Pi_j)$ monotonically increases with k, echoing the fact that more branches of the quantum algorithm will halt as the computation proceeds toward completion.

We call Π_b and its complement $\overline{\Pi_b} = I - \Pi_b$ the *flag projections*. In the context of variable time amplification, $\mathbf{Im}(\overline{\Pi_b})$ is the part of space where the desired output state resides, corresponding to the success of algorithm. Axiom (ii) requires that Π_b commutes with all Π_j, so these projections may be simultaneously measured in a quantum computation. In Appendix A.2, we tabulate the meaning of different outcomes from the simultaneous measurement of $\{\Pi_j - \Pi_{j-1}\}_{j=1}^m$ and $\{\Pi_b, I - \Pi_b\}$. Of particular interest is the outcome corresponding to $\overline{\Pi_j\Pi_b}$, which represents that the computation can potentially succeed at stage j. These potentially good projections form a monotonically decreasing sequence

$$I = \overline{\Pi_0\Pi_b} \geq \overline{\Pi_1\Pi_b} \geq \cdots \geq \overline{\Pi_m\Pi_b} = \overline{\Pi_b}, \tag{39}$$

and they satisfy

$$\overline{\Pi_k\Pi_b} = \overline{\Pi_j\Pi_b} \cdot \overline{\Pi_k\Pi_b} = \overline{\Pi_k\Pi_b} \cdot \overline{\Pi_j\Pi_b}, \qquad 0 \leq j \leq k \leq m. \tag{40}$$

We call A_j the *input algorithms*, which act trivially on $\mathbf{Im}(\Pi_{j-1})$ as is required by Axiom (iii). In Appendix A.3, we present a number of equivalent characterizations of the property $A_j\Pi_{j-1} = \Pi_{j-1}$, finding that $A_j = \Pi_{j-1} + \overline{\Pi_{j-1}}A_j\overline{\Pi_{j-1}}$ are necessarily controlled unitaries controlled by $\overline{\Pi_{j-1}}$. This implies

$$A_j\Pi_l = \Pi_l = \Pi_l A_j, \qquad A_j\overline{\Pi_l\Pi_b} = \overline{\Pi_l\Pi_b}A_j, \qquad 0 \leq l < j \leq m. \tag{41}$$

As an immediate consequence of this property, we have that the potentially good amplitudes

$$\left\|\overline{\Pi_j\Pi_b}A_j\cdots A_1|\psi_0\rangle\right\| = \left\|\overline{\Pi_j\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\|, \qquad j=0,\ldots,m \tag{42}$$

form a monotonic sequence decreasing from 1 to $\sqrt{p_{\text{succ}}}$:

$$1 = \left\|\overline{\Pi_0\Pi_b}|\psi_0\rangle\right\| \geq \left\|\overline{\Pi_1\Pi_b}A_1|\psi_0\rangle\right\| \geq \cdots \geq \left\|\overline{\Pi_m\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\| = \left\|\overline{\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\| = \sqrt{p_{\text{succ}}}. \tag{43}$$

Finally, we call $\widetilde{A}_j$ the *amplified algorithms*. At each stage j, we would like the amplification to preserve the part of state potentially leading to success, while suppressing the part that has already resulted in failure. This requirement is captured by Axiom (iv), which can be succinctly represented as $\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle \propto \overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle$. We then prove in Appendix A.4 that

$$\frac{\left\|\overline{\Pi_h\Pi_b}A_h\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|} = \frac{\left\|\overline{\Pi_h\Pi_b}A_h\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle\right\|}, \qquad 0\leq l,j\leq k,h\leq m. \tag{44}$$

That is, the transition of potentially good amplitudes remains the same, regardless of whether we consider the pre- or post-amplified algorithms.

Until this point, we have not specified the amplification algorithms $\widetilde{A}_j$ other than the requirement that they should preserve the potentially good outcomes. In what follows, we consider the case where $\widetilde{A}_j$ are constructed by the standard amplitude amplification toward the potentially good subspaces $\widetilde{A}_j = \left(-\left(I-2A_j\widetilde{A}_{j-1}|\psi_0\rangle\langle\psi_0|\widetilde{A}_{j-1}^\dagger A_j^\dagger\right)\left(I-2\overline{\Pi_j\Pi_b}\right)\right)^{r_j}A_j\widetilde{A}_{j-1}$. Here, r_j are nonnegative integers and we call $2r_j+1$ *amplification schedules* or *amplification step numbers*. The query complexity of $\widetilde{A}_j$ then satisfies the recurrence

$$\mathbf{Cost}\left(\widetilde{A}_j|\psi_0\rangle\right) = (2r_j+1)\left(\mathbf{Cost}(A_j)+\mathbf{Cost}\left(\widetilde{A}_{j-1}|\psi_0\rangle\right)\right), \qquad j=1,\ldots,m. \tag{45}$$

For a given choice of r_j, we can unwrap the recursion to obtain a *variable time nested amplitude amplification* $\widetilde{A}_m$, which has a query cost of

$$\mathbf{Cost}\left(\widetilde{A}_m|\psi_0\rangle\right) = \mathbf{Cost}(|\psi_0\rangle)\prod_{k=1}^{m}(2r_k+1) + \sum_{j=1}^{m}\mathbf{Cost}(A_j)\prod_{k=j}^{m}(2r_k+1). \tag{46}$$

Formally:

Definition 2 (Variable time nested amplitude amplification)**.** *A variable time nested amplitude amplification is a* 5*-tuple* $\left(\{\Pi_j\}_{j=0}^m,\Pi_b,\{A_j\}_{j=0}^m,\{r_j\}_{j=1}^m,|\psi_0\rangle\right)$ *that additionally satisfies*

(iv′) r_j *are nonnegative integers for* $j=1,\ldots,m$*, which define*

$$\widetilde{A}_j = \begin{cases}\left(-\left(I-2A_j\widetilde{A}_{j-1}|\psi_0\rangle\langle\psi_0|\widetilde{A}_{j-1}^\dagger A_j^\dagger\right)\left(I-2\overline{\Pi_j\Pi_b}\right)\right)^{r_j}A_j\widetilde{A}_{j-1}, & j=1,\ldots,m,\\ I, & j=0.\end{cases} \tag{47}$$

When $(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\leq 1$, we have $(2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\right)\leq\frac{\pi}{2}$ and there is no over amplification/overshoot. Consequently, the pre- and post-amplified amplitudes of stage j are related by the analytic expression

$$\left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\| = \sin\left((2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\right)\right). \tag{48}$$

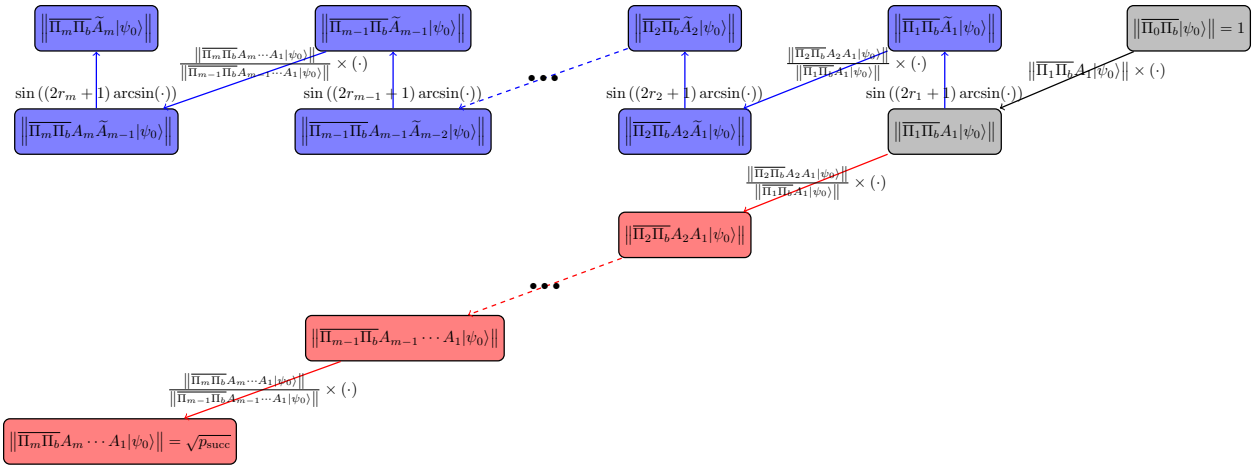


Figure 2: Illustration of the transition of amplitudes in a VTAA/nested amplitude amplification. The flow of algorithms is represented by arrows, with the input branch colored in red and amplified one in blue. Assuming there is no over amplification, for any pair of connected nodes, the amplitude from the bottom node is less than or equal to that from the top node.

Note we have almost amplified the amplitude by $2r_j+1$, up to the *loss factor*

$$\frac{\sin\left((2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\right)\right)}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}. \tag{49}$$

Carrying out this analysis to all the stages, we have the total loss factor

$$\prod_{j=1}^{m}\left(\frac{\sin\left((2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\right)\right)}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}\right). \tag{50}$$

See Figure 2 for an illustration of the amplitude transitions in a nested amplitude amplification.

In Ambainis' *Variable Time Amplitude Amplification* (VTAA) [2], the schedules are determined as follows: we choose r_j as the smallest nonnegative integer satisfying $(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\| \geq \frac{1}{3\sqrt{m}}$. This choice of r_j necessarily means that $(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\| \leq \frac{1}{\sqrt{m}}$, and one can show via an induction that there is no over amplification at all stages. On the other hand, we have $\sum_{j=1}^{m}(2r_j+1)^2\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|^2 \leq 1$, which yields $\prod_{j=1}^{m}\left(\frac{\sin\left((2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\right)\right)}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}\right) = \boldsymbol{\Omega}(1)$. The conclusion is that Ambainis' VTAA is a nested amplitude amplification with no overshoot and constant loss factor. Note our above definition of r_j uses the precise value of the amplitude $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|$. This is only for presentational purpose. In practice, one would estimate it to a constant muliplicative precision by running VTAA up to stage $j-1$ [12, Theorem 23]. Then one compares the upper bound with the threshold value to determine the amplification schedule, which enlarges the loss factor by only a constant amount.

Using the state-of-the-art analysis, VTAA has the query complexity

$$\mathbf{O}\left(\frac{\sqrt{m}}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(|\psi_0\rangle\right)+\frac{\sqrt{m}}{\sqrt{p_{\text{succ}}}}\sum_{j=1}^{m}\left\|\overline{\Pi_{j-1}\Pi_b}A_{j-1}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_j\right)\right). \tag{51}$$

Here, the cost of input algorithms has the ℓ_1-norm scaling with an additional $\sqrt{m}$ factor, which can be relaxed to the ℓ_2-norm scaling introducing another $\sqrt{m}$ factor. This complexity analysis can be further improved—we present the tightened analysis in Section 3 for Tunable VTAA which subsumes Ambainis' scheme as a special case. In any event, the above complexity is achievable only when all the amplification schedules are pre-determined during compilation. Otherwise, we need to compile the algorithm by frequently invoking amplitude estimations. This makes VTAA substantially more complicated, and introduces polylogarithmic factors to the cost of both input algorithms and the initial state preparation.

3 Tunable variable time amplitude amplification

We now introduce *Tunable Variable Time Amplitude Amplification* (Tunable VTAA) and analyze its performance. Specifically, we formally define Tunable VTAA in Section 3.1 and show its equivalence to a generic nested amplitude amplification algorithm reflecting toward the potentially good subspaces. We then analyze the query complexity of Tunable VTAA in Section 3.2. We present an optimized amplification schedule in Section 3.3 under which Tunable VTAA has the lowest possible query cost.

For presentation purpose, we assume that values of all norms in our discussion are known a prior. We will comment on how this assumption can be relaxed at the end of Section 3.3.

3.1 Definition and universal property

We begin with the formal definition of Tunable VTAA.

Definition 3 (Tunable variable time amplitude amplification)**.** *A tunable variable time amplitude amplification is a* 5*-tuple* $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m, \{\alpha_j\}_{j=0}^m, |\psi_0\rangle\right)$ *that additionally satisfies*

(iv′) α_j *are nonnegative real numbers for* $j = 1, \ldots, m$*, and* $\alpha_0 = 1$*, which define*

$$\begin{aligned} r_j &= \min\left\{ r \in \mathbb{Z}_{\geq 0} \;\middle|\; (2r+1)\left\|\overline{\Pi_j\Pi_b} A_j \widetilde{A}_{j-1}|\psi_0\rangle\right\| \geq \frac{1}{3}\sqrt{\alpha_j} \right\}, \qquad j = 1, \ldots, m \\ \widetilde{A}_j &= \begin{cases} \left(-\left(I - 2A_j\widetilde{A}_{j-1}|\psi_0\rangle\langle\psi_0|\widetilde{A}_{j-1}^\dagger A_j^\dagger\right)\left(I - 2\overline{\Pi_j\Pi_b}\right)\right)^{r_j} A_j\widetilde{A}_{j-1}, & j = 1, \ldots, m, \\ I, & j = 0. \end{cases} \end{aligned} \tag{52}$$

Compared with the standard VTAA, we have introduced the *amplification thresholds* α_j in Tunable VTAA. These thresholds determine the amplification schedules and amplified algorithms via the specified recurrence. That is, the schedule r_j is the smallest nonnegative integer satisfying $(2r_j+1)\left\|\overline{\Pi_j\Pi_b} A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\| \geq \frac{1}{3}\sqrt{\alpha_j}$; it has the following closed-form representation

$$r_j = \max\left\{ \mathbf{Ceil}\left(\frac{\sqrt{\alpha_j}}{6\left\|\overline{\Pi_j\Pi_b} A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|} - \frac{1}{2} \right), 0 \right\}. \tag{53}$$

When it happens $\left\|\overline{\Pi_j\Pi_b} A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\| \geq \frac{1}{3}\sqrt{\alpha_j}$, we choose $r_j = 0$. In this case, we have a trivial step $\widetilde{A}_j = A_j\widetilde{A}_{j-1}$. Otherwise if $0 < \left\|\overline{\Pi_j\Pi_b} A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\| < \frac{1}{3}\sqrt{\alpha_j}$, we have a nontrivial amplification with $r_j \geq 1$. $\widetilde{A}_j$ is then obtained by performing $2r_j + 1$ steps of amplitude amplifications toward the potentially good subspace $\mathbf{Im}\left(\overline{\Pi_j\Pi_b}\right)$.

Apparently, Tunable VTAA offers more flexibility than the VTAA of Ambainis, due to the introduction of tunable threshold values α_j that can be optimized for different input algorithms and initial states, though it may not be immediately clear how much more powerful this modification is. In the following, we show that Tunable VTAA is universal in the sense that it is equivalent to a generic nested amplitude amplification with no overshoot and constant loss factor. This provides a compelling reason for us to examine Tunable VTAA in greater detail.

In the forward direction, let us suppose we have a Tunable VTAA with amplification thresholds $0 \leq \alpha_j \leq 1$ for all $j = 1, \ldots, m$ and $\sum_{j=1}^{m} \alpha_j = \mathbf{O}(1)$. Then, in case we have a nontrivial amplification with $r_j \geq 1$ and $0 < \left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\| < \frac{1}{3}\sqrt{\alpha_j}$,

$$\begin{aligned}(2r_j+1)\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\| &< \left(2\left(\frac{\sqrt{\alpha_j}}{6\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\|} + \frac{1}{2} \right) + 1 \right) \left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\| \\ &= \frac{\sqrt{\alpha_j}}{3} + 2\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\| \\ &< \sqrt{\alpha_j} \leq 1,\end{aligned} \tag{54}$$

so there is no over amplification. If on the other hand $r_j = 0$, then it trivially holds $(2 \cdot 0 + 1)\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\| \leq 1$. We thus conclude that there is no overshoot in all the stages $j = 1, \ldots, m$. Now we lower bound the loss factor. Applying Proposition 16 from Appendix B only to the nontrivial stages,

$$\begin{aligned}&\prod_{r_j \geq 1} \left(\frac{\sin\left((2r_j+1) \arcsin\left(\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\| \right) \right)}{(2r_j+1)\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\|} \right) \\ &\geq \prod_{r_j \geq 1} \left(1 - \frac{1}{6}(2r_j+1)^2 \left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\|^2 \right) \geq \exp\left(\sum_{r_j \geq 1} \ln\left(1 - \frac{\alpha_j}{6} \right) \right) \\ &\geq \exp\left(\sum_{r_j \geq 1} \ln\left(\frac{5}{6} \right) \alpha_j \right) = \left(\frac{5}{6} \right)^{\sum_{r_j \geq 1} \alpha_j} \geq \left(\frac{5}{6} \right)^{\sum_{j=1}^{m} \alpha_j} = \mathbf{\Omega}(1),\end{aligned} \tag{55}$$

where in the third inequality we have used the fact that $\ln(1-x) \geq \ln(1-c)\frac{x}{c}$ for $0 < x \leq c < 1$. Hence, the total loss factor of the nested amplitude amplification is at least constant.

Conversely, suppose we have a nested amplitude amplification with schedules r_j with no overshoot $\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\| \leq \frac{1}{3(2r_j+1)}$, and constant total loss factor $\prod_{j=1}^{m} \frac{\left\| \overline{\Pi_j \Pi_b} \widetilde{A}_j |\psi_0\rangle \right\|}{(2r_j+1)\left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\|} = \mathbf{\Omega}(1)$. Then, let us choose the following thresholds

$$\alpha_j = \begin{cases} 9(2r_j+1)^2 \left\| \overline{\Pi_j \Pi_b} A_j \widetilde{A}_{j-1} |\psi_0\rangle \right\|^2, & r_j \geq 1, \\ 0, & r_j = 0. \end{cases} \tag{56}$$

Clearly, $\alpha_j \leq 1$ which follows from the no overshoot condition. It is also verifiable that Tunable VTAA with thresholds α_j implements exactly a nested amplitude amplification with schedules r_j.

Moreover, we upper bound the loss factor by applying Proposition 16 to all the nontrivial stages

$$\begin{aligned}
&\prod_{r_j\geq 1}\left(\frac{\sin\left((2r_j+1)\arcsin\left(\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\right)\right)}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}\right)\\
&\leq \prod_{r_j\geq 1}\left(1-\frac{4\pi-8}{\pi^3}(2r_j+1)^2\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|^2\right)=\exp\left(\sum_{r_j\geq 1}\ln\left(1-\frac{4\pi-8}{9\pi^3}\alpha_j\right)\right) \qquad (57)\\
&\leq \exp\left(-\sum_{r_j\geq 1}\frac{4\pi-8}{9\pi^3}\alpha_j\right)=\frac{1}{\left(\exp\left(\frac{4\pi-8}{9\pi^3}\right)\right)^{\sum_{r_j\geq 1}\alpha_j}},
\end{aligned}$$

where in the second inequality we use the fact that $\ln(1-x)\leq -x$ for $x<1$. The assumption that the total loss factor is constant then implies that $\sum_{j=1}^m \alpha_j=\sum_{r_j\geq 1}\alpha_j=\mathbf{O}(1)$. We have thus established:

Proposition 4 (Universality of Tunable VTAA)**.** *The following correspondence holds between Tunable VTAA and variable time nested amplitude amplification.*

(i) A Tunable VTAA $\left(\{\Pi_j\}_{j=0}^m,\Pi_b,\{A_j\}_{j=0}^m,\{\alpha_j\}_{j=0}^m,|\psi_0\rangle\right)$ *according to Definition 3 where* $\alpha_j\leq 1$ *and* $\sum_{j=1}^m\alpha_j=\mathbf{O}(1)$*, is a variable time nested amplitude amplification according to Definition 2 where* $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\leq\frac{1}{2r_j+1}$ *and* $\prod_{j=1}^m\frac{\left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\|}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}=\mathbf{\Omega}(1)$.

(ii) A variable time nested amplitude amplification $\left(\{\Pi_j\}_{j=0}^m,\Pi_b,\{A_j\}_{j=0}^m,\{r_j\}_{j=1}^m,|\psi_0\rangle\right)$ *according to Definition 2 where* $\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|\leq\frac{1}{3(2r_j+1)}$ *and* $\prod_{j=1}^m\frac{\left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\|}{(2r_j+1)\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}=\mathbf{\Omega}(1)$*, is some Tunable VTAA according to Definition 3 where* $\alpha_j\leq 1$ *and* $\sum_{j=1}^m\alpha_j=\mathbf{O}(1)$.

3.2 Query complexity

We now analyze the performance of Tunable VTAA, taking into account the fact that not all stages have nontrivial amplitude amplifications.

Specifically, suppose there are nontrivial amplifications at stages $1\leq s_1\leq\cdots\leq s_l\leq m$, with the corresponding amplification step numbers $r_{s_1},\ldots,r_{s_l}\geq 1$. Then, the cost of a nested amplitude amplification algorithm previously stated in Eq. (46) should be revised to

$$\mathbf{Cost}(\widetilde{A}_m|\psi_0\rangle)=\mathbf{Cost}(|\psi_0\rangle)\prod_{u=1}^{l}(2r_{s_u}+1)+\sum_{v=1}^{l+1}\mathbf{Cost}(A_{s_v}\cdots A_{s_{v-1}+1})\prod_{u=v}^{l}(2r_{s_u}+1), \qquad (58)$$

where we adopt the convention that $s_0=0$, $s_{l+1}=m+1$ and $A_{m+1}=I$. That is, the query cost is multiplied by $2r_{s_u}+1$ only at the l nontrivial stages, and we merge the cost of input algorithms $\mathbf{Cost}(A_k)$ for $k=s_{v-1}+1,\ldots,s_v$, associated with the same query product $\prod_{u=v}^{l}(2r_{s_u}+1)$.

Let us consider a general query product from stage s_v to s_w and re-express it as

$$\begin{aligned}
&\prod_{u=v}^{w}(2r_{s_u}+1)=\prod_{k=s_v}^{s_w}(2r_k+1)\\
&=\prod_{k=s_v}^{s_w}\left(\frac{1}{\frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A}_k|\psi_0\rangle\right\|}{(2r_k+1)\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|}}\right)\prod_{k=s_v}^{s_w}\left(\frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A}_k|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|}\right)\\
&=\prod_{k=s_v}^{s_w}\left(\frac{1}{\frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A}_k|\psi_0\rangle\right\|}{(2r_k+1)\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|}}\right)\frac{\left\|\overline{\Pi_{s_w}\Pi_b}\widetilde{A}_{s_w}|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_v-1}\Pi_b}\widetilde{A}_{s_v-1}|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{s_v-1}\Pi_b}A_{s_v-1}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_w}\Pi_b}A_{s_w}\cdots A_1|\psi_0\rangle\right\|}\\
&=\prod_{u=v}^{w}\left(\frac{1}{\frac{\left\|\overline{\Pi_{s_u}\Pi_b}\widetilde{A}_{s_u}|\psi_0\rangle\right\|}{(2r_{s_u}+1)\left\|\overline{\Pi_{s_u}\Pi_b}A_{s_u}\widetilde{A}_{s_u-1}|\psi_0\rangle\right\|}}\right)\frac{\left\|\overline{\Pi_{s_w}\Pi_b}\widetilde{A}_{s_w}|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_v-1}\Pi_b}\widetilde{A}_{s_v-1}|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{s_v-1}\Pi_b}A_{s_v-1}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_w}\Pi_b}A_{s_w}\cdots A_1|\psi_0\rangle\right\|}.
\end{aligned}\tag{59}$$

Here, the first equality uses the observation that for $s_v\le k\le s_w$, either we have a nontrivial stage with $k=s_u$ for $v\le u\le w$, or we have a trivial stage with $r_k=0$. The second equality is a direct rewriting. The third equality can be reasoned using Eq. (44) in a similar way as [12, Lemma 16]:

$$\begin{aligned}
\prod_{k=s_v}^{s_w}\left(\frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A}_k|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|}\right)&=\prod_{k=s_v}^{s_w}\left(\frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A}_k|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{k-1}\Pi_b}\widetilde{A}_{k-1}|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{k-1}\Pi_b}\widetilde{A}_{k-1}|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|}\right)\\
&=\prod_{k=s_v}^{s_w}\left(\frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A}_k|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{k-1}\Pi_b}\widetilde{A}_{k-1}|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{k-1}\Pi_b}A_{k-1}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_1|\psi_0\rangle\right\|}\right)\\
&=\frac{\left\|\overline{\Pi_{s_w}\Pi_b}\widetilde{A}_{s_w}|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_v-1}\Pi_b}\widetilde{A}_{s_v-1}|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{s_v-1}\Pi_b}A_{s_v-1}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_w}\Pi_b}A_{s_w}\cdots A_1|\psi_0\rangle\right\|}.
\end{aligned}\tag{60}$$

The justification of the last equality requires some more efforts. In the first factor, we have changed the multiplication variable from k back to u, using again the observation that for $s_v\le k\le s_w$, either we have a nontrivial stage with $k=s_u$ for $v\le u\le w$, or we have a trivial stage with $r_k=0$ and loss factor $\frac{\left\|\overline{\Pi_k\Pi_b}\widetilde{A}_k|\psi_0\rangle\right\|}{(2r_k+1)\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|}=1$. Now recall that $s_{v-1}+1,\ldots,s_v-1$ are all trivial stages by definition. Thus, invoking Eq. (44),

$$\frac{\left\|\overline{\Pi_{s_v-1}\Pi_b}A_{s_v-1}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_v-1}\Pi_b}\widetilde{A}_{s_v-1}|\psi_0\rangle\right\|}=\frac{\left\|\overline{\Pi_{s_v-1}\Pi_b}A_{s_v-1}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_v-1}\Pi_b}A_{s_v-1}\cdots A_{s_{v-1}+1}\widetilde{A}_{s_{v-1}}|\psi_0\rangle\right\|}=\frac{\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_v-1}\Pi_b}\widetilde{A}_{s_{v-1}}|\psi_0\rangle\right\|}.\tag{61}$$

This proves the claimed representation of query product. As this representation will be used multiple times within the paper, let us encapsulate it into a lemma.

Lemma 5 (Representation of query product)**.** *Let* $\left(\{\Pi_j\}_{j=0}^m,\Pi_b,\{A_j\}_{j=0}^m,\{r_j\}_{j=1}^m,|\psi_0\rangle\right)$ *be a variable time nested amplitude amplification according to Definition 2. Suppose that* $r_k\ge 1$ *happens*

only at l stages $1 \le s_1 \le \cdots \le s_l \le m$. Then,

$$\prod_{u=v}^{w}(2r_{s_u}+1) = \prod_{u=v}^{w}\left(\frac{1}{\frac{\left\|\overline{\Pi_{s_u}\Pi_b}\widetilde{A}_{s_u}|\psi_0\rangle\right\|}{(2r_{s_u}+1)\left\|\overline{\Pi_{s_u}\Pi_b}A_{s_u}\widetilde{A}_{s_u-1}|\psi_0\rangle\right\|}}\right)\frac{\left\|\overline{\Pi_{s_w}\Pi_b}\widetilde{A}_{s_w}|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_{v-1}}\Pi_b}\widetilde{A}_{s_{v-1}}|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_w}\Pi_b}A_{s_w}\cdots A_1|\psi_0\rangle\right\|} \tag{62}$$

for $1 \le v \le w \le l$, under the convention that $s_0 = 0$.

We now apply this representation to the query products in Eq. (58), all of which end with $w = l$. By definition, s_l is the last nontrivial stage so $s_l + 1, \ldots, m$ are all trivial. Invoking Eq. (44) once more,

$$\frac{\left\|\overline{\Pi_{s_l}\Pi_b}A_{s_l}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_l}\Pi_b}\widetilde{A}_{s_l}|\psi_0\rangle\right\|} = \frac{\left\|\overline{\Pi_m\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_m\Pi_b}A_m\cdots A_{s_l+1}\widetilde{A}_{s_l}|\psi_0\rangle\right\|} = \frac{\left\|\overline{\Pi_m\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_m\Pi_b}\widetilde{A}_m|\psi_0\rangle\right\|}, \tag{63}$$

giving

$$\prod_{u=v}^{l}(2r_{s_u}+1) = \prod_{u=v}^{l}\left(\frac{1}{\frac{\left\|\overline{\Pi_{s_u}\Pi_b}\widetilde{A}_{s_u}|\psi_0\rangle\right\|}{(2r_{s_u}+1)\left\|\overline{\Pi_{s_u}\Pi_b}A_{s_u}\widetilde{A}_{s_u-1}|\psi_0\rangle\right\|}}\right)\frac{\left\|\overline{\Pi_m\Pi_b}\widetilde{A}_m|\psi_0\rangle\right\|}{\left\|\overline{\Pi_{s_{v-1}}\Pi_b}\widetilde{A}_{s_{v-1}}|\psi_0\rangle\right\|}\frac{\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|}{\left\|\overline{\Pi_m\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\|}. \tag{64}$$

To proceed, note that the first factor is the inverse loss factor, which can be bounded by $\left(\frac{6}{5}\right)^{\sum_{j=1}^m \alpha_j}$. Then, we have the amplitude of post-amplified algorithm trivially bounded by $\left\|\overline{\Pi_m\Pi_b}\widetilde{A}_m|\psi_0\rangle\right\| \le 1$. In practice, we would also adjust the last stage of VTAA so that the amplitude is at least constant. Next, $\left\|\overline{\Pi_m\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\| = \sqrt{p_{\text{succ}}}$ by definition. We now lower bound $\left\|\overline{\Pi_{s_{v-1}}\Pi_b}\widetilde{A}_{s_{v-1}}|\psi_0\rangle\right\|$:

$$\begin{aligned}\left\|\overline{\Pi_{s_{v-1}}\Pi_b}\widetilde{A}_{s_{v-1}}|\psi_0\rangle\right\| &= \sin\left((2r_{s_{v-1}}+1)\arcsin\left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\widetilde{A}_{s_{v-1}-1}|\psi_0\rangle\right\|\right)\right)\\ &\ge \left(\frac{5}{6}\right)^{\alpha_{s_{v-1}}}(2r_{s_{v-1}}+1)\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\widetilde{A}_{s_{v-1}-1}|\psi_0\rangle\right\|\\ &= \mathbf{\Omega}\left(\sqrt{\alpha_{s_{v-1}}}\right).\end{aligned} \tag{65}$$

This analysis holds for all query products starting at $v = 1, \ldots, l$. In particular, for the cost of initial state preparation, we have $v = 1$, $\alpha_0 = 1$ and $\||\psi_0\rangle\| = 1$. Altogether, this gives the query complexity of Tunable VTAA

$$\begin{aligned}\mathbf{Cost}(\widetilde{A}_m|\psi_0\rangle) = \mathbf{O}\Bigg(&\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(A_{s_1}\cdots A_1|\psi_0\rangle)\\ &+\frac{1}{\sqrt{p_{\text{succ}}}}\sum_{v=2}^{l+1}\frac{1}{\sqrt{\alpha_{s_{v-1}}}}\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\Bigg).\end{aligned} \tag{66}$$

Note that the cost of VTAA would remain the same if we were to pre-merge the input algorithms $A_{s_v}\cdots A_{s_{v-1}+1}$ during compilation, resulting in a $(l+1)$-stage algorithm. We now claim that $l = \mathbf{O}\left(\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right)$. Indeed, this follows from the observation that

$$3^l \le \prod_{u=1}^{l}(2r_{s_u}+1) \le \left(\frac{6}{5}\right)^{\sum_{j=1}^m \alpha_j}\frac{1}{\sqrt{p_{\text{succ}}}} = \mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right) \quad\Rightarrow\quad l = \mathbf{O}\left(\log_3\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right). \tag{67}$$

Therefore, in the interesting regime where $\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right) \ll m$, majority of the algorithms can be pre-merged yielding a significantly simplified VTAA.

3.3 $l_{\frac{2}{3}}$-quasinorm scaling

Having established the universal property and query complexity of Tunable VTAA, we now explore its limitation. In particular, we will show that complexity of Tunable VTAA can be optimized to scale with $l_{\frac{2}{3}}$-quasinorm of the input cost.

To this end, we note that the query cost of initial state preparation and algorithm $A_{s_1}\cdots A_1$ has the scaling $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$ independent of the amplification schedules, so it suffices to optimize query complexity of the remaining input algorithms. Up to a rescaling, this reduces to solving the following problem:

$$\begin{aligned} \text{minimize} \quad & \sum_{v=2}^{l+1} \frac{1}{\sqrt{\alpha_{s_{v-1}}}} \left\|\overline{\Pi_{s_{v-1}}\Pi_b} A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\| \mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right) \\ \text{subject to} \quad & \sum_{v=2}^{l+1} \alpha_{s_{v-1}} = 1, \\ & \alpha_{s_{v-1}} > 0, \quad v = 2,\ldots,l+1. \end{aligned} \tag{68}$$

This optimization problem can be readily solved by the weighted mean inequality. Specifically, given positive real numbers $\{w_v\}_{v=1}^l$ and $\{x_v\}_{v=1}^l$, the inequality asserts that the weighted harmonic mean is always upper bounded by the weighted quadratic mean [52, Problem 8.3]:

$$\frac{\sum_{v=1}^{l} w_v}{\sum_{v=1}^{l} w_v \frac{1}{x_v}} \le \sqrt{\frac{\sum_{v=1}^{l} w_v x_v^2}{\sum_{v=1}^{l} w_v}} \quad \Leftrightarrow \quad \sum_{v=1}^{l} w_v \frac{1}{x_v} \ge \frac{\left(\sum_{v=1}^{l} w_v\right)^{\frac{3}{2}}}{\sqrt{\sum_{v=1}^{l} w_v x_v^2}} \tag{69}$$

with equality if and only if $x_1 = \cdots = x_l$. Specialized to our problem, let us choose

$$\begin{aligned} w_{v-1} &= \left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b} A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\| \mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}, \\ x_{v-1} &= \sqrt{\alpha_{s_v-1}} \left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b} A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\| \mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{-\frac{1}{3}}, \end{aligned} \tag{70}$$

for $v = 2,\ldots,l+1$. Then, the weighted mean inequality implies that

$$\begin{aligned} &\sum_{v=2}^{l+1} \frac{1}{\sqrt{\alpha_{s_{v-1}}}} \left\|\overline{\Pi_{s_{v-1}}\Pi_b} A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\| \mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right) \\ &\ge \left(\sum_{v=2}^{l+1} \left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b} A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\| \mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}\right)^{\frac{3}{2}} \end{aligned} \tag{71}$$

with the lower bound attained when

$$\alpha_{s_v-1} \propto \left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b} A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\| \mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}. \tag{72}$$

We have thus established:

Proposition 6 (Query complexity of Tunable VTAA). *Let* $\left(\{\Pi_j\}_{j=0}^m, \Pi_b, \{A_j\}_{j=0}^m, \{\alpha_j\}_{j=0}^m, |\psi_0\rangle\right)$ *be a Tunable VTAA according to Definition 3. Assume that the threshold values satisfy* $0 \le \alpha_j \le 1$ *and* $\sum_{j=1}^m \alpha_j = \mathbf{O}(1)$. *Then,*

(i) Nontrivial amplifications happen only at l *stages* $1 \le s_1 \le \cdots \le s_l \le m$, *where*

$$l = \mathbf{O}\left(\log_3\left(\frac{1}{\sqrt{p_{succ}}}\right)\right). \tag{73}$$

Under the convention that $s_{l+1} = m+1$ *and* $A_{m+1} = I$, *Tunable VTAA has query complexity*

$$\begin{aligned}\mathbf{Cost}(\widetilde{A}_m|\psi_0\rangle) = \mathbf{O}\Bigg(&\frac{1}{\sqrt{p_{succ}}}\mathbf{Cost}(A_{s_1}\cdots A_1|\psi_0\rangle)\\ &+\frac{1}{\sqrt{p_{succ}}}\sum_{v=2}^{l+1}\frac{1}{\sqrt{\alpha_{s_{v-1}}}}\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\Bigg).\end{aligned} \tag{74}$$

(ii) Pre-merging trivial stages and using thresholds

$$\alpha_{s_{v-1}} \propto \left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}, \tag{75}$$

the complexity of Tunable VTAA can be minimized to attain the $\ell_{\frac{2}{3}}$-quasinorm *scaling*

$$\begin{aligned}\mathbf{Cost}(\widetilde{A}_m|\psi_0\rangle) = \mathbf{O}\Bigg(&\frac{1}{\sqrt{p_{succ}}}\mathbf{Cost}(A_{s_1}\cdots A_1|\psi_0\rangle)\\ &+\frac{1}{\sqrt{p_{succ}}}\left(\sum_{v=2}^{l+1}\left(\left\|\overline{\Pi_{s_{v-1}}\Pi_b}A_{s_{v-1}}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_{s_v}\cdots A_{s_{v-1}+1}\right)\right)^{\frac{2}{3}}\right)^{\frac{3}{2}}\Bigg).\end{aligned} \tag{76}$$

In our above discussion, we have used precise values of all the norms involved to simplify our presentation, but they can be replaced by their constant multiplicative approximators without affecting the asymptotic analysis. If this prior knowledge is not available, one could perform nested amplitude estimations like in [2, 12], introducing logarithmic overhead and substantially complicating the structure of algorithm. However, the advantage of using Tunable VTAA is that this nested amplitude estimation may be completely avoided if we choose the threshold values analytically. We will demonstrate this feature in Section 4 for solving the quantum linear system problem.

4 Discretized inverse state

In this section, we introduce the *discretized inverse state* for solving the quantum linear system problem, which can be efficiently prepared by Tunable VTAA. We begin with its construction in Section 4.1 by performing *Gapped Phase Estimation* (GPE) on quantum walk, taking special care of the issue that eigenphases of the walk operator are split into two branches with opposite +/- signs. We then present a deterministic amplification schedule in Section 4.2 that significantly simplifies the structure of VTAA.

To further reduce the query complexity, we present an improved analysis of VTAA in Section 4.3 by projecting error onto the potentially good subspaces. Finally, we develop a simple solution norm estimation algorithm in Section 4.4 to estimate p_{succ} when a constant multiplicative approximation of it is unavailable a prior.

4.1 Gapped phase estimation with branch marking

In the quantum linear system problem, the coefficient matrix is block encoded by oracle O_A as A/α_A with normalization factor $\alpha_A \geq \|A\|$, and an upper bound on its inverse $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ is known a prior. We now make a few more assumptions to simplify the discussion without affecting generality. First, we assume that A is Hermitian and O_A is a Hermitian unitary; we can always fulfill this requirement by considering the Hermitian dilation $|0\rangle\langle 1|\otimes A+|1\rangle\langle 0|\otimes A^\dagger$ block encoded by $|0\rangle\langle 1|\otimes O_A+|1\rangle\langle 0|\otimes O_A^\dagger$, with the corresponding initial state $|0\rangle|b\rangle$. Second, we assume $\alpha_A \geq 2\|A\|$, which can be achieved by block encoding a constant factor to artificially increase the normalization factor. Third, we assume that both α_A and $\alpha_{A^{-1}}$ are integer powers of 3. Again the former can be satisfied by rescaling the block encoding, whereas the latter can be realized using another bound at most a constant factor 3 larger.

As an immediate consequence, the upper bound $\kappa = \alpha_A\alpha_{A^{-1}}$ on the condition number is also an integer power of 3, and thus the number of VTAA stages $m = \log_3(\kappa)$ is an integer. Moreover, for any eigenvalue λ_u of A, the block encoded operator $\frac{A}{\alpha_A}$ has an eigenvalue $\frac{\lambda}{\alpha_A}$ such that $\left|\frac{\lambda}{\alpha_A}\right| \in \left[\frac{1}{3^m}, 1\right)$.

Our algorithm proceeds by applying quantum signal processing to the quantum walk operator. Specifically, assume that the coefficient matrix is block encoded as $\frac{A}{\alpha_A} = G^\dagger O_A G$ for some isometry G. Then, the quantum walk operator is defined by $W = \left(2GG^\dagger - I\right) O_A$. In Appendix C, we give a self-contained exposition of the qubitization result [38] linking a block encoding to its walk operator. In particular, we prove the following spectrum correspondence: if the input block encoding has the spectral decomposition

$$A = \sum_u \lambda_u |\phi_u\rangle\langle\phi_u|, \tag{77}$$

then the walk operator has the corresponding spectral decomposition

$$W = \sum_u \left(e^{+i \arccos\left(\frac{\lambda_u}{\alpha_A}\right)} |\phi_{u,+}\rangle\langle\phi_{u,+}| + e^{-i \arccos\left(\frac{\lambda_u}{\alpha_A}\right)} |\phi_{u,-}\rangle\langle\phi_{u,-}| \right) \tag{78}$$

when restricted to $\mathbf{Im}\left(GG^\dagger\right) + \mathbf{Im}\left(O_A GG^\dagger O_A^\dagger\right)$, where

$$|\phi_{u,0}\rangle = G|\phi_u\rangle, \qquad |\phi_{u,1}\rangle = \frac{O_A G|\phi_u\rangle - \lambda_u G|\phi_u\rangle}{\sqrt{1-\lambda_u^2}}, \qquad |\phi_{u,\pm}\rangle = \frac{|\phi_{u,0}\rangle \pm i|\phi_{u,1}\rangle}{\sqrt{2}}. \tag{79}$$

Note that we have omitted the degenerate 1-dimensional subspaces in qubitization, because all our eigenvalues satisfy $\left|\frac{\lambda_u}{\alpha_A}\right| < 1$ following the rescaling assumption at the beginning of this subsection.

Now, suppose that the initial state can be expanded in the eigenbasis of A as $|b\rangle = \sum_u \gamma_u |\phi_u\rangle$. We can then apply G and expand it in the eigenbasis of W as

$$G|b\rangle = \sum_u \gamma_u G|\phi_u\rangle = \sum_u \gamma_u |\phi_{u,0}\rangle = \sum_u \gamma_u \frac{|\phi_{u,+}\rangle + |\phi_{u,-}\rangle}{\sqrt{2}}. \tag{80}$$

Next, we apply GPE to label the interval to which every eigenvalue belongs. In earlier work [15], GPE is performed on the time evolution operator e^{iA}, which introduces the overhead of Hamiltonian simulation. More recent work [13] suggested applying GPE directly on the quantum walk operator

W. In our language, this would produce an unnormalized state of the form

$$\frac{1}{\sqrt{2}}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}\left(\left(\zeta_{k+1,u,+}\frac{3^{k+1}}{3^m}|k\rangle+\zeta_{k,u,+}\frac{3^k}{3^m}|k-1\rangle\right)\gamma_u|\phi_{u,+}\rangle\right.\\ \left.+\left(\zeta_{k+1,u,-}\frac{3^{k+1}}{3^m}|k\rangle+\zeta_{k,u,-}\frac{3^k}{3^m}|k-1\rangle\right)\gamma_u|\phi_{u,-}\rangle\right). \tag{81}$$

As the coefficients $\zeta_{k,u,+}$ are different from $\zeta_{k,u,-}$ in general, one finds that the original uniform superposition of $|\phi_{u,+}\rangle$ and $|\phi_{u,-}\rangle$ is "distorted" by GPE into a non-uniform superposition. Moreover, this distortion on the $\pm$ branches of quantum walk is not recorded by any ancilla state, and so it is not possible to perform amplitude amplification. This then leads to a highly inaccurate solution state, a critical issue left unattended in [13].

We address this issue by a gadget called *branch marking*. Up to a controllable error, branch marking realizes the transformation

$$|+\rangle|\phi_{u,\pm}\rangle\mapsto|\pm\rangle|\phi_{u,\pm}\rangle, \tag{82}$$

recording information about the signs of eigenphases of the walk operator into an ancilla register. Given thresholds $\gamma,\frac{\gamma}{\rho}$, we then perform GPE controlled by the branch register to transform

$$|\pm\rangle|0\rangle|\phi_{u,\pm}\rangle\mapsto|\pm\rangle|\xi_u\rangle|\phi_{u,\pm}\rangle,\qquad|\xi_u\rangle=\xi_{u,0}|0\rangle+\xi_{u,1}|1\rangle, \tag{83}$$

such that

$$|\xi_u\rangle\approx\begin{cases}|0\rangle, & \frac{\lambda_u}{\alpha_A}\in[\gamma,1),\\ i|1\rangle, & \frac{\lambda_u}{\alpha_A}\in\left[-\frac{\gamma}{\rho},\frac{\gamma}{\rho}\right],\\ -|0\rangle, & \frac{\lambda_u}{\alpha_A}\in(-1,-\gamma],\end{cases} \tag{84}$$

Importantly, the output state $|\xi_u\rangle$ has *no* dependence on the specific $\pm$ branch of quantum walk, regardless of whether the original eigenvalue λ_u is in the passband, stopband or transition band of GPE—this is the primary purpose of using branch marking.

We now describe a variable time quantum algorithm for preparing the discretized inverse state. Our algorithm acts on a clock register $|j\rangle$ holding values $j=0,\ldots,m-1$, a two-qubit flag register with possible states

$$|\text{good}\rangle=|00\rangle,\qquad|\text{bad}\rangle=|10\rangle,\qquad|\text{cont}'\text{d}\rangle=|01\rangle, \tag{85}$$

and a system register holding the solution state, ancilla state for block encoding, a single qubit $|\pm\rangle$ for branch marking, and any additional ancillas consumed by the branch marking and GPE. We introduce accuracy parameters $\epsilon_{\text{bm}},\epsilon_{\text{gpe},j}$ to be specified later. The algorithm is then constructed as follows.

1. Initialization:
   (a) Set the state of clock register to be $|0\rangle$.
   (b) Set the state of flag register to be $|\text{cont}'\text{d}\rangle$.
   (c) Apply the branch marking unitary from Proposition 22 to implement the transformation $|+\rangle|\phi_{u,\pm}\rangle\mapsto|\pm\rangle|\phi_{u,\pm}\rangle$ with accuracy ϵ_{bm}.
2. Discretized inversion: for $j=1,\ldots,m-1$,

(a) Controlled on the clock state $|j-1\rangle$, flip the second qubit of the flag register and apply GPE from Proposition 23 with $\gamma = \frac{1}{3^j}$, $\rho = 3$ and accuracy $\epsilon_{\mathrm{gpe},j}$, storing outcome back to the second qubit of flag register. Up to error $\epsilon_{\mathrm{gpe},j}$, this implements the transformation $|\mathrm{cont'd}\rangle|\pm\rangle|\phi_{u,\pm}\rangle \mapsto (\xi_{j,u,0}|\mathrm{good}\rangle + \xi_{j,u,1}|\mathrm{cont'd}\rangle)\,|\pm\rangle|\phi_{u,\pm}\rangle$, where $\xi_{j,u,1} = 0$ if $\frac{1}{3^j} \leq \left|\frac{\lambda_u}{\alpha_A}\right| < \frac{1}{2}$, and $\xi_{j,u,0} = 0$ if $0 \leq \left|\frac{\lambda_u}{\alpha_A}\right| < \frac{1}{3^{j+1}}$. Note that both $\pm$ branches pick up the same coefficients $\xi_{j,u,0}$ and $\xi_{j,u,1}$ in the outcome state.

(b) Controlled on the state $|j-1\rangle$, implement the mapping $|\mathrm{cont'd}\rangle \mapsto |\mathrm{cont'd}\rangle, |\mathrm{good}\rangle \mapsto \frac{3^j}{3^m}|\mathrm{good}\rangle + \sqrt{1-\frac{9^j}{9^m}}|\mathrm{bad}\rangle$. This can be achieved by applying a Pauli-Y rotation on the first qubit of flag register, controlled by the second qubit in state $|1\rangle$.

(c) Controlled on the flag state $|\mathrm{cont'd}\rangle$, increment the clock state $|j\rangle \mapsto |j+1\rangle$.

3. Finalization:

(a) Controlled on the clock state $|m-1\rangle$, apply the transformation $|\mathrm{cont'd}\rangle \mapsto |\mathrm{good}\rangle$. This can be achieved by applying a Pauli-X gate on the second qubit of flag register.

(b) Undo the branch marking by invoking the reversal of Proposition 22.

Let us first confirm that the above is indeed a variable time quantum algorithm in the sense of Definition 1. To this end, we define the clock projections

$$\Pi_j = \sum_{0 \leq x \leq j-1} |x\rangle\langle x| \otimes (|\mathrm{good}\rangle\langle\mathrm{good}| + |\mathrm{bad}\rangle\langle\mathrm{bad}|) \tag{86}$$

for $j = 0, \ldots, m$. Note that $\Pi_m = I$ holds effectively since the last step of our algorithm always maps $|\mathrm{cont'd}\rangle$ to $|\mathrm{good}\rangle$. The flag projection is naturally selected to be

$$\Pi_b = |\mathrm{bad}\rangle\langle\mathrm{bad}|. \tag{87}$$

Finally, we have the following input algorithm at stage j (only showing its nontrivial actions):

$$\begin{aligned} C_j = {} & |j\rangle\langle j-1| \otimes \sum_u \xi_{j,u,1} |\mathrm{cont'd}\rangle\langle\mathrm{cont'd}| \otimes |\pm, \phi_{u,\pm}\rangle\langle\pm, \phi_{u,\pm}| \\ & + |j-1\rangle\langle j-1| \otimes \sum_u \xi_{j,u,0} \left(\frac{3^j}{3^m}|\mathrm{good}\rangle + \sqrt{1-\frac{9^j}{9^m}}|\mathrm{bad}\rangle \right) \langle\mathrm{cont'd}| \otimes |\pm, \phi_{u,\pm}\rangle\langle\pm, \phi_{u,\pm}|. \end{aligned} \tag{88}$$

It is then a routine verification that $C_j \Pi_{j-1} = \Pi_{j-1}$ holds for all $j = 1, \ldots, m-1$. This is also true for $j = m$ if we define $C_m = |m-1\rangle\langle m-1| \otimes |\mathrm{good}\rangle\langle\mathrm{cont'd}|$ (corresponding to setting $\xi_{m,u,0} = 1$ for all u). For notational convenience, we may assume that branch marking and its inverse are incorporated into C_1 and C_m, respectively.

To simplify the discussion, we consider the case where the branch marking can be performed perfectly, corresponding to input algorithms $B_1, \ldots, B_m$. This produces the output state

$$\frac{1}{\sqrt{2}} \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \sum_{j=1}^{m} \zeta_{j,u} |j-1\rangle \left(\frac{3^j}{3^m}|\mathrm{good}\rangle + \sqrt{1-\frac{9^j}{9^m}}|\mathrm{bad}\rangle \right) \gamma_u \left(|+, \phi_{u,+}\rangle + |-, \phi_{u,-}\rangle \right), \tag{89}$$

where $\zeta_{j,u}$ are the cumulative coefficients at stages j

$$\zeta_{j,u} = \xi_{j,u,0} \prod_{l=1}^{j-1} \xi_{l,u,1}. \tag{90}$$

Moreover, we know that GPE has the action:

(i) For $\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right) \subseteq \left[\frac{1}{3^l}, 1\right)$, or equivalently, $l \geq k+1$,

$$\xi_{l,u,1} \approx 0, \qquad \xi_{l,u,0} \approx \pm 1. \tag{91}$$

(ii) For $\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right) = \left[\frac{1}{3^{l+1}}, \frac{1}{3^l}\right)$, or equivalently, $l = k$,

$$|\xi_{l,u,0}|^2 + |\xi_{l,u,1}|^2 = 1. \tag{92}$$

(iii) For $\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right) \subseteq \left[0, \frac{1}{3^{l+1}}\right)$, or equivalently, $l \leq k-1$,

$$\xi_{l,u,0} \approx 0, \qquad \xi_{l,u,1} \approx i. \tag{93}$$

This means that the cumulative coefficients satisfy $\zeta_{j,u} \approx 0$ if $j \geq k+2$ or $j \leq k-1$. So if we further assume that GPE can be performed perfectly in the passband and stopband, corresponding to input algorithms $A_1, \ldots, A_m$, then we get

$$\begin{aligned}\frac{1}{\sqrt{2}} \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} & \left(\zeta_{k+1,u} |k\rangle \left(\frac{3^{k+1}}{3^m} |\text{good}\rangle + \sqrt{1 - \frac{9^{k+1}}{9^m}} |\text{bad}\rangle \right) \right. \\ & \left. + \zeta_{k,u} |k-1\rangle \left(\frac{3^k}{3^m} |\text{good}\rangle + \sqrt{1 - \frac{9^k}{9^m}} |\text{bad}\rangle \right) \right) \gamma_u \left(|+, \phi_{u,+}\rangle + |-, \phi_{u,-}\rangle \right).\end{aligned} \tag{94}$$

Compared with Eq. (81), our Eq. (94) maintains the uniform superposition of the eigenstates $|\phi_{u,\pm}\rangle$ of the quantum walk operator. At the end, the branch marking register is uncomputed, so the two branches can be merged back to recover the original eigenstate $|\phi_u\rangle$ of the input matrix. In the next subsection, we will describe a deterministic amplification schedule for Tunable VTAA to amplify such a state.

4.2 Deterministic amplification schedule

We now describe a deterministic amplification schedule for Tunable VTAA to prepare a state proportional to

$$\begin{aligned}\psi_{\text{d-inv}} &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \left(\zeta_{k+1,u} \frac{3^{k+1}}{3^m} |k\rangle + \zeta_{k,u} \frac{3^k}{3^m} |k-1\rangle \right) \gamma_u |\phi_u\rangle, \\ \|\psi_{\text{d-inv}}\|^2 &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \left(|\zeta_{k+1,u}|^2 \frac{9^{k+1}}{9^m} + |\zeta_{k,u}|^2 \frac{9^k}{9^m} \right) = p_{\text{succ,d-inv}}.\end{aligned} \tag{95}$$

For presentational purposes, we assume that we know the precise value of $p_{\text{succ,d-inv}}$. This assumption is without loss of generality as the asymptotic scaling of algorithm is not affected if a constant multiplicative approximation of $p_{\text{succ,d-inv}}$ is used instead, which can be constructed from a constant multiplicative estimate of p_{succ}, because $p_{\text{succ,d-inv}} = \boldsymbol{\Theta}(p_{\text{succ}})$ holds per Proposition 25 of Appendix E.2.

Following the discussion in Section 1.4, we consider the thresholds

$$\alpha_j = \begin{cases} c^2 9^{j-m+l} \left\| \overline{\Pi_j \Pi_b} A_j \cdots A_1 |\psi_0\rangle \right\|^2, & j = m-l+1, \ldots, m, \\ 0, & j = 1, \ldots, m-l, \end{cases} \tag{96}$$

for some constant $c \gtrapprox 1$ (say $c = 1.001$) and integer $0 \leq l \leq m$ to be determined later.

Analysis of amplification threshold We know from Proposition 4 that $\sum_{j=m-l+1}^{m}\alpha_j=\mathbf{O}(1)$ must hold to avoid a large loss factor. Here, we can use Eq. (94) to evaluate the potentially good probabilities as

$$\begin{aligned}
&\left\|\overline{\Pi_j\Pi_b}A_j\cdots A_1|\psi_0\rangle\right\|^2\\
&=\left\|\overline{\Pi_j\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\|^2\\
&=\sum_{k=j}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k+1,u}|^2+\sum_{k=j+1}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k,u}|^2\\
&\quad+\sum_{k=0}^{j-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k+1,u}|^2\,\frac{9^{k+1}}{9^m}+\sum_{k=0}^{j}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k,u}|^2\,\frac{9^k}{9^m}.
\end{aligned}\tag{97}$$

In doing so, we have assumed again that GPE and branch marking can be performed perfectly. See Proposition 26 from Appendix E.3 for the complete analysis.

Using this, we compute the sum of threshold values:

$$\begin{aligned}
&\sum_{j=m-l+1}^{m}\alpha_j\\
&\le c^2\sum_{j=m-l+1}^{m}\left(\sum_{k=j}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k+1,u}|^2+\sum_{k=0}^{j-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k+1,u}|^2\,\frac{9^{k+1}}{9^m}\right.\\
&\qquad\left.+\sum_{k=j+1}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k,u}|^2+\sum_{k=0}^{j}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k,u}|^2\,\frac{9^k}{9^m}\right)9^{j-m+l}\\
&=c^2\sum_{j=1}^{l}\left(\sum_{k=j+m-l}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k+1,u}|^2+\sum_{k=0}^{j+m-l-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k+1,u}|^2\,\frac{9^{k+1}}{9^m}\right.\\
&\qquad\left.+\sum_{k=j+m-l+1}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k,u}|^2+\sum_{k=0}^{j+m-l}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\,|\zeta_{k,u}|^2\,\frac{9^k}{9^m}\right)9^j.
\end{aligned}\tag{98}$$

By exchanging the order of summation, we may bound first line of the result as

$$\begin{aligned}
&\sum_{j=1}^{l}\Bigg(\sum_{k=j+m-l}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2+\sum_{k=0}^{j+m-l-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2\frac{9^{k+1}}{9^m}\Bigg)9^j\\
&=\sum_{k=m-l+1}^{m-1}\sum_{j=1}^{k-m+l}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2 9^j+\sum_{k=0}^{m-1}\sum_{j=k-m+l+1}^{l}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2\frac{9^{k+1}}{9^m}9^j\\
&\le\frac{9}{8}\sum_{k=m-l+1}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2 9^{k-m+l}+\frac{9}{8}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2\frac{9^{k+1}}{9^m}9^l\\
&\le\frac{5}{4}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2\frac{9^{k+1}}{9^m}9^l.
\end{aligned}\tag{99}$$

Similarly,

$$\begin{aligned}
&\sum_{j=1}^{l}\Bigg(\sum_{k=j+m-l+1}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2+\sum_{k=0}^{j+m-l}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2\frac{9^k}{9^m}\Bigg)9^j\\
&=\sum_{k=m-l+2}^{m-1}\sum_{j=1}^{k-m+l-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2 9^j+\sum_{k=0}^{m}\sum_{j=k-m+l}^{l}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2\frac{9^k}{9^m}9^j\\
&\le\frac{9}{8}\sum_{k=m-l+2}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2 9^{k-m+l-1}+\frac{9}{8}\sum_{k=0}^{m}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2\frac{9^k}{9^m}9^l\\
&\le\frac{5}{4}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2\frac{9^k}{9^m}9^l.
\end{aligned}\tag{100}$$

Altogether, we obtain

$$\begin{aligned}
&c^2p_{\text{succ,d-inv}}9^l=\alpha_m\le\sum_{j=m-l+1}^{m}\alpha_j=\sum_{j=m-l+1}^{m}c^2\left\|\overline{\Pi_j\Pi_b}A_j\cdots A_1|\psi_0\rangle\right\|^2 9^{j-m+l}\\
&\le\frac{5c^2}{4}\left(\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k+1,u}|^2\frac{9^{k+1}}{9^m}+\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2|\zeta_{k,u}|^2\frac{9^k}{9^m}\right)9^l\\
&=\frac{5c^2}{4}p_{\text{succ,d-inv}}9^l.
\end{aligned}\tag{101}$$

We thus have $\sum_{j=m-l+1}^{m}\alpha_j\le 1$ as long as $l\le\mathbf{Floor}\left(\log_3\left(\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\right)\right)=\mathbf{O}\left(\log_3\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right)$. In particular, there is no over amplification and the loss factor of VTAA is constant.

Analysis of amplification schedule Next, let us derive the deterministic amplification schedule. At the beginning, $j=m-l+1$, and we compare

$$\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}\widetilde{A}_{m-l}|\psi_0\rangle\right\|=\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}A_{m-l}\cdots A_1|\psi_0\rangle\right\|\tag{102}$$

and

$$\frac{1}{3}\sqrt{\alpha_{m-l+1}} = c\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}A_{m-l}\cdots A_1|\psi_0\rangle\right\|. \tag{103}$$

Assuming $c \gtrapprox 1$ (say $c = 1.001$), we find that

$$\begin{aligned}\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}\widetilde{A}_{m-l}|\psi_0\rangle\right\| &< \frac{1}{3}\sqrt{\alpha_{m-l+1}},\\ 3\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}\widetilde{A}_{m-l}|\psi_0\rangle\right\| &\geq \frac{1}{3}\sqrt{\alpha_{m-l+1}},\end{aligned} \tag{104}$$

so 3 amplification steps are needed for $j = m - l + 1$. After that,

$$\begin{aligned}&\left(1 - \frac{1}{6}9\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}\cdots A_1|\psi_0\rangle\right\|^2\right)3\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}\cdots A_1|\psi_0\rangle\right\|\\ &\leq \left\|\overline{\Pi_{m-l+1}\Pi_b}\widetilde{A}_{m-l+1}|\psi_0\rangle\right\|\\ &\leq 3\left\|\overline{\Pi_{m-l+1}\Pi_b}A_{m-l+1}\cdots A_1|\psi_0\rangle\right\|.\end{aligned} \tag{105}$$

By induction, suppose we have

$$\begin{aligned}&\prod_{k=m-l+1}^{j}\left(1 - \frac{1}{6}9\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|^2\right)\left\|\overline{\Pi_j\Pi_b}A_j\cdots A_1|\psi_0\rangle\right\|3^{j-m+l}\\ &\leq \left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\|\\ &\leq \left\|\overline{\Pi_j\Pi_b}A_j\cdots A_1|\psi_0\rangle\right\|3^{j-m+l}\end{aligned} \tag{106}$$

after stage j. For $j+1$, we want to compare

$$\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\| \tag{107}$$

and

$$\frac{1}{3}\sqrt{\alpha_{j+1}} = c3^{j-m+l}\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}A_j\cdots A_1|\psi_0\rangle\right\|. \tag{108}$$

Using the inductive hypothesis and Eq. (44),

$$\begin{aligned}&\prod_{k=m-l+1}^{j}\left(1 - \frac{1}{6}9\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|^2\right)\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\cdots A_1|\psi_0\rangle\right\|3^{j-m+l}\\ &\leq \left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|\\ &\leq \left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\cdots A_1|\psi_0\rangle\right\|3^{j-m+l}.\end{aligned} \tag{109}$$

Then since the loss factor $\geq \left(\frac{5}{6}\right)^{\sum_{j=1}^{m}\alpha_j} \geq \frac{5}{6}$, the amplitude satisfies

$$\begin{aligned}\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\| &< \frac{1}{3}\sqrt{\alpha_{j+1}},\\ 3\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\| &\geq \frac{1}{3}\sqrt{\alpha_{j+1}}.\end{aligned} \tag{110}$$

So again we need 3 amplification steps for $j+1$. After that,

$$\begin{aligned}
&\prod_{k=m-l+1}^{j+1}\left(1-\frac{1}{6}9\left\|\overline{\Pi_k\Pi_b}A_k\widetilde{A}_{k-1}|\psi_0\rangle\right\|^2\right)\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\cdots A_1|\psi_0\rangle\right\|3^{j+1-m+l}\\
&\leq\left(1-\frac{1}{6}9\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|^2\right)3\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|\\
&\leq\left\|\overline{\Pi_{j+1}\Pi_b}\widetilde{A}_{j+1}|\psi_0\rangle\right\|\\
&\leq 3\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|\\
&\leq\left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\cdots A_1|\psi_0\rangle\right\|3^{j+1-m+l}.
\end{aligned}\tag{111}$$

The induction is now complete. We obtain the following deterministic amplification schedule as desired:

$$2r_j+1=\begin{cases}3, & j=m-l+1,\ldots,m,\\ 1, & j=1,\ldots,m-l.\end{cases}\tag{112}$$

Analysis of query complexity We now consider the query complexity. Invoking Proposition 6, we have the cost

$$\begin{aligned}
&\mathbf{O}\Bigg(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(A_{m-l+1}\cdots A_1|\psi_0\rangle\right)\\
&\quad+\frac{1}{\sqrt{p_{\text{succ}}}}\sum_{j=m-l+2}^{m}\frac{1}{\sqrt{\alpha_{j-1}}}\left\|\overline{\Pi_{j-1}\Pi_b}A_{j-1}\cdots A_1|\psi_0\rangle\right\|\mathbf{Cost}\left(A_j\right)\Bigg)\\
&=\mathbf{O}\Bigg(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(O_b\right)+\frac{1}{\sqrt{p_{\text{succ}}}}\log\left(\frac{1}{\epsilon_{\text{bm}}}\right)\mathbf{Cost}\left(O_A\right)+\frac{1}{\sqrt{p_{\text{succ}}}}\sum_{j=1}^{m-l+1}3^j\log\left(\frac{1}{\epsilon_{\text{gpe},j}}\right)\mathbf{Cost}(O_A)\\
&\quad+\frac{1}{\sqrt{p_{\text{succ}}}}\sum_{j=m-l+2}^{m}3^{m-l-j}3^j\log\left(\frac{1}{\epsilon_{\text{gpe},j}}\right)\mathbf{Cost}\left(O_A\right)\Bigg),
\end{aligned}\tag{113}$$

where we have again used $p_{\text{succ,d-inv}}=\mathbf{\Theta}\left(p_{\text{succ}}\right)$. Here, the cost of GPE grows exponentially with the stage number j like $\sim 3^j$, suggesting a non-uniform error schedule of the following form

$$\epsilon_{\text{gpe},j}=\begin{cases}\frac{\epsilon_{\text{gpe}}}{l}, & j=m-l+2,\ldots,m,\\ \frac{\epsilon_{\text{gpe}}}{l\cdot 2^{m-l+1-j}}, & j=1,\ldots,m-l+1.\end{cases}\tag{114}$$

This then leads to the cost

$$\begin{aligned}
&\mathbf{O}\Bigg(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(O_b\right)+\frac{1}{\sqrt{p_{\text{succ}}}}\log\left(\frac{1}{\epsilon_{\text{bm}}}\right)\mathbf{Cost}\left(O_A\right)\\
&\quad+\frac{1}{\sqrt{p_{\text{succ}}}}\sum_{j=1}^{m-l+1}3^j(m-l+1-j)\log\left(\frac{l}{\epsilon_{\text{gpe}}}\right)\mathbf{Cost}(O_A)\\
&\quad+\frac{1}{\sqrt{p_{\text{succ}}}}\sum_{j=m-l+2}^{m}3^{m-l-j}3^j\log\left(\frac{l}{\epsilon_{\text{gpe}}}\right)\mathbf{Cost}\left(O_A\right)\Bigg)\\
&=\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(O_b\right)+\frac{1}{\sqrt{p_{\text{succ}}}}\left(\log\left(\frac{1}{\epsilon_{\text{bm}}}\right)+\frac{l\kappa}{3^l}\log\left(\frac{l}{\epsilon_{\text{gpe}}}\right)\right)\mathbf{Cost}\left(O_A\right)\right),
\end{aligned}\tag{115}$$

which decreases as l increases. So we should choose the largest possible l:

$$l = \mathbf{Floor}\left(\log_3\left(\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\right)\right) = \mathbf{\Theta}\left(\log_3\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right). \tag{116}$$

Analysis of success probability As an immediate consequence of the choice of schedule, we obtain

$$\begin{aligned}\left\|\overline{\Pi_m\Pi_b}\widetilde{A}_m|\psi_0\rangle\right\| &\geq \frac{5}{6}3^l\left\|\overline{\Pi_m\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\| \\ &\geq \frac{5}{6}3^{\log_3\left(\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\right)-1}\sqrt{p_{\text{succ,d-inv}}} \\ &= \frac{5}{6}\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\frac{1}{3}\sqrt{p_{\text{succ,d-inv}}} = \frac{\sqrt{5}}{9c}.\end{aligned} \tag{117}$$

So VTAA outputs the normalized version of discretized inverse state $\psi_{\text{d-inv}}$ with a constant success probability.

Analysis of error With this deterministic schedule, the initial state $|b\rangle$ and input algorithms $A_1,\ldots,A_{m-l+1}$ are invoked 3^l times. After that, algorithm A_j is invoked 3^{m-j+1} times for $j = m-l+2,\ldots,m$. So the total error of preparing the discretized inverse state is at most

$$3^l\epsilon_{\text{bm}} + 3^l\sum_{j=1}^{m-l+1}\frac{\epsilon_{\text{gpe}}}{l\cdot 2^{m-l+1-j}} + \sum_{j=m-l+2}^{m}3^{m-j+1}\frac{\epsilon_{\text{gpe}}}{l} = \mathbf{O}\left(3^l\left(\epsilon_{\text{bm}}+\epsilon_{\text{gpe}}\right)\right) = \mathbf{O}\left(\frac{\epsilon_{\text{bm}}+\epsilon_{\text{gpe}}}{\sqrt{p_{\text{succ}}}}\right). \tag{118}$$

Thus, by setting

$$\epsilon_{\text{bm}}, \epsilon_{\text{gpe}} = \mathbf{\Theta}\left(\sqrt{p_{\text{succ}}}\epsilon\right), \tag{119}$$

we can generate the discretized inverse state $\psi_{\text{d-inv}}$ with constant probability, accuracy ϵ and query complexity

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(O_b\right) + \kappa\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\log\left(\frac{1}{\sqrt{p_{\text{succ}}}\epsilon}\right)\mathbf{Cost}\left(O_A\right)\right). \tag{120}$$

In the next subsection, we show how to project the error into potentially good subspaces to improve the query complexity to

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(O_b\right) + \kappa\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\log\left(\frac{\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)}{\epsilon}\right)\mathbf{Cost}\left(O_A\right)\right). \tag{121}$$

4.3 Projecting error toward potentially good subspaces

In this subection, we present an improved error analysis for preparing the normalized discretized inverse state by projecting error onto potentially good subspaces.

To begin with, recall that if branch marking were perfectly performed, our discretized inverse state would take the form

$$\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}\sum_{j=1}^{m}\zeta_{j,u}|j-1\rangle\left(\frac{3^j}{3^m}|\text{good}\rangle + \sqrt{1-\frac{9^j}{9^m}}|\text{bad}\rangle\right)\gamma_u|\phi_u\rangle, \tag{122}$$

where $\zeta_{j,u} = \xi_{j,u,0}\prod_{l=1}^{j-1}\xi_{l,u,1}$ are the cumulative coefficients, such that the normalization condition $|\xi_{l,u,0}|^2 + |\xi_{l,u,1}|^2 = 1$ holds at all stages l. If GPEs were also performed perfectly, then majority of $\zeta_{j,u}$ would be zero unless $j = k, k+1$ and we get

$$\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}\left(\zeta_{k+1,u}|k\rangle\left(\frac{3^{k+1}}{3^m}|\text{good}\rangle + \sqrt{1-\frac{9^{k+1}}{9^m}}|\text{bad}\rangle\right)\right.$$
$$\left. + \zeta_{k,u}|k-1\rangle\left(\frac{3^k}{3^m}|\text{good}\rangle + \sqrt{1-\frac{9^k}{9^m}}|\text{bad}\rangle\right)\right)\gamma_u|\phi_u\rangle. \tag{123}$$

Let us compare these two states, but only within the potentially good subspaces. That is, we compare

$$\psi_{\text{d-inv},m} = \sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}\sum_{j=1}^{m}\zeta_{j,u}|j-1\rangle\frac{3^j}{3^m}\gamma_u|\phi_u\rangle, \qquad \|\psi_{\text{d-inv},m}\|^2 = p_{\text{succ,d-inv},m},$$
$$\psi_{\text{d-inv}} = \sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}\left(\zeta_{k+1,u}|k\rangle\frac{3^{k+1}}{3^m} + \zeta_{k,u}|k-1\rangle\frac{3^k}{3^m}\right)\gamma_u|\phi_u\rangle, \qquad \|\psi_{\text{d-inv}}\|^2 = p_{\text{succ,d-inv}}. \tag{124}$$

Then their squared Euclidean distance is bounded by

$$\|\psi_{\text{d-inv},m} - \psi_{\text{d-inv}}\|^2 \le \sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j\neq k,k+1}|\zeta_{j,u}|^2\frac{9^j}{9^m}. \tag{125}$$

We divide the analysis into two cases.

(i) $j \le k-1$: In this case, we use the bound

$$\begin{cases}|\xi_{l,u,1}| \le 1, & l = 1,\dots,j-1,\\ |\xi_{j,u,0}| \le \epsilon_{\text{qpe},j}.\end{cases} \tag{126}$$

to obtain

$$|\zeta_{j,u}| \le \epsilon_{\text{gpe},j}. \tag{127}$$

(ii) $j \ge k+2$: In this case, we use the bound

$$\begin{cases}|\xi_{l,u,1}| \le 1, & l = 1,\dots,k,\\ |\xi_{l,u,1}| \le \epsilon_{\text{gpe},l}, & l = k+1,\dots,j-1,\\ |\xi_{j,u,0}| \le 1,\end{cases} \tag{128}$$

to obtain

$$|\zeta_{j,u}| \le \prod_{l=k+1}^{j-1}\epsilon_{\text{gpe},l}. \tag{129}$$

This bounds the squared distance by

$$\begin{aligned}\|\psi_{\text{d-inv},m}-\psi_{\text{d-inv}}\|^2 &\le \sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\left(\sum_{j=1}^{k-1}\epsilon_{\text{gpe},j}^2\frac{9^j}{9^m}+\sum_{j=k+2}^{m}\prod_{l=k+1}^{j-1}\epsilon_{\text{gpe},l}^2\frac{9^j}{9^m}\right)\\ &\le \sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\left(\sum_{j=1}^{k-1}\epsilon_{\text{gpe},j}^2\frac{9^{k-1}}{9^m}+\sum_{j=k+2}^{m}\epsilon_{\text{gpe},j-1}^2\frac{9^{k+2}}{9^m}\right)\\ &\le \left(\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\frac{9^{k+2}}{9^m}\right)\left(\sum_{j=1}^{m}\epsilon_{\text{gpe},j}\right)^2\le 9p_{\text{succ,d-inv},1}\epsilon_{\text{gpe}}^2,\end{aligned}\tag{130}$$

where

$$p_{\text{succ,d-inv},1}=\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\frac{9^{k+1}}{9^m}.\tag{131}$$

Here, we assume all $\epsilon_{\text{gpe},l}\le\frac{1}{3}$ in the second inequality and upper bound the ℓ_2-norm by ℓ_1-norm in the third inequality.

Now we take the error of branch marking into account. If the actual output state is $\psi_{\text{d-inv,bm}}$ with $\|\psi_{\text{d-inv,bm}}\|^2=p_{\text{succ,d-inv,bm}}$, then

$$\|\psi_{\text{d-inv,bm}}-\psi_{\text{d-inv},m}\|\le 2\epsilon_{\text{bm}},\tag{132}$$

implying through the triangle inequality that

$$\|\psi_{\text{d-inv,bm}}-\psi_{\text{d-inv}}\|=\mathbf{O}\left(\sqrt{p_{\text{succ,d-inv},1}}\epsilon_{\text{gpe}}+\epsilon_{\text{bm}}\right).\tag{133}$$

Hence, by [39, Lemma 24],

$$\left\|\frac{\psi_{\text{d-inv,bm}}}{\sqrt{p_{\text{succ,d-inv,bm}}}}-\frac{\psi_{\text{d-inv}}}{\sqrt{p_{\text{succ,d-inv}}}}\right\|\le\frac{2\|\psi_{\text{d-inv,bm}}-\psi_{\text{d-inv}}\|}{\sqrt{p_{\text{succ,d-inv}}}}=\mathbf{O}\left(\epsilon_{\text{gpe}}+\frac{\epsilon_{\text{bm}}}{\sqrt{p_{\text{succ}}}}\right),\tag{134}$$

where we have used $p_{\text{succ,d-inv}}=\mathbf{\Theta}(p_{\text{succ,d-inv},1})=\mathbf{\Theta}(p_{\text{succ}})$ from Proposition 25. We see that the error of GPE shrinks by a factor of $\mathbf{\Theta}\left(\sqrt{p_{\text{succ}}}\right)$ when projected onto the potentially good subspaces, which cancels with the state normalization factor $\mathbf{\Theta}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$. Hence, we choose

$$\epsilon_{\text{gpe}}=\mathbf{\Theta}(\epsilon),\qquad \epsilon_{\text{bm}}=\mathbf{\Theta}\left(\sqrt{p_{\text{succ}}}\epsilon\right),\tag{135}$$

obtaining the claimed query complexity of preparing discretized inverse state

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}\left(O_b\right)+\kappa\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\log\left(\frac{\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)}{\epsilon}\right)\mathbf{Cost}\left(O_A\right)\right).\tag{136}$$

4.4 Solution norm estimation

Up to this point, we have assumed that a constant multiplicative estimate of the solution norm $\left\||A^{-1}|b\rangle\right\|$ is available a prior. This assumption is equivalent to knowing p_{succ} and $p_{\text{succ,d-inv}}$ to a constant multiplicative accuracy, which is necessary to achieve the $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$ scaling for initial state preparation. In this subsection, we show that this strictly linear scaling can be attained even with an unknown solution norm, when p_{succ} is replaced by its lower bound $\alpha_{p_{\text{succ}}} \leq p_{\text{succ}}$. In fact, we will describe a solution norm estimation algorithm whose query complexity of the initial state oracle has the scaling $\mathbf{O}\left(\frac{1}{\sqrt{\alpha_{p_{\text{succ}}}}}\right)$.

We begin by introducing the amplitude estimation algorithm.

Lemma 7 (Amplitude estimation)**.** *Let U be a unitary and Π be an orthogonal projection. Then for any $\epsilon, \delta > 0$, there exists a quantum algorithm that outputs y with*

$$\mathbf{P}\left(|y - \|\Pi U|0\rangle\|| \geq \epsilon\right) < \delta \tag{137}$$

using

$$\mathbf{O}\left(\frac{1}{\epsilon}\log\left(\frac{1}{\delta}\right)\right) \tag{138}$$

queries to controlled-U, controlled-reflection $I - 2\Pi$ and their inverses.

Note that in applications where the target accuracy $\epsilon = \mathbf{\Theta}(1)$ is constant, it is possible to perform amplitude estimation by repeatedly measuring outcome of the algorithm and applying Chernoff bound. Otherwise if a smaller ϵ is desired, we can use quantum phase estimation [11].

To simplify the discussion, we first consider the discretized inverse state with an unknown $p_{\text{succ,d-inv}} = \|\psi_{\text{d-inv}}\|^2$ but a known lower bound $\alpha_{p_{\text{succ,d-inv}}} \leq p_{\text{succ,d-inv}}$. We then (mathematically) define

$$l^* = \mathbf{Floor}\left(\log_3\left(\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\right)\right). \tag{139}$$

If we run Tunable VTAA with l^*, our analysis in Section 4.2 then shows that the success amplitude is at least

$$\left\|\overline{\Pi_m\Pi_b}\widetilde{A}_m|\psi_0\rangle\right\| \geq \frac{\sqrt{5}}{9c}. \tag{140}$$

Our claim is that if $l < l^*$ is sufficiently small, the success amplitude is constant gapped below $\frac{\sqrt{5}}{9c}$. Indeed, if $l \leq l^* - 2$,

$$\begin{aligned}\left\|\overline{\Pi_m\Pi_b}\widetilde{A}_m|\psi_0\rangle\right\| &\leq 3^l\left\|\overline{\Pi_m\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\| \leq 3^{l^*-2}\sqrt{p_{\text{succ,d-inv}}} \\ &\leq 3^{\log_3\left(\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\right)-2}\sqrt{p_{\text{succ,d-inv}}} \\ &= \frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\frac{1}{9}\sqrt{p_{\text{succ,d-inv}}} = \frac{2\sqrt{5}}{45c}.\end{aligned} \tag{141}$$

Our solution norm estimation algorithm proceeds as follows. We run Tunable VTAA for pre-merging parameter $l = 1, 2, \ldots$ and perform amplitude estimation with a sufficiently small (yet constant) accuracy and failure probability δ_l. When iterating through $l = 1, 2, \ldots$, with high probability we see the estimated amplitude below $\frac{\sqrt{5}}{15c}$ for all $l = 1, \ldots, l^* - 2$ and amplitude above

$\frac{4\sqrt{5}}{45c}$ for either $l = l^* - 1, l^*$. Supposing the first time we see the amplitude above $\frac{4\sqrt{5}}{45c}$ is at some l, we consider the value $l+1$. Then, we have

$$\log_3\left(\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\right) - 1 \leq l^* \leq l+1 \leq l^*+1 \leq \log_3\left(\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}\right) + 1, \tag{142}$$

which implies

$$\frac{1}{3}\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}} \leq 3^{l+1} \leq 3\frac{2}{\sqrt{5}c\sqrt{p_{\text{succ,d-inv}}}}. \tag{143}$$

Hence, $\frac{2}{\sqrt{5}3^{l+1}c}$ is a 3-multiplicative approximation of $\sqrt{p_{\text{succ,d-inv}}}$.

Note that for each fixed l, Tunable VTAA has query complexity

$$\begin{aligned}
&3^l\mathbf{Cost}\left(A_{m-l+1}\cdots A_1|\psi_0\rangle\right) + \sum_{j=m-l+2}^{m} 3^{m-j+1}\mathbf{Cost}\left(A_j\right) \\
&= \mathbf{O}\left(3^l\mathbf{Cost}(O_b) + 3^l\sum_{j=1}^{m-l+1} 3^j \log\left(2^{m-l+1-j}l\right)\mathbf{Cost}(O_A) + \sum_{j=m-l+2}^{m} 3^{m-j+1}3^j\log(l)\mathbf{Cost}(O_A)\right) \\
&= \mathbf{O}\left(3^l\mathbf{Cost}(O_b) + \kappa l\log(l)\mathbf{Cost}(O_A)\right).
\end{aligned} \tag{144}$$

The total query complexity is thus bounded by

$$\mathbf{O}\left(\sum_{l=1}^{l^*} 3^l \log\left(\frac{1}{\delta_l}\right)\mathbf{Cost}(O_b) + \sum_{l=1}^{l^*}\kappa l\log(l)\log\left(\frac{1}{\delta_l}\right)\mathbf{Cost}(O_A)\right). \tag{145}$$

Let us choose the schedule of failure probabilities

$$\delta_l = \frac{1}{(\alpha_{l^*} - l + 3)^2}, \qquad \alpha_{l^*} = \mathbf{Floor}\left(\log_3\left(\frac{2}{\sqrt{5}c\sqrt{\alpha_{p_{\text{succ,d-inv}}}}}\right)\right) \geq l^*. \tag{146}$$

By the union bound, the total failure probability is at most

$$\sum_{l=1}^{l^*}\frac{1}{(\alpha_{l^*}-l+3)^2} \leq \sum_{l=1}^{\alpha_{l^*}}\frac{1}{(\alpha_{l^*}-l+3)^2} \leq \frac{1}{3^2} + \frac{1}{4^2} + \cdots = \frac{\pi^2}{6} - \frac{5}{4} \approx 0.4. \tag{147}$$

The query complexity then becomes

$$\begin{aligned}
&\mathbf{O}\left(\sum_{l=1}^{l^*} 3^l \log\left(\alpha_{l^*}-l+3\right)\mathbf{Cost}(O_b) + \sum_{l=1}^{l^*}\kappa l\log(l)\log\left(\alpha_{l^*}-l+3\right)\mathbf{Cost}(O_A)\right) \\
&= \mathbf{O}\left(\frac{1}{\sqrt{\alpha_{p_{\text{succ,d-inv}}}}}\mathbf{Cost}(O_b) + \kappa\log^2\left(\frac{1}{\sqrt{\alpha_{p_{\text{succ,d-inv}}}}}\right)\log\log^2\left(\frac{1}{\sqrt{\alpha_{p_{\text{succ,d-inv}}}}}\right)\mathbf{Cost}(O_A)\right).
\end{aligned} \tag{148}$$

The above analysis assumes that GPE and branch marking can be performed perfectly, and that a lower bound $\alpha_{p_{\text{succ,d-inv}}} \leq p_{\text{succ,d-inv}}$ is known a prior. In practice, the output state is $\psi_{\text{d-inv,bm}}$ with $\|\psi_{\text{d-inv,bm}}\|^2 = p_{\text{succ,d-inv,bm}}$, but we would still proceed as above, except we replace all $p_{\text{succ,d-inv}}$ by $p_{\text{succ,d-inv,bm}}$. As per Proposition 26, we can also convert the given lower bound $\alpha_{p_{\text{succ}}} \leq p_{\text{succ}}$ into some $\alpha_{p_{\text{succ,d-inv,bm}}} \leq p_{\text{succ,d-inv,bm}}$. Consequently, what we actually obtain is a 3-multiplicative approximation of $\sqrt{p_{\text{succ,d-inv,bm}}}$. Invoking Proposition 26 once more, we obtain a constant multiplicative approximation of $\sqrt{p_{\text{succ}}}$.

Theorem 1 (Solution norm estimation with optimal initial state preparation)**.** *Let A be the coefficient matrix such that A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Let $|b\rangle$ be the initial state prepared by oracle O_b. Then the solution norm $\left\|A^{-1}|b\rangle\right\|$, and hence the success amplitude $\sqrt{p_{succ}} = \frac{\left\|A^{-1}|b\rangle\right\|}{\alpha_{A^{-1}}}$, can be estimated to a constant multiplicative accuracy and success probability $> \frac{1}{2}$ with query complexity*

$$\mathbf{O}\left(\frac{1}{\sqrt{\alpha_{p_{succ}}}}\mathbf{Cost}(O_b) + \kappa \log^2\left(\frac{1}{\sqrt{\alpha_{p_{succ}}}}\right) \log\log^2\left(\frac{1}{\sqrt{\alpha_{p_{succ}}}}\right) \mathbf{Cost}(O_A)\right), \tag{149}$$

where $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ is a norm upper bound on the inverse matrix, $\kappa = \alpha_A \alpha_{A^{-1}}$ is an upper bound on the spectral condition number, and $\alpha_{p_{succ}} \leq p_{succ}$ is a lower bound on the success probability. The algorithm uses a nested amplitude amplification with deterministic schedule Eq. (112) and an increasing choice of pre-merging parameter $l = 1, 2, \ldots$

5 Solving linear systems

We now consider the quantum linear system problem. In Section 5.1, we show that the problem can be solved by inverting the input matrix over the discretized inverse state. Combining with the VTAA algorithm from the previous section, we provide a simplified quantum linear system algorithm with optimal queries to the initial state preparation and nearly optimal queries to the coefficient matrix block encoding. We provide a proof of the optimality in Section 5.2.

5.1 Matrix inversion over discretized inverse state

Recall from Section 4 that we can use Tunable VTAA to prepare the following state with accuracy ϵ and constant success probability

$$\begin{aligned}
&\frac{1}{\sqrt{p_{\text{succ,d-inv}}}} \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \left(\zeta_{k+1,u} \frac{3^{k+1}}{3^m}|k\rangle + \zeta_{k,u} \frac{3^k}{3^m}|k-1\rangle\right) \gamma_u |\phi_u\rangle, \\
&p_{\text{succ,d-inv}} = \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \left(|\zeta_{k+1,u}|^2 \frac{9^{k+1}}{9^m} + |\zeta_{k,u}|^2 \frac{9^k}{9^m}\right).
\end{aligned} \tag{150}$$

This state differs from the solution state in that the reciprocal of eigenvalues are replaced by the discrete values $\frac{3^k}{3^m}$. To get the actual solution state, we perform a block-encoded matrix inversion based on the following gadget.

Lemma 8 (Block encoding inversion [22, Corollary 69])**.** *Let A be a matrix such that A/α_A is block encoded by O_A with some normalization factor $\alpha_A \geq \|A\|$. Then the operator*

$$\frac{A^{-1}}{2\alpha_{A^{-1}}} \tag{151}$$

can be block encoded with accuracy ϵ using

$$\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right)\right) \tag{152}$$

queries to the controlled-O_A and its inverse, where $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ is a norm upper bound on the inverse matrix, and $\kappa = \alpha_A \alpha_{A^{-1}}$ is an upper bound on the spectral condition number.

In our case, we apply the block-encoded inversion of the input matrix with accuracy ϵ_{blk}, controlled by the clock register:

$$\sum_{k=0}^{m-1} |k\rangle\langle k| \otimes \frac{\Pi_{[\frac{1}{3^{k+2}},1)} A^{-1} \Pi_{[\frac{1}{3^{k+2}},1)}}{2 \cdot \frac{3^{k+2}}{\alpha_A}}, \qquad \Pi_{\mathcal{S}} = \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \mathcal{S}} |\phi_u\rangle\langle\phi_u|. \tag{153}$$

Here, the clock state $|k\rangle$ corresponds to eigenvalues of the input matrix within the interval $\left|\frac{\lambda_u}{\alpha_A}\right| \in [\frac{1}{3^{k+2}}, \frac{1}{3^k})$, thus to fully cover all possible eigenvalues, we perform the block-encoded matrix inversion over $[\frac{1}{3^{k+2}}, 1)$. This block encoding can be implemented with query complexity $\mathbf{O}\left(\kappa \log(1/\epsilon_{\text{blk}})\right)$, by using the $|k\rangle$ register to control the rotations in the quantum signal processing circuit [37], and the entire operator has accuracy ϵ_{blk} due to its block diagonal structure. With a high probability, we then transform the state into

$$\begin{aligned} &\frac{1}{\sqrt{p_{\text{succ,blk}}}} \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \left(\zeta_{k+1,u} \frac{1}{3^m} \frac{\alpha_A}{2 \cdot 3} \frac{1}{\lambda_u} |k\rangle + \zeta_{k,u} \frac{1}{3^m} \frac{\alpha_A}{2 \cdot 3} \frac{1}{\lambda_u} |k-1\rangle\right) \gamma_u |\phi_u\rangle, \\ &p_{\text{succ,blk}} = \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \left|\frac{1}{3^m} \frac{\alpha_A}{2 \cdot 3} \frac{1}{\lambda_u}\right|^2 = \frac{p_{\text{succ}}}{36}. \end{aligned} \tag{154}$$

This state can be produced with probability close to unity by another amplitude amplification. Finally, we uncompute the clock register by reversing the GPE (without performing rotations $|\text{good}\rangle \mapsto \frac{3^j}{3^m}|\text{good}\rangle + \sqrt{1 - \frac{9^j}{9^m}}|\text{bad}\rangle$ or VTAA), obtaining the normalized solution state

$$\frac{\sum_u \frac{1}{\lambda_u} \gamma_u |\phi_u\rangle}{\sqrt{\sum_u \left|\frac{\gamma_u}{\lambda_u}\right|^2}} = \frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}. \tag{155}$$

As per Proposition 25, we only need at most constant rounds of amplitude amplifications beyond the preparation of discretized inverse state by Tunable VTAA. We have thus established the main algorithm.

Theorem 2 (Quantum linear system algorithm with optimal initial state preparation)**.** *Let A be the coefficient matrix such that A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Let $|b\rangle$ be the initial state prepared by oracle O_b. Suppose that the solution norm $\left\|A^{-1}|b\rangle\right\|$, and hence the success amplitude $\sqrt{p_{succ}} = \frac{\left\|A^{-1}|b\rangle\right\|}{\alpha_{A^{-1}}}$, can be estimated to a constant multiplicative accuracy. Then the quantum state*

$$\frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|} \tag{156}$$

can be prepared to accuracy ϵ and success probability $> \frac{1}{2}$ with query complexity

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{succ}}} \mathbf{Cost}\left(O_b\right) + \kappa \log\left(\frac{1}{\sqrt{p_{succ}}}\right) \log\left(\frac{\log\left(\frac{1}{\sqrt{p_{succ}}}\right)}{\epsilon}\right) \mathbf{Cost}\left(O_A\right)\right), \tag{157}$$

where $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ is a norm upper bound on the inverse matrix, and $\kappa = \alpha_A \alpha_{A^{-1}}$ is an upper bound on the spectral condition number. The algorithm uses a nested amplitude amplification with deterministic schedule Eq. (112) and pre-merging parameter $l = \mathbf{\Theta}\left(\log_3\left(\frac{1}{\sqrt{p_{succ}}}\right)\right)$.

5.2 Lower bound

Now, we prove that our query complexity to the initial state preparation is optimal, by showing how the quantum search problem [24] can be solved by a linear system algorithm. We note that a similar lower bound [51] have been previously derived.

Specifically, consider the d-dimensional initial state

$$|b\rangle = \frac{1}{\sqrt{d}} \left(\sum_{j \neq w} |j\rangle - |w\rangle \right), \tag{158}$$

where $|w\rangle \in \{|0\rangle, |1\rangle, \ldots, |d-1\rangle\}$ is an arbitrary unknown basis state. This state can be (and only be) prepared by making 1 query to the Grover oracle with the uniform superposition state. We can perform an orthogonal decomposition of $|b\rangle$ along the uniform superposition state to get

$$|b\rangle = \frac{d-2}{d}|+\rangle + \frac{2\sqrt{d-1}}{d}|-_b\rangle, \tag{159}$$

where

$$|+\rangle = \frac{1}{\sqrt{d}} \sum_j |j\rangle, \qquad |-_b\rangle = \sum_{j \neq w} \frac{1}{\sqrt{(d-1)d}}|j\rangle + \sqrt{\frac{d-1}{d}}|w\rangle. \tag{160}$$

Now, we define the coefficient matrix

$$A = \frac{1}{\sqrt{d}} \left(I - |+\rangle\langle+| \right) + |+\rangle\langle+|. \tag{161}$$

Note that A can be block encoded with normalization factor $\alpha_A = 1$ and zero query to the Grover oracle. Then the inverse matrix

$$A^{-1} = \sqrt{d} \left(I - |+\rangle\langle+| \right) + |+\rangle\langle+| \tag{162}$$

has spectral norm

$$\left\| A^{-1} \right\| = \sqrt{d} = \alpha_{A^{-1}}. \tag{163}$$

We now invoke the quantum linear system algorithm, obtaining a state within Euclidean distance ϵ_{lin} to the solution

$$\frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}. \tag{164}$$

Here,

$$\begin{aligned} A^{-1}|b\rangle &= 2\sqrt{\frac{d-1}{d}}|-_b\rangle + \frac{d-2}{d}|+\rangle \\ &= \sum_{j \neq w} \frac{d-2+2\sqrt{d}}{d\sqrt{d}}|j\rangle - \frac{2(d-1)\sqrt{d}-d+2}{d\sqrt{d}}|w\rangle, \end{aligned} \tag{165}$$

which implies

$$\left\| A^{-1}|b\rangle \right\| = \sqrt{\frac{5d^2-8d+4}{d^2}}. \tag{166}$$

So the inverse success amplitude is

$$\frac{\left\| A^{-1} \right\|}{\|A^{-1}|b\rangle\|} = \sqrt{d}\sqrt{\frac{d^2}{5d^2-8d+4}} = \mathbf{\Theta}\left(\sqrt{d}\right). \tag{167}$$

Meanwhile, the absolute value of amplitude of the desired $|w\rangle$ is lower bounded by

$$\begin{aligned}\left|\frac{\langle w|A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}\right| - \epsilon_{\text{lin}} &\geq \left|\frac{\frac{2(d-1)\sqrt{d}+d-2}{d\sqrt{d}}}{\sqrt{\frac{5d^2-8d+4}{d^2}}}\right| - \epsilon_{\text{lin}} \geq \frac{\frac{2(d-1)\sqrt{d}-2}{d\sqrt{d}}}{\sqrt{\frac{5d^2+4}{d^2}}} - \epsilon_{\text{lin}} \\ &\geq \frac{\frac{2(d-2)\sqrt{d}}{d\sqrt{d}}}{\sqrt{\frac{5(d+1)^2}{d^2}}} - \epsilon_{\text{lin}} = \frac{2}{\sqrt{5}}\frac{d-2}{d+1} - \epsilon_{\text{lin}} \\ &= \frac{2}{\sqrt{5}}\left(1 - \frac{3}{d+1}\right) - \epsilon_{\text{lin}} \geq \frac{13}{8\sqrt{5}} - \epsilon_{\text{lin}}\end{aligned} \tag{168}$$

when $d \geq 15$. Let us choose

$$\epsilon_{\text{lin}} = \frac{1}{60}, \tag{169}$$

so the probability of outcome $|w\rangle$ is at least

$$\left(\frac{13}{8\sqrt{5}} - \frac{1}{60}\right)^2 > 0.504. \tag{170}$$

This thus solves the Grover search problem, and we conclude that the query complexity to the initial state is at least

$$\mathbf{\Omega}\left(\sqrt{d}\right) = \mathbf{\Omega}\left(\frac{\left\|A^{-1}\right\|}{\|A^{-1}|b\rangle\|}\right). \tag{171}$$

Theorem 3 (Lower bound on initial state queries)**.** *For any $0 < p < 1$, there exists a linear system with coefficient matrix A and initial state $|b\rangle$ satisfying*

$$\frac{\left\|A^{-1}|b\rangle\right\|^2}{\left\|A^{-1}\right\|^2} = \mathbf{\Theta}(p), \tag{172}$$

such that

$$\begin{aligned}\min\Bigg\{g \in \mathbb{Z}_{\geq 0} \;\Big|\; &\exists \text{ a quantum algorithm } V \text{ making } g \text{ queries to } O_b, \\ &\left\|V|0\rangle - \frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}\right\| = \mathbf{O}(1) \text{ sufficiently small}\Bigg\} = \mathbf{\Omega}\left(\frac{1}{\sqrt{p}}\right).\end{aligned} \tag{173}$$

6 Block preconditioning

In this section, we introduce the technique of *block preconditioning*, and describe its algorithmic applications. Specifically, we present a simple quantum linear system solver in Section 6.1 with optimal queries to the coefficient matrix block encoding by choosing the initial state itself as the preconditioner. Combining block preconditioning with our main algorithms Theorem 1 and Theorem 2, we then show how to reduce the cost of initial state preparation in solving differential equations (Section 6.2), estimating real eigenvalues of non-normal matrices (Section 6.3), transforming matrices with real spectra and preparing their ground states (Section 6.4). Additionally, we apply block preconditioning to improve the block encoded version of quantum eigenvalue transformer in Section 6.5.

6.1 Quantum linear system solver

Our quantum linear system algorithm in Theorem 2 has an optimal query complexity of the initial state preparation, and a nearly optimal cost of the coefficient block encoding. We now show that it is possible to solve a preconditioned quantum linear system problem, producing the same solution state but making an optimal number of queries to the block encoding oracle.

Specifically, we choose the preconditioner to be the initial state $|b\rangle$ itself and define the scaling matrix

$$S = s|b\rangle\langle b| + (I - |b\rangle\langle b|), \qquad S^{-1} = \frac{1}{s}|b\rangle\langle b| + (I - |b\rangle\langle b|), \qquad 0 < s < 1. \tag{174}$$

Then solving the preconditioned quantum linear system problem would produce the same solution state because

$$\frac{(SA)^{-1}|b\rangle}{\|(SA)^{-1}|b\rangle\|} = \frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|}. \tag{175}$$

We can block encode S with normalization factor 1 using 2 queries to the initial state oracle O_b through the linear combination

$$S = O_b \left(\frac{1-s}{2}(I - 2|0\rangle\langle 0|) + \frac{1+s}{2} I \right) O_b^\dagger. \tag{176}$$

This then yields a block encoding of SA/α_A with normalization factor α_A same as that of the input matrix. Meanwhile, the inverse matrix has the norm bound

$$\begin{aligned} \left\|(SA)^{-1}\right\| = \left\|A^{-1}S^{-1}\right\| &= \left\| \frac{1}{s} A^{-1}|b\rangle\langle b| + A^{-1}\left(I - |b\rangle\langle b|\right) \right\| \\ &\leq \sqrt{\frac{\left\|A^{-1}|b\rangle\right\|^2}{s^2} + \left\|A^{-1}\left(I - |b\rangle\langle b|\right)\right\|^2} \leq \sqrt{\frac{\left\|A^{-1}|b\rangle\right\|^2}{s^2 \left\|A^{-1}\right\|^2} + 1}\left\|A^{-1}\right\|, \end{aligned} \tag{177}$$

whereas the solution norm becomes

$$\left\|(SA)^{-1}|b\rangle\right\| = \frac{1}{s}\left\|A^{-1}|b\rangle\right\|. \tag{178}$$

Suppose we have a constant multiplicative approximation of the solution norm $\left\|A^{-1}|b\rangle\right\|$, say

$$\frac{t}{2} < \left\|A^{-1}|b\rangle\right\| < 2t \quad \Leftrightarrow \quad \frac{\left\|A^{-1}|b\rangle\right\|}{2} < t < 2\left\|A^{-1}|b\rangle\right\| \tag{179}$$

without loss of generality. Then we choose

$$s = \frac{t}{2\alpha_{A^{-1}}}, \qquad \frac{\left\|A^{-1}|b\rangle\right\|}{4\alpha_{A^{-1}}} < s < \frac{\left\|A^{-1}|b\rangle\right\|}{\alpha_{A^{-1}}} \leq \frac{\left\|A^{-1}|b\rangle\right\|}{\|A^{-1}\|} \leq 1. \tag{180}$$

This choice gives the norm bound on the preconditioned inverse matrix

$$\left\|(SA)^{-1}\right\| \leq \sqrt{\frac{\left\|A^{-1}|b\rangle\right\|^2}{s^2} + \|A^{-1}\|^2} \leq \sqrt{17}\alpha_{A^{-1}}, \tag{181}$$

and the preconditioned solution norm

$$\left\|(SA)^{-1}|b\rangle\right\| = \frac{1}{s}\left\|A^{-1}|b\rangle\right\| = \frac{2\alpha_{A^{-1}}\left\|A^{-1}|b\rangle\right\|}{t}, \tag{182}$$

which implies the success amplitude after block preconditioning

$$\sqrt{p_{\text{succ}}} = \frac{\left\|(SA)^{-1}|b\rangle\right\|}{\sqrt{17}\alpha_{A^{-1}}} = \frac{2\left\|A^{-1}|b\rangle\right\|}{\sqrt{17}t} \geq \frac{1}{\sqrt{17}}. \tag{183}$$

Thus, by invoking the QSVT-based linear system solver (Lemma 8), we can obtain the solution state to accuracy ϵ and constant success probability with query complexity

$$\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right) \mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\epsilon}\right) \mathbf{Cost}(O_A)\right). \tag{184}$$

Moreover, this approach does not require padding the target linear system, and appears to be conceptually even simpler than the kernel reflection method [19] which in turn simplifies the discrete adiabatic method [18].

Theorem 4 (Quantum linear system algorithm with optimal coefficient block encoding)**.** *Let A be the coefficient matrix such that A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Let $|b\rangle$ be the initial state prepared by oracle O_b. Suppose that the solution norm $\left\|A^{-1}|b\rangle\right\|$, and hence the success amplitude $\sqrt{p_{succ}} = \frac{\left\|A^{-1}|b\rangle\right\|}{\alpha_{A^{-1}}}$, can be estimated to a constant multiplicative accuracy. Then the quantum state*

$$\frac{A^{-1}|b\rangle}{\|A^{-1}|b\rangle\|} \tag{185}$$

can be prepared to accuracy ϵ and success probability $> \frac{1}{2}$ with query complexity

$$\mathbf{O}\left(\kappa \log\left(\frac{1}{\epsilon}\right) \mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\epsilon}\right) \mathbf{Cost}(O_A)\right), \tag{186}$$

where $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ is a norm upper bound on the inverse matrix, and $\kappa = \alpha_A \alpha_{A^{-1}}$ is an upper bound on the spectral condition number. The algorithm solves a preconditioned linear system specified by the scaling operator Eq. (174) and parameter Eq. (179) and Eq. (180).

6.2 Quantum differential equation solver

Differential equations arise naturally in a broad range of scientific disciplines including engineering, physics, economics, and biology. However, classical differential equation solvers can struggle to handle problems of large dimensions, which motivates the development of quantum algorithms. To be concrete, consider the system of first-order linear differential equations

$$\frac{\mathrm{d}}{\mathrm{d}t}x(t) = Ax(t), \qquad x(0) = b, \tag{187}$$

whose solution is given formally by

$$x(t) = e^{tA}b. \tag{188}$$

Here, we assume the coefficient matrix A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$ and the initial state $|b\rangle$ is prepared through the oracle O_b.

Many previous quantum differential equation algorithms proceed by recasting the problem as solving a system of linear equations, and then solve the recast problem using a quantum linear system solver. However, such algorithms have query complexity of O_b comparable to that of O_A, and are thus inefficient when preparing initial states incurs a substantial cost. We show that the cost of initial state preparation can be lowered using our Theorem 2 and block preconditioning,

nearly matching or outperforming the performance of alternative solvers for differential equations, and attaining the query complexity lower bound established in [20]. For illustration purposes, we focus on the differential equation solvers implementing the truncated Taylor series [9], although similar improvements can be achieved for other linear-system-based solvers.

In this algorithm, the coefficient matrix is given by

$$\begin{aligned}
&C_{n,k,p}\left(\frac{A}{\alpha_A}\right) \\
&= \sum_{i=0}^{n-1}\left(\sum_{j=0}^{k}|i(k+1)+j\rangle\langle i(k+1)+j|\otimes I - \sum_{j=1}^{k}|i(k+1)+j\rangle\langle i(k+1)+j-1|\otimes\frac{A}{j\alpha_A}\right) \\
&\quad - \sum_{i=0}^{n-1}\sum_{j=0}^{k}|(i+1)(k+1)\rangle\langle i(k+1)+j|\otimes I \\
&\quad + \left(\sum_{j=0}^{p}|n(k+1)+j\rangle\langle n(k+1)+j|\otimes I - \sum_{j=1}^{p}|n(k+1)+j\rangle\langle n(k+1)+j-1|\otimes I\right),
\end{aligned} \tag{189}$$

whereas the initial state is $|0\rangle|b\rangle$. Here, the parameter n denotes the total number of time steps in the algorithm, k denotes the Taylor truncate order of the evolution operator within each step, p refers to the additional padding steps required to boost the success probability, with the successful outcomes labeled by $n(k+1),\ldots,n(k+1)+p$. We set $p=n=\mathbf{\Theta}(\alpha_A t)$. Then after running the quantum linear system solver and measuring the ancilla register of the output state, we get the outcomes $n(k+1),\ldots,n(k+1)+p$ with probability at least

$$\mathbf{\Omega}\left(\frac{\left\|e^{tA}|b\rangle\right\|}{\max_{0\le\tau\le t}\left\|e^{\tau A}|b\rangle\right\|}\right), \tag{190}$$

which can then be boosted to unity by amplitude amplification. The resulting state has error at most $\mathbf{O}\left(\frac{\alpha_A t}{k!}\right)$, which implies the choice of $k=\mathbf{O}\left(\log\left(\frac{\alpha_A t}{\epsilon}\right)\right)$ to achieve a target accuracy ϵ. Meanwhile, the coefficient matrix satisfies

$$\left\|C_{n,k,p}\left(\frac{A}{\alpha_A}\right)\right\| = \mathbf{O}\left(\sqrt{k}\right), \qquad \left\|C^{-1}_{n,k,p}\left(\frac{A}{\alpha_A}\right)\right\| = \mathbf{O}\left(\max_{0\le\tau\le t}\left\|e^{\tau A}\right\|\sqrt{kn}\right). \tag{191}$$

With the quantum linear system solver [18], this method has a query complexity of

$$\mathbf{O}\left(\frac{\max_{0\le\tau\le t}\left\|e^{\tau A}|b\rangle\right\|}{\left\|e^{tA}|b\rangle\right\|}\max_{0\le\tau\le t}\left\|e^{\tau A}\right\|\alpha_A t\log\left(\frac{\alpha_A t}{\epsilon}\right)\log\left(\frac{\max_{0\le\tau\le t}\left\|e^{\tau A}|b\rangle\right\|}{\left\|e^{tA}|b\rangle\right\|\epsilon}\right)\left(\mathbf{Cost}\left(O_b\right)+\mathbf{Cost}\left(O_A\right)\right)\right). \tag{192}$$

The state-of-the-art result [8, 32] has a slightly better query complexity of O_b by shaving off the logarithmic factor $\log\left(\frac{\alpha_A t}{\epsilon}\right)$.

To improve over this result, we perform the following block preconditioning. We choose the preconditioner based on the initial ancilla state $\Pi_{\text{cond}}=|0\rangle\langle 0|\otimes I$ and set the scaling parameter $s=\frac{1}{\sqrt{kn}}$, defining the scaling operator

$$S=\frac{1}{\sqrt{kn}}|0\rangle\langle 0|\otimes I+(I-|0\rangle\langle 0|)\otimes I, \qquad S^{-1}=\sqrt{kn}|0\rangle\langle 0|\otimes I+(I-|0\rangle\langle 0|)\otimes I. \tag{193}$$

Note in particular that our preconditioner depends only on the ancilla state, and can be implemented without querying the oracle O_b. This block preconditioning increases the solution norm of quantum linear system algorithm to

$$\left\| \left(SC_{n,k,p}\left(\frac{A}{\alpha_A}\right)\right)|0\rangle|b\rangle \right\| = \frac{1}{s}\left\| C^{-1}_{n,k,p}\left(\frac{A}{\alpha_A}\right)|0\rangle|b\rangle \right\| = \sqrt{kn}\left\| C^{-1}_{n,k,p}\left(\frac{A}{\alpha_A}\right)|0\rangle|b\rangle \right\|, \tag{194}$$

while the norm of the inverse matrix remains asymptotically unaffected

$$\begin{aligned}\left\| \left(SC_{n,k,p}\left(\frac{A}{\alpha_A}\right)\right)^{-1} \right\| &\leq \sqrt{\frac{\left\| C^{-1}_{n,k,p}\left(\frac{A}{\alpha_A}\right)|0\rangle \otimes I \right\|^2}{s^2} + \left\| C^{-1}_{n,k,p}\left(\frac{A}{\alpha_A}\right)\right\|^2} \\ &= \mathbf{O}\left(\sqrt{kn}\sqrt{n}\max_{0\leq\tau\leq t}\left\|e^{\tau A}\right\| + \max_{0\leq\tau\leq t}\left\|e^{\tau A}\right\|\sqrt{kn}\right) \\ &= \mathbf{O}\left(\max_{0\leq\tau\leq t}\left\|e^{\tau A}\right\|\sqrt{kn}\right).\end{aligned} \tag{195}$$

Suppose that the norm of solution state of the quantum linear system solver can be estimated to a constant multiplicative accuracy. Then, the query complexity of the initial state preparation is bounded by

$$\mathbf{O}\left(\frac{\max_{0\leq\tau\leq t}\left\|e^{\tau A}\right\|\sqrt{kn}}{\sqrt{kn}\left\| C^{-1}_{n,k,p}\left(\frac{A}{\alpha_A}\right)|0\rangle|b\rangle\right\|}\cdot\frac{\sqrt{kn}\left\| C^{-1}_{n,k,p}\left(\frac{A}{\alpha_A}\right)|0\rangle|b\rangle\right\|}{\sqrt{kn}\sqrt{n}\left\|e^{tA}|b\rangle\right\|}\right) = \mathbf{O}\left(\frac{\max_{0\leq\tau\leq t}\left\|e^{\tau A}\right\|}{\left\|e^{tA}|b\rangle\right\|}\right). \tag{196}$$

We have thus obtained:

Theorem 5 (Quantum differential equation solver with optimal initial state preparation)**.** *Let A be a matrix such that A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Let $|b\rangle$ be the initial state prepared by oracle O_b. Then the quantum state*

$$\frac{e^{tA}|b\rangle}{\left\|e^{tA}|b\rangle\right\|} \tag{197}$$

can be prepared to accuracy ϵ and success probability $> \frac{1}{2}$ with query complexity

$$\begin{aligned}&\mathbf{O}\Bigg(\frac{\alpha_{\exp}}{\alpha_{\exp,b}}\mathbf{Cost}\left(O_b\right) \\ &\quad + \frac{\alpha_{\exp,b,\max}}{\alpha_{\exp,b}}\alpha_{\exp}\alpha_A t\log\left(\frac{\alpha_{\exp}}{\alpha_{\exp,b}}\right)\log\left(\frac{\log\left(\frac{\alpha_{\exp}}{\alpha_{\exp,b}}\right)}{\epsilon}\right)\log\left(\frac{\alpha_A t}{\epsilon}\right)\mathbf{Cost}\left(O_A\right)\Bigg),\end{aligned} \tag{198}$$

where $\alpha_{\exp} \geq \max_{0\leq\tau\leq t}\left\|e^{\tau A}\right\|$ is an upper bound on norm of the evolution operator, $\alpha_{\exp,b,\max} \geq \max_{0\leq\tau\leq t}\left\|e^{\tau A}|b\rangle\right\|$ is a norm upper bound on solution state of the differential equation, and $\alpha_{\exp,b} \leq \left\|e^{tA}|b\rangle\right\|$ is a lower bound on norm of the solution state.

Remark. The query complexity quoted above follows from Theorem 2, assuming that a constant multiplicative estimate of the solution norm is available for the linear system problem. Without this assumption, we can obtain such an estimate using Theorem 1, with the complexity of initial state preparation $\mathbf{O}\left(\frac{\alpha_{\exp}}{\alpha_{\exp,b}}\right)$ remaining the same. When $\alpha_{\exp} = \mathbf{O}(1)$ as is commonly assumed by recent work on differential equations [4, 8, 32], our complexity of initial state preparation matches the lower bound of [20, Theorem 10].

6.3 Quantum eigenvalue estimator

The efficient solution of the eigenvalue estimation problem underlies the quantum speedups for factoring integers [50] and elucidating chemical reactions [33, 57]. Here, an initial state $|\psi\rangle$ close to an eigenvector of the input matrix A is given, and the goal is to estimate the corresponding eigenvalue. If A is Hermitian, this can be solved using the quantum singular value estimation algorithm with optimal query complexity [12, 22, 30]. However, the problem becomes considerably more difficult when A is a nonnormal matrix, relevant for applications in simulating transcorrelated quantum chemistry [42] and non-Hermitian physics [6, 7].

For presentational purposes, we assume $A = S\Lambda S^{-1}$ is diagonalizable with real spectra, and A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Suppose that oracle $O_\psi|0\rangle = |\psi\rangle$ prepares an initial state close to an eigenstate $|\psi_j\rangle$ such that $A|\psi_j\rangle = \lambda_j|\psi_j\rangle$. Then recent work provides a linear-system-based quantum algorithm that estimates λ_j [39, Theorem 3] to accuracy ϵ and success probability $> \frac{1}{2}$, with query complexity

$$\mathbf{O}\left(\frac{\alpha_A \kappa_S}{\epsilon}\left(\mathbf{Cost}\left(O_A\right) + \mathbf{Cost}\left(O_\psi\right)\right)\right), \tag{199}$$

where $\kappa_S \geq \|S\|\left\|S^{-1}\right\|$ is an upper bound on the spectral condition number of the basis transformation. This improves over previous results [47, 48] based on differential equation solvers. A related result is obtained in [39, Theorem 12] for eigenvalue estimation on the unit circle, which is more recently generalized by [1]. In terms of the query complexity of O_A, the complexity quoted above exactly matches the Heisenberg scaling [23, 61] and is provably optimal for eigenvalue estimation. However, the algorithm uses the same number of queries to the initial state, which underperforms alternative methods [60] in this regard.

In the algorithm of [39], the eigenvalue estimation problem is solved by generating a Chebyshev history state, which is in turn realized by solving a linear system. Specifically, we introduce

$$\mathbf{Pad}(A) = \begin{bmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{bmatrix}$$

$$= \left[\begin{array}{ccccccc:ccc} I & 0 & \cdots & \cdots & \cdots & 0 & 0 & \cdots & \cdots \\ -\frac{2A}{\alpha_A} & I & \ddots & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ I & -\frac{2A}{\alpha_A} & I & \ddots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & I & \frac{-2A}{\alpha_A} & I & \ddots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & I & -\frac{2A}{\alpha_A} & I & 0 & \cdots & \cdots \\ \hdashline 0 & \cdots & \cdots & 0 & 0 & -I & I & 0 & \cdots \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & -I & I & \ddots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \ddots & \ddots \end{array}\right], \tag{200}$$

where A_{11} is n-by-n and A_{22} is ηn-by-ηn for some nonnegative integers η and n. The inverse of

this matrix is

$$\mathbf{Pad}(A)^{-1} = \left[\begin{array}{cccc:ccc} \mathbf{U}_0\left(\frac{A}{\alpha_A}\right) & 0 & \cdots & 0 & 0 & \cdots & \cdots \\ \mathbf{U}_1\left(\frac{A}{\alpha_A}\right) & \mathbf{U}_0\left(\frac{A}{\alpha_A}\right) & \ddots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & \vdots & \vdots & \vdots \\ \mathbf{U}_{n-1}\left(\frac{A}{\alpha_A}\right) & \cdots & \mathbf{U}_1\left(\frac{A}{\alpha_A}\right) & \mathbf{U}_0\left(\frac{A}{\alpha_A}\right) & 0 & \cdots & \cdots \\ \hdashline \mathbf{U}_{n-1}\left(\frac{A}{\alpha_A}\right) & \cdots & \mathbf{U}_1\left(\frac{A}{\alpha_A}\right) & \mathbf{U}_0\left(\frac{A}{\alpha_A}\right) & I & 0 & \cdots \\ \mathbf{U}_{n-1}\left(\frac{A}{\alpha_A}\right) & \cdots & \mathbf{U}_1\left(\frac{A}{\alpha_A}\right) & \mathbf{U}_0\left(\frac{A}{\alpha_A}\right) & I & I & \ddots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{array}\right], \tag{201}$$

with $\mathbf{U}_j$ Chebyshev polynomials of the second kind. Here, the coefficent matrix and its inverse have norm bounds

$$\|\mathbf{Pad}(A)\| \leq 4, \qquad \left\|\mathbf{Pad}^{-1}(A)\right\| = \mathbf{O}(n\kappa_S). \tag{202}$$

For the eigenvalue estimation problem, we choose the initial state $\frac{|0\rangle-|2\rangle}{\sqrt{2}}|\psi\rangle$ and let $\eta = 0$. Then the solution state becomes

$$\frac{\mathbf{Pad}(A)^{-1}\frac{|0\rangle-|2\rangle}{\sqrt{2}}|\psi\rangle}{\left\|\mathbf{Pad}(A)^{-1}\frac{|0\rangle-|2\rangle}{\sqrt{2}}|\psi\rangle\right\|} = \frac{\sum_{l=0}^{n-1}|l\rangle\widetilde{\mathbf{T}}_l\left(\frac{A}{\alpha_A}\right)|\psi\rangle}{\left\|\sum_{l=0}^{n-1}|l\rangle\widetilde{\mathbf{T}}_l\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|}, \tag{203}$$

with $\widetilde{\mathbf{T}}_l$ rescaled Chebyshev polynomials of the second kind. As long as this state is produced with constant accuracy and $n = \mathbf{\Theta}\left(\frac{\alpha_A}{\epsilon}\right)$, we can estimate the eigenvalue to accuracy ϵ. This gives the query complexity claimed above.

We can improve the cost of initial state preparation using block preconditioning. Specifically, we choose $\Pi_{\text{cond}} = \frac{|0\rangle-|2\rangle}{\sqrt{2}}\frac{\langle 0|-\langle 2|}{\sqrt{2}} \otimes I$, $s = \frac{1}{\sqrt{n}}$, and define the scaling operator

$$\begin{aligned} S &= \frac{1}{\sqrt{n}}\frac{|0\rangle-|2\rangle}{\sqrt{2}}\frac{\langle 0|-\langle 2|}{\sqrt{2}} \otimes I + \left(I - \frac{|0\rangle-|2\rangle}{\sqrt{2}}\frac{\langle 0|-\langle 2|}{\sqrt{2}}\right) \otimes I, \\ S^{-1} &= \sqrt{n}\frac{|0\rangle-|2\rangle}{\sqrt{2}}\frac{\langle 0|-\langle 2|}{\sqrt{2}} \otimes I + \left(I - \frac{|0\rangle-|2\rangle}{\sqrt{2}}\frac{\langle 0|-\langle 2|}{\sqrt{2}}\right) \otimes I. \end{aligned} \tag{204}$$

Note again that our preconditioner depends only on the ancilla state, and can be implemented without consuming queries to O_ψ. After block preconditioning, the solution norm is increased to [39, Lemma 17]

$$\begin{aligned} \left\|(S\mathbf{Pad}(A))^{-1}\frac{|0\rangle-|2\rangle}{\sqrt{2}}|\psi\rangle\right\| &= \sqrt{n}\left\|\mathbf{Pad}^{-1}(A)\frac{|0\rangle-|2\rangle}{\sqrt{2}}|\psi\rangle\right\| = \mathbf{\Theta}\left(\sqrt{n}\left\|\sum_{l=0}^{n-1}|l\rangle\widetilde{\mathbf{T}}_l\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|\right) \\ &= \mathbf{\Theta}\left(\sqrt{n}\sqrt{\sum_{l=0}^{n-1}\widetilde{\mathbf{T}}_l^2\left(\frac{\lambda_j}{\alpha_A}\right)}\right) = \mathbf{\Theta}(n), \end{aligned} \tag{205}$$

while the norm upper bound on the inverse matrix remains asymptotically unchanged:

$$\begin{aligned}\left\|(S\mathbf{Pad}(A))^{-1}\right\| &\leq \sqrt{\frac{\left\|\mathbf{Pad}^{-1}(A)\frac{|0\rangle-|2\rangle}{\sqrt{2}}\otimes I\right\|^2}{s^2}+\left\|\mathbf{Pad}^{-1}(A)\right\|^2} \\ &= \mathbf{O}\left(\sqrt{n}\left\|\sum_{l=0}^{n-1}|l\rangle\widetilde{\mathbf{T}}_l\left(\frac{A}{\alpha_A}\right)\right\|+n\kappa_S\right)=\mathbf{O}(n\kappa_S).\end{aligned} \tag{206}$$

Invoking Theorem 2, we have:

Theorem 6 (Quantum eigenvalue estimator with improved initial state preparation)**.** *Let $A = S\Lambda S^{-1}$ be a diagonalizable matrix with real spectra and upper bound $\kappa_S \geq \|S\|\left\|S^{-1}\right\|$ on the condition number, such that A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Suppose that oracle $O_\psi|0\rangle = |\psi\rangle$ prepares an initial state within distance $\||\psi\rangle - |\psi_j\rangle\| = \mathbf{O}\left(\frac{1}{\kappa_S}\right)$ to an eigenstate such that $A|\psi_j\rangle = \lambda_j|\psi_j\rangle$. Then λ_j can be estimate to accuracy ϵ and success probability $> \frac{1}{2}$ with query complexity*

$$\mathbf{O}\left(\kappa_S\mathbf{Cost}\left(O_\psi\right)+\frac{\alpha_A\kappa_S}{\epsilon}\log\left(\kappa_S\right)\log\log\left(\kappa_S\right)\mathbf{Cost}\left(O_A\right)\right). \tag{207}$$

Remark. The query complexity quoted above follows from Theorem 2 assuming a constant multiplicative approximation of the solution norm is available. Without this prior knowledge, one can use Theorem 1 to get such an estimate with the same cost of initial state preparation, while the cost of block encoding remains the Heisenberg-limited scaling [23, 61].

When A is Hermitian, $\kappa_S = 1$. In this case, we only need a constant number of queries to the initial state oracle and a constant overlap with the target eigenstate. Our result then recovers the performance of standard quantum phase estimation. For estimating real eigenvalues, the above result outperforms the one reported in [60]. Note that the approach of [60] works by implementing projections onto $\mathbf{Ker}(A-\mu I)$ with different shifting values μ, assuming oracular queries to initial states that are μ-dependent. This input model appears to be quite different from ours.

6.4 Quantum eigenvalue transformer and ground state preparator

We obtain an analogous improvement for the quantum eigenvalue transformation of nonnormal matrices. For a diagonalizable input matrix block enocded as A/α_A with normalization factor $\alpha_A \geq \|A\|$, this means performing a polynomial transformation on the eigenvalues

$$\frac{A}{\alpha_A} = S\frac{\Lambda}{\alpha_A}S^{-1} \mapsto p\left(\frac{A}{\alpha_A}\right) = Sp\left(\frac{\Lambda}{\alpha_A}\right)S^{-1}, \tag{208}$$

applied to an initial state $|\psi\rangle$. This covers the special case where A is Hermitian, which is relevant for applications including Hamiltonian simulation [38], ground state [34] and thermal state preparation [22]. More generally, efficient algorithms for the eigenvalue transformation of nonnormal matrices can be applied to solve differential equations and prepare ground states of non-Hermitian matrices with real spectra.

A linear-system-based quantum algorithm was recently developed to transform eigenvalues of nonnormal matrices [39]. For a diagonalizable input matrix $A = S\Lambda S^{-1}$ with only real eigenvalues, the previous method has query complexity

$$\mathbf{O}\left(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S^2 n}{\left\|p\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|}\log\left(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S\log(n)}{\left\|p\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|\epsilon}\right)\log\left(n\right)\left(\mathbf{Cost}\left(O_A\right)+\mathbf{Cost}\left(O_\psi\right)\right)\right), \tag{209}$$

where ϵ is the accuracy of the output state, κ_S is the condition number of the basis transformation, p is the target polynomial with norm $\|p\|_{\max,[-\frac{1}{2},\frac{1}{2}]} = \max_{x\in[-\frac{1}{2},\frac{1}{2}]} |p(x)|$ and $n-1$ is its degree. This is achieved by inverting the coefficient matrix $\mathbf{Pad}^{-1}(A)$ introduced in the previous subsection with $\eta = 1$, which has the norm bounds $\|\mathbf{Pad}(A)\| \leq 4$ and $\left\|\mathbf{Pad}^{-1}(A)\right\| = \mathbf{O}(n\kappa_S)$ same as before. For the eigenvalue transformation problem, supposing that the target polynomial is represented under the (rescaled) Chebyshev basis as $p(x) = \sum_{j=0}^{n-1} \widetilde{\beta}_j \widetilde{\mathbf{T}}_j(x)$, we choose the initial state $|0\rangle|\beta\rangle|\psi\rangle$ where

$$|\beta\rangle = \frac{1}{\alpha_{\widetilde{\beta}}} \sum_{k=0}^{n-1} (\widetilde{\beta}_k - \widetilde{\beta}_{k+2}) |n-1-k\rangle, \qquad \alpha_{\widetilde{\beta}} = \sqrt{\sum_{k=0}^{n-1} |\widetilde{\beta}_k - \widetilde{\beta}_{k+2}|^2} = \mathbf{\Theta}\left(\|p(\cos)\sin\|_{2,[-\pi,\pi]}\right), \quad (210)$$

where $\|p(\cos)\sin\|_{2,[-\pi,\pi]} = \sqrt{\int_{-\pi}^{\pi} \mathrm{d}\theta \;\; |p(\cos\theta)\sin\theta|^2}$. Applying the quantum linear system solver then produces the Chebyshev history state

$$\frac{|0\rangle \sum_{l=0}^{n-1} |l\rangle \sum_{k=n-1-l}^{n-1} \widetilde{\beta}_k \widetilde{\mathbf{T}}_{k+l-n+1}\left(\frac{A}{\alpha_A}\right)|\psi\rangle + |1\rangle \sum_{l=0}^{n-1} |l\rangle \sum_{k=0}^{n-1} \widetilde{\beta}_k \widetilde{\mathbf{T}}_k \left(\frac{A}{\alpha_A}\right)|\psi\rangle}{\sqrt{\sum_{l=0}^{n-1} \left\|\sum_{k=n-1-l}^{n-1} \widetilde{\beta}_k \widetilde{\mathbf{T}}_{k+l-n+1}\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|^2 + n\left\|\sum_{k=0}^{n-1} \widetilde{\beta}_k \widetilde{\mathbf{T}}_k\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|^2}}. \quad (211)$$

Here, the component flagged by the ancilla state $|1\rangle$ is desired, and the success amplitude is at least

$$\mathbf{\Omega}\left(\frac{\left\|p\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|}{\|p\|_{\max,[-\frac{1}{2},\frac{1}{2}]}\,\kappa_S \log(n)}\right). \quad (212)$$

This can be boosted close to 1 by amplitude amplification, leading to the query complexity cited above. Note that compared to [39], we have changed the domain from $[-1,1]$ to $\left[-\frac{1}{2},\frac{1}{2}\right]$ which is without loss of generality by rescaling the input block encoding.

Once again, we can improve the query complexity of the initial state preparation using block preconditioning. To this end, we choose $\Pi_{\text{cond}} = |\beta\rangle\langle\beta| \otimes I$, $s = \frac{\|p\|_{\max,[-\frac{1}{2},\frac{1}{2}]}}{\sqrt{n}\alpha_{\widetilde{\beta}}}$, and define the scaling operator

$$\begin{aligned} S &= \frac{\|p\|_{\max,[-\frac{1}{2},\frac{1}{2}]}}{\sqrt{n}\alpha_{\widetilde{\beta}}} |0,\beta\rangle\langle 0,\beta| \otimes I + (I - |0,\beta\rangle\langle 0,\beta|) \otimes I, \\ S^{-1} &= \frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,[-\frac{1}{2},\frac{1}{2}]}} |0,\beta\rangle\langle 0,\beta| \otimes I + (I - |0,\beta\rangle\langle 0,\beta|) \otimes I. \end{aligned} \quad (213)$$

Recall that in order for this to be a valid block preconditioning, S must be invertible which further requires $0 < s < 1$. This is confirmed by the following lemma.

Lemma 9. *Let $p(x)$ be a polynomial with the expansion $p(x) = \sum_{j=0}^{n-1} \widetilde{\beta}_j \widetilde{\mathbf{T}}_j(x)$ into rescaled Chebyshev polynomials of the first kind. It holds*

$$\|p\|_{\max,[-\frac{1}{2},\frac{1}{2}]} < \sqrt{n}\sqrt{\sum_{k=0}^{n-1} |\widetilde{\beta}_k - \widetilde{\beta}_{k+2}|^2}. \quad (214)$$

Proof. Extending the definition of coefficients by setting $\widetilde{\beta}_n = \widetilde{\beta}_{n+1} = \cdots = 0$, we have

$$\begin{aligned} p(x) &= \sum_{j=0}^{\infty} \widetilde{\beta}_j \widetilde{\mathbf{T}}_j(x) = \sum_{j=2}^{\infty} \widetilde{\beta}_j \frac{\mathbf{U}_j(x) - \mathbf{U}_{j-2}(x)}{2} + \widetilde{\beta}_1 \widetilde{\mathbf{T}}_1(x) + \widetilde{\beta}_0 \widetilde{\mathbf{T}}_0(x) \\ &= \frac{1}{2} \sum_{j=2}^{\infty} \widetilde{\beta}_j \mathbf{U}_j(x) - \frac{1}{2} \sum_{j=0}^{\infty} \widetilde{\beta}_{j+2} \mathbf{U}_j(x) + \widetilde{\beta}_1 \widetilde{\mathbf{T}}_1(x) + \widetilde{\beta}_0 \widetilde{\mathbf{T}}_0(x) \\ &= \frac{1}{2} \sum_{j=0}^{\infty} \left(\widetilde{\beta}_j - \widetilde{\beta}_{j+2} \right) \mathbf{U}_j(x) - \frac{1}{2} \widetilde{\beta}_1 \mathbf{U}_1(x) - \frac{1}{2} \widetilde{\beta}_0 \mathbf{U}_0(x) + \widetilde{\beta}_1 \widetilde{\mathbf{T}}_1(x) + \widetilde{\beta}_0 \widetilde{\mathbf{T}}_0(x) \\ &= \frac{1}{2} \sum_{j=0}^{\infty} \left(\widetilde{\beta}_j - \widetilde{\beta}_{j+2} \right) \mathbf{U}_j(x) = \frac{1}{2} \sum_{j=0}^{n-1} \left(\widetilde{\beta}_j - \widetilde{\beta}_{j+2} \right) \mathbf{U}_j(x). \end{aligned} \tag{215}$$

Now by the Cauchy-Schwarz inequality,

$$|p(x)| \leq \frac{1}{2} \sqrt{\sum_{j=0}^{n-1} \mathbf{U}_j^2(x)} \sqrt{\sum_{k=0}^{n-1} |\widetilde{\beta}_k - \widetilde{\beta}_{k+2}|^2}. \tag{216}$$

We now claim that $\sum_{j=0}^{n-1} \mathbf{U}_j^2(x) \leq \frac{2}{3}n + 1$ for $-\frac{1}{2} \leq x \leq \frac{1}{2}$. To this end, let us introduce the angular variable $\phi = \frac{1}{2\pi} \arccos(x)$ and consider

$$\begin{aligned} \mathbf{U}_{n-1}^2(x) + \cdots + \mathbf{U}_0^2(x) &= \sum_{j=0}^{n-1} \frac{\sin^2(2\pi(j+1)\phi)}{\sin^2(2\pi\phi)} = \frac{1}{\sin^2(2\pi\phi)} \sum_{j=1}^{n} \frac{1 - \cos(4\pi j\phi)}{2} \\ &= \frac{1}{\sin^2(2\pi\phi)} \left(\frac{n}{2} - \frac{1}{4} \sum_{j=0}^{n-1} \left(e^{i4\pi j\phi} + e^{-i4\pi j\phi} \right) \right) \\ &= \frac{1}{\sin^2(2\pi\phi)} \left(\frac{n}{2} - \frac{1}{2} \frac{\sin(2\pi n\phi)}{\sin(2\pi\phi)} \cos((n-1)2\pi\phi) \right). \end{aligned} \tag{217}$$

By our assumption, $\frac{\sqrt{3}}{2} \leq \sin(2\pi\phi) \leq 1$ and hence

$$\frac{n}{2} - \frac{\sqrt{3}}{3} \leq \mathbf{U}_{n-1}^2(x) + \cdots + \mathbf{U}_0^2(x) \leq \frac{4}{3} \left(\frac{n}{2} + \frac{\sqrt{3}}{3} \right). \tag{218}$$

□

Similar as before, our preconditioner is defined by the initial ancilla state, and its implementation uses no query to O_ψ. After block preconditioning, the solution norm becomes

$$\left\| (S\mathbf{Pad}(A))^{-1} |0\rangle|\beta\rangle|\psi\rangle \right\| = \frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}} \left\| \mathbf{Pad}^{-1}(A)|0\rangle|\beta\rangle|\psi\rangle \right\|, \tag{219}$$

whereas the upper bound on the norm of inverse matrix is asymptotically:

$$\begin{aligned} \left\| (S\mathbf{Pad}(A))^{-1} \right\| &\leq \sqrt{\frac{\left\| \mathbf{Pad}^{-1}(A)|0,\beta\rangle \otimes I \right\|^2}{s^2} + \left\| \mathbf{Pad}^{-1}(A) \right\|^2} \\ &= \mathbf{O}\left(\frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}} \frac{\sqrt{n} \max_l \left\| \sum_{k=l}^{n-1} \widetilde{\beta}_k \widetilde{\mathbf{T}}_{k-l}\left(\frac{A}{\alpha_A}\right) \right\|}{\alpha_{\widetilde{\beta}}} + n\kappa_S \right) \\ &= \mathbf{O}\left(n\kappa_S \log(n) \right). \end{aligned} \tag{220}$$

This is larger by a factor of $\log(n)$ in the worst case, but one can potentially remove this logarithmic factor when running the algorithm over an average input.

Suppose that the norm of solution state from the quantum linear system solver (Theorem 2) can be estimated to a constant multiplicative accuracy. Then the query complexity of the initial state oracle is bounded by

$$\begin{aligned}
&\mathbf{O}\left(\frac{n\kappa_S \log(n)}{\frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}}\left\|\mathbf{Pad}^{-1}(A)|0\rangle|\beta\rangle|\psi\rangle\right\|}\frac{\frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}}\left\|\mathbf{Pad}^{-1}(A)|0\rangle|\beta\rangle|\psi\rangle\right\|}{\frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}}\frac{\sqrt{n}}{\alpha_{\widetilde{\beta}}}\left\|p\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|}\right) \\
&= \mathbf{O}\left(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S \log(n)}{\left\|p\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|}\right).
\end{aligned} \tag{221}$$

We thus obtain:

Theorem 7 (Quantum eigenvalue transformer with improved initial state preparation)**.** *Let $A = S\Lambda S^{-1}$ be a diagonalizable matrix with real spectra and upper bound $\kappa_S \geq \|S\|\left\|S^{-1}\right\|$ on the condition number, such that A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Let $O_\psi|0\rangle = |\psi\rangle$ be the oracle preparing the initial state, and $p(x) = \sum_{k=0}^{n-1}\widetilde{\beta}_k\widetilde{\mathbf{T}}_k(x) = \sum_{k=0}^{n-1}\beta_k\mathbf{T}_k(x)$ be the Chebyshev expansion of a degree-$(n-1)$ polynomial p. Then, the quantum state*

$$\frac{p\left(\frac{A}{\alpha_A}\right)|\psi\rangle}{\left\|p\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|} \tag{222}$$

can be prepared with accuracy ϵ and success probability $> \frac{1}{2}$ with query complexity

$$\begin{aligned}
&\mathbf{O}\Bigg(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S \log(n)}{\alpha_{p,\psi}}\mathbf{Cost}\left(O_\psi\right) \\
&\quad + \frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S^2 n\log(n)}{\alpha_{p,\psi}}\log\left(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S\log(n)}{\alpha_{p,\psi}}\right)\log\left(\frac{\log\left(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S\log(n)}{\alpha_{p,\psi}}\right)}{\epsilon}\right)\mathbf{Cost}\left(O_A\right)\Bigg),
\end{aligned} \tag{223}$$

where $\alpha_{p,\psi} \leq \left\|p\left(\frac{A}{\alpha_A}\right)|\psi\rangle\right\|$ is a norm lower bound on the transformed state.

As an immediate application, we also obtain an improved quantum algorithm for ground state preparation. In the case where the input operator is a Hermitian Hamiltonian, this problem has been extensively studied by previous work [21, 43], and can be solved near optimally on a quantum computer [34]. Here, we consider the general case where inputs are non-normal matrices with real eigenvalues whose ground states are still well defined, which are relevant to applications in non-Hermitian physics and transcorrelated quantum chemistry. Specifically, let $A = S\Lambda S^{-1}$ be a diagonalizable matrix with only real eigenvalues and an upper bound κ_S on the condition number of its basis transformation, such that A/α_A is block encoded by oracle O_A with normalization factor α_A. Suppose that λ_0 is the smallest eigenvalue of A with the corresponding eigenstate $|\psi_0\rangle$, which is separated from the next eigenvalue λ_1:

$$\lambda_0 \leq -\frac{\delta_A}{2} < 0 < \frac{\delta_A}{2} \leq \lambda_1 \tag{224}$$

for some spectral gap $\delta_A > 0$. Then our goal is to prepare a quantum state that ϵ-approximates the ground state $|\psi_0\rangle$ up to a global phase, given an initial state $|\psi\rangle = \gamma_0|\psi_0\rangle + \sum_{l=1}^{d-1} \gamma_l|\psi_l\rangle$ prepared by oracle O_ψ.

The best previous quantum ground state preparation algorithm proceeds by implementing a degree

$$n = \mathbf{O}\left(\frac{\alpha_A}{\delta_A} \log\left(\frac{\kappa_S}{|\gamma_0|\epsilon}\right)\right). \tag{225}$$

polynomial using quantum eigenvalue transformer [39, Theorem 8], which leads to the query complexity

$$\mathbf{O}\left(\frac{\kappa_S^2}{|\gamma_0|}\frac{\alpha_A}{\delta_A} \log^2\left(\frac{\kappa_S}{|\gamma_0|\epsilon}\right)\left(\mathbf{Cost}\left(O_A\right) + \mathbf{Cost}\left(O_\psi\right)\right)\right). \tag{226}$$

Using block preconditioning and our quantum linear system solver with optimal initial state preparation, we improve this to:

Theorem 8 (Quantum ground state preparator with improved initial state preparation)**.** *Let $A = S\Lambda S^{-1}$ be a diagonalizable matrix with real spectra and upper bound $\kappa_S \geq \|S\| \left\|S^{-1}\right\|$ on the condition number, such that A/α_A is block encoded by O_A with normalization factor $\alpha_A \geq \|A\|$. Let eigenvalues of A be ordered nondecreasingly, with λ_0 the smallest one corresponding to eigenstate $|\psi_0\rangle$, satisfying the condition*

$$\lambda_0 \leq -\frac{\delta_A}{2} < 0 < \frac{\delta_A}{2} \leq \lambda_1 \tag{227}$$

for some spectral gap $\delta_A > 0$. Let $O_\psi|0\rangle = |\psi\rangle$ be the oracle preparing the initial state with the eigenbasis expansion

$$|\psi\rangle = \gamma_0|\psi_0\rangle + \sum_{l=1}^{d-1} \gamma_l|\psi_l\rangle. \tag{228}$$

Then, the ground state $|\psi_0\rangle$ can be produced with accuracy ϵ, success probability $> \frac{1}{2}$ and global phase factor $\gamma_0/|\gamma_0|$, with query complexity

$$\mathbf{O}\left(\frac{\kappa_S}{|\gamma_0|} \log\left(\frac{\alpha_A}{\delta_A} \log\left(\frac{\kappa_S}{|\gamma_0|\epsilon}\right)\right) \mathbf{Cost}\left(O_\psi\right) + \frac{\kappa_S^2}{|\gamma_0|}\frac{\alpha_A}{\delta_A} \operatorname{polylog}\left(\frac{\alpha_A}{\delta_A} \log\left(\frac{\kappa_S}{|\gamma_0|\epsilon}\right)\right) \mathbf{Cost}\left(O_A\right)\right). \tag{229}$$

6.5 Block encoded quantum eigenvalue transformer

We report another improvement to the block-encoding version of the quantum eigenvalue transformation algorithm. This is similar to the application from the previous subsection, except we construct the block encoding of the transformed matrix, rather than applying it to an initial state, making it more versatile when quantum eigenvalue transformation is used as a subroutine in constructing other quantum algorithms.

Specifically, we set $\eta = 1$ and use Lemma 8 to construct a block encoding of

$$\frac{\mathbf{Pad}^{-1}(A)}{2\alpha_{\mathbf{Pad}^{-1}(\mathbf{A})}} \tag{230}$$

with a normalization factor $\alpha_{\mathbf{Pad}^{-1}(\mathbf{A})} = \mathbf{O}\left(n\kappa_S\right)$. Together with the preparation of initial state $|0\rangle|\beta\rangle$ and unpreparation of $|1\rangle\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}|k\rangle$ where $|\beta\rangle$ is given by Eq. (210), we obtain the block

encoding

$$\left(\langle 1|\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\langle k|\otimes I\right)\frac{\mathbf{Pad}^{-1}(A)}{2\alpha_{\mathbf{Pad}^{-1}(\mathbf{A})}}\left(|0\rangle|\beta\rangle\otimes I\right)=\frac{p\left(\frac{A}{\alpha_A}\right)}{\alpha_{p,\mathrm{pre}}} \tag{231}$$

with

$$\alpha_{p,\mathrm{pre}}=\frac{\alpha_{\mathbf{Pad}(\mathbf{A})^{-1}}\alpha_{\widetilde{\beta}}}{\sqrt{n}}=\mathbf{O}\left(\sqrt{n}\kappa_S\alpha_{\widetilde{\beta}}\right), \tag{232}$$

Therefore, we have block encoded the target polynomial but with a larger normalization factor. In prior art [39], further amplification is performed to reduce the normalization factor, giving

$$\frac{p\left(\frac{A}{\alpha_A}\right)}{2\left\|p\left(\frac{A}{\alpha_A}\right)\right\|}. \tag{233}$$

The overall query complexity is then

$$\mathbf{O}\left(\frac{\|p(\cos)\sin\|_{2,[-\pi,\pi]}\,n^{\frac{3}{2}}\kappa_S^2}{\left\|p\left(\frac{A}{\alpha_A}\right)\right\|}\log\left(\frac{\|p(\cos)\sin\|_{2,[-\pi,\pi]}\,\sqrt{n}\kappa_S}{\left\|p\left(\frac{A}{\alpha_A}\right)\right\|\epsilon}\right)\log\left(\frac{1}{\epsilon}\right)\right). \tag{234}$$

We improve this block encoding cost using the block preconditioning technique. Specifically, we define the scaling operator

$$\begin{aligned}S&=\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}}{\sqrt{n}\alpha_{\widetilde{\beta}}}|0,\beta\rangle\langle 0,\beta|\otimes I+(I-|0,\beta\rangle\langle 0,\beta|)\otimes I,\\ S^{-1}&=\frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}}|0,\beta\rangle\langle 0,\beta|\otimes I+(I-|0,\beta\rangle\langle 0,\beta|)\otimes I\end{aligned} \tag{235}$$

same as in the previous subsection. Recall that this does not increase the asymptotic scaling of the condition number. However, the block encoding now becomes

$$\left(\langle 1|\frac{1}{\sqrt{n}}\sum_{k=0}^{n-1}\langle k|\otimes I\right)\frac{\mathbf{Pad}^{-1}(A)S^{-1}}{2\alpha_{\mathbf{Pad}^{-1}(\mathbf{A})}}\left(|0\rangle|\beta\rangle\otimes I\right)=\frac{\sqrt{n}\alpha_{\widetilde{\beta}}}{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}}\frac{p\left(\frac{A}{\alpha_A}\right)}{\alpha_{p,\mathrm{pre}}}=\frac{p\left(\frac{A}{\alpha_A}\right)}{\alpha_{p,\mathrm{cond}}} \tag{236}$$

for

$$\alpha_{p,\mathrm{cond}}=\mathbf{O}\left(\sqrt{n}\kappa_S\alpha_{\widetilde{\beta}}\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}}{\sqrt{n}\alpha_{\widetilde{\beta}}}\right)=\mathbf{O}\left(\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S\right). \tag{237}$$

So the query complexity to the block encoding is improved from $\mathbf{O}(n^{1.5})$ to $\mathbf{O}(n)$.

Theorem 9 (Block encoded quantum eigenvalue transformer with linear degree cost)**.** *Let $A=S\Lambda S^{-1}$ be a diagonalizable matrix with real spectra and upper bound $\kappa_S\geq\|S\|\left\|S^{-1}\right\|$ on the condition number, such that A/α_A is block encoded by O_A with normalization factor $\alpha_A\geq\|A\|$. Let $p(x)=\sum_{k=0}^{n-1}\widetilde{\beta}_k\widetilde{\mathbf{T}}_k(x)=\sum_{k=0}^{n-1}\beta_k\mathbf{T}_k(x)$ be the Chebyshev expansion of a degree-$(n-1)$ polynomial p. Then for any $\alpha_p\geq\left\|p\left(\frac{A}{\alpha_A}\right)\right\|$, the operator*

$$\frac{p\left(\frac{A}{\alpha_A}\right)}{2\alpha_p} \tag{238}$$

can be block encoded with accuracy ϵ using

$$\mathbf{O}\left(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]} n\kappa_S^2}{\alpha_p}\log\left(\frac{\|p\|_{\max,\left[-\frac{1}{2},\frac{1}{2}\right]}\kappa_S}{\alpha_p\epsilon}\right)\log\left(\frac{1}{\epsilon}\right)\right) \tag{239}$$

queries to O_A.

7 Discussion

In this work, we have developed a quantum linear system algorithm that achieves an optimal query complexity of the initial state preparation, while maintaining a nearly optimal cost scaling of the coefficient block encoding. This outperforms recent linear system solvers that make optimal uses of one resource, but far too many queries to the other. Our algorithm employs a nested amplitude amplification with a deterministic amplification schedule, which substantially simplifies prior approaches based on VTAA while improving their asymptotic query complexity. We also present a quantum algorithm that produces a constant multiplicative approximation of norm of the solution state, again using optimal number of queries to the initial state oracle, improving over all previous approaches where finding such an estimate incurs polylogarithmic overhead.

Our state preparation cost scales strictly linear in the inverse success amplitude $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$. We further develop a block preconditioning technique that boosts p_{succ} for practical applications of quantum linear system solvers. As a result, we obtain quantum algorithms for solving differential equations, for preparing ground states of non-Hermitian operators, and for processing eigenvalues of non-normal matrices, all with reduced query complexity of the initial state preparation, nearly matching or outperforming alternative methods for the same tasks. Of independent interest, our block preconditioning technique also allows for the improvement of other scaling parameters. We present an extremely simple quantum linear system solver with an optimal block encoding cost by choosing the initial state as the preconditioner. We also give a preconditioned quantum eigenvalue transformer that implements a degree-n polynomial using $\mathbf{O}(n)$ queries to the block encoding oracle, whereas the best prior scaling was $\mathbf{O}(n^{1.5})$.

We have examined variable time amplitude amplifications with tunable threshold values that capture the power of generic nested amplitude amplifications. We have shown that Tunable VTAA makes $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)$ queries to the initial state matching the performance of standard amplitude amplification, while only contains at most $\mathbf{O}\left(\log\left(\frac{1}{\sqrt{p_{\text{succ}}}}\right)\right)$ nontrivial amplification stages allowing majority of the algorithms to be pre-merged. Specialized to the quantum linear system problem, we have designed a discretized inverse state which can be prepared by Tunable VTAA with a complexity scaling with ℓ_2-norm of the input costs. We show that this scheme translates naturally to a deterministic amplification schedule, avoiding the significant setup overhead of prior VTAA approaches. However, we prove that one can in principle optimize the amplification thresholds so that the cost of Tunable VTAA attains $\ell_{\frac{2}{3}}$-quasinorm of the input costs. This improves over the ℓ_1-norm result of Ambainis and the more commonly used ℓ_2-norm result. With more prior knowledge about the target linear system, this result can be utilized to further improve the query complexity of quantum solvers. For instance, consider the coefficient matrix

$$A = \begin{bmatrix} \frac{1}{3} & & & \\ & \frac{1}{9} & & \\ & & \ddots & \\ & & & \frac{1}{3^m} \end{bmatrix} \tag{240}$$

and initial state $|b\rangle = |m-l-1\rangle$, such that $A|b\rangle = \frac{1}{3^{m-l}}|b\rangle$. Here A can be block encoded with normalization factor $\alpha_A = 1$, whereas $\left\|A^{-1}\right\| \leq 3^m = \alpha_{A^{-1}} = \kappa$. Meanwhile, we have $\left\|A^{-1}|b\rangle\right\| = 3^{m-l}$, so $\sqrt{p_{\text{succ}}} = \frac{\left\|A^{-1}|b\rangle\right\|}{\alpha_{A^{-1}}} = \frac{1}{3^l}$. Choosing a non-uniform schedule of GPE accuracies $\epsilon_{\text{gpe},j} = \frac{\epsilon}{2^{m-j+2}}$ for $j = m-l+1, \ldots, m$ and $\epsilon_{\text{gpe},j} = \frac{\epsilon}{2^{m-l-j+2}}$ for $j = 1, \ldots, m-l$, one can attain the improved query complexity $\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\epsilon}\right)\mathbf{Cost}\left(O_A\right)\right)$. It would be interesting to identify more algorithmic applications where the $\ell_{\frac{2}{3}}$-quasinorm can be tightly bounded without introducing additional scaling factors.

Related to the above discussion, an obvious open question is whether it is possible for a quantum linear system algorithm to achieve the query complexity

$$\mathbf{O}\left(\frac{1}{\sqrt{p_{\text{succ}}}}\mathbf{Cost}(O_b) + \kappa \log\left(\frac{1}{\epsilon}\right)\mathbf{Cost}\left(O_A\right)\right) \tag{241}$$

for a generic linear system problem. From an algorithmic perspective, our Tunable VTAA matches the above query scaling of O_b and, theoretically, the cost of O_A scales strictly linear with the $\ell_{\frac{2}{3}}$-quasinorm of input costs, although upper bounding it in terms of κ without logarithmic overhead can be difficult in general. On the other hand, there may also be room to further improve the lower bound when one considers the oracles O_b and O_A simultaneously.

We have mainly focused on applications of quantum linear system solvers where the target outputs are quantum states or block encoded operators. In this case, we have introduced a scaling matrix such that, when it is inverted from the right side of coefficient matrix, the success amplitude can be boosted up and complexity of the algorithm can be reduced. However, there also exist other linear-system related problems such as computing Green's functions, where the goal is to estimate the expectation value of an observable. We leave it as a subject for future work to explore whether algorithms for such problems can be improved by implementing generalized versions of block preconditioning on both sides of the coefficient matrix.

A number of other questions call for more investigations. Our preconditioned quantum linear system algorithm has an optimal query complexity of block encoding and appears to be conceptually simpler than recent optimal methods based on the discrete adiabatic evolution and kernel reflection. It may be fruitful to assess the resources required to implement this algorithm, for both near-term and fault-tolerant quantum devices. It is also possible to improve the constant prefactor of the complexity of VTAA: for instance, one may use the trigonometric identity $\frac{\sin(3\theta)}{3\sin(\theta)} = 1 - \frac{4}{3}\sin^2(\theta)$ as opposed to its bounds when the amplification step number is exactly 3, or one may replace the first $m-l$ stages of GPE by a single GPE covering a larger interval of eigenvalues (although this modification could significantly increase the success probability and result in over amplification). Our solution norm estimation algorithm maintains the optimal scaling of query cost of O_b, but introduces additional logarithmic factors to the complexity of O_A, which may be improved by solving an augmented linear system similar to [19]. Finally, it would be of interest to explore other algorithmic applications of quantum linear system solvers beyond those studied here.

Acknowledgements

We thank Rolando Somma for helpful comments on an earlier draft.

A Axiomatic definition of variable time amplification

In this appendix, we collect mathematical results useful for formulating the axiomatic definition of variable time amplification in Section 2.2. We assume throughout this appendix that an underlying Hilbert space is fixed, on which all operators act. We denote the complementary operator of X by $\overline{X} = I - X$.

A.1 Clock projections

Lemma 10 (Partially ordered projections)**.** *Let* $\Pi_1^2 = \Pi_1 = \Pi_1^\dagger$ *and* $\Pi_2^2 = \Pi_2 = \Pi_2^\dagger$ *be orthogonal projections. The following statements are all equivalent:*

1. $\mathbf{Im}(\Pi_1) \subseteq \mathbf{Im}(\Pi_2)$.
2. *(a)* $\Pi_2\Pi_1 = \Pi_1$.
   (b) $\Pi_1\Pi_2 = \Pi_1$.
   (c) $(\Pi_2 - \Pi_1)^2 = \Pi_2 - \Pi_1$ *is an orthogonal projection.*
   (d) $\Pi_1\Pi_2\Pi_1 = \Pi_1$.
3. *(a)* $\Pi_1 \leq \Pi_2$ *as Hermitian operators.*
   (b) $\|\Pi_1|\psi\rangle\| \leq \|\Pi_2|\psi\rangle\|$ *for all states* $|\psi\rangle$.

Proof. We begin with the equivalence of 3(a) and 3(b) which follows from a direct verification:

$$\begin{aligned} \|\Pi_1|\psi\rangle\| \leq \|\Pi_2|\psi\rangle\| \quad &\Leftrightarrow \quad \|\Pi_1|\psi\rangle\|^2 \leq \|\Pi_2|\psi\rangle\|^2 \quad &\Leftrightarrow \quad \langle\psi|\Pi_1^\dagger\Pi_1|\psi\rangle \leq \langle\psi|\Pi_2^\dagger\Pi_2|\psi\rangle \\ &\Leftrightarrow \quad \langle\psi|\Pi_1|\psi\rangle \leq \langle\psi|\Pi_2|\psi\rangle \quad &\Leftrightarrow \quad \Pi_1 \leq \Pi_2. \end{aligned} \tag{242}$$

The equivalence of 2(a) and 2(b) is also immediate:

$$\Pi_1 = \Pi_2\Pi_1 \quad \Leftrightarrow \quad \Pi_1^\dagger = (\Pi_2\Pi_1)^\dagger = \Pi_1^\dagger\Pi_2^\dagger \quad \Leftrightarrow \quad \Pi_1 = \Pi_1\Pi_2. \tag{243}$$

Assuming 2(b) (and hence 2(a)) holds, 2(c) follows from the calculation

$$(\Pi_2 - \Pi_1)^2 = \Pi_2^2 + \Pi_1^2 - \Pi_1\Pi_2 - \Pi_2\Pi_1 = \Pi_2 + \Pi_1 - \Pi_1 - \Pi_1 = \Pi_2 - \Pi_1. \tag{244}$$

Now if 2(c) is true, we have

$$\begin{aligned} &\Pi_2 + \Pi_1 - \Pi_1\Pi_2 - \Pi_2\Pi_1 = (\Pi_2 - \Pi_1)^2 = \Pi_2 - \Pi_1 \\ \Leftrightarrow \quad &2\Pi_1 = \Pi_1\Pi_2 + \Pi_2\Pi_1 \quad \Rightarrow \quad \Pi_1 = \Pi_1\Pi_2\Pi_1, \end{aligned} \tag{245}$$

where in the last step we perform Π_1 on both sides of the operators. This proves 2(d).

The implication 2(d) $\Rightarrow$ 2(a) can be established by a trace argument. Note that

$$\mathbf{Tr}\left((I - \Pi_2)\Pi_1\right) = \mathbf{Tr}\left(\Pi_1 - \Pi_2\Pi_1\right) = \mathbf{Tr}\left(\Pi_1 - \Pi_1\Pi_2\Pi_1\right) = 0, \tag{246}$$

where the second equality follows from cyclic property of the trace function. Since $I - \Pi_2$ and Π_1 are both positive semidefinite, this necessarily means that $(I - \Pi_2)\Pi_1 = 0$ which is equivalent to 2(a).

Suppose that $\mathbf{Im}(\Pi_1) \subseteq \mathbf{Im}(\Pi_2)$. For any state $|\psi\rangle$, we have $\Pi_1|\psi\rangle \in \mathbf{Im}(\Pi_1) \subseteq \mathbf{Im}(\Pi_2)$, which gives $\Pi_2\Pi_1|\psi\rangle = \Pi_1|\psi\rangle$ and thus $\Pi_1 = \Pi_2\Pi_1$. This shows 1 $\Rightarrow$ 2(a). The implication 2(c) $\Rightarrow$ 3(a) is trivial. Finally, assume that 3(b) is true. For any $|\psi\rangle \in \mathbf{Ker}(\Pi_2)$, we have $0 \leq \|\Pi_1|\psi\rangle\| \leq \|\Pi_2|\psi\rangle\| = 0$, which forces $\|\Pi_1|\psi\rangle\| = 0$ and hence $\Pi_1|\psi\rangle = 0$. This means $\mathbf{Ker}(\Pi_2) \subseteq \mathbf{Ker}(\Pi_1)$, which is equivalent to Claim 1 after taking the orthogonal complement. □

The above list can be expanded to include other equivalent characterizations, but this is not needed for our work. See [29, Section 2.5] and [31, Section 9.6] for further discussions about the partial ordering of orthogonal projections.

Proposition 11. *Let $\{\Pi_j\}_{j=0}^m$ be orthogonal projections partially ordered as $0 = \Pi_0 \leq \Pi_1 \leq \cdots \leq \Pi_m = I$. Then, we have $I = \overline{\Pi_0} \geq \overline{\Pi_1} \geq \cdots \geq \overline{\Pi_m} = 0$ and*

$$\Pi_j = \Pi_k \Pi_j = \Pi_j \Pi_k, \qquad \overline{\Pi_k} = \overline{\Pi_j} \cdot \overline{\Pi_k} = \overline{\Pi_k} \cdot \overline{\Pi_j}, \qquad 0 \leq j \leq k \leq m, \tag{247}$$

whereas $\Pi_k - \Pi_j$ are all orthogonal projections.

A.2 Flag projections

Proposition 12. *Let $\{\Pi_j\}_{j=0}^m$ be orthogonal projections partially ordered as $0 = \Pi_0 \leq \Pi_1 \leq \cdots \leq \Pi_m = I$. Let Π_b be an orthogonal projection commuting with all Π_j: $\Pi_b \Pi_j = \Pi_j \Pi_b$ for $j = 0, \ldots, m$. Then, the following two resolutions of identity hold*

$$I = \sum_{j=1}^{m} (\Pi_j - \Pi_{j-1}) = \Pi_b + \overline{\Pi_b}, \tag{248}$$

where all $\{\Pi_j - \Pi_{j-1}\}_{j=1}^m$ and $\{\Pi_b, \overline{\Pi_b}\}$ are orthogonal projections and pairwise commute. Moreover, we have

$$0 = \Pi_0 \Pi_b \leq \Pi_1 \Pi_b \leq \cdots \leq \Pi_m \Pi_b = \Pi_b, \qquad I = \overline{\Pi_0 \Pi_b} \geq \overline{\Pi_1 \Pi_b} \geq \cdots \geq \overline{\Pi_m \Pi_b} = \overline{\Pi_b}, \tag{249}$$

with

$$\Pi_j \Pi_b = \Pi_k \Pi_b \cdot \Pi_j \Pi_b = \Pi_j \Pi_b \cdot \Pi_k \Pi_b, \qquad \overline{\Pi_k \Pi_b} = \overline{\Pi_j \Pi_b} \cdot \overline{\Pi_k \Pi_b} = \overline{\Pi_k \Pi_b} \cdot \overline{\Pi_j \Pi_b}, \qquad 0 \leq j \leq k \leq m. \tag{250}$$

The above proposition shows that the two sets of orthogonal projections $\{\Pi_j - \Pi_{j-1}\}_{j=1}^m$ and $\{\Pi_b, I - \Pi_b\}$ are complete and pairwise commute, so they can be simultaneously measured in a quantum computation. We tabulate the meaning of different outcomes from such a simultaneous measurement in Table 3. Note that logical operations correspond directly to arithmetics of the orthogonal projections. For instance, the statement "the algorithm does not fail at stage j" is logically equivalent to "the algorithm succeeds before or at stage j, or it is still running", which can be represented in terms of operators as

$$\overline{\Pi_j \overline{\Pi_b}} = I - \Pi_j \Pi_b = \Pi_j (I - \Pi_b) + (I - \Pi_j) = \Pi_j \overline{\Pi_b} + \overline{\Pi_j}. \tag{251}$$

A.3 Input algorithms

Lemma 13 (Controlled unitaries)**.** *Let $U^\dagger U = U U^\dagger = I$ be a unitary operator and $\Pi^2 = \Pi = \Pi^\dagger$ be an orthogonal projection. The following statements are all equivalent:*

1. *(a)* $U\Pi = \Pi$.
   (b) $\Pi U = \Pi$.
   (c) $\Pi U \Pi = \Pi$.
2. $U = \Pi + (I - \Pi) U (I - \Pi)$.

Projection	Meaning of the image space
Π_j	Algorithm stops before or at stage j
$\Pi_j\overline{\Pi_b}$	Algorithm succeeds before or at stage j
$\Pi_j\Pi_b$	Algorithm fails before or at stage j
$\Pi_j - \Pi_{j-1}$	Algorithm stops exactly at stage j
$(\Pi_j - \Pi_{j-1})\,\overline{\Pi_b}$	Algorithm succeeds exactly at stage j
$(\Pi_j - \Pi_{j-1})\,\Pi_b$	Algorithm fails exactly at stage j
$\overline{\Pi_j}$	Algorithm is still running at stage j
$\overline{\Pi_j\overline{\Pi_b}}$	Algorithm does not succeed at stage j
$\overline{\Pi_j\Pi_b}$	Algorithm does not fail at stage j

Table 3: Orthogonal projections and meaning of their image spaces in the description of variable time algorithms.

Proof. We start with the equivalence 1(a) $\Leftrightarrow$ 1(b) which follows from a direct verification

$$U\Pi = \Pi \quad \Leftrightarrow \quad \Pi U^\dagger = \Pi \quad \Leftrightarrow \quad \Pi = \Pi U. \tag{252}$$

Now assuming either 1(a) or 1(b), proving 1(c) is trivial.

The implication 1(c) $\Rightarrow$ 1(a) can be established by a trace argument. Note that

$$\begin{aligned}\mathbf{Tr}\left(\Pi(U-I)^\dagger(U-I)\Pi\right) &= \mathbf{Tr}\left(\Pi\left(2I-U-U^\dagger\right)\Pi\right) = \mathbf{Tr}\left(2\Pi - \Pi U\Pi - (\Pi U\Pi)^\dagger\right)\\ &= \mathbf{Tr}\,(2\Pi - \Pi - \Pi) = 0.\end{aligned} \tag{253}$$

This means that $(U-I)\Pi = 0$, which is equivalent to 1(a).

Finally, if 1(a) (hence also 1(b) and 1(c)) is true,

$$\begin{aligned}U &= (\Pi + (I-\Pi))\,U\,(\Pi + (I-\Pi))\\ &= \Pi U\Pi + \Pi U(I-\Pi) + (I-\Pi)U\Pi + (I-\Pi)U(I-\Pi)\\ &= \Pi + \Pi(I-\Pi) + (I-\Pi)\Pi + (I-\Pi)U\Pi + (I-\Pi)U(I-\Pi)\\ &= \Pi + (I-\Pi)U(I-\Pi),\end{aligned} \tag{254}$$

which proves Claim 2. The reverse direction follows from a direct calculation. □

Proposition 14. *Let $\{\Pi_j\}_{j=0}^m$ be orthogonal projections partially ordered as $0 = \Pi_0 \le \Pi_1 \le \cdots \le \Pi_m = I$. Let Π_b be an orthogonal projection commuting with all Π_j: $\Pi_b\Pi_j = \Pi_j\Pi_b$ for $j = 0, \ldots, m$. Let A_j be unitaries such that $A_j\Pi_{j-1} = \Pi_{j-1}$ for all $j = 1, \ldots, m$, and $A_0 = I$. Then,*

$$\begin{aligned}&A_j\Pi_l = \Pi_l = \Pi_l A_j, \qquad && A_j\Pi_l\Pi_b = \Pi_l\Pi_b A_j,\\ &A_j\overline{\Pi_l} = \overline{\Pi_l} = \overline{\Pi_l}A_j, \qquad && A_j\overline{\Pi_l\Pi_b} = \overline{\Pi_l\Pi_b}A_j, \qquad 0 \le l < j \le m.\end{aligned} \tag{255}$$

Moreover, for any quantum state $|\psi\rangle$,

$$\left\|\overline{\Pi_j\Pi_b}|\psi\rangle\right\| = \left\|\overline{\Pi_j\Pi_b}A_k\cdots A_{j+1}|\psi\rangle\right\|, \qquad 0 \le j \le k \le m, \tag{256}$$

and for any state $|\psi_0\rangle$,

$$1 = \left\|\overline{\Pi_0\Pi_b}|\psi_0\rangle\right\| \ge \left\|\overline{\Pi_1\Pi_b}A_1|\psi_0\rangle\right\| \ge \cdots \ge \left\|\overline{\Pi_m\Pi_b}A_k\cdots A_1|\psi_0\rangle\right\| = \left\|\overline{\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\|. \tag{257}$$

Proof. We will only prove $A_j\Pi_l = \Pi_l = \Pi_l A_j$ and $A_j\overline{\Pi_l\Pi_b} = \overline{\Pi_l\Pi_b}A_j$ as verifications of other operator identities proceed in a similar way. They follow from

$$A_j\Pi_l = A_j\Pi_{j-1}\Pi_l = \Pi_{j-1}\Pi_l = \Pi_l = \Pi_l\Pi_{j-1} = \Pi_l\Pi_{j-1}A_j = \Pi_l A_j \tag{258}$$

and

$$A_j\overline{\Pi_l\Pi_b} = A_j(1-\Pi_l\Pi_b) = A_j - \Pi_l\Pi_b = A_j - \Pi_b\Pi_l = (1-\Pi_b\Pi_l)A_j = \overline{\Pi_l\Pi_b}A_j. \tag{259}$$

The norm identity then follows directly as

$$\left\|\overline{\Pi_j\Pi_b}|\psi\rangle\right\| = \left\|A_k\cdots A_{j+1}\overline{\Pi_j\Pi_b}|\psi\rangle\right\| = \left\|\overline{\Pi_j\Pi_b}A_k\cdots A_{j+1}|\psi\rangle\right\|. \tag{260}$$

Using the above identity, the claim about monotonicity is established:

$$\begin{aligned}\left\|\overline{\Pi_j\Pi_b}A_j\cdots A_1|\psi_0\rangle\right\| &= \left\|\overline{\Pi_j\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\| \\ &\geq \left\|\overline{\Pi_{j+1}\Pi_b}A_m\cdots A_1|\psi_0\rangle\right\| = \left\|\overline{\Pi_{j+1}\Pi_b}A_{j+1}\cdots A_1|\psi_0\rangle\right\|.\end{aligned} \tag{261}$$

□

A.4 Amplified algorithms

Proposition 15. *Let $\{\Pi_j\}_{j=0}^m$ be orthogonal projections partially ordered as $0=\Pi_0\leq\Pi_1\leq\cdots\leq\Pi_m=I$. Let Π_b be an orthogonal projection commuting with all Π_j: $\Pi_b\Pi_j=\Pi_j\Pi_b$ for $j=0,\ldots,m$. Let A_j be unitaries such that $A_j\Pi_{j-1}=\Pi_{j-1}$ for all $j=1,\ldots,m$, and $A_0=I$. Finally, let $\widetilde{A}_j$ be unitaries such that $\frac{\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle}{\left\|\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle\right\|} = \frac{\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle}{\left\|\overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle\right\|}$ for all $j=1,\ldots,m$, and $\widetilde{A}_0=I$. Then, for any quantum state $|\psi_0\rangle$,*

$$\frac{\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|} = \frac{\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle\right\|}, \qquad 0\leq l,j\leq k\leq m, \tag{262}$$

and

$$\frac{\left\|\overline{\Pi_h\Pi_b}A_h\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|} = \frac{\left\|\overline{\Pi_h\Pi_b}A_h\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle\right\|}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle\right\|}, \qquad 0\leq l,j\leq k,h\leq m. \tag{263}$$

Proof. By exchanging the two sides of both equations and taking reciprocal of the second one, we may assume without loss of generality that $0\leq l\leq j\leq k\leq h\leq m$. Recall that the defining property of $\widetilde{A}_j$ can be succinctly represented as $\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle \propto \overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle$. We then have

$$\begin{aligned}\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle &\propto \overline{\Pi_j\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle = \overline{\Pi_j\Pi_b}\cdot\overline{\Pi_{j-1}\Pi_b}A_j\widetilde{A}_{j-1}|\psi_0\rangle = \overline{\Pi_j\Pi_b}A_j\overline{\Pi_{j-1}\Pi_b}\widetilde{A}_{j-1}|\psi_0\rangle \\ &\propto \overline{\Pi_j\Pi_b}A_j\overline{\Pi_{j-1}\Pi_b}A_{j-1}\widetilde{A}_{j-2}|\psi_0\rangle \propto \cdots \\ &\propto \overline{\Pi_j\Pi_b}A_j\cdots\overline{\Pi_{l+1}\Pi_b}A_{l+1}\overline{\Pi_l\Pi_b}\widetilde{A}_l|\psi_0\rangle = \overline{\Pi_j\Pi_b}\cdots\overline{\Pi_{l+1}\Pi_b}A_j\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle \\ &= \overline{\Pi_j\Pi_b}A_j\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle,\end{aligned} \tag{264}$$

which proves the first claim through

$$\begin{aligned}\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle &= \overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\overline{\Pi_j\Pi_b}\widetilde{A}_j|\psi_0\rangle \\ &\propto \overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\overline{\Pi_j\Pi_b}A_j\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle = \overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle.\end{aligned} \tag{265}$$

As for the second claim, we use the first one to rewrite

$$\begin{aligned}\frac{\overline{\Pi_h\Pi_b}A_h\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|} &= \overline{\Pi_h\Pi_b}A_h\cdots A_{k+1}\frac{\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{j+1}\widetilde{A}_j|\psi_0\rangle\right\|}\\ &= \overline{\Pi_h\Pi_b}A_h\cdots A_{k+1}\frac{\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle\right\|}\\ &= \frac{\overline{\Pi_h\Pi_b}A_h\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle}{\left\|\overline{\Pi_k\Pi_b}A_k\cdots A_{l+1}\widetilde{A}_l|\psi_0\rangle\right\|}.\end{aligned} \tag{266}$$

The claim is then justified by taking the norm on both sides of the equation. □

B Tight bounds on the Dirichlet kernel

In this appendix, we prove the following bounds useful for analyzing the loss factor of amplitude amplification in Section 3.1. Up to a change of variables, they are essentially bounds on the Dirichlet kernel that arises in the Fourier analysis.

Proposition 16 (Tight bounds on the Dirichlet kernel)**.** *For any integer $\rho \geq 3$ and real number $\theta \geq 0$ such that $0 \leq \rho\theta \leq \frac{\pi}{2}$, we have*

$$1 - \frac{1}{6}\rho^2\sin^2(\theta) \leq \frac{\sin(\rho\theta)}{\rho\sin(\theta)} \leq 1 - \frac{4\pi-8}{\pi^3}\rho^2\sin^2(\theta). \tag{267}$$

The constant factors are tight in the sense that

$$\begin{aligned}\frac{1}{6} &= \inf\left\{c > 0 \;\middle|\; 1 - c\rho^2\sin^2(\theta) \leq \frac{\sin(\rho\theta)}{\rho\sin(\theta)},\ 0 \leq \rho\theta \leq \frac{\pi}{2}\right\},\\ \frac{4\pi-8}{\pi^3} &= \sup\left\{c > 0 \;\middle|\; \frac{\sin(\rho\theta)}{\rho\sin(\theta)} \leq 1 - c\rho^2\sin^2(\theta),\ 0 \leq \rho\theta \leq \frac{\pi}{2}\right\}.\end{aligned} \tag{268}$$

In particular, when $\rho = 3$,

$$\frac{\sin(3\theta)}{3\sin(\theta)} = 1 - \frac{4}{3}\sin^2(\theta). \tag{269}$$

B.1 Proof of lower bound

Let us consider the lower bound first. We start by reorganizing it as

$$\left(1 - \frac{1}{6}\rho^2\sin^2(\theta)\right) \overset{?}{\leq} \frac{\sin(\rho\theta)}{\rho\sin(\theta)} \qquad \Leftrightarrow \qquad \frac{\rho\sin(\theta) - \sin(\rho\theta)}{\rho^3\sin^3(\theta)} \overset{?}{\leq} \frac{1}{6}. \tag{270}$$

Note that

$$\lim_{\theta\to 0}\frac{\rho\sin(\theta) - \sin(\rho\theta)}{\rho^3\sin^3(\theta)} = \lim_{\theta\to 0}\frac{-\frac{2r+1}{6}\theta^3 + \frac{\rho^3}{6}\theta^3}{\rho^3\theta^3} = \frac{1}{6} - \frac{1}{6\rho^2} \nearrow \frac{1}{6}, \tag{271}$$

which shows that the lower bound is tight. So it remains to prove that the function on the left hand side is monotonically decreasing over $0 \leq \theta \leq \frac{\pi}{2\rho}$.

To this end, we differentiate it with respect to θ

$$\left(\frac{\rho\sin(\theta)-\sin(\rho\theta)}{\rho^3\sin^3(\theta)}\right)' \overset{?}{\leq} 0, \tag{272}$$

which simplifies to

$$3\sin(\rho\theta)\cos(\theta) \overset{?}{\leq} 2\rho\sin(\theta)\cos(\theta)+\rho\sin(\theta)\cos(\rho\theta). \tag{273}$$

Using the trigonometric inequalities

$$\begin{aligned}\theta-\frac{\theta^3}{6}+\frac{\theta^5}{120}-\frac{\theta^7}{5040} \leq \sin(\theta) \leq \theta-\frac{\theta^3}{6}+\frac{\theta^5}{120},\\ 1-\frac{\theta^2}{2}+\frac{\theta^4}{24}-\frac{\theta^6}{720} \leq \cos(\theta) \leq 1-\frac{\theta^2}{2}+\frac{\theta^4}{24},\end{aligned} \tag{274}$$

justified via Taylor's theorem with Lagrange's remainder, we want

$$\begin{aligned}3\left(\rho\theta-\frac{\rho^3\theta^3}{6}+\frac{\rho^5\theta^5}{120}\right)\left(1-\frac{\theta^2}{2}+\frac{\theta^4}{24}\right) \overset{?}{\leq} 2\rho\left(\theta-\frac{\theta^3}{6}+\frac{\theta^5}{120}-\frac{\theta^7}{5040}\right)\left(1-\frac{\theta^2}{2}+\frac{\theta^4}{24}-\frac{\theta^6}{720}\right)\\ +\rho\left(\theta-\frac{\theta^3}{6}+\frac{\theta^5}{120}-\frac{\theta^7}{5040}\right)\left(1-\frac{\rho^2\theta^2}{2}+\frac{\rho^4\theta^4}{24}-\frac{\rho^6\theta^6}{720}\right)\end{aligned} \tag{275}$$

or equivalently

$$\begin{aligned}3\left(1-\frac{\rho^2\theta^2}{6}+\frac{\rho^4\theta^4}{120}\right)\left(1-\frac{\theta^2}{2}+\frac{\theta^4}{24}\right) \overset{?}{\leq} 2\left(1-\frac{\theta^2}{6}+\frac{\theta^4}{120}-\frac{\theta^6}{5040}\right)\left(1-\frac{\theta^2}{2}+\frac{\theta^4}{24}-\frac{\theta^6}{720}\right)\\ +\left(1-\frac{\theta^2}{6}+\frac{\theta^4}{120}-\frac{\theta^6}{5040}\right)\left(1-\frac{\rho^2\theta^2}{2}+\frac{\rho^4\theta^4}{24}-\frac{\rho^6\theta^6}{720}\right).\end{aligned} \tag{276}$$

The left hand side of the above equation expands to

$$3+\frac{-\rho^2-3}{2}\theta^2+\frac{\rho^4+10\rho^2+5}{40}\theta^4+\frac{-\rho^2(3\rho^2+5)}{240}\theta^6+\frac{\rho^4}{960}\theta^8, \tag{277}$$

whereas the right hand side expands to

$$\begin{aligned}&3+\frac{-\rho^2-3}{2}\theta^2+\frac{5\rho^4+10\rho^2+33}{120}\theta^4\\ &+\frac{-7\rho^6-35\rho^4-21\rho^2-129}{5040}\theta^6+\frac{14\rho^6+21\rho^4+6\rho^2+82}{60480}\theta^8\\ &+\frac{-7\rho^6-5\rho^4-24}{604800}\theta^{10}+\frac{\rho^6+2}{3628800}\theta^{12}.\end{aligned} \tag{278}$$

It thus suffices to show that

$$\begin{aligned}\frac{\rho^4+10\rho^2+5}{40}\theta^4+\frac{\rho^4}{960}\theta^8 \overset{?}{\leq} \frac{5\rho^4+10\rho^2+33}{120}\theta^4\\ +\frac{-7\rho^6-35\rho^4-21\rho^2-129}{5040}\theta^6+\frac{-7\rho^6-5\rho^4-24}{604800}\theta^{10}.\end{aligned} \tag{279}$$

Note that $\theta^2 \leq \frac{\pi^2}{4\rho^2} < \frac{5}{2\rho^2}$. This upper bounds the left hand side of Eq. (279) by

$$\left(\frac{\rho^4}{40}+\frac{\rho^2}{4}+\frac{101}{768}\right)\theta^4. \tag{280}$$

Meanwhile, using $\theta^2 < \frac{5}{2\rho^2} \le \frac{5}{32}$ for $\rho \ge 4$, we lower bound the right hand side of Eq. (279) by

$$\begin{aligned}
&\frac{5\rho^4+10\rho^2+33}{120}\theta^4 + \frac{-7\rho^6-35\rho^4-21\rho^2-129}{5040}\theta^6 + \frac{-7\cdot\frac{25}{4}\rho^2-5\cdot\frac{25}{4}-24\cdot\frac{25}{1024}}{604800}\theta^6 \\
\ge\ &\frac{5\rho^4+10\rho^2+33}{120}\theta^4 + \frac{-8\rho^6-36\rho^4-22\rho^2-130}{5040}\theta^6 \\
\ge\ &\frac{5\rho^4+10\rho^2+33}{120}\theta^4 + \frac{-8\cdot\frac{5}{2}\rho^4-36\cdot\frac{5}{2}\rho^2-22\cdot\frac{5}{2}-130\cdot\frac{5}{32}}{5040}\theta^4 \\
\ge\ &\frac{190\rho^4+330\rho^2+1331-130\cdot\frac{5}{32}}{5040}\theta^4 \ge \frac{190\rho^4+330\rho^2+1310}{5040}\theta^4.
\end{aligned} \tag{281}$$

We thus want to prove that

$$\frac{\rho^4}{40}+\frac{\rho^2}{4}+\frac{101}{768} \overset{?}{\le} \frac{19\rho^4+33\rho^2+131}{504} \tag{282}$$

which can be directly verified to hold for $\rho \ge 4$.

B.2 Proof of upper bound

Recall that in the proof of lower bound, we have actually shown that when $\rho \ge 4$, $\frac{\rho\sin(\theta)-\sin(\rho\theta)}{\rho^3\sin^3(\theta)}$ monotonically decreases as a function of θ over $0 \le \theta \le \frac{\pi}{2\rho}$. Hence,

$$\frac{\rho\sin(\theta)-\sin(\rho\theta)}{\rho^3\sin^3(\theta)} \ge \frac{\rho\sin\left(\frac{\pi}{2\rho}\right)-1}{\rho^3\sin^3\left(\frac{\pi}{2\rho}\right)}. \tag{283}$$

Denoting $x = \frac{1}{\rho}$, our goal is to lower bound

$$\frac{\frac{\sin\left(\frac{\pi}{2}x\right)}{x}-1}{\frac{\sin^3\left(\frac{\pi}{2}x\right)}{x^3}} = \frac{x^2\sin\left(\frac{\pi}{2}x\right)-x^3}{\sin^3\left(\frac{\pi}{2}x\right)} \tag{284}$$

for $0 < x \le \frac{1}{3}$.

Note that

$$\lim_{x\to 0}\frac{x^2\sin\left(\frac{\pi}{2}x\right)-x^3}{\sin^3\left(\frac{\pi}{2}x\right)} = \frac{\frac{\pi}{2}-1}{\frac{\pi^3}{2^3}} = \frac{4\pi-8}{\pi^3}, \tag{285}$$

which shows that the claimed constant factor is tight. It remains to prove that the function on the left hand side is monotonically increasing over $0 < x \le \frac{1}{3}$.

To this end, we differentiate it with respect to x:

$$\left(\frac{x^2\sin\left(\frac{\pi}{2}x\right)-x^3}{\sin^3\left(\frac{\pi}{2}x\right)}\right)' = \frac{x}{2}\left(\pi x\cot\left(\frac{\pi}{2}x\right)-2\right)\csc^2\left(\frac{\pi}{2}x\right)\left(3x\csc\left(\frac{\pi}{2}x\right)-2\right). \tag{286}$$

Then, it suffices to show that

$$\pi x\cot\left(\frac{\pi}{2}x\right)-2 \overset{?}{\le} 0, \qquad 3x\csc\left(\frac{\pi}{2}x\right)-2 \overset{?}{\le} 0. \tag{287}$$

The first inequality is equivalent to $\frac{\pi}{2}x \overset{?}{\leq} \tan\left(\frac{\pi}{2}x\right)$ and becomes trivial. The second inequality is equivalent to $\frac{3}{2}x \overset{?}{\leq} \sin\left(\frac{\pi}{2}x\right)$ which also holds trivially when $x \leq \frac{1}{3}$.

Altogether, we have shown that

$$\frac{\rho\sin(\theta) - \sin(\rho\theta)}{\rho^3\sin^3(\theta)} \geq \frac{4\pi - 8}{\pi^3}. \tag{288}$$

This is equivalent to the claimed bound.

C Hermitian qubitization

In this appendix, we review results on Hermitian qubitization [38] useful for constructing the branch marking and gapped phase estimation algorithms in Section 4.1. We use $\mathcal{G}$ and $\mathcal{H}$ to represent finite-dimensional Hilbert spaces, on which all operators act.

C.1 U-cyclic subspaces

Lemma 17 (U-cyclic subspaces)**.** *Let $U : \mathcal{H} \to \mathcal{H}$ be a unitary operator and $G : \mathcal{G} \to \mathcal{H}$ be an isometry, such that $G^\dagger UG$ is Hermitian. Let $|\phi_u\rangle$ be eigenvectors of $G^\dagger UG$ with corresponding eigenvalues λ_u, and define the cyclic subspaces*

$$\mathcal{H}_u = \mathbf{Span}\left\{\ldots, U^{\dagger 2}G|\phi_u\rangle, U^\dagger G|\phi_u\rangle, G|\phi_u\rangle, UG|\phi_u\rangle, U^2G|\phi_u\rangle, \ldots\right\}. \tag{289}$$

Then:

1. *The following conditions are equivalent:*
   (a) $U^2G = G$;
   (b) $G^\dagger U^2 G = I$.

   When any of the conditions is satisfied, all $\mathcal{H}_u = \mathbf{Span}\{G|\phi_u\rangle, UG|\phi_u\rangle\}$ have dimensions $\dim(\mathcal{H}_u) = 1, 2$ and are invariant under U, $U^\dagger$ and $GG^\dagger$.

2. *The following conditions are equivalent for 1D subspaces:*
   (a) $\dim(\mathcal{H}_u) = 1$;
   (b) $\{G|\phi_u\rangle\}$ *is a basis for* $\mathcal{H}_u$;
   (c) $\lambda_u = \pm 1$;

   in which case we have the matrix representation

$$U = \left[\lambda_u\right], \qquad GG^\dagger = \left[1\right]. \tag{290}$$

3. *The following conditions are equivalent for 2D subspaces:*
   (a) $\dim(\mathcal{H}_u) = 2$;
   (b) $\{G|\phi_u\rangle, UG|\phi_u\rangle\}$ *is a basis for* $\mathcal{H}_u$;
   (c) $-1 < \lambda_u < 1$;

in which case we have the matrix representation

$$U = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad GG^\dagger = \begin{bmatrix} 1 & \lambda_u \\ 0 & 0 \end{bmatrix}. \tag{291}$$

Proof. Applying Lemma 13 to unitary U^2 and orthogonal projection $GG^\dagger$, we have

$$U^2G = G \quad \Leftrightarrow \quad U^2GG^\dagger = GG^\dagger \quad \Leftrightarrow \quad GG^\dagger U^2 GG^\dagger = GG^\dagger \quad \Leftrightarrow \quad G^\dagger U^2 G = I. \tag{292}$$

One concludes from $\left\|G^\dagger UG\right\| \leq \left\|G^\dagger\right\| \|U\| \|G\| = 1$ that the real eigenvalues of $G^\dagger UG$ must satisfy $-1 \leq \lambda_u \leq 1$. As a cyclic subspace, $\mathcal{H}_u$ is clearly invariant under U and $U^\dagger$. Furthermore, due to the equivalence $G^\dagger U^2 G = I \Leftrightarrow U^2 G = G \Leftrightarrow U^{\dagger 2} G = G$, all $U^l G|\phi_u\rangle$ can be reduced to either $G|\phi_u\rangle$ or $UG|\phi_u\rangle$, implying $\mathcal{H}_u = \mathbf{Span}\left\{G|\phi_u\rangle, UG|\phi_u\rangle\right\}$ and $\dim(\mathcal{H}_u) = 1, 2$.

Note that $G|\phi_u\rangle$ is a vector of unit length, so it must be a basis of $\mathcal{H}_u$ if $\dim(\mathcal{H}_u) = 1$. When this happens, $UG|\phi_u\rangle = \mu_u G|\phi_u\rangle$ for some complex number μ_u. But this means $\lambda_u G|\phi_u\rangle = GG^\dagger UG|\phi_u\rangle = \mu_u GG^\dagger G|\phi_u\rangle = \mu_u G|\phi_u\rangle$ and hence $|\lambda_u| = \|\lambda_u G|\phi_u\rangle\| = \|\mu_u G|\phi_u\rangle\| = \|UG|\phi_u\rangle\| = 1$, giving $\lambda_u = \pm 1$. Now assuming $\lambda_u = \pm 1$, we consider the orthogonal decomposition $UG|\phi_u\rangle = \left(GG^\dagger\right) UG|\phi_u\rangle + \left(I - GG^\dagger\right) UG|\phi_u\rangle = \lambda_u G|\phi_u\rangle + \left(I - GG^\dagger\right) UG|\phi_u\rangle$. By the Pythagorean theorem, $\left\|\left(I - GG^\dagger\right) UG|\phi_u\rangle\right\| = 0$, which implies that $\left(I - GG^\dagger\right) UG|\phi_u\rangle$ and $UG|\phi_u\rangle = \lambda_u G|\phi_u\rangle$, establishing the equivalence of three conditions. The matrix representation of U and $GG^\dagger$ then follows from a direct calculation.

When $\dim(\mathcal{H}_u) = 2$, the span set $\{G|\phi_u\rangle, UG|\phi_u\rangle\}$ is naturally a basis for $\mathcal{H}_u$. In this case, we know that $-1 \leq \lambda_u \leq 1$ and $\lambda_u \neq \pm 1$ simultaneously hold, so it must be that $-1 < \lambda_u < 1$. Similarly, assuming $-1 < \lambda_u < 1$, then it simultaneously holds that $\dim(\mathcal{H}_u) = 1, 2$ and $\dim(\mathcal{H}_u) \neq 1$, implying $\dim(\mathcal{H}_u) = 2$. The matrix representation of U and $GG^\dagger$ follows again from a direct calculation. □

C.2 Qubitization and quantum walk operator

Now, we perform the orthogonal decompositions

$$\begin{aligned} \mathcal{H} &= \mathbf{Im}\left(GG^\dagger + UGG^\dagger U^\dagger\right) \oplus \mathbf{Ker}\left(GG^\dagger + UGG^\dagger U^\dagger\right) \\ &= \left(\mathbf{Im}\left(GG^\dagger\right) + \mathbf{Im}\left(UGG^\dagger U^\dagger\right)\right) \oplus \left(\mathbf{Ker}\left(GG^\dagger\right) \cap \mathbf{Ker}\left(UGG^\dagger U^\dagger\right)\right), \end{aligned} \tag{293}$$

which are equivalent since for any two positive semidefinite operators P, Q and state $|\phi\rangle$,

$$(P+Q)|\phi\rangle = 0 \quad \Leftrightarrow \quad \langle\phi|(P+Q)|\phi\rangle = 0 \quad \Leftrightarrow \quad \langle\phi|P|\phi\rangle = \langle\phi|Q|\phi\rangle = 0 \quad \Leftrightarrow \quad P|\phi\rangle = Q|\phi\rangle = 0. \tag{294}$$

For the purpose of qubitization, we further decompose the first term into 1D and 2D subspaces introduced in the previous subsection

$$\mathbf{Im}\left(GG^\dagger\right) + \mathbf{Im}\left(UGG^\dagger U^\dagger\right) = \bigoplus_u \mathcal{H}_u. \tag{295}$$

This gives:

Proposition 18 (Hermitian qubitization)**.** *Let $U : \mathcal{H} \to \mathcal{H}$ be a unitary operator and $G : \mathcal{G} \to \mathcal{H}$ be an isometry, such that $G^\dagger UG$ is Hermitian and $G^\dagger U^2 G = I$. Let $|\phi_u\rangle$ be eigenvectors of $G^\dagger UG$ with corresponding eigenvalues λ_u, and define the cyclic subspaces*

$$\mathcal{H}_u = \mathbf{Span}\left\{\ldots, U^{\dagger 2} G|\phi_u\rangle, U^\dagger G|\phi_u\rangle, G|\phi_u\rangle, UG|\phi_u\rangle, U^2 G|\phi_u\rangle, \ldots\right\}. \tag{296}$$

Then:

1. *$\mathcal{H}$ admits the orthogonal decomposition*

$$\mathcal{H} = \bigoplus_u \mathcal{H}_u \oplus \mathcal{H}_\perp, \tag{297}$$

where $\mathcal{H}_u = \mathbf{Span}\{G|\phi_u\rangle, UG|\phi_u\rangle\}$, such that $\bigoplus_u \mathcal{H}_u = \mathbf{Im}\left(GG^\dagger\right) + \mathbf{Im}\left(UGG^\dagger U^\dagger\right)$ and $\mathcal{H}_\perp = \ker\left(GG^\dagger\right) \cap \ker\left(UGG^\dagger U^\dagger\right)$. All subspaces are invariant under U, $U^\dagger$ and $GG^\dagger$.

2. *When $\lambda_u = \pm 1$, $\{G|\phi_u\rangle\}$ is an orthonormal basis for $\mathcal{H}_u$, under which*

$$U = \left[\lambda_u\right], \qquad GG^\dagger = \left[1\right]. \tag{298}$$

3. *When $-1 < \lambda_u < 1$, $\left\{G|\phi_u\rangle, \frac{UG|\phi_u\rangle - \lambda_u G|\phi_u\rangle}{\sqrt{1-\lambda_u^2}}\right\}$ is an orthonormal basis for $\mathcal{H}_u$, under which*

$$U = \begin{bmatrix} \lambda_u & \sqrt{1-\lambda_u^2} \\ \sqrt{1-\lambda_u^2} & -\lambda_u \end{bmatrix}, \qquad GG^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{299}$$

4. *Restricted to $\mathcal{H}_\perp$, U is still a unitary and $GG^\dagger = 0$.*

Proof. We start by checking the pairwise orthogonality of $\mathcal{H}_u$ and $\mathcal{H}_v$ for $u \neq v$:

$$\begin{aligned} \langle\phi_v|G^\dagger G|\phi_u\rangle = 0, \qquad & \langle\phi_v|G^\dagger UG|\phi_u\rangle = \lambda_u\langle\phi_v|\phi_u\rangle = 0, \\ \langle\phi_v|G^\dagger U^\dagger G|\phi_u\rangle = \lambda_v\langle\phi_v|\phi_u\rangle = 0, \qquad & \langle\phi_v|G^\dagger U^\dagger UG|\phi_u\rangle = \langle\phi_v|\phi_u\rangle = 0. \end{aligned} \tag{300}$$

Moreover, we have $\bigoplus_u \mathbf{Span}\{G|\phi_u\rangle\} = \mathbf{Im}(G) = \mathbf{Im}(GG^\dagger)$ and $\bigoplus_u \mathbf{Span}\{UG|\phi_u\rangle\} = \mathbf{Im}(UG) = \mathbf{Im}(UGG^\dagger U^\dagger)$. Combining with the analysis proceeding this theorem, we have established the claimed orthogonal decomposition. Since $\bigoplus_u \mathcal{H}_u$ is invariant under the normal operators U and $GG^\dagger$, its orthogonal complement $\mathcal{H}_\perp$ is also invariant under U and $GG^\dagger$.

The statement about 1D subspaces is already proved in Lemma 17. When $\dim(\mathcal{H}_u) = 2$, we can construct an orthonormal basis by applying the Gram-Schmidt process to $\{G|\phi_u\rangle, UG|\phi_u\rangle\}$. This produces the unit basis vector

$$\begin{aligned} \frac{UG|\phi_u\rangle - G|\phi_u\rangle\langle\phi_u|G^\dagger UG|\phi_u\rangle}{\|UG|\phi_u\rangle - G|\phi_u\rangle\langle\phi_u|G^\dagger UG|\phi_u\rangle\|} &= \frac{UG|\phi_u\rangle - \lambda_u G|\phi_u\rangle}{\|UG|\phi_u\rangle - \lambda_u G|\phi_u\rangle\|} \\ &= \frac{UG|\phi_u\rangle - \lambda_u G|\phi_u\rangle}{\sqrt{1+\lambda_u^2 - \lambda_u\langle\phi_u|G^\dagger UG|\phi_u\rangle - \lambda_u\langle\phi_u|G^\dagger U^\dagger G|\phi_u\rangle}} \\ &= \frac{UG|\phi_u\rangle - \lambda_u G|\phi_u\rangle}{\sqrt{1-\lambda_u^2}} \end{aligned} \tag{301}$$

orthogonal to $G|\phi_u\rangle$. The matrix representation then follows from a direct calculation. □

For a 2D subspace $\mathcal{H}_u$, we see that $\left\{|\phi_{u,0}\rangle = G|\phi_u\rangle, |\phi_{u,1}\rangle = \frac{UG|\phi_u\rangle - \lambda_u G|\phi_u\rangle}{\sqrt{1-\lambda_u^2}}\right\}$ is an orthonormal basis, under which

$$U = \begin{bmatrix} \lambda_u & \sqrt{1-\lambda_u^2} \\ \sqrt{1-\lambda_u^2} & -\lambda_u \end{bmatrix}, \qquad GG^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{302}$$

This means that the quantum walk operator defined by $W = \left(2GG^\dagger - I\right)U$ has the matrix representation

$$W = \left(2GG^\dagger - I\right)U = \begin{bmatrix} \lambda_u & \sqrt{1-\lambda_u^2} \\ -\sqrt{1-\lambda_u^2} & \lambda_u \end{bmatrix} = \lambda_u I + i\sqrt{1-\lambda_u^2}Y. \tag{303}$$

expressed as a linear combination of Pauli operators. Therefore, for each u, W has two eigenvalues

$$\lambda_{u,\pm} = \lambda_u \pm i\sqrt{1-\lambda_u^2} = e^{\pm i \arccos(\lambda_u)} \tag{304}$$

with the associated eigenvectors

$$|\phi_{u,\pm}\rangle = \frac{|\phi_{u,0}\rangle \pm i|\phi_{u,1}\rangle}{\sqrt{2}}. \tag{305}$$

We thus obtain the following spectral decomposition of the walk operator.

Corollary 19 (Quantum walk)**.** *Let $U : \mathcal{H} \to \mathcal{H}$ be a unitary operator and $G : \mathcal{G} \to \mathcal{H}$ be an isometry, such that $G^\dagger UG$ is Hermitian and $G^\dagger U^2 G = I$. If $G^\dagger UG$ has the spectral decomposition*

$$G^\dagger UG = \sum_u \lambda_u |\phi_u\rangle\langle\phi_u|, \tag{306}$$

then the quantum walk operator $W = \left(2GG^\dagger - I\right) U$ has the spectral decomposition

$$\begin{aligned} W &= \left(2GG^\dagger - I\right) U \\ &= \sum_{|\lambda_u|=1} \lambda_u |\phi_{u,0}\rangle\langle\phi_{u,0}| + \sum_{|\lambda_u|<1} \left(e^{+i\arccos(\lambda_u)} |\phi_{u,+}\rangle\langle\phi_{u,+}| + e^{-i\arccos(\lambda_u)} |\phi_{u,-}\rangle\langle\phi_{u,-}|\right) \end{aligned} \tag{307}$$

within the space $\mathbf{Im}\left(GG^\dagger\right) + \mathbf{Im}\left(UGG^\dagger U^\dagger\right)$, where

$$|\phi_{u,0}\rangle = G|\phi_u\rangle, \qquad |\phi_{u,1}\rangle = \frac{UG|\phi_u\rangle - \lambda_u G|\phi_u\rangle}{\sqrt{1-\lambda_u^2}}, \qquad |\phi_{u,\pm}\rangle = \frac{|\phi_{u,0}\rangle \pm i|\phi_{u,1}\rangle}{\sqrt{2}}. \tag{308}$$

D Gapped phase estimation with branch marking

In this appendix, we provide details on the construction of the gapped phase estimation algorithm with branch marking, which are used in Section 4.1 for preparing the discretized inverse state.

D.1 Simultaneous Fourier approximation of even and odd functions

We begin by considering the general problem of applying functions to the eigenphases of an input unitary. Specifically, suppose that the given unitary has the spectral decomposition $U = \sum_v e^{i\theta_v} |\phi_v\rangle\langle\phi_v|$. Our goal is to obtain an operator V close to $\sum_v \left(f_a(\theta_v) I + i f_c(\theta_v) X\right) \otimes |\phi_v\rangle\langle\phi_v|$ acting jointly on the input register and an ancilla, for some desired even/odd periodic functions f_a and f_c, using oracular queries to the unitary operator U and its inverse. This problem is solved by the quantum signal processing technique [37, 40], which we review below.

Lemma 20 (Quantum signal processing)**.** *Let $n > 0$ be an integer, $f_a(\theta) = \sum_{k=0}^n a_j \cos(k\theta)$ and $f_c(\theta) = \sum_{k=1}^n a_j \sin(k\theta)$ be real Fourier series, such that $f_a^2(\theta) + f_c^2(\theta) \le 1$ for all $\theta \in [-\pi, \pi]$ and $f_a(0) = 1$. Given oracular access to $U = \sum_v e^{i\theta_v} |\phi_v\rangle\langle\phi_v|$, there exists a quantum circuit V acting on the system register and a single-qubit ancilla as*

$$V = \sum_v \left(f_a(\theta_v) I + i f_b(\theta_v) Z + i f_c(\theta_v) X + i f_d(\theta_v) Y\right) \otimes |\phi_v\rangle\langle\phi_v|, \tag{309}$$

with some even/odd real 2π-periodic functions f_b/f_d satisfying $f_a^2(\theta) + f_b^2(\theta) + f_c^2(\theta) + f_d^2(\theta) = 1$ for all $\theta \in [-\pi, \pi]$, using at most $2n$ queries to U and $U^\dagger$.

Note that this is slightly different from the setting of QSVT where we are often interested in a single function with a fixed parity—here we want to construct a unitary V that simultaneously implements both even and odd functions. In practice, the desired f_a and f_c can be obtained by truncating an infinite Fourier series. We then slightly modify the functions to satisfy the prerequisites $f_a^2(\theta) + f_c^2(\theta) \leq 1$ and $f_a(0) = 1$. We will come back to this point momentarily.

Now we describe how to construct the Fourier approximation. Our starting point is the following Chebyshev approximation of the sign function [36, 58]

$$\sum_{\substack{j=0 \\ j \text{ odd}}}^{n} \beta_j \mathbf{T}_j(x) \in \begin{cases} [-1, -1+\epsilon], & x \in (-\infty, -\nu], \\ [-1, 1], & x \in [-\nu, \nu], \\ [1-\epsilon, 1], & x \in [\nu, +\infty), \end{cases} \tag{310}$$

for arbitrary margin $\nu > 0$, accuracy $\epsilon > 0$ and some coefficients β_j. Here, one can choose the polynomial degree to scale like

$$n = \mathbf{O}\left(\frac{1}{\nu} \log\left(\frac{1}{\epsilon}\right)\right). \tag{311}$$

From this, we then construct a Fourier approximation of the periodic sign function by substituting $x = \sin(\theta)$ and $\nu = \sin(\varphi)$, assuming $0 < \varphi \leq \frac{\pi}{2}$. Within the period of $[-\pi, \pi]$, it has the following behavior

$$g(\theta) = \sum_{\substack{j=0 \\ j \text{ odd}}}^{n} \beta_j \mathbf{T}_j(\sin(\theta)) \in \begin{cases} [-1, 1], & x \in [-\pi, -\pi+\varphi], \\ [-1, -1+\epsilon], & x \in [-\pi+\varphi, -\varphi], \\ [-1, 1], & x \in [-\varphi, \varphi], \\ [1-\epsilon, 1], & x \in [\varphi, \pi-\varphi], \\ [-1, 1], & x \in [\pi-\varphi, \pi]. \end{cases} \tag{312}$$

Now assuming $0 < \varphi \leq \theta_0 \leq \frac{\pi}{2}$, we reflect and shift the periodic sign function to construct the threshold functions for GPE. Specifically, we define

$$\begin{aligned} f_c(\theta) &= \frac{g(\theta - \theta_0) - g(-\theta - \theta_0)}{2} = \frac{\sum_{\substack{j=0 \\ j \text{ odd}}}^{n} \beta_j \mathbf{T}_j(\sin(\theta - \theta_0)) - \sum_{\substack{j=0 \\ j \text{ odd}}}^{n} \beta_j \mathbf{T}_j(\sin(-\theta - \theta_0))}{2} \\ &= \frac{\sum_{\substack{j=0 \\ j \text{ odd}}}^{n} \beta_j \sin(j\theta - j\theta_0)(-1)^{\frac{j-1}{2}} - \sum_{\substack{j=0 \\ j \text{ odd}}}^{n} \beta_j \sin(-j\theta - j\theta_0)(-1)^{\frac{j-1}{2}}}{2} \\ &= \sum_{\substack{j=0 \\ j \text{ odd}}}^{n} \beta_j \sin(j\theta)\cos(j\theta_0)(-1)^{\frac{j-1}{2}}, \end{aligned} \tag{313}$$

θ	$g(\theta-\theta_0)$	$g(-\theta-\theta_0)$	$f_c(\theta)$	$f_a(\theta)$
$[-\pi, -\pi+\theta_0-\varphi]$	$[1-\epsilon, 1]$	$[1-\epsilon, 1]$	$[-\frac{\epsilon}{2}, \frac{\epsilon}{2}]$	$[-1, -1+\epsilon]$
$[-\pi+\theta_0-\varphi, -\pi+\theta_0+\varphi]$	$[-1, 1]$	$[1-\epsilon, 1]$	$[-1, \frac{\epsilon}{2}]$	$[-1, \frac{\epsilon}{2}]$
$[-\pi+\theta_0+\varphi, -\theta_0-\varphi]$	$[-1, -1+\epsilon]$	$[1-\epsilon, 1]$	$[-1, -1+\epsilon]$	$[-\frac{\epsilon}{2}, \frac{\epsilon}{2}]$
$[-\theta_0-\varphi, -\theta_0+\varphi]$	$[-1, -1+\epsilon]$	$[-1, 1]$	$[-1, \frac{\epsilon}{2}]$	$[-\frac{\epsilon}{2}, 1]$
$[-\theta_0+\varphi, \theta_0-\varphi]$	$[-1, -1+\epsilon]$	$[-1, -1+\epsilon]$	$[-\frac{\epsilon}{2}, \frac{\epsilon}{2}]$	$[1-\epsilon, 1]$
$[\theta_0-\varphi, \theta_0+\varphi]$	$[-1, 1]$	$[-1, -1+\epsilon]$	$[-\frac{\epsilon}{2}, 1]$	$[-\frac{\epsilon}{2}, 1]$
$[\theta_0+\varphi, \pi-\theta_0-\varphi]$	$[1-\epsilon, 1]$	$[-1, -1+\epsilon]$	$[1-\epsilon, 1]$	$[-\frac{\epsilon}{2}, \frac{\epsilon}{2}]$
$[\pi-\theta_0-\varphi, \pi-\theta_0+\varphi]$	$[1-\epsilon, 1]$	$[-1, 1]$	$[-\frac{\epsilon}{2}, 1]$	$[-1, \frac{\epsilon}{2}]$
$[\pi-\theta_0+\varphi, \pi]$	$[1-\epsilon, 1]$	$[1-\epsilon, 1]$	$[-\frac{\epsilon}{2}, \frac{\epsilon}{2}]$	$[-1, -1+\epsilon]$

Table 4: Qualitative behavior of the Fourier approximation of threshold functions used in branch marking and GPE.

and

$$\begin{aligned} f_a(\theta) = \frac{-g(\theta-\theta_0)-g(-\theta-\theta_0)}{2} &= \frac{-\sum\limits_{\substack{j=0\\ j \text{ odd}}}^{n} \beta_j \mathbf{T}_j(\sin(\theta-\theta_0)) - \sum\limits_{\substack{j=0\\ j \text{ odd}}}^{n} \beta_j \mathbf{T}_j(\sin(-\theta-\theta_0))}{2} \\ &= \frac{-\sum\limits_{\substack{j=0\\ j \text{ odd}}}^{n} \beta_j \sin(j\theta - j\theta_0)(-1)^{\frac{j-1}{2}} - \sum\limits_{\substack{j=0\\ j \text{ odd}}}^{n} \beta_j \sin(-j\theta - j\theta_0)(-1)^{\frac{j-1}{2}}}{2} \\ &= \sum_{\substack{j=0\\ j \text{ odd}}}^{n} \beta_j \cos(j\theta)\sin(j\theta_0)(-1)^{\frac{j-1}{2}}. \end{aligned} \tag{314}$$

The above functions actually depend on various parameters such as θ_0, φ and ϵ, but when their values are clear from the context, we will suppress these dependences for presentational purposes. The qualitative behaviors of these functions are shown in Table 4 and Figure 3.

D.2 Branch marking

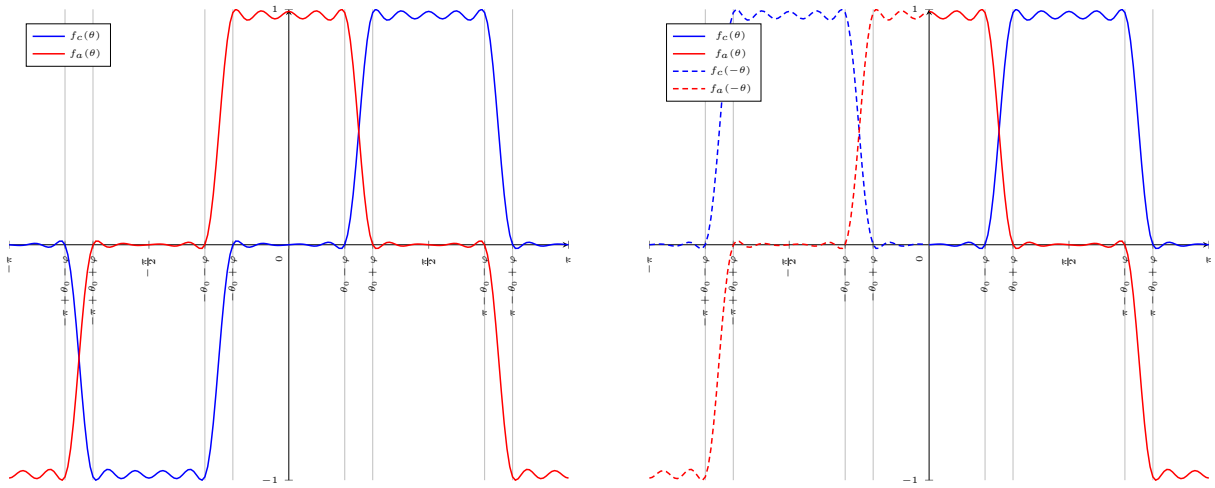
Recall from Section 4.1 that our preparation of the discretized inverse state uses the quantum walk operator. When the input matrix has the spectral decomposition

$$A = \sum_u \lambda_u |\phi_u\rangle\langle\phi_u|, \tag{315}$$

the walk operator has the corresponding spectral decomposition

$$W = \sum_u \left(e^{i\theta_{u,+}} |\phi_{u,+}\rangle\langle\phi_{u,+}| + e^{i\theta_{u,-}} |\phi_{u,-}\rangle\langle\phi_{u,-}| \right), \qquad \theta_{u,\pm} = \pm \arccos\left(\frac{\lambda_u}{\alpha_A}\right). \tag{316}$$

Here, each eigenvector $|\phi_u\rangle$ of the coefficient matrix splits into 2 eigenvectors $|\phi_{u,\pm}\rangle$ of the quantum walk opearator, and our primary goal is to ensure that these two branches undergo the exact same computation in GPE. This way, they can be merged back to recover $|\phi_{u,0}\rangle$, on which we perform the eigenvalue inversion $\frac{1}{\lambda_u}$.



(a) (b)

Figure 3: Illustration of the qualitative behavior of functions used in branch marking (left panel) and GPE (right panel).

To this end, we choose the block encoding normalization factor $\alpha_A \geq 2\|A\|$, so that

$$\frac{\lambda_u}{\alpha_A} \in \left[-\frac{1}{2}, \frac{1}{2}\right]. \tag{317}$$

Then in the quantum walk operator, this eigenvalue corresponds to two eigenphases

$$\arccos\left(\frac{\lambda_u}{\alpha_A}\right) \in \left[\frac{\pi}{3}, \frac{2\pi}{3}\right], \qquad -\arccos\left(\frac{\lambda_u}{\alpha_A}\right) \in \left[-\frac{2\pi}{3}, -\frac{\pi}{3}\right]. \tag{318}$$

To distinguish between these cases, we can implement the periodic threshold functions with $\theta_0 = \varphi = \frac{\pi}{6}$. This gives

$$f_c\left(\pm\arccos\left(\frac{\lambda_u}{\alpha_A}\right)\right) \approx \pm 1, \qquad f_a\left(\pm\arccos\left(\frac{\lambda_u}{\alpha_A}\right)\right) \approx 0. \tag{319}$$

We now carefully analyze the error. Specifically, we consider the two periodic threshold functions $f_c(\theta)$ and $f_a(\theta)$ with $\theta_0 = \varphi = \frac{\pi}{6}$ whose actions are given by Table 4 and Figure 3. By symmetry, it suffices to focus on the interval $\left[\frac{\pi}{3}, \frac{2\pi}{3}\right]$. Recall that $f_c(\theta) = \frac{g(\theta-\theta_0)-g(-\theta-\theta_0)}{2}$ and $f_a(\theta) = \frac{-g(\theta-\theta_0)-g(-\theta-\theta_0)}{2}$, which implies

$$f_c^2(\theta) + f_a^2(\theta) = \frac{g^2(\theta-\theta_0) + g^2(-\theta-\theta_0)}{2}. \tag{320}$$

For all $-\pi \leq \theta \leq \pi$, we have from Table 4 that $|g(\theta-\theta_0)|, |g(-\theta-\theta_0)| \leq 1$, so $f_c^2(\theta) + f_a^2(\theta) \leq 1$ always holds. Also we know from Table 4 that $1-\epsilon \leq f_a(0) \leq 1$. Furthermore, for the target interval where $\frac{\pi}{3} \leq \theta \leq \frac{2\pi}{3}$, we have $1-\epsilon \leq g(\theta-\theta_0) \leq 1$ and $-1 \leq g(-\theta-\theta_0) \leq -1+\epsilon$, which implies $1-\epsilon \leq f_c(\theta) \leq 1$. To summarize,

$$\begin{cases} f_a^2(\theta) + f_c^2(\theta) \leq 1, & \forall \theta \in [-\pi, \pi], \\ 1-\epsilon \leq f_a(0) \leq 1, & \\ 1-\epsilon \leq f_c(\theta) \leq 1, & \forall \theta \in \left[\frac{\pi}{3}, \frac{2\pi}{3}\right], \\ -1 \leq f_c(\theta) \leq -1+\epsilon, & \forall \theta \in \left[-\frac{2\pi}{3}, -\frac{\pi}{3}\right]. \end{cases} \tag{321}$$

We now explain how to satisfy the requirements of Lemma 20. We first run the sum-of-squares method [40] to obtain functions $f_{b,1}(\theta)$, and $f_{d,1}(\theta)$ from

$$f_{a,1}(\theta) = f_a(\theta), \qquad f_{c,1}(\theta) = f_c(\theta), \tag{322}$$

where

$$\begin{cases} f_{a,1}^2(\theta) + f_{b,1}^2(\theta) + f_{c,1}^2(\theta) + f_{d,1}^2(\theta) = 1, \quad \forall \theta \in [-\pi, \pi], \\ f_{a,1}^2(0) + f_{b,1}^2(0) = 1. \end{cases} \tag{323}$$

Then, we let

$$f_{a,2}(\theta) = f_{a,1}(\theta) f_{a,1}(0) + f_{b,1}(\theta) f_{b,1}(0), \qquad f_{c,2}(\theta) = f_{c,1}(\theta). \tag{324}$$

Note that by the Cauchy-Schwarz inequality,

$$\begin{aligned} f_{a,2}^2(\theta) + f_{c,2}^2(\theta) &= \left(f_{a,1}(\theta) f_{a,1}(0) + f_{b,1}(\theta) f_{b,1}(0)\right)^2 + f_{c,1}^2(\theta) \\ &\leq \left(f_{a,1}^2(\theta) + f_{b,1}^2(\theta)\right)\left(f_{a,1}^2(0) + f_{b,1}^2(0)\right) + f_{c,1}^2(\theta) \\ &= f_{a,1}^2(\theta) + f_{b,1}^2(\theta) + f_{c,1}^2(\theta) \leq 1. \end{aligned} \tag{325}$$

Thus, we can rerun the sum-of-squares method to get functions $f_{b,2}(\theta)$, and $f_{d,2}(\theta)$. These functions now satisfy

$$\begin{cases} f_{a,2}^2(\theta) + f_{b,2}^2(\theta) + f_{c,2}^2(\theta) + f_{d,2}^2(\theta) = 1, \quad \forall \theta \in [-\pi, \pi], \\ f_{a,2}(0) = 1, \end{cases} \tag{326}$$

as desired, resulting in the QSP operator

$$\sum_v \left(f_{a,2}(\theta_v) I + i f_{b,2}(\theta_v) Z + i f_{c,2}(\theta_v) X + i f_{d,2}(\theta_v) Y\right) \otimes |\phi_v\rangle\langle\phi_v|, \tag{327}$$

where v goes through all 2-tuple $(u, \pm)$.

It is clear that $f_{c,2} = f_{c,1} = f_c$, so we introduce no additional error to the function f_c:

$$\begin{cases} 1 - \epsilon \leq f_{c,2}(\theta) \leq 1, & \forall \theta \in \left[\frac{\pi}{3}, \frac{2\pi}{3}\right], \\ -1 \leq f_{c,2}(\theta) \leq -1 + \epsilon, & \forall \theta \in \left[-\frac{2\pi}{3}, -\frac{\pi}{3}\right]. \end{cases} \tag{328}$$

As for the remaining components,

$$f_{a,2}^2(\theta) + f_{b,2}^2(\theta) + f_{d,2}^2(\theta) = 1 - f_{c,2}^2(\theta) \leq 1 - (1-\epsilon)^2 = 2\epsilon - \epsilon^2, \qquad \forall \theta \in \left[-\frac{2\pi}{3}, -\frac{\pi}{3}\right] \cup \left[\frac{\pi}{3}, \frac{2\pi}{3}\right]. \tag{329}$$

To proceed, we need the following distance formula for matrices expanded in the Pauli basis.

Lemma 21 (Matrix distance in the Pauli basis)**.** *For real vectors $\beta = [\beta_a\ \beta_b\ \beta_c\ \beta_d]$ and $\gamma = [\gamma_a\ \gamma_b\ \gamma_c\ \gamma_d]$,*

$$\left\|(\beta_a I + i\beta_b Z + i\beta_c X + i\beta_d Y) - (\gamma_a I + i\gamma_b Z + i\gamma_c X + i\gamma_d Y)\right\| = \|\beta - \gamma\|. \tag{330}$$

Proof. We know that the Hermitian matrix

$$(\beta_b - \gamma_b) Z + (\beta_c - \gamma_c) X + (\beta_d - \gamma_d) Y \tag{331}$$

has eigenvalues

$$\pm\sqrt{(\beta_b - \gamma_b)^2 + (\beta_c - \gamma_c)^2 + (\beta_d - \gamma_d)^2}. \tag{332}$$

The claim then follows by rescaling by i and shifting by $\beta_a - \gamma_a$. □

When $\theta \in \left[\frac{\pi}{3}, \frac{2\pi}{3}\right]$, our ideal operator is iX, so the error is bounded by

$$\begin{aligned} &\|iX - ((f_{a,2}(\theta)I + if_{b,2}(\theta)Z + if_{c,2}(\theta)X + if_{d,2}(\theta)Y))\| \\ \leq\ &\sqrt{(1 - f_{c,2}(\theta))^2 + f_{a,2}^2(\theta) + f_{b,2}^2(\theta) + f_{d,2}^2(\theta)} \\ \leq\ &\sqrt{\epsilon^2 + 2\epsilon - \epsilon^2} = \sqrt{2\epsilon}. \end{aligned} \tag{333}$$

Similarly, when $\theta \in \left[-\frac{2\pi}{3}, -\frac{\pi}{3}\right]$, our ideal operator is $-iX$, so the error is again bounded by

$$\begin{aligned} &\|-iX - ((f_{a,2}(\theta)I + if_{b,2}(\theta)Z + if_{c,2}(\theta)X + if_{d,2}(\theta)Y))\| \\ \leq\ &\sqrt{(-1 - f_{c,2}(\theta))^2 + f_{a,2}^2(\theta) + f_{b,2}^2(\theta) + f_{d,2}^2(\theta)} \\ \leq\ &\sqrt{\epsilon^2 + 2\epsilon - \epsilon^2} = \sqrt{2\epsilon}. \end{aligned} \tag{334}$$

This establishes the following:

Proposition 22 (Branch marking)**.** *Let A/α_A be block encoded by O_A with normalization factor $\alpha_A \geq 2\|A\|$. Let $A|\phi_u\rangle = \lambda_u|\phi_u\rangle$ and $W|\phi_{u,\pm}\rangle = e^{\pm i \arccos\left(\frac{\lambda_u}{\alpha_A}\right)}|\phi_{u,\pm}\rangle$ be the corresponding eigenpairs of A and the quantum walk operator W. For any $\epsilon > 0$, the isometry*

$$|+\rangle|+\rangle|\phi_{u,\pm}\rangle \mapsto |\xi_{u,\pm}\rangle|\phi_{u,\pm}\rangle, \qquad \||\xi_{u,\pm}\rangle - |+\rangle|\pm\rangle\| \leq \epsilon \tag{335}$$

can be implemented using

$$\mathbf{O}\left(\log\left(\frac{1}{\epsilon}\right)\right) \tag{336}$$

queries to O_A.

Proof. By Lemma 20 and the proceeding analysis, we can implement the operator

$$V = \sum_u \left(F(\theta_{u,+}) \otimes |\phi_{u,+}\rangle\langle\phi_{u,+}| + F(\theta_{u,-}) \otimes |\phi_{u,-}\rangle\langle\phi_{u,-}|\right), \tag{337}$$

such that $\|F(\theta_{u,\pm}) - (\pm iX)\| \leq \sqrt{2\epsilon}$ for all u. Then, the controlled unitary

$$|0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes (-iV) \tag{338}$$

implements the mapping

$$|+\rangle|+\rangle|\phi_{u,\pm}\rangle \mapsto |\pm\rangle|+\rangle|\phi_{u,\pm}\rangle \tag{339}$$

to accuracy $\sqrt{2\epsilon}$. The query complexity follows now from Eq. (311) and the rescaling $\sqrt{2\epsilon} \mapsto \epsilon$. □

D.3 Branch marked gapped phase estimation

In GPE, our goal is to distinguish between eigenvalues of the input matrix within intervals

$$\frac{\lambda_u}{\alpha_A} \in \left[-\frac{\gamma}{\rho}, \frac{\gamma}{\rho}\right] \quad \text{and} \quad \frac{\lambda_u}{\alpha_A} \in (-1, -\gamma] \cup [\gamma, 1). \tag{340}$$

When we construct the quantum walk operator, each eigenvalue is split into two eigenphases. For the positive branch, our goal is to distinguish between

$$\arccos\left(\frac{\lambda_u}{\alpha_A}\right) \in \left[\arccos\left(\frac{\gamma}{\rho}\right), \pi - \arccos\left(\frac{\gamma}{\rho}\right)\right] \tag{341}$$

and

$$\arccos\left(\frac{\lambda_u}{\alpha_A}\right) \in (0, \arccos(\gamma)] \cup [\pi - \arccos(\gamma), \pi)\,, \tag{342}$$

whereas for the negative branch, we need to differentiate

$$-\arccos\left(\frac{\lambda_u}{\alpha_A}\right) \in \left[-\pi + \arccos\left(\frac{\gamma}{\rho}\right), -\arccos\left(\frac{\gamma}{\rho}\right)\right] \tag{343}$$

and

$$-\arccos\left(\frac{\lambda_u}{\alpha_A}\right) \in (-\pi, -\pi + \arccos(\gamma)] \cup [-\arccos(\gamma), 0)\,. \tag{344}$$

Additionally, we need to ensure that GPE has exactly the same behavior on eigenphases with opposite signs.

Let us consider the positive branch first (corresponding to $\arccos\left(\frac{\lambda_u}{\alpha_A}\right)$). To this end, we choose

$$\theta_0 = \frac{\arccos(\gamma) + \arccos\left(\frac{\gamma}{\rho}\right)}{2}, \qquad \varphi = \frac{\arccos\left(\frac{\gamma}{\rho}\right) - \arccos(\gamma)}{2}. \tag{345}$$

Then, we implement

$$\sum_u (f_a(\theta_{u,+})I + if_b(\theta_{u,+})Z + if_c(\theta_{u,+})X + if_d(\theta_{u,+})Y) \otimes |\phi_{u,+}\rangle\langle\phi_{u,+}|, \tag{346}$$

where

$$f_c(\theta) \approx \begin{cases} 0, & \theta \in (0, \arccos(\gamma)]\,, \\ 1, & \theta \in \left[\arccos\left(\frac{\gamma}{\rho}\right), \pi - \arccos\left(\frac{\gamma}{\rho}\right)\right], \\ 0, & \theta \in [\pi - \arccos(\gamma), \pi)\,, \end{cases} \tag{347}$$

and

$$f_a(\theta) \approx \begin{cases} 1, & \theta \in (0, \arccos(\gamma)]\,, \\ 0, & \theta \in \left[\arccos\left(\frac{\gamma}{\rho}\right), \pi - \arccos\left(\frac{\gamma}{\rho}\right)\right], . \\ -1, & \theta \in [\pi - \arccos(\gamma), \pi)\,. \end{cases} \tag{348}$$

Note that $f_a(\theta_{u,+}) \approx \pm 1$ has different signs for eigenvalue of the input matrix $\lambda_u \gtrless 0$ with opposite signs, and the function values are also quite different in the transition band. But this is acceptable since this auxiliary information will be uncomputed in the end.

However, this choice of functions will have a different behavior on the negative branch. To address this issue, we instead aim to perform

$$\sum_v (f_a(-\theta_{u,-})I + if_b(-\theta_{u,-})Z + if_c(-\theta_{u,-})X + if_d(-\theta_{u,-})Y) \otimes |\phi_{u,-}\rangle\langle\phi_{u,-}|. \tag{349}$$

That is, the action of this new operator on negative eigenphases is exactly the same as that of the above operator on positive eigenphases. Moreover, the new polynomials automatically satisfy the requirements. In particular, it holds that $f_c(-\theta) = -f_c(\theta)$, $f_a(-\theta) = f_a(\theta)$.

Now we discuss error analysis. Without loss of generality, we focus on the distinguishment of $[0, \arccos(\gamma)]$ and $\left[\arccos\left(\frac{\gamma}{\rho}\right), \frac{\pi}{2}\right]$, since the other cases have exactly the same error by symmetry. Proceeding as in the previous subsection, we consider $f_c(\theta) = \frac{g(\theta-\theta_0)-g(-\theta-\theta_0)}{2}$ and $f_a(\theta) = \frac{-g(\theta-\theta_0)-g(-\theta-\theta_0)}{2}$ where

$$f_c^2(\theta) + f_a^2(\theta) = \frac{g^2(\theta-\theta_0) + g^2(-\theta-\theta_0)}{2}. \tag{350}$$

We also know from Table 4 that $f_c^2(\theta) + f_a^2(\theta) \leq 1$ and $1 - \epsilon \leq f_a(0) \leq 1$. Furthermore, we have $-\frac{\epsilon}{2} \leq f_c(\theta) \leq \frac{\epsilon}{2}$ and $1 - \epsilon \leq f_a(\theta) \leq 1$ for the target interval $0 \leq \theta \leq \arccos(\gamma)$, and $1 - \epsilon \leq f_c(\theta) \leq 1$ and $-\frac{\epsilon}{2} \leq f_a(\theta) \leq \frac{\epsilon}{2}$ for the target interval $\arccos\left(\frac{\gamma}{\rho}\right) \leq \theta \leq \frac{\pi}{2}$. To summarize,

$$\begin{cases} f_a^2(\theta) + f_c^2(\theta) \leq 1, & \forall \theta \in [-\pi, \pi], \\ 1 - \epsilon \leq f_a(0) \leq 1, & \\ |f_c(\theta)| \leq \frac{\epsilon}{2},\ 1 - \epsilon \leq f_a(\theta) \leq 1, & \forall \theta \in [0, \arccos(\gamma)], \\ 1 - \epsilon \leq f_c(\theta) \leq 1,\ |f_a(\theta)| \leq \frac{\epsilon}{2}, & \forall \theta \in \left[\arccos\left(\frac{\gamma}{\rho}\right), \frac{\pi}{2}\right]. \end{cases} \tag{351}$$

When $\theta \in \left[\arccos\left(\frac{\gamma}{\rho}\right), \frac{\pi}{2}\right]$, our ideal operator is iX, so the error can be bounded in a similar way as in the previous subsection

$$\begin{aligned} & \|iX - ((f_{a,2}(\theta)I + if_{b,2}(\theta)Z + if_{c,2}(\theta)X + if_{d,2}(\theta)Y))\| \\ \leq\ & \sqrt{(1 - f_{c,2}(\theta))^2 + f_{a,2}^2(\theta) + f_{b,2}^2(\theta) + f_{d,2}^2(\theta)} \\ \leq\ & \sqrt{\epsilon^2 + 2\epsilon - \epsilon^2} = \sqrt{2\epsilon}. \end{aligned} \tag{352}$$

Let us now consider $\theta \in [0, \arccos(\gamma)]$. We let

$$f_{a,1}(\theta) = f_a(\theta), \qquad f_{c,1}(\theta) = f_c(\theta), \tag{353}$$

and run the sum-of-squares method [40] to obtain

$$\begin{cases} f_{a,1}^2(\theta) + f_{b,1}^2(\theta) + f_{c,1}^2(\theta) + f_{d,1}^2(\theta) = 1, & \forall \theta \in [-\pi, \pi], \\ f_{a,1}^2(0) + f_{b,1}^2(0) = 1. \end{cases} \tag{354}$$

Continuing, we define

$$f_{a,2}(\theta) = f_{a,1}(\theta)f_{a,1}(0) + f_{b,1}(\theta)f_{b,1}(0), \qquad f_{c,2}(\theta) = f_{c,1}(\theta). \tag{355}$$

and rerun the sum-of-squares method to get

$$\begin{cases} f_{a,2}^2(\theta) + f_{b,2}^2(\theta) + f_{c,2}^2(\theta) + f_{d,2}^2(\theta) = 1, & \forall \theta \in [-\pi, \pi], \\ f_{a,2}(0) = 1. \end{cases} \tag{356}$$

This results in

$$\sum_u (f_{a,2}(\theta_{u,+})I + if_{b,2}(\theta_{u,+})Z + if_{c,2}(\theta_{u,+})X + if_{d,2}(\theta_{u,+})Y) \otimes |\phi_{u,+}\rangle\langle\phi_{u,+}|. \tag{357}$$

Now we have

$$\begin{aligned} |f_{a,2}(\theta) - f_{a,1}(\theta)| &= |f_{a,1}(\theta)(f_{a,1}(0) - 1) + f_{b,1}(\theta)f_{b,1}(0)| \\ &\leq |f_{a,1}(\theta)|\,|f_{a,1}(0) - 1| + \sqrt{1 - f_{a,1}^2(\theta)}\sqrt{1 - f_{a,1}^2(0)} \\ &\leq 1 \cdot \epsilon + \sqrt{1 - (1 - \epsilon)^2}\sqrt{1 - (1 - \epsilon)^2} = 3\epsilon - \epsilon^2, \end{aligned} \tag{358}$$

which implies

$$|f_{a,2}(\theta) - 1| \leq 4\epsilon - \epsilon^2. \tag{359}$$

Combining with the obvious requirement that $|f_{a,2}| \leq 1$, we arrive at

$$1 - 4\epsilon \leq 1 - 4\epsilon + \epsilon^2 \leq f_{a,2}(\theta) \leq 1. \tag{360}$$

When $\theta \in [0, \arccos(\gamma)]$, our ideal operator is I, so the error can be bounded as

$$\begin{aligned} & \|I - ((f_{a,2}(\theta)I + if_{b,2}(\theta)Z + if_{c,2}(\theta)X + if_{d,2}(\theta)Y))\| \\ \leq & \sqrt{(1 - f_{a,2}(\theta))^2 + f_{b,2}^2(\theta) + f_{c,2}^2(\theta) + f_{d,2}^2(\theta)} \\ \leq & \sqrt{(4\epsilon)^2 + 1 - (1 - 4\epsilon)^2} = 2\sqrt{2\epsilon}. \end{aligned} \tag{361}$$

Altogether, we have established the following proposition. Note it is important that the output state $|\xi_u\rangle$ of gapped phase estimation has *no* dependence on the $\pm$ branches of eigenphases. This is the primary goal of branch marking.

Proposition 23 (Branch marked gapped phase estimation)**.** *Let A/α_A be block encoded by O_A with normalization factor $\alpha_A \geq 2\|A\|$. Let $A|\phi_u\rangle = \lambda_u|\phi_u\rangle$ and $W|\phi_{u,\pm}\rangle = e^{\pm i \arccos\left(\frac{\lambda_u}{\alpha_A}\right)}|\phi_{u,\pm}\rangle$ be the corresponding eigenpairs of A and the quantum walk operator W. For any $\epsilon > 0$, $0 < \gamma < \frac{\alpha_A}{2}$ and constant $\rho > 0$, the isometry*

$$|\pm\rangle|0\rangle|\phi_{u,\pm}\rangle \mapsto |\pm\rangle|\xi_u\rangle|\phi_{u,\pm}\rangle, \qquad \begin{cases} \||\xi_u\rangle - |0\rangle\| \leq \epsilon, & \frac{\lambda_u}{\alpha_A} \in [\gamma, 1), \\ \||\xi_u\rangle - i|1\rangle\| \leq \epsilon, & \frac{\lambda_u}{\alpha_A} \in \left[-\frac{\gamma}{\rho}, \frac{\gamma}{\rho}\right], \\ \||\xi_u\rangle - (-|0\rangle)\| \leq \epsilon, & \frac{\lambda_u}{\alpha_A} \in (-1, -\gamma], \end{cases} \tag{362}$$

can be implemented using

$$\mathbf{O}\left(\frac{1}{\gamma}\log\left(\frac{1}{\epsilon}\right)\right) \tag{363}$$

queries to O_A.

Proof. By Lemma 20 and the proceeding analysis, we can implement the operators

$$V_+ = \sum_u F(\theta_{u,+}) \otimes |\phi_{u,+}\rangle\langle\phi_{u,+}|, \qquad V_- = \sum_u F(-\theta_{u,-}) \otimes |\phi_{u,-}\rangle\langle\phi_{u,-}| \tag{364}$$

respectively, such that

$$\begin{cases} \|F(\theta) - I\| \leq 2\sqrt{2\epsilon}, & \theta \in (0, \arccos(\gamma)], \\ \|F(\theta) - iX\| \leq \sqrt{2\epsilon}, & \theta \in \left[\arccos\left(\frac{\gamma}{\rho}\right), \pi - \arccos\left(\frac{\gamma}{\rho}\right)\right], \\ \|F(\theta) - (-I)\| \leq 2\sqrt{2\epsilon}, & \theta \in [\pi - \arccos(\gamma), \pi). \end{cases} \tag{365}$$

Then, the controlled unitary

$$|+\rangle\langle+| \otimes V_+ + |-\rangle\langle-| \otimes V_- \tag{366}$$

implements the mapping

$$|\pm\rangle|0\rangle|\phi_{u,\pm}\rangle \mapsto \begin{cases} |\pm\rangle|0\rangle|\phi_{u,\pm}\rangle, & \frac{\lambda_u}{\alpha_A} \in [\gamma, 1), \\ i|\pm\rangle|1\rangle|\phi_{u,\pm}\rangle, & \frac{\lambda_u}{\alpha_A} \in \left[-\frac{\gamma}{\rho}, \frac{\gamma}{\rho}\right], \\ -|\pm\rangle|0\rangle|\phi_{u,\pm}\rangle, & \frac{\lambda_u}{\alpha_A} \in (-1, -\gamma], \end{cases} \tag{367}$$

to accuracy $2\sqrt{2\epsilon}$. The query complexity follows now from Eq. (311) and the rescaling $2\sqrt{2\epsilon} \mapsto \epsilon$. □

E Multiplicative approximation of success probabilities

In this appendix, we will consider a number of success probabilities involved in the analysis of our quantum linear system algorithm in Section 4. Specifically, we will analyze five different probabilities and show that they are in fact all constant multiplicative approximations of one another.

E.1 Definition of success probabilities

The first probability corresponds to the success probability of applying a block-encoded inverse of the input matrix to the initial state. Suppose that our input matrix is block encoded as A/α_A with normalization factor $\alpha_A \geq \|A\|$ and upper bound $\alpha_{A^{-1}} \geq \left\|A^{-1}\right\|$ on norm of the inverse matrix. Without loss of generality, we assume that both α_A and $\alpha_{A^{-1}}$ are powers of 3, so that $\alpha_A \alpha_{A^{-1}} = 3^m$ for some positive integer m. By switching to the Hermitian dilation $|0\rangle\langle 1| \otimes A + |1\rangle\langle 0| \otimes A^\dagger$, we further assume that A is Hermitian with the spectral decomposition $A = \sum_u \lambda_u |\phi_u\rangle\langle\phi_u|$ where $\frac{1}{\alpha_{A^{-1}}} \leq |\lambda_u| \leq \alpha_A$, while the initial state can be expanded in the eigenbasis of A as $|b\rangle = \sum_u \gamma_u |\phi_u\rangle$. Then, we define

$$\psi_{\text{inv}} = \frac{A^{-1}}{\alpha_{A^{-1}}}|b\rangle = \sum_u \frac{1}{\alpha_{A^{-1}}\lambda_u}\gamma_u|\phi_u\rangle, \qquad p_{\text{succ}} = \frac{\left\|A^{-1}|b\rangle\right\|^2}{\alpha_{A^{-1}}^2} = \frac{1}{\alpha_{A^{-1}}^2}\left(\sum_u \left|\frac{\gamma_u}{\lambda_u}\right|^2\right). \tag{368}$$

The second probability is an auxiliary probability for a version of discretized inverse state supported only on one basis state of the time register. We let

$$\psi_{\text{d-inv},1} = \sum_{k=0}^{m-1} \frac{3^{k+1}}{3^m}|k\rangle \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \gamma_u|\phi_u\rangle, \qquad p_{\text{succ,d-inv},1} = \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \frac{9^{k+1}}{9^m}. \tag{369}$$

The third probability corresponds to the probability of the discretized inverse state supported on two basis states of the time register. Concretely,

$$\begin{aligned} \psi_{\text{d-inv}} &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \left(\zeta_{k+1,u}\frac{3^{k+1}}{3^m}|k\rangle + \zeta_{k,u}\frac{3^k}{3^m}|k-1\rangle\right)\gamma_u|\phi_u\rangle, \\ p_{\text{succ,d-inv}} &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{j=k,k+1} |\zeta_{j,u}|^2 \frac{9^j}{9^m}. \end{aligned} \tag{370}$$

The fourth probability is also an auxiliary probability for a version of discretized inverse state with full support on all basis states of the time register, assuming branch marking is performed perfectly:

$$\begin{aligned} \psi_{\text{d-inv},m} &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} \sum_{j=1}^{m} \zeta_{j,u}\frac{3^j}{3^m}|j-1\rangle\gamma_u|\phi_u\rangle, \\ p_{\text{d-inv},m} &= \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{j=1}^{m} |\zeta_{j,u}|^2 \frac{9^j}{9^m}. \end{aligned} \tag{371}$$

Finally, the fifth probability is for the actual state $\psi_{\text{d-inv,bm}}$ prepared by the erroneous GPE and branch marking. We denote this probability by $p_{\text{succ,d-inv,bm}}$.

E.2 Multiplicative bounds on success probabilities

We now give multiplicative bounds on the success probabilities introduced in the previous subsection. Our analysis uses the following lemma.

Lemma 24. *The following estimates hold:*

$$\begin{aligned}
&\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{j \neq k, k+1} |\zeta_{j,u}|^2 \frac{9^j}{9^m} \leq \left(\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \frac{9^{k+2}}{9^m} \right) \epsilon_{gpe}^2, \\
&\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{j \neq k, k+1} |\zeta_{j,u}|^2 \frac{9^k}{9^m} \leq \left(\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \frac{9^k}{9^m} \right) \epsilon_{gpe}^2.
\end{aligned} \tag{372}$$

Proof. Recall that the cumulative coefficients $\zeta_{j,u}$ have the following bound

$$|\zeta_{j,u}| \leq \begin{cases} \epsilon_{\mathrm{gpe},j}, & j \leq k-1, \\ \prod_{l=k+1}^{j-1} \epsilon_{\mathrm{gpe},l}, & j \geq k+2. \end{cases} \tag{373}$$

Therefore, we can prove the first claim as

$$\begin{aligned}
&\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \left(\sum_{j=1}^{k-1} \epsilon_{\mathrm{gpe},j}^2 \frac{9^j}{9^m} + \sum_{j=k+2}^{m} \prod_{l=k+1}^{j-1} \epsilon_{\mathrm{gpe},l}^2 \frac{9^j}{9^m} \right) \\
&\leq \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \left(\sum_{j=1}^{k-1} \epsilon_{\mathrm{gpe},j}^2 \frac{9^{k-1}}{9^m} + \sum_{j=k+2}^{m} \epsilon_{\mathrm{gpe},j-1}^2 \frac{9^{k+2}}{9^m} \right) \\
&\leq \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{j=1}^{m-1} \epsilon_{\mathrm{gpe},j}^2 \frac{9^{k+2}}{9^m} \leq \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \frac{9^{k+2}}{9^m} \underbrace{\left(\sum_{j=1}^{m-1} \epsilon_{\mathrm{gpe},j} \right)^2}_{\epsilon_{\mathrm{gpe}}^2},
\end{aligned} \tag{374}$$

where we have assumed all $\epsilon_{\mathrm{gpe},l} \leq \frac{1}{3}$. Similarly for the second claim,

$$\begin{aligned}
&\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \left(\sum_{j=1}^{k-1} \epsilon_{\mathrm{gpe},j}^2 \frac{9^k}{9^m} + \sum_{j=k+2}^{m} \prod_{l=k+1}^{j-1} \epsilon_{\mathrm{gpe},l}^2 \frac{9^k}{9^m} \right) \\
&\leq \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \left(\sum_{j=1}^{k-1} \epsilon_{\mathrm{gpe},j}^2 \frac{9^k}{9^m} + \sum_{j=k+2}^{m} \epsilon_{\mathrm{gpe},j-1}^2 \frac{9^k}{9^m} \right) \\
&\leq \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{j=1}^{m-1} \epsilon_{\mathrm{gpe},j}^2 \frac{9^k}{9^m} \leq \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \frac{9^k}{9^m} \underbrace{\left(\sum_{j=1}^{m-1} \epsilon_{\mathrm{gpe},j} \right)^2}_{\epsilon_{\mathrm{gpe}}^2}.
\end{aligned} \tag{375}$$

□

Let us begin with the relation between p_{succ} and $p_{\text{succ,d-inv,1}}$ which is easy to describe. Following

$$\begin{aligned}
\frac{1}{\alpha_{A^{-1}}^2}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}\left|\frac{\gamma_u}{\lambda_u}\right|^2 &\leq \frac{1}{\alpha_{A^{-1}}^2}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\frac{9^{k+1}}{\alpha_A^2}=\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\frac{9^{k+1}}{9^m},\\
\frac{1}{\alpha_{A^{-1}}^2}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}\left|\frac{\gamma_u}{\lambda_u}\right|^2 &\geq \frac{1}{\alpha_{A^{-1}}^2}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\frac{9^{k}}{\alpha_A^2}=\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\frac{9^{k}}{9^m},
\end{aligned} \tag{376}$$

we conclude that

$$\frac{p_{\text{succ,d-inv,1}}}{9}\leq p_{\text{succ}}\leq p_{\text{succ,d-inv,1}}. \tag{377}$$

Then, we consider $p_{\text{succ,d-inv,1}}$ and $p_{\text{succ,d-inv}}$. It is fairly straightforward to derive one side of the bound

$$\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=k,k+1}|\zeta_{j,u}|^2\frac{9^j}{9^m}\leq\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\frac{9^{k+1}}{9^m}. \tag{378}$$

Analysis of the other side however is more involved and relies on Lemma 24:

$$\begin{aligned}
&\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=k,k+1}|\zeta_{j,u}|^2\frac{9^j}{9^m}\geq\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=k,k+1}|\zeta_{j,u}|^2\frac{9^k}{9^m}\\
&=\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=1}^{m}|\zeta_{j,u}|^2\frac{9^k}{9^m}-\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j\neq k,k+1}|\zeta_{j,u}|^2\frac{9^k}{9^m}\\
&\geq\frac{p_{\text{succ,d-inv,1}}}{9}-\frac{p_{\text{succ,d-inv,1}}}{9}\epsilon_{\text{gpe}}^2.
\end{aligned} \tag{379}$$

This gives

$$\frac{p_{\text{succ,d-inv,1}}}{9}\left(1-\epsilon_{\text{gpe}}^2\right)\leq p_{\text{succ,d-inv}}\leq p_{\text{succ,d-inv,1}}. \tag{380}$$

Next, we discuss $p_{\text{succ,d-inv}}$ and $p_{\text{succ,d-inv},m}$. Again, one side of the bound follows trivially from the definition

$$\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=k,k+1}|\zeta_{j,u}|^2\frac{9^j}{9^m}\leq\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=1}^{m}|\zeta_{j,u}|^2\frac{9^j}{9^m}. \tag{381}$$

As for the other side, we apply Lemma 24 to compute

$$\begin{aligned}
&\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=k,k+1}|\zeta_{j,u}|^2\frac{9^j}{9^m}\\
&=\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j=1}^{m}|\zeta_{j,u}|^2\frac{9^j}{9^m}-\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)}|\gamma_u|^2\sum_{j\neq k,k+1}|\zeta_{j,u}|^2\frac{9^j}{9^m}\\
&\geq p_{\text{succ,d-inv},m}-9^2p_{\text{succ,d-inv,1}}\epsilon_{\text{gpe}}^2\geq p_{\text{succ,d-inv},m}-9^3p_{\text{succ,d-inv}}\left(\frac{\epsilon_{\text{gpe}}}{1-\epsilon_{\text{gpe}}}\right)^2.
\end{aligned} \tag{382}$$

We thus obtain

$$p_{\text{succ,d-inv}} \leq p_{\text{succ,d-inv},m} \leq p_{\text{succ,d-inv}} \left(1 + 9^3 \frac{\epsilon_{\text{gpe}}^2}{(1-\epsilon_{\text{gpe}})^2}\right). \tag{383}$$

Finally, we examine $p_{\text{succ,d-inv},m}$ and $p_{\text{succ,d-inv,bm}}$. As branch marking is invoked twice, we have the additive approximation

$$|p_{\text{succ,d-inv},m} - p_{\text{succ,d-inv,bm}}| \leq 2\left|\sqrt{p_{\text{succ,d-inv},m}} - \sqrt{p_{\text{succ,d-inv,bm}}}\right| \leq 4\epsilon_{\text{bm}}. \tag{384}$$

Let us summarize these estimates as follows.

Proposition 25 (Multiplicative bounds on success probabilities)**.** *Let the five success probabilities* p_{succ}, $p_{succ,d\text{-}inv,1}$, $p_{succ,d\text{-}inv}$, $p_{succ,d\text{-}inv,m}$ *and* $p_{succ,d\text{-}inv,bm}$ *be defined as in Appendix E.1. Then,*

$$\begin{aligned}
\frac{p_{succ,d\text{-}inv,1}}{9} &\leq p_{succ} \leq p_{succ,d\text{-}inv,1},\\
\frac{p_{succ,d\text{-}inv,1}}{9}\left(1-\epsilon_{gpe}^2\right) &\leq p_{succ,d\text{-}inv} \leq p_{succ,d\text{-}inv,1},\\
p_{succ,d\text{-}inv} \leq p_{succ,d\text{-}inv,m} &\leq p_{succ,d\text{-}inv}\left(1+9^3\frac{\epsilon_{gpe}^2}{(1-\epsilon_{gpe})^2}\right),\\
|p_{succ,d\text{-}inv,m} - p_{succ,d\text{-}inv,bm}| &\leq 4\epsilon_{bm}.
\end{aligned} \tag{385}$$

Hence, as long as

$$\epsilon_{gpe} = \mathbf{O}(1), \qquad \epsilon_{bm} = \mathbf{O}\left(p_{succ}\right), \tag{386}$$

all success probabilities are constant multiplicative approximations of one another

$$p_{succ,d\text{-}inv,1},\ p_{succ,d\text{-}inv},\ p_{succ,d\text{-}inv,m},\ p_{succ,d\text{-}inv,bm} = \mathbf{\Theta}\left(p_{succ}\right). \tag{387}$$

E.3 Multiplicative bounds on sum of amplification thresholds

In Section 4.2, we have derived the following multiplicative bounds on sum of amplification thresholds

$$p_{\text{succ,d-inv}} 9^l \leq \sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j \Pi_b} A_j \cdots A_1 |\psi_0\rangle\right\|^2 9^{j-m+l} \leq \frac{5}{4} p_{\text{succ,d-inv}} 9^l, \tag{388}$$

where we assume VTAA with input algorithms $A_1, \ldots, A_m$ produces $\psi_{\text{d-inv}}$. We now incorporate the error of GPE and branch marking into the analysis.

When we consider erroneous GPE and perfect branch marking, VTAA with input algorithms $B_1, \ldots, B_m$ produces $\psi_{\text{d-inv},m}$ and the potentially good probabilities take the form

$$\begin{aligned}
\left\|\overline{\Pi_j \Pi_b} B_j \cdots B_1 |\psi_0\rangle\right\|^2 = &\sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{h=j+1}^{m} |\zeta_{h,u}|^2\\
&+ \sum_{k=0}^{m-1} \sum_{\left|\frac{\lambda_u}{\alpha_A}\right| \in \left[\frac{1}{3^{k+1}}, \frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{h=1}^{j} |\zeta_{h,u}|^2 \frac{9^h}{9^m}.
\end{aligned} \tag{389}$$

Correspondingly, the sum of amplification thresholds are split into two terms:

$$\sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}B_j\cdots B_1|\psi_0\rangle\right\|^2 9^{j-m+l} = \sum_{j=m-l+1}^{m}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{h=j+1}^{m} |\zeta_{h,u}|^2\, 9^{j-m+l} + \sum_{j=m-l+1}^{m}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{h=1}^{j} |\zeta_{h,u}|^2 \frac{9^h}{9^m} 9^{j-m+l}. \tag{390}$$

We exchange the summation order in the first term to get

$$\sum_{j=m-l+1}^{m}\sum_{h=j+1}^{m} 9^j = \sum_{h=m-l+2}^{m}\sum_{j=m-l+1}^{h-1} 9^j = \frac{9}{8}\sum_{h=m-l+2}^{m} 9^{h-1}, \tag{391}$$

whereas for the second term

$$\sum_{j=m-l+1}^{m}\sum_{h=1}^{j} 9^j = \sum_{h=1}^{m}\sum_{j=h}^{m} 9^j = \frac{9}{8}\sum_{h=1}^{m} 9^m. \tag{392}$$

Altogether,

$$\begin{aligned} p_{\text{d-inv},m}9^l &\le \sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}B_j\cdots B_1|\psi_0\rangle\right\|^2 9^{j-m+l} \\ &\le \frac{5}{4}\sum_{k=0}^{m-1}\sum_{\left|\frac{\lambda_u}{\alpha_A}\right|\in\left[\frac{1}{3^{k+1}},\frac{1}{3^k}\right)} |\gamma_u|^2 \sum_{h=1}^{m} |\zeta_{h,u}|^2\, 9^{h-m+l} \le \frac{5}{4} p_{\text{d-inv},m}9^l. \end{aligned} \tag{393}$$

Finally, we consider the general case where both GPE and branch marking are imperfect. In this case, VTAA with input algorithms $C_1,\ldots,C_m$ produces $\psi_{\text{d-inv,bm}}$ with $\|\psi_{\text{d-inv,bm}}\|^2 = p_{\text{succ,d-inv,bm}}$. Then, the sum of amplification thresholds has error

$$\left|\sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}B_j\cdots B_1|\psi_0\rangle\right\|^2 9^{j-m+l} - \sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}C_j\cdots C_1|\psi_0\rangle\right\|^2 9^{j-m+l}\right| \le \frac{9}{8}4\epsilon_{\text{bm}}9^l. \tag{394}$$

We thus obtain:

Proposition 26 (Multiplicative bounds on sum of amplification thresholds)**.** *Let the input algorithms $A_1,\ldots,A_m$, $B_1,\ldots,B_m$ and $C_1,\ldots,C_m$ be defined as in Appendix E.3. Then,*

$$\begin{aligned} p_{succ,d\text{-}inv}9^l &\le \sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}A_j\cdots A_1|\psi_0\rangle\right\|^2 9^{j-m+l} \le \frac{5}{4}p_{succ,d\text{-}inv}9^l, \\ p_{d\text{-}inv,m}9^l &\le \sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}B_j\cdots B_1|\psi_0\rangle\right\|^2 9^{j-m+l} \le \frac{5}{4}p_{d\text{-}inv,m}9^l, \\ \left|\sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}B_j\cdots B_1|\psi_0\rangle\right\|^2 9^{j-m+l} - \sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j\Pi_b}C_j\cdots C_1|\psi_0\rangle\right\|^2 9^{j-m+l}\right| &\le \frac{9}{8}4\epsilon_{bm}9^l. \end{aligned} \tag{395}$$

Hence, as long as

$$\epsilon_{gpe} = \mathbf{O}(1), \qquad \epsilon_{bm} = \mathbf{O}\left(p_{succ}\right), \tag{396}$$

all sums of amplification thresholds have the scaling

$$\begin{aligned}
\sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j \Pi_b} A_j \cdots A_1 |\psi_0\rangle\right\|^2 9^{j-m+l} &= \mathbf{\Theta}\left(p_{succ} 9^l\right), \\
\sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j \Pi_b} B_j \cdots B_1 |\psi_0\rangle\right\|^2 9^{j-m+l} &= \mathbf{\Theta}\left(p_{succ} 9^l\right), \\
\sum_{j=m-l+1}^{m} \left\|\overline{\Pi_j \Pi_b} C_j \cdots C_1 |\psi_0\rangle\right\|^2 9^{j-m+l} &= \mathbf{\Theta}\left(p_{succ} 9^l\right).
\end{aligned} \tag{397}$$

References

[1] Abhijeet Alase and Salini Karuvade, *Resolvent-based quantum phase estimation: Towards estimation of parametrized eigenvalues*, 2024, arXiv:2410.04837.

[2] Andris Ambainis, *Variable time amplitude amplification and quantum algorithms for linear algebra problems*, 29th International Symposium on Theoretical Aspects of Computer Science (STACS 2012), vol. 14, pp. 636–647, 2012, arXiv:1010.4458.

[3] Andris Ambainis, Martins Kokainis, and Jevgēnijs Vihrovs, *Improved algorithm and lower bound for variable time quantum search*, 18th Conference on the Theory of Quantum Computation, Communication and Cryptography (TQC 2023), vol. 266, pp. 7:1–7:18, 2023, arXiv:2302.06749.

[4] Dong An, Andrew M. Childs, and Lin Lin, *Quantum algorithm for linear non-unitary dynamics with near-optimal dependence on all parameters*, 2023, arXiv:2312.03916.

[5] Dong An and Lin Lin, *Quantum linear system solver based on time-optimal adiabatic quantum computing and quantum approximate optimization algorithm*, ACM Transactions on Quantum Computing **3** (2022), 1–28, arXiv:1909.05500.

[6] Yuto Ashida, Zongping Gong, and Masahito Ueda, *Non-Hermitian physics*, Advances in Physics **69** (2020), no. 3, 249–435, arXiv:2006.01837.

[7] Carl M. Bender, Dorje C. Brody, Hugh F. Jones, and Bernhard K. Meister, *Faster than Hermitian quantum mechanics*, Physical Review Letters **98** (2007), 040403, arXiv:quant-ph/0609032.

[8] Dominic W. Berry and Pedro C. S. Costa, *Quantum algorithm for time-dependent differential equations using Dyson series*, Quantum **8** (2024), 1369, arXiv:2212.03544.

[9] Dominic W. Berry, Andrew M. Childs, Aaron Ostrander, and Guoming Wang, *Quantum algorithm for linear differential equations with exponentially improved dependence on precision*, Communications in Mathematical Physics **356** (2017), 1057–1081, arXiv:1701.03684.

[10] Dominic W. Berry, Yu Tong, Tanuj Khattar, Alec White, Tae In Kim, Sergio Boixo, Lin Lin, Seunghoon Lee, Garnet Kin-Lic Chan, Ryan Babbush, and Nicholas C. Rubin, *Rapid initial state preparation for the quantum simulation of strongly correlated molecules*, 2024, arXiv:2409.11748.

[11] Gilles Brassard, Peter Hoyer, Michele Mosca, and Alain Tapp, *Quantum amplitude amplification and estimation*, Contemporary Mathematics **305** (2002), 53–74, arXiv:quant-ph/0005055.

[12] Shantanav Chakraborty, András Gilyén, and Stacey Jeffery, *The power of block-encoded matrix powers: Improved regression techniques via faster Hamiltonian simulation*, 46th International Colloquium on Automata, Languages, and Programming (ICALP 2019), vol. 132, pp. 33:1–33:14, 2019, arXiv:1804.01973.

[13] Shantanav Chakraborty, Aditya Morolia, and Anurudh Peduri, *Quantum regularized least squares*, Quantum **7** (2023), 988, arXiv:2206.13143.

[14] Sourav Chakraborty, Arkadev Chattopadhyay, Peter Høyer, Nikhil S. Mande, Manaswi Paraashar, and Ronald de Wolf, *Symmetry and quantum query-to-communication simulation*, 39th International Symposium on Theoretical Aspects of Computer Science (STACS 2022), vol. 219, pp. 20:1–20:23, 2022, arXiv:2012.05233.

[15] Andrew M. Childs, Robin Kothari, and Rolando D. Somma, *Quantum algorithm for systems of linear equations with exponentially improved dependence on precision*, SIAM Journal on Computing **46** (2017), 1920–1950, arXiv:1511.02306.

[16] B. D. Clader, B. C. Jacobs, and C. R. Sprouse, *Preconditioned quantum linear system algorithm*, Physical Review Letters **110** (2013), 250504, arXiv:1301.2340.

[17] Keith Conrad, *L^p-spaces for $0 < p < 1$*, 2019, https://kconrad.math.uconn.edu/blurbs/analysis/lpspace.pdf, Last accessed on 2024-10-14.

[18] Pedro C.S. Costa, Dong An, Yuval R. Sanders, Yuan Su, Ryan Babbush, and Dominic W. Berry, *Optimal scaling quantum linear-systems solver via discrete adiabatic theorem*, PRX Quantum **3** (2022), 040303, arXiv:2111.08152.

[19] Alexander M. Dalzell, *A shortcut to an optimal quantum linear system solver*, 2024, arXiv:2406.12086.

[20] Di Fang, Lin Lin, and Yu Tong, *Time-marching based quantum solvers for time-dependent linear differential equations*, Quantum **7** (2023), 955, arXiv:2208.06941.

[21] Yimin Ge, Jordi Tura, and J. Ignacio Cirac, *Faster ground state preparation and high-precision ground energy estimation with fewer qubits*, Journal of Mathematical Physics **60** (2019), no. 2, 022202, arXiv:1712.03193.

[22] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe, *Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics*, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, pp. 193–204, 2019, arXiv:1806.01838.

[23] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone, *Quantum metrology*, Physical Review Letters **96** (2006), 010401, arXiv:quant-ph/0509179.

[24] Lov K. Grover, *A fast quantum mechanical algorithm for database search*, Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, p. 212219, 1996, arXiv:quant-ph/9605043.

[25] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd, *Quantum algorithm for linear systems of equations*, Physical Review Letters **103** (2009), 150502, arXiv:0811.3171.

[26] Roger A. Horn and Charles R. Johnson, *Matrix analysis*, Cambridge University Press, 2012.

[27] Peter Høyer, Michele Mosca, and Ronald de Wolf, *Quantum search on bounded-error inputs*, Automata, Languages and Programming, pp. 291–299, 2003, arXiv:quant-ph/0304052.

[28] Martin Idel, *A review of matrix scaling and Sinkhorn's normal form for matrices and positive maps*, 2016, arXiv:1609.06349.

[29] Richard V Kadison and John R Ringrose, *Fundamentals of the theory of operator algebras. volume i: Elementary theory*, Academic press New York, 1983.

[30] Iordanis Kerenidis and Anupam Prakash, *Quantum recommendation systems*, 8th Innovations in Theoretical Computer Science Conference (ITCS 2017), vol. 67, pp. 49:1–49:21, 2017, arXiv:1603.08675.

[31] Erwin Kreyszig, *Introductory functional analysis with applications*, vol. 17, John Wiley & Sons, 1991.

[32] Hari Krovi, *Improved quantum algorithms for linear and nonlinear differential equations*, Quantum **7** (2023), 913, arXiv:2202.01054.

[33] Joonho Lee, Dominic W. Berry, Craig Gidney, William J. Huggins, Jarrod R. McClean, Nathan Wiebe, and Ryan Babbush, *Even more efficient quantum computations of chemistry through tensor hypercontraction*, PRX Quantum **2** (2021), 030305, arXiv:2011.03494.

[34] Lin Lin and Yu Tong, *Near-optimal ground state preparation*, Quantum **4** (2020), 372, arXiv:2002.12508.

[35] Lin Lin and Yu Tong, *Optimal polynomial based quantum eigenstate filtering with application to solving quantum linear systems*, Quantum **4** (2020), 361, arXiv:1910.14596.

[36] Guang Hao Low and Isaac L. Chuang, *Hamiltonian simulation by uniform spectral amplification*, 2017, arXiv:1707.05391.

[37] Guang Hao Low and Isaac L. Chuang, *Optimal hamiltonian simulation by quantum signal processing*, Physical Review Letters **118** (2017), 010501, arXiv:1606.02685.

[38] Guang Hao Low and Isaac L. Chuang, *Hamiltonian simulation by qubitization*, Quantum **3** (2019), 163, arXiv:1610.06546.

[39] Guang Hao Low and Yuan Su, *Quantum eigenvalue processing*, 2024, arXiv:2401.06240.

[40] Guang Hao Low, Theodore J. Yoder, and Isaac L. Chuang, *Methodology of resonant equiangular composite quantum gates*, Physical Review X **6** (2016), 041067, arXiv:1603.03996.

[41] John M. Martyn, Zane M. Rossi, Andrew K. Tan, and Isaac L. Chuang, *Grand unification of quantum algorithms*, PRX Quantum **2** (2021), 040203, arXiv:2105.02859.

[42] Sam McArdle and David P. Tew, *Improving the accuracy of quantum computational chemistry using the transcorrelated method*, 2020, arXiv:2006.11181.

[43] David Poulin and Pawel Wocjan, *Preparing ground states of quantum many-body systems on a quantum computer*, Physical Review Letters **102** (2009), 130503, arXiv:0809.2705.

[44] Matt Rosenzweig, *L^p spaces for $0 < p < 1$*, 2012, https://matthewhr.wordpress.com/wp-content/uploads/2012/09/lp-spaces-for-p-in-01.pdf, Last accessed on 2024-10-14.

[45] Artur Scherer, Benoît Valiron, Siun-Chuon Mau, Scott Alexander, Eric van den Berg, and Thomas E. Chapuran, *Concrete resource analysis of the quantum linear-system algorithm used to compute the electromagnetic scattering cross section of a 2D target*, Quantum Information Processing **16** (2017), no. 3, 60, arXiv:1505.06552.

[46] Andr Schrottenloher and Marc Stevens, *Quantum procedures for nested search problems*, IACR Communications in Cryptology **1** (2024), no. 3, 1–39.

[47] Changpeng Shao, *Computing eigenvalues of diagonalizable matrices in a quantum computer*, 2019, arXiv:1912.08015.

[48] Changpeng Shao and Jin-Peng Liu, *Solving generalized eigenvalue problems by ordinary differential equations on a quantum computer*, 2020, arXiv:2010.15027.

[49] Changpeng Shao and Hua Xiang, *Quantum circulant preconditioner for a linear system of equations*, Physical Review A **98** (2018), 062321, arXiv:1807.04563.

[50] Peter W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM Journal on Computing **26** (1997), 1484–1509, arXiv:quant-ph/9508027.

[51] Rolando D. Somma and Yiğit Subaşı, *Complexity of quantum state verification in the quantum linear systems problem*, PRX Quantum **2** (2021), 010315, arXiv:2007.15698.

[52] J Michael Steele, *The Cauchy-Schwarz master class: an introduction to the art of mathematical inequalities*, Cambridge University Press, 2004.

[53] Yiğit Subaşı, Rolando D. Somma, and Davide Orsucci, *Quantum algorithms for systems of linear equations inspired by adiabatic quantum computing*, Physical Review Letters **122** (2019), 060504, arXiv:1805.10549.

[54] Souichi Takahira, Asuka Ohashi, Tomohiro Sogabe, and Tsuyoshi S. Usuda, *Quantum algorithm for matrix functions by Cauchy's integral formula*, Quantum Information and Computation **20** (2020), 14–36, arXiv:2106.08075.

[55] Souichi Takahira, Asuka Ohashi, Tomohiro Sogabe, and Tsuyoshi Sasaki Usuda, *Quantum algorithms based on the block-encoding framework for matrix functions by contour integrals*, 2021, 2106.08076.

[56] Yu Tong, Dong An, Nathan Wiebe, and Lin Lin, *Fast inversion, preconditioned quantum linear system solvers, fast Green's-function computation, and fast evaluation of matrix functions*, Physical Review A **104** (2021), 032422, arXiv:2008.13295.

[57] Vera von Burg, Guang Hao Low, Thomas Häner, Damian S. Steiger, Markus Reiher, Martin Roetteler, and Matthias Troyer, *Quantum computing enhanced computational catalysis*, Physical Review Research **3** (2021), 033055, arXiv:2007.14460.

[58] Kianna Wan, Mario Berta, and Earl T. Campbell, *Randomized quantum algorithm for statistical phase estimation*, Physical Review Letters **129** (2022), 030503, arXiv:2110.12071.

[59] Guoming Wang, *Efficient quantum algorithms for analyzing large sparse electrical networks*, Quantum Information and Computation **17** (2017), no. 1112, 9871026, arXiv:1311.1851.

[60] Xiao-Ming Zhang, Yukun Zhang, Wenhao He, and Xiao Yuan, *Quantum eigensolver for general matrices*, 2024, arXiv:2401.12091v1.

[61] Marcin Zwierz, Carlos A. Pérez-Delgado, and Pieter Kok, *General optimality of the Heisenberg limit for quantum metrology*, Physical Review Letters **105** (2010), 180402, arXiv:1004.3944.