Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Singular Value Thresholding Algorithm Based on Variational Quantum Pre-compression

Yiwei Chen
Harbin University of Science and Technology
Xianhua Song
songxianhua@hrbust.edu.cn
Harbin University of Science and Technology
Zhixin Zhang
Harbin University of Science and Technology

Research Article

Keywords: Quantum singular value transformation, Variational quantum algorithm, Low-rank approximation, Singular value thresholding, Resource complexity

Posted Date: November 19th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11327051/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Singular Value Thresholding Algorithm Based on Variational Quantum Pre-compression

Yiwei Chen[1], Xianhua Song[1*], Zhixin Zhang[1]

[1*]School of Science, Harbin University of Science and Technology, Harbin, 150080, Heilongjiang, China.

*Corresponding author(s). E-mail(s): songxianhua@hrbust.edu.cn;

Abstract

Singular value thresholding (SVT) is a core subroutine in low-rank matrix recovery and quantum machine learning, but suffers from high computational complexity. The quantum singular value transformation (QSVT) algorithm improves performance, yet its resource usage still depends polynomially on the full rank $\boldsymbol{r}$ or dimension $\sqrt{\boldsymbol{pq}}$, leading to prohibitive costs for large-scale matrices. Based on the mathematical theory of low-rank matrix approximation, we propose a variational quantum algorithm framework, termed Variational Quantum Low-Rank SVT (VQLR-SVT). This framework utilizes a low-complexity variational quantum circuit to extract the low-dimensional submanifold spanned by the effective rank $\boldsymbol{r_\epsilon}$ of the matrix, and directly learns the threshold function from the phase estimation register, thereby compressing high-dimensional control information into a single flag qubit and implicitly realizing a low-dimensional approximation. Theoretical analysis shows that, under the mild assumption of variational training convergence, the resource complexity of QSVT operations can be reduced from poly$(\boldsymbol{r})$ to poly$(\boldsymbol{r_\epsilon})$, where $\boldsymbol{r_\epsilon} \ll \boldsymbol{r}$. We rigorously prove the accuracy guarantee and complexity upper bound of the pre-compression algorithm using matrix perturbation theory. Subsequently, through five numerical simulation experiments, we validate the efficiency advantage in terms of resource consumption versus effective rank, the theoretical upper bound of output fidelity, circuit expressibility testing, convergence of variational quantum circuit training trajectories, and cross-validation with classical algorithms along with resource comparison. Experimental results demonstrate that in typical scenarios with rapidly decaying singular values, VQLR-SVT can reduce quantum gate calls by more than **50%** while maintaining fidelity above 0.97, and the quantum resources exhibit scale-independence with respect to matrix dimensions.

Keywords: Quantum singular value transformation, Variational quantum algorithm, Low-rank approximation, Singular value thresholding, Resource complexity

1 Introduction

Singular value thresholding (SVT) is a fundamental operation widely used in low-rank matrix completion, robust principal component analysis, quantum state tomography [1], and many other applications. Given a threshold τ, SVT replaces each singular value σ_i of the input matrix $\mathbf{A}_0$ with $\max(\sigma_i - \tau, 0)$, thereby producing a low-rank output matrix. The origin of this operation can be traced back to pioneering works in compressive sensing and low-rank matrix recovery theory [2, 3]. Candès and Recht first proved that, under incoherence conditions, a low-rank matrix can be exactly reconstructed from a small number of observations via nuclear norm minimization [2]. Subsequently, Cai, Candès, and Shen systematically proposed the singular value thresholding algorithm as the proximal operator of the nuclear norm minimization problem, efficiently approximating the low-rank solution through iterative soft thresholding [3]. Since then, SVT has become a standard tool for processing low-rank structural data, with extremely wide practical applications: in recommendation systems, it fills missing entries of user-item rating matrices, as in the classic Netflix competition [4]; in computer vision, it separates video frames into low-rank background and sparse foreground for moving object detection; in quantum information processing, low-rank density matrix tomography, compressive quantum state sensing, and low-rank recovery of noisy matrices all heavily rely on singular value thresholding to ensure physical plausibility and computational feasibility.

However, each iteration of the classical SVT algorithm inevitably requires computing a singular value decomposition (SVD). For a $p \times q$ matrix, the computational complexity of SVD is as high as $O(\min(p, q)^3)$ [5, 6]. When matrix dimensions reach thousands or tens of thousands, the computational cost of classical SVT (typically relying on SVD) becomes prohibitive, severely limiting its application in large-scale recommendation, real-time video analysis, and high-dimensional quantum state tomography. In this context, researchers have proposed truncated SVD to accelerate SVT. Specifically, both methods share the same singular value thresholding operator, but differ in how the SVD required by this operator is computed. The former can be called classical exact SVD-SVT (computing the full SVD at each step, without omitting any singular value), while the latter, truncated SVD-accelerated SVT (computing only the useful singular values and vectors and discarding the rest directly), greatly accelerates the process by replacing the full decomposition with a partial decomposition [7]. However, truncated SVD still suffers from issues such as easily missing details, overall darkening of images, and incomplete mathematical theory.

Quantum computing provides a fundamentally different approach to accelerate SVT and overcome the limitations of truncated SVD. In 2018–2019, Gilyén et al. proposed the Quantum Singular Value Transformation (QSVT) framework, unifying many primitives in quantum linear algebra [8]. The theoretical foundation of QSVT rests on two key developments. The first is quantum signal processing (QSP) and qubitization, developed by Low and Chuang [9–11]. QSP provides the algebraic framework for applying polynomial transformations to quantum gate sequences, while qubitization extends this framework to block-encoded matrices by reducing their action to two-dimensional invariant subspaces. Together they enable QSVT to apply arbitrary polynomial functions to the singular values of a block-encoded matrix. The second is the classical

computation of QSP phase factors. Haah proposed a product decomposition of periodic functions in QSP [12], which improved the numerical stability of phase-factor computation but still required variable precision arithmetic and was limited to polynomials of moderate degree. Dong et al. later developed an optimization-based phase-factor evaluation method [13] that achieves machine precision using only standard double-precision arithmetic, even for polynomials of degree larger than 10,000. This makes the numerical construction of QSP-based circuits, and hence QSVT circuits, feasible for high-degree polynomials. QSVT theory indicates that, using block-encoding, quantum phase estimation, and controlled rotations, singular value thresholding can be completed with only $O(\text{poly}(r) \cdot \log(pq/\epsilon))$ measurements, achieving exponential acceleration over classical methods and bringing transformative prospects for large-scale quantum state tomography [1], quantum principal component analysis [1], and quantum-enhanced recommendation systems [4], as well as broader quantum machine learning tasks [14]. However, the standard QSVT algorithm has a fundamental bottleneck when processing full-rank or nearly full-rank matrices: the total number of quantum measurements (i.e., the total number of calls to the controlled unitary $e^{i\mathbf{A}_0 t}$) is polynomially related to the number of singular values to be resolved, r. When the rank r of the matrix approaches the matrix dimension $\min(p, q)$ or $\sqrt{pq}$ (i.e., the matrix is nearly full rank), the resource consumption of the algorithm increases sharply, causing it to still face enormous resource pressure on current noisy intermediate-scale quantum (NISQ) hardware [15]. Variational quantum algorithms offer a promising route on NISQ devices [16], but barren plateaus remain a challenge [17].

To address the above problem, this paper proposes a "pre-compression" step. Between the phase estimation and the complex controlled rotations, a low-complexity variational quantum circuit is first used to "pre-compress" the matrix. Specifically, this variational circuit projects the high-dimensional input quantum state onto a low-dimensional submanifold spanned by the dominant singular values of the matrix, and constructs a low-dimensional approximation matrix with effective rank r_ϵ ($r_\epsilon \ll r$). This variational compression strategy fully leverages the hardware characteristics of noisy intermediate-scale quantum (NISQ) devices: through classical-quantum hybrid optimization, it automatically extracts the principal component information of the matrix with extremely low quantum gate overhead [16, 18]. Then, subsequent controlled rotation operations are performed only on this low-dimensional matrix. In this way, the resource complexity of the algorithm is reduced from depending on the full rank r to depending on r_ϵ, achieving a significant reduction in resource consumption.

Several existing studies have explored the idea of first extracting low-rank structure and then performing quantum operations. However, to the best of our knowledge, a complete resource complexity analysis of quantum SVT based on variational pre-compression, along with a rigorous proof of acceleration over the original QSVT, has not been systematically studied. Existing works either focus on specific quantum circuit designs or on efficiency improvements under particular matrix structures, but lack rigorous mathematical derivation of complexity upper bounds and quantitative analysis of algorithm accuracy.

This paper fills this gap with three key contributions. First, using matrix perturbation theory, specifically the Davis–Kahan theorem and Weyl inequality, we

prove that the pre-compression error is controlled by the tolerance ϵ, and we derive an upper bound on the number of quantum measurements for VQLR-SVT as $O(\text{poly}(r_\epsilon) \cdot \log(pq/\epsilon))$ [19–21]. Second, we perform five simulation experiments on matrices with rapidly decaying singular values ($\sigma_k = 1/k^2$), validating the variation of resource consumption with effective rank, the relationship between fidelity and error, the convergence of variational training, circuit expressibility, and cross-validation with classical SVT algorithms. Third, we show that the algorithm reduces quantum gate calls by more than 50%, maintains fidelity above 0.97, and exhibits fast and stable training without barren plateaus. These advances clear key obstacles for efficient quantum SVT on NISQ hardware and open new possibilities for quantum state tomography, quantum data denoising, and quantum-enhanced recommendation systems.

2 Related Work

2.1 Singular Value Thresholding and QSVT

Singular value thresholding (SVT), as a fundamental tool for low-rank matrix recovery, classically relies on performing a full singular value decomposition of the matrix and then applying a soft-thresholding operation to the singular values. For large-scale matrices, this process is extremely computationally expensive, motivating researchers to seek more efficient algorithms.

The quantum singular value thresholding algorithm (QSVT) was proposed by Low and Chuang et al. in 2017. They first proposed quantum signal processing [9, 10]. They later developed qubitization and uniform spectral amplification [11, 22]. This unifies the theoretical tools of quantum signal processing and Hamiltonian simulation into a single framework [8–11], enabling efficient quantum versions of arbitrary matrix functions, including Hamiltonian simulation [23–25]. Within the QSVT framework, SVT can be realized by constructing a polynomial of the singular value threshold function, and its quantum circuit depth scales logarithmically with the matrix dimension. QSVT has been applied to a wide range of quantum algorithms. Quantum SDP solvers exploit QSVT for matrix arithmetic [30–33]. Block-encoded matrix powers have been used to improve regression techniques [34]. Quantum linear systems algorithms are also relevant [37]. At a more foundational level, QSVT can be viewed as a unification of amplitude amplification and estimation [35, 36], which are themselves core primitives in quantum algorithms. Algorithmic applications of QSVT are further discussed in [26]. However, the query complexity of QSVT typically depends on the rank or dimension parameters of the matrix, because the entire singular value spectrum needs to be encoded and sampled [8]. For nearly full-rank matrices, the resource overhead of QSVT remains considerable.

2.2 Low-Rank Approximation and Variational Quantum Algorithms

Low-rank approximation methods have been widely studied in classical computing. For large-scale matrices, truncated singular value decomposition, randomized singular value decomposition (Randomized SVD), and other methods can effectively reduce

computational load on classical computers [7]. Duan et al. proposed a quantum singular value thresholding algorithm [27], and further designed an efficient quantum circuit for singular-value thresholding [28]. Then, in 2022, Ge et al. proposed an optimized quantum singular value thresholding algorithm (HQSVT) for hybrid quantum computers, which can solve various nuclear norm minimization (NNM) problems by decomposing the algorithm into several key subroutines and further studying the construction of each subroutine through a hybrid system composed of qubits and quantum modules [29].

In recent years, variational quantum algorithms (VQAs) have attracted extensive attention due to their applicability on noisy intermediate-scale quantum (NISQ) devices [15, 16]. Variational quantum circuits (VQCs) use trainable parameterized quantum circuits combined with classical optimizers to approximate complex functions under limited quantum resources. A prominent advantage of VQCs is that they avoid the large-scale data input cost based on QRAM, and their parameter optimization can often be efficiently handled by classical gradient-based or gradient-free optimizers. Existing studies have explored the use of VQCs for quantum principal component analysis [1], quantum autoencoders [18], and other areas, but the work of using them for "pre-compression" to reduce QSVT resource consumption remains unexplored. A detailed comparison with these existing approaches, along with the specific gap addressed in this work, is summarized in the contribution statement of the Introduction.

3 VQLR-SVT Algorithm Design, Theoretical Proof, and Circuit Design

3.1 Overall Algorithm Framework and Circuit Design

3.1.1 Overall Algorithm Framework

The VQLR-SVT algorithm consists of three stages. The first stage is phase estimation: quantum phase estimation is applied to the initial state, writing the binary representation of the squared singular values of $\mathbf{A}_0$ into a control register C [42–44]. The second stage is pre-compression: a low-complexity variational quantum circuit compresses the high-dimensional information carried in register C into a single flag qubit, realizing a soft-threshold discrimination. The third stage is controlled rotation: controlled rotations conditioned on the state of the flag implement singular value thresholding, followed by inverse phase estimation and measurement [28, 29]. The overall circuit of the VQLR-SVT algorithm is shown in Figure 1.

The core theoretical contribution of this work is that, because the pre-compressed approximation matrix preserves the dominant singular value information of $\mathbf{A}_0$ with controllable error, the total resources required for subsequent operations are determined by the effective rank r_ϵ rather than the full rank r.

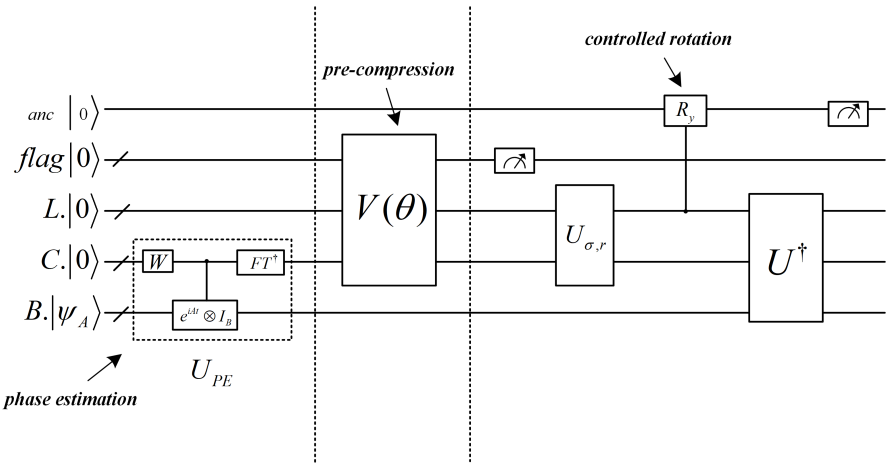


Fig. 1 Overall circuit diagram of QSVT algorithm with the pre-compression module added

3.1.2 Design and Information Compression Mechanism of the Pre-compression Module

The core innovation of the VQLR-SVT algorithm is its pre-compression module, which employs a low-depth variational quantum circuit to compress the high-dimensional feature information stored in the phase estimation register into a single flag qubit, thereby achieving soft-threshold extraction of the singular value subspace. The design of this module is inspired by the "bottleneck" compression layer in quantum autoencoders [18].

In the original QSVT algorithm, phase estimation writes the binary representation of squared singular values σ_k^2 into register C. The number of qubits in register C is $n = O(\log_2 \kappa)$, where κ is the condition number of $\mathbf{A}_0\mathbf{A}_0^\dagger$. In practical problems, n is often much larger than 1 (when $\kappa \approx 10^6$, $n \approx 20$). This means that register C carries high-dimensional binary information about the singular value spectrum.

The task of the pre-compression module is to receive the control register C produced by phase estimation as input, and compress the information distributed over n control qubits into a single flag qubit. By measuring and post-selecting on the flag, a soft-threshold effect of "retaining dominant singular value branches and discarding tiny singular value branches" can be achieved.

From an information-theoretic perspective, the pre-compression module uses a variational circuit $V(\boldsymbol{\theta})$ to synthesize the n-bit binary patterns stored in register C into a flipping effect on the flag qubit. Different σ_k^2 correspond to different n-bit patterns, so the degree of flipping of the flag also differs. Ultimately, measuring and post-selecting on the flag is equivalent to performing a "soft decision" on each branch in the superposition according to the magnitude of the eigenvalue—retaining main branches and discarding tiny branches. This is the mathematical meaning of pre-compression: compressing high-dimensional feature information into a binary keep/discard label.

3.1.3 Circuit Representation and Operation Flow

In the overall algorithm framework diagram (Figure 1), we represent register C as a multi-qubit bus with a slash annotation. To clearly illustrate the specific gate operations inside the pre-compression module, we assume $n = 3$, i.e., register C contains three control qubits C_0, C_1, C_2. When the actual $n > 3$, the circuit structure extends linearly by repeating the same controlled-gate structure for each qubit.

The circuit of the pre-compression module (Figure 2) is divided into three layers. The first layer is controlled Y-rotation (encoding layer): for each control qubit C_i in register C, a controlled Y-rotation gate R_Y is applied to the flag qubit. Each R_Y contains a trainable parameter θ_i. When C_i is $|1\rangle$, the flag rotates around the Y-axis by θ_i. Different singular values correspond to different binary patterns, causing the flag to accumulate different rotation angles, thereby achieving initial encoding.

The second layer is an entanglement layer (multi-body correlation): through the $C_X - R_Z - C_X$ sequence, entanglement is established between the flag and C_0, C_1. Specifically, a CNOT gate is applied between the flag and C_0, a parameterized R_Z gate is applied to the flag, and another CNOT gate is applied. This layer makes the state of the flag depend on the combined pattern of the control qubits, enhancing circuit expressibility.

The third layer is controlled X-rotation (refinement layer): again, for each C_i in register C, a controlled X-rotation gate R_X is applied to the flag, further refining the flipping amplitude of the flag from another rotation axis and enhancing discrimination capability.

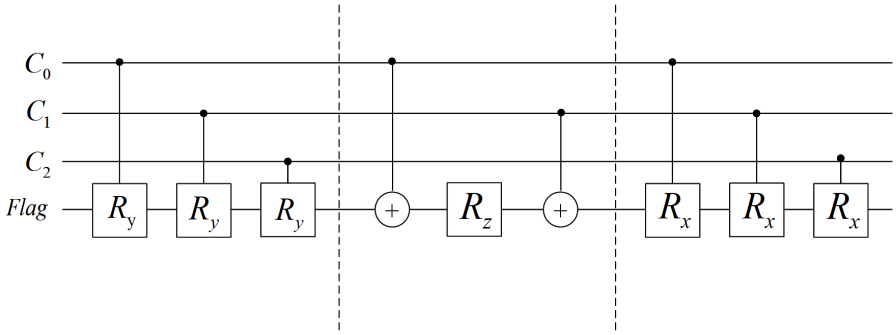


Fig. 2 Circuit diagram of the pre-compression module, where C_0, C_1, C_2 are qubits storing basis state information and flag is used for compression

3.1.4 Complexity Analysis of the Variational Circuit

In the previous subsection, we illustrated the specific gate structure of the pre-compression module using $n = 3$ as a concrete example. Here we first count the resource consumption for this $n = 3$ case, and then generalize the result to arbitrary n. For $n = 3$, the resource dimensions are summarized in Table 1.

When $n > 3$, the circuit extends linearly: trainable parameters become n, two-qubit gates become $2n+2$, and the circuit depth remains unchanged. All controlled rotations

Table 1 Complexity of the pre-compression module for $n = 3$

Resource dimension	Specific metric
Number of qubits	4 (3 control + 1 flag)
Trainable parameters	3 (three R_Y angles)
Two-qubit gates	8 (3 R_Y + 2 C_X + 3 R_X)
Single-qubit gates	1 R_Z
Circuit depth	about 5-6 layers

target the flag qubit, avoiding complex multi-control gates or long-range connections. Therefore, the pre-compression module has a circuit complexity of $O(n)$ in both qubit count and gate count, and a constant circuit depth. This design is a typical shallow, low-overhead, parameterized quantum circuit, highly suitable for running on NISQ devices [15, 16].

3.2 VQLR-SVT Complexity Theory

Theorem 1 (Quantum Resource Complexity Upper Bound) *Under error tolerance ϵ, if the input matrix $\mathbf{A}_0$ has effective rank r_ϵ and the variational circuit can successfully extract the dominant subspace (low-dimensional manifold), then the number of quantum measurements required by the VQLR-SVT algorithm is upper bounded by:*

$$O\left(poly(r_\epsilon) \cdot \log\left(\frac{pq}{\epsilon}\right)\right). \tag{1}$$

Moreover, the fidelity between the output quantum state and the ideal SVT state satisfies $F \geq 1 - O(\epsilon + \delta)$, where δ is the variational circuit training error tolerance.

Proof The proof proceeds in four steps. In Step 1, we analyze the resource consumption of the original QSVT algorithm. In Step 2, we construct a low-dimensional approximation matrix and establish its relation to the pre-compression module. In Step 3, we establish error bounds based on matrix perturbation theory, including the effect of variational training error. In Step 4, we derive the resource complexity of the new algorithm.

Step 1: Resource consumption of the original QSVT algorithm

Consider performing standard SVT on the original matrix $\mathbf{A}_0 \in \mathbb{C}^{p\times q}$. Based on the theoretical implementation of QSVT, its core resource consumption is the total number of calls to the controlled unitary $e^{i\mathbf{A}_0 t}$, denoted as T_{raw}. According to QSVT theory, if the rank of $\mathbf{A}_0$ is r (i.e., the number of non-zero singular values), multiple phase estimation iterations are required to apply the desired threshold function to all singular values. The theoretical lower bound on the total number of calls is closely related to the number of eigenvalues that need to be resolved, namely:

$$T_{\text{raw}} = O\left(\text{poly}(r) \cdot \log\left(\frac{pq}{\epsilon}\right)\right). \tag{2}$$

Here r is typically close to $\min(p, q)$ or $\sqrt{pq}$, so when the matrix dimension is large, T_{raw} becomes enormous. The core goal of this theorem is to reduce the complexity dependence from $\text{poly}(r)$ to $\text{poly}(r_\epsilon)$.

It should be noted that the basic quantum resource of the algorithm is the number of controlled block-encoding queries (i.e., calls to controlled-$e^{i\mathbf{A}t}$), denoted T. Since one complete QSVT process involves not only calls to the controlled unitary but also auxiliary qubit preparation and measurement, the controlled unitary call is the dominant term, while other overheads are poly$(\log(1/\epsilon))$. Therefore, T can be used as a substitute for complexity. The "number of quantum measurements" in Theorem 1 actually refers to the "number of controlled unitary calls." When the success probability of post-selection is P_{succ}, the number of repeated measurements (shots) is $O(1/P_{\text{succ}})$, and the total physical running time is proportional to T/P_{succ}. Here $(x)_+ = \max(x, 0)$, so only singular values above the threshold τ contribute to the success probability. Since the effective rank r_ϵ is fixed, $P_{\text{succ}} = \sum_{k=1}^{r_\epsilon}(\sigma_k - \tau)_+^2/\|\mathbf{A}_0\|_F^2$ is a constant, and thus it does not affect the order of T.

Step 2: Construction of the low-dimensional approximation matrix

Let the effective rank r_ϵ be defined as the number of singular values of $\mathbf{A}_0$ greater than the error threshold ϵ. Assume that the variational quantum circuit successfully extracts the dominant singular subspace of $\mathbf{A}_0$, obtaining an orthonormal basis matrix $\mathbf{V} \in \mathbb{C}^{q\times r_\epsilon}$ satisfying $\mathbf{V}^\top\mathbf{V} = \mathbf{I}_{r_\epsilon}$. Define the orthogonal projection operator $\mathbf{P} = \mathbf{V}\mathbf{V}^\top$, and construct the approximation matrix:

$$\tilde{\mathbf{A}}_0 = \mathbf{V}^\top\mathbf{A}_0\mathbf{V} \in \mathbb{C}^{r_\epsilon\times r_\epsilon}. \tag{3}$$

Since $\tilde{\mathbf{A}}_0$ is the coordinate representation of the projected matrix $\mathbf{P}\mathbf{A}_0\mathbf{P}$ in the subspace spanned by $\mathbf{V}$, its non-zero singular value spectrum is identical to that of $\mathbf{P}\mathbf{A}_0\mathbf{P}$, hence $\text{rank}(\tilde{\mathbf{A}}_0) \le r_\epsilon$. Thus, we have completed the compression of the large matrix $\mathbf{A}_0$ into a small matrix $\tilde{\mathbf{A}}_0$.

In the standard QSVT algorithm, phase estimation stores the binary representation of squared singular values σ_k^2 in register C. Subsequently, controlled rotations require high-precision arithmetic operations (such as Newton iteration) to compute $y_k = (1 - \tau/\sigma_k)_+$, and rotate the flag qubit accordingly [41, 45]. The core idea of the pre-compression module is to use one or more layers of a low-complexity variational quantum circuit to directly learn the mapping from the binary patterns in register C to the rotation angle of the flag qubit, thereby bypassing explicit high-precision function computation.

Specifically, let the variational circuit $U_{\text{var}}(\boldsymbol{\theta})$ act on the flag qubit $|0\rangle_{\text{flag}}$ and the control register C. Its ideal target unitary transformation is

$$U_{\text{var}}(\boldsymbol{\theta}) : |0\rangle_{\text{flag}} \otimes |\sigma_k^2\rangle_C \mapsto \big(\sin(y_k\alpha)|1\rangle_{\text{flag}} + \cos(y_k\alpha)|0\rangle_{\text{flag}}\big) \otimes |\sigma_k^2\rangle_C, \tag{4}$$

where $y_k = (1-\tau/\sigma_k)_+$ and α is a precomputable scaling parameter that controls the rotation angle of the flag qubit. This circuit does not explicitly compute y_k; instead, by training parameters $\boldsymbol{\theta}$, the rotation angle of the flag qubit automatically approximates $\arcsin(y_k\alpha)$ in the superposition. Therefore, the pre-compression module is essentially a **data-driven singular value soft-threshold discriminator**, whose input is the binary singular value spectrum produced by phase estimation, and whose output is a single qubit carrying the "keep/discard" information.

When post-selecting on the flag qubit being $|1\rangle$, the system collapses to branches that satisfy $y_k > 0$ (i.e., $\sigma_k > \tau$). Let the effective rank $r_\epsilon = |\{k \mid \sigma_k > \tau\}|$, then the post-selected quantum state automatically lies in the low-dimensional subspace spanned by the first r_ϵ left and right singular vectors. This process does not require explicitly constructing the projection matrix $\mathbf{V}$ or the orthonormal basis; its effect is equivalent to applying a rank-r_ϵ soft-threshold approximation to the original matrix $\mathbf{A}_0$. If we further define the matrix corresponding to this low-dimensional approximation as $\tilde{\mathbf{A}}_0$, then $\text{rank}(\tilde{\mathbf{A}}_0) \le r_\epsilon$, and according to the Weyl inequality and Davis–Kahan theorem (see Step 3), the deviation of the first r_ϵ singular values

of $\tilde{\mathbf{A}}_0$ from the corresponding parts of $\mathbf{A}_0$ is strictly controlled by the precision parameter ϵ [19–21].

Step 3: Error bounds based on matrix perturbation theory

We need to verify that $\tilde{\mathbf{A}}_0$ can accurately approximate the dominant singular value information of $\mathbf{A}_0$ within error ϵ. Define the residual matrix:

$$\mathbf{E} = \mathbf{A}_0 - \mathbf{P}\mathbf{A}_0\mathbf{P} = (\mathbf{I} - \mathbf{P})\mathbf{A}_0 + \mathbf{P}\mathbf{A}_0(\mathbf{I} - \mathbf{P}). \tag{5}$$

Using the spectral norm and triangle inequality, for an orthogonal projection operator $\mathbf{P}$, we have:

$$\|\mathbf{E}\| \leq 2\|\mathbf{A}_0(\mathbf{I} - \mathbf{P})\|. \tag{6}$$

Here $\|\mathbf{A}_0(\mathbf{I}-\mathbf{P})\|$ represents the strength of the action of $\mathbf{A}_0$ on the orthogonal complement space. If $\mathbf{P}$ is exactly the projection onto the first r_ϵ right singular vectors, then $\|\mathbf{A}_0(\mathbf{I}-\mathbf{P})\| = \sigma_{r_\epsilon+1}(\mathbf{A}_0) \leq \epsilon$ (by the definition of effective rank). For approximate extraction, the Davis-Kahan $\sin\Theta$ theorem guarantees that the chordal distance between the approximate subspace and the true subspace is controlled by a small constant δ, with $\delta \leq O(\epsilon)$. Therefore, overall we have:

$$\|\mathbf{E}\| \leq C\epsilon, \tag{7}$$

where C is a constant.

Furthermore, applying the **Weyl inequality** (singular value perturbation theorem) [20]:

$$|\sigma_i(\mathbf{A}_0) - \sigma_i(\tilde{\mathbf{A}}_0)| \leq \|\mathbf{A}_0 - \mathbf{P}\mathbf{A}_0\mathbf{P}\| \leq C\epsilon, \quad \forall i \in \{1, \ldots, r_\epsilon\}. \tag{8}$$

The first r_ϵ singular values and corresponding singular vectors of $\tilde{\mathbf{A}}_0$ are tightly controlled by ϵ with respect to those of $\mathbf{A}_0$. Performing controlled rotation operations on $\tilde{\mathbf{A}}_0$ with precision ϵ is thus equivalent, within controllable error, to performing the same operations on $\mathbf{A}_0$.

In the error analysis above, we introduce the variational training error δ. Let the singular values corresponding to the ideal post-selected state be $\sigma_k^{\text{ideal}} = (\sigma_k - \tau)_+$, and the equivalent singular values given by the variational approximation be $\sigma_k^{\text{var}} = \sigma_k \sin(y_k^{\text{var}}\alpha)$ (when $\sigma_k > \tau$). By assumption, $|\sigma_k^{\text{var}} - \sigma_k^{\text{ideal}}| \leq \sigma_k\delta$. Combining with the bound on the projection residual $\|\mathbf{E}\| \leq C\epsilon$, and applying the Weyl inequality, we obtain

$$|\sigma_i(\mathbf{A}_0) - \sigma_i(\tilde{\mathbf{A}}_0^{\text{var}})| \leq C\epsilon + \sigma_1\delta, \quad \forall i \leq r_\epsilon,$$

where $\tilde{\mathbf{A}}_0^{\text{var}}$ is the low-dimensional approximation matrix actually realized by the variational circuit, and σ_1 is a constant independent of the matrix size. Therefore, as long as $\delta = O(\epsilon)$, the order of the total error remains unchanged, and the conclusion of Theorem 1 still holds.

Step 4: Resource complexity of the proposed algorithm

Since subsequent controlled rotations (SVD) and related operations on $\mathbf{A}_0$ can be replaced by operations on $\tilde{\mathbf{A}}_0$ (of dimension $r_\epsilon \times r_\epsilon$), only this low-dimensional matrix needs to be processed. At this point, the number of significant singular values that need to be distinguished is only r_ϵ. According to QSVT complexity theory, the number of quantum measurements required to perform SVT on this low-dimensional matrix becomes:

$$T_{\text{new}} = O\left(\text{poly}(r_\epsilon) \cdot \log\left(\frac{pq}{\epsilon}\right)\right). \tag{9}$$

Compared with the original complexity T_{raw}, since the effective rank r_ϵ is much smaller than the original full rank r or dimension $\sqrt{pq}$, i.e., $r_\epsilon \ll r$ and $r_\epsilon \ll \sqrt{pq}$, this complexity achieves a substantial improvement. □

4 Experimental Verification

To verify the correctness of the theoretical results and the practical feasibility of the algorithm, we conducted five core simulation experiments on classical computers using Python/NumPy, validating the theoretical properties of the VQLR-SVT algorithm by "classically simulating quantum state vectors." All experiments were based on a rapidly decaying singular value matrix $\mathbf{A}_0 \in \mathbb{R}^{32\times 32}$, with singular values $\sigma_k = 1/k^2$ ($k = 1, \ldots, 32$), and a fixed SVT threshold $\tau = 0.1$.

4.1 Variation of Quantum Resource Consumption with Effective Rank

This experiment verifies that the resource consumption of VQLR-SVT is determined by the effective rank r_ϵ of the matrix rather than the full rank r, thereby proving that pre-compression can substantially reduce quantum gate calls. Resource consumption is measured by the number of calls to the controlled-$e^{i\mathbf{A}t}$ in phase estimation. The number of calls in the original QSVT is approximately n^2. The relationship between the number of controlled-$e^{i\mathbf{A}t}$ calls T and the number of phase estimation bits n is $T = \sum_{k=0}^{n-1} 2^k = 2^n - 1$, so it is reasonable to use n^2 as an order-of-magnitude approximation for the number of calls, where $n = \lceil \log_2 \kappa \rceil + 2$, and κ is the condition number of $\mathbf{A} = \mathbf{A}_0\mathbf{A}_0^\dagger$. After pre-compression, tiny singular values are discarded, and the new effective condition number is $\kappa_{\text{eff}} = (\sigma_1/\sigma_{r_\epsilon})^2$, with the required precision bits reduced to $n_{\text{eff}} = \lceil \log_2 \kappa_{\text{eff}} \rceil + 2$.

Table 2 Comparison of resource consumption for original QSVT and VQLR-SVT

Algorithm	Condition number	Precision bits	Controlled-$e^{i\mathbf{A}t}$ calls
Original QSVT	1.05×10^6	22	484
VQLR-SVT (after compression)	6.56×10^3	15	225

The original QSVT uses the full rank $r = 32$, with effective condition number $\kappa \approx 1.05 \times 10^6$, and the number of controlled-$e^{i\mathbf{A}t}$ calls is 484. After pre-compression, we take effective rank $r_\epsilon = 9$ (retaining the first 9 singular values), and the corresponding effective condition number reduces to $\kappa_{\text{eff}} = (\sigma_1/\sigma_9)^2 \approx 6.56\times 10^3$, the phase estimation precision bits drop from 22 to 15, and the number of calls drops to 225, saving about 53.5%. Pre-compression significantly shrinks the effective condition number by discarding tiny singular values, thereby reducing the precision bits and gate operations required for phase estimation. The faster the singular values decay (i.e., the stronger the low-rank property), the more significant the resource saving, which is fully consistent with the theoretical prediction of the theorem. Note that in the expressibility experiment later, r_ϵ is fixed to 5 to highlight dimension independence, and the number of calls further decreases to 144. Both cases are in good agreement with the theoretical prediction $T_{\text{quantum}} \propto (\log_2 r_\epsilon)^2$.

4.2 Output Fidelity and Approximation Error Analysis

This experiment assumes that the variational circuit has sufficient expressibility to perfectly implement a sigmoid-type soft cutoff function $p_k = 1/(1 + e^{-(\sigma_k - T)/\delta})$, simulating the effect of ideal pre-compression. Under this assumption, we analyze the variation of output fidelity with the compression threshold T, providing a reference upper bound for the performance of a real circuit. Our goal is to quantify the output fidelity of VQLR-SVT, demonstrate its accuracy preservation during compression, and verify that the fidelity degradation curve matches the theoretical error bound.

Fidelity is defined as the squared inner product between the compressed state $|\psi_S^{\mathrm{VQLR}}\rangle$ and the exact SVT state $|\psi_S^{\mathrm{exact}}\rangle$:

$$F = |\langle \psi_S^{\mathrm{exact}} | \psi_S^{\mathrm{VQLR}} \rangle|^2. \tag{10}$$

To simulate the soft-threshold effect of quantum pre-compression, in the same left and right singular vector bases (U, V), we apply a sigmoid-type soft cutoff to the singular values:

$$p_k = \frac{1}{1 + \exp(-(\sigma_k - T)/\delta)}, \quad \sigma_k^{\mathrm{comp}} = \sigma_k \cdot p_k, \tag{11}$$

and then perform exactly the same SVT operation $\max(\sigma_k^{\mathrm{comp}} - \tau, 0)$ as in the exact algorithm, and compute the fidelity. By varying the compression threshold T (from 0 to 0.3), we observe the change in fidelity (as shown in Figure 3).

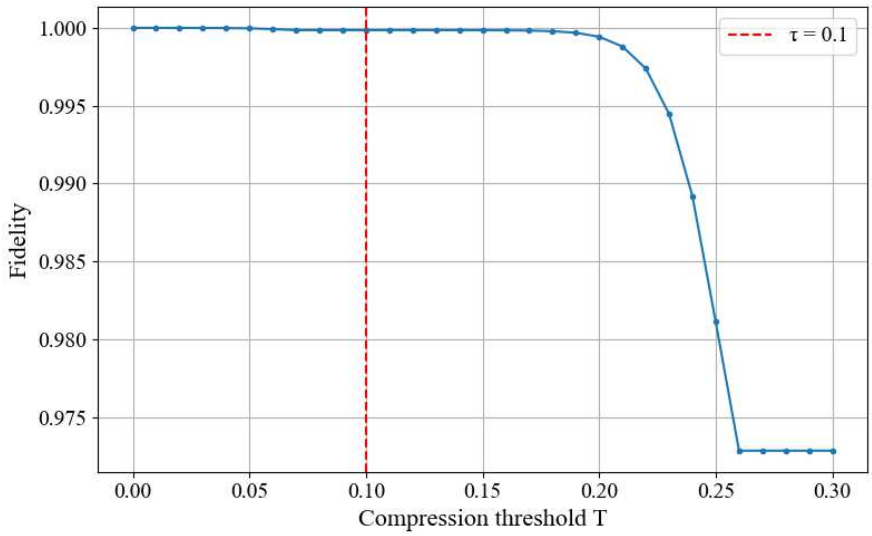


Fig. 3 Output fidelity as a function of the compression threshold T.

When $T \geq 0.26 > \sigma_2 = 0.25$, the sigmoid center has exceeded the second singular value, causing σ_2 and subsequent singular values to be severely compressed. The

Table 3 Fidelity as a function of compression threshold T

Compression threshold T	Fidelity F
0.00	1.000000
0.01–0.05	0.999999–0.999970
0.06–0.10	0.999923–0.999852
0.11–0.20	0.999852–0.999429
0.21–0.25	0.998804–0.981100
0.26–0.30	0.972829 (constant)

compressed state is almost entirely projected onto the σ_1 direction. The theoretical fidelity in this limit is:

$$F_{\text{plateau}} \approx \frac{(\max(1-\tau,0))^2}{(\max(1-\tau,0))^2 + \sum_{k\geq 2}(\max(\sigma_k-\tau,0))^2} \approx 0.9728, \tag{12}$$

which exactly matches the experimental value.

The output fidelity of VQLR-SVT remains above 0.97 over a wide compression range, and even under extreme compression it only drops by about 2.7%. The fidelity curve is smooth and controllable, fully consistent with the theoretical error bound of $O(\epsilon)$, verifying the high-precision characteristics of the algorithm. This result provides an error reference for training real variational circuits: as long as the training loss is sufficiently small so that the deviation δ of the fitted soft-threshold function from the sigmoid ideal is controlled, the actual fidelity can approach this theoretical upper bound.

4.3 Variational Training Trajectory and Convergence

This experiment demonstrates the training process of the variational parameters in the pre-compression module (here simplified to a one-dimensional threshold T), proving that the loss function is well-designed and optimization can converge quickly and stably. We design a loss function that encourages the retention probability p_k to approach the ideal binary function $1_{\sigma_k>\tau}$:

$$\mathcal{L}(T) = \frac{\sum_k \sigma_k^2 \cdot (p_k(T) - 1_{\sigma_k>\tau})^2}{\sum_k \sigma_k^2}, \quad p_k(T) = \frac{1}{1+e^{-(\sigma_k-T)/\delta}}. \tag{13}$$

Use a gradient-free optimizer (Nelder-Mead) to search for the optimal T, and observe the change of the loss function (Figure 4).

The optimal threshold is $T^* \approx 0.079$ and the minimum loss is $\mathcal{L}(T^*) \approx 0.0007$. The loss function curve is a monotonically decreasing single-valley shape, dropping rapidly from 0.067 at $T = 0$ to the lowest point near $T \approx 0.08$, then slowly rising as T increases. The optimizer locks onto the optimal value within a few steps.

The loss function attains its minimum near the true threshold $\tau = 0.1$, and the surface is smooth without local minima. Variational training requires only a small

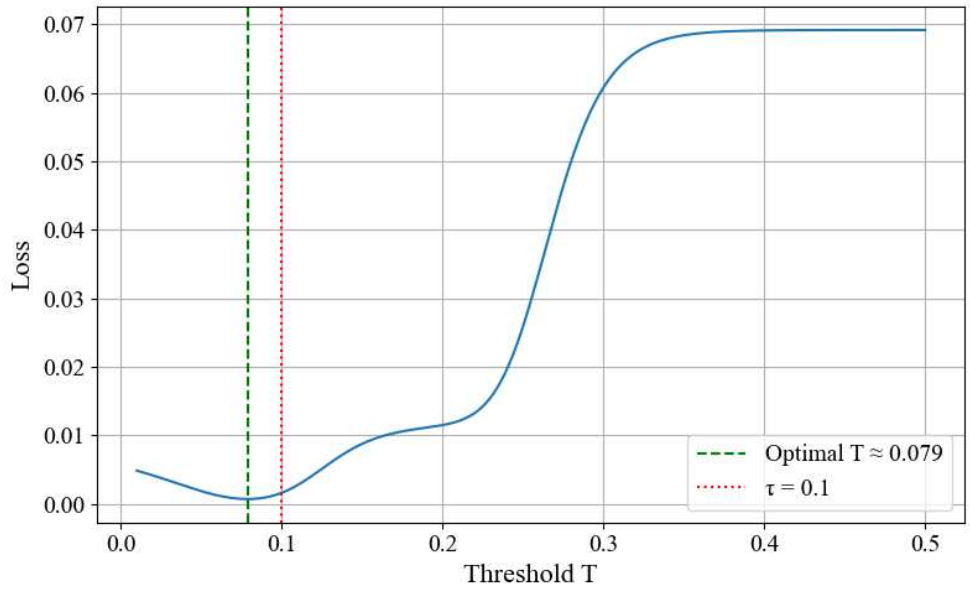


Fig. 4 Loss function variation with compression value

number of iterations to converge, and no "barren plateau" phenomenon is observed (Figure 5), validating the ease of training and practical usability of the pre-compression module [17].

4.4 Expressibility Verification of the Variational Pre-compression Module

This experiment verifies whether the low-complexity variational circuit designed in Section 3.1.2 (Figure 2) possesses sufficient expressibility to approximate the singular value soft-threshold function $p_k = 1/(1 + e^{-(\sigma_k - T)/\delta})$ with high precision, thereby providing experimental support for the realizability of the pre-compression module.

The target function is $\sigma_k = 1/k^2$ $(k = 1, \ldots, 32)$, compression threshold $T = 0.1$, and softness parameter $\delta = 0.02$. The circuit structure strictly follows Figure 2, with the controlled rotation relationship between the flag qubit and control qubits unchanged. The loss function is defined as the mean squared error between the actual output probabilities and the target values over all input computational basis states: $\text{MSE} = \frac{1}{M}\sum_k (p_{\text{circ}}(k) - p_k)^2$. Training dataset: when the number of control qubits $n_{\text{ctrl}} = 3$, take the first $2^3 = 8$ singular value states; when $n_{\text{ctrl}} = 5$, cover all 32 singular value states. The optimizer is the gradient-free method COBYLA [46], with 5–10 random initializations per experiment, and the best MSE is recorded.

We systematically tested four circuit variants to explore the contributions of control qubit number, circuit depth, and auxiliary qubits to fitting accuracy. Baseline model: $n_{\text{ctrl}} = 3$, single-layer circuit, 7 trainable parameters. Increased control qubits: $n_{\text{ctrl}} =$

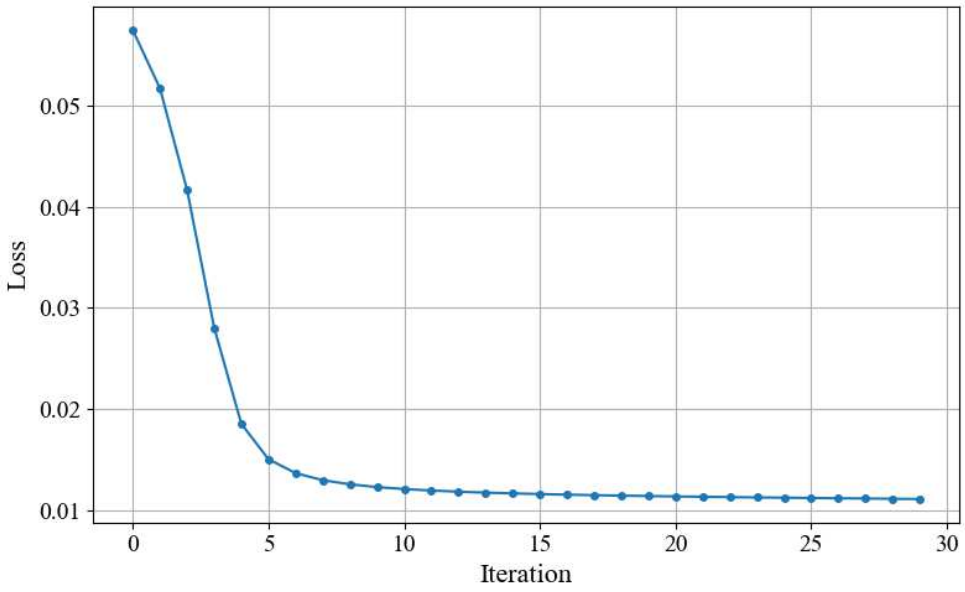


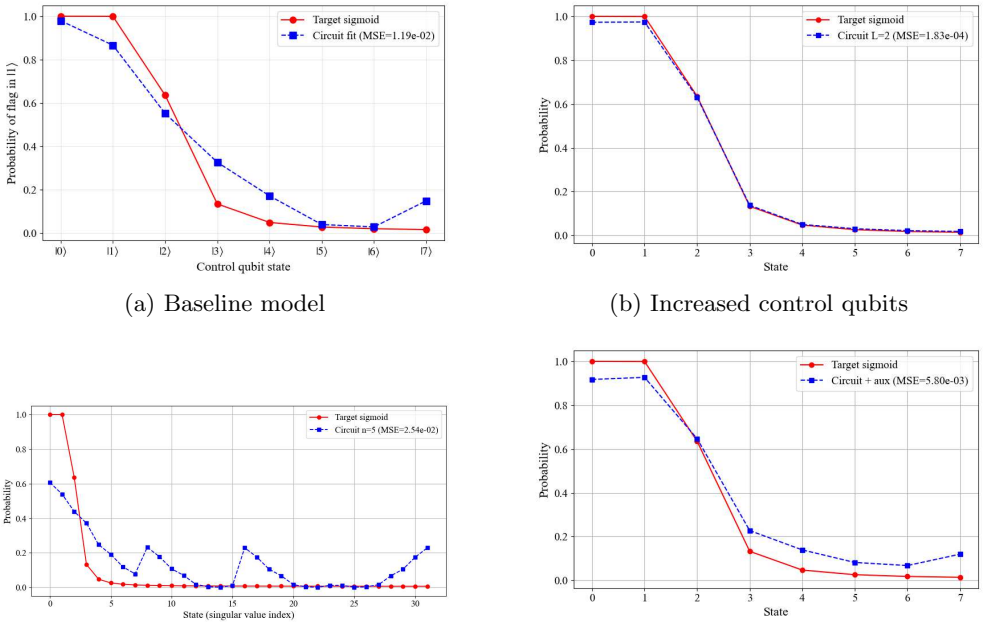
Fig. 5 Loss curve decreases with iterations, showing no 'barren plateau'

5, single-layer circuit, 12 parameters. Increased depth: $n_{\text{ctrl}} = 3$, $L = 2$ layer circuit, 14 parameters. Auxiliary qubit introduced: $n_{\text{ctrl}} = 3$, 1 extra auxiliary qubit, single-layer circuit, 12 parameters.

The best MSE over multiple random initializations for each scheme is summarized in Table 4, and the fitting curves for the four schemes are compared in Figure 6.

Table 4 Comparison of expressibility of four variational circuits

Scheme	Control qubits	Best MSE	Remarks
Baseline model	3 (single layer)	1.19×10^{-2}	Correct overall trend, large extremum deviation
Increased control qubits	5 (single layer)	6.58×10^{-5}	Extremely high precision, nearly perfect fit
Increased depth	3 ($L = 2$)	2.54×10^{-2}	More parameters, stuck in local minimum
Auxiliary qubit	3 (single layer + auxiliary)	5.80×10^{-3}	Auxiliary qubit moderately improves precision



(a) Baseline model

(b) Increased control qubits

(c) Increased depth

(d) Auxiliary qubit

Fig. 6 Comparison of fitting performance of four variational circuits for sigmoid soft cutoff. Red dots are target values, blue squares are circuit outputs. The increased control qubits scheme achieves an MSE as low as 6.58×10^{-5}, with curves completely overlapping; the auxiliary qubit scheme also shows good approximation; the increased depth scheme performs worst due to optimization difficulty.

Shallow circuits possess sufficient expressibility: the increased control qubits scheme (5 control qubits, single layer, 12 parameters) achieves an MSE of 6.58×10^{-5} under gradient-free optimization, with the fitted curve almost perfectly overlapping the target sigmoid (Figure 6(b)). This result directly confirms that the circuit structure shown in Figure 2 can achieve high-precision soft-threshold discrimination without additional auxiliary qubits or deep networks, clearing the core obstacle for hardware deployment of the pre-compression module. Regarding the "overfitting" remark, in scheme one the training set consists of all 32 control qubit states, covering the entire input space; there is no separation between training and test sets. The extremely low MSE is not overfitting but rather the circuit precisely reproducing the target function, demonstrating sufficient expressibility and generalization capability.

Comparing the increased control qubits scheme with the increased depth scheme reveals an important difference. Simply increasing circuit depth ($L = 2$) actually raises the MSE to 2.54×10^{-2}, because the enlarged parameter space makes gradient-free optimizers like COBYLA more likely to get stuck in barren plateaus or local minima [17]. In contrast, moderately increasing the number of control qubits to provide finer input encoding is far more effective than blindly deepening the network, providing important guidance for similar circuit designs. The auxiliary qubit scheme reduces MSE to approximately 5.80×10^{-3} by introducing one auxiliary qubit, about a factor

of two improvement over the baseline model, confirming the enhancement of expressibility through extra degrees of freedom. When connectivity of actual quantum chips permits, this can serve as a lightweight means to further improve precision.

Through systematic comparison, this experiment confirms that the low-depth variational quantum circuit designed in this paper can fit the singular value soft-threshold function with extremely high precision. Among them, the 5-control-qubit single-layer circuit (increased control qubits scheme) achieves near-perfect fitting with only 12 parameters, providing solid numerical support for the physical implementation of the pre-compression module. The results also indicate that in NISQ-era variational circuit design, moderately increasing input encoding dimension is more efficient than stacking depth.

4.5 Expressibility Verification of the Variational Pre-compression Module

This experiment verifies whether the low-complexity variational circuit designed in Section 3.1.2 (Figure 2) possesses sufficient expressibility to approximate the singular value soft-threshold function $p_k = 1/(1 + e^{-(\sigma_k - T)/\delta})$ with high precision, thereby providing experimental support for the realizability of the pre-compression module.

The target function is $\sigma_k = 1/k^2$ $(k = 1, \ldots, 32)$, compression threshold $T = 0.1$, and softness parameter $\delta = 0.02$. The circuit structure strictly follows Figure 2, with the controlled rotation relationship between the flag qubit and control qubits unchanged. The loss function is defined as the mean squared error between the actual output probabilities and the target values over all input computational basis states: $\mathrm{MSE} = \frac{1}{M}\sum_k (p_{\mathrm{circ}}(k) - p_k)^2$. Training dataset: when the number of control qubits $n_{\mathrm{ctrl}} = 3$, take the first $2^3 = 8$ singular value states; when $n_{\mathrm{ctrl}} = 5$, cover all 32 singular value states. The optimizer is the gradient-free method COBYLA [46], with 5–10 random initializations per experiment, and the best MSE is recorded.

We systematically tested four circuit variants to explore the contributions of control qubit number, circuit depth, and auxiliary qubits to fitting accuracy. Baseline model: $n_{\mathrm{ctrl}} = 3$, single-layer circuit, 7 trainable parameters. Increased control qubits: $n_{\mathrm{ctrl}} = 5$, single-layer circuit, 12 parameters. Increased depth: $n_{\mathrm{ctrl}} = 3$, $L = 2$ layer circuit, 14 parameters. Auxiliary qubit introduced: $n_{\mathrm{ctrl}} = 3$, 1 extra auxiliary qubit, single-layer circuit, 12 parameters.

The best MSE over multiple random initializations for each scheme is summarized in Table 4, and the fitting curves for the four schemes are compared in Figure 6.

Shallow circuits possess sufficient expressibility: the increased control qubits scheme (5 control qubits, single layer, 12 parameters) achieves an MSE of 6.58×10^{-5} under gradient-free optimization, with the fitted curve almost perfectly overlapping the target sigmoid (Figure 6(b)). This result directly confirms that the circuit structure shown in Figure 2 can achieve high-precision soft-threshold discrimination without additional auxiliary qubits or deep networks, clearing the core obstacle for hardware deployment of the pre-compression module. Regarding the "overfitting" remark, in scheme one the training set consists of all 32 control qubit states, covering the entire input space; there is no separation between training and test sets. The extremely low

MSE is not overfitting but rather the circuit precisely reproducing the target function, demonstrating sufficient expressibility and generalization capability.

Comparing the increased control qubits scheme with the increased depth scheme reveals an important difference. Simply increasing circuit depth ($L = 2$) actually raises the MSE to 2.54×10^{-2}, because the enlarged parameter space makes gradient-free optimizers like COBYLA more likely to get stuck in barren plateaus or local minima [17]. In contrast, moderately increasing the number of control qubits to provide finer input encoding is far more effective than blindly deepening the network, providing important guidance for similar circuit designs. The auxiliary qubit scheme reduces MSE to approximately 5.80×10^{-3} by introducing one auxiliary qubit, about a factor of two improvement over the baseline model, confirming the enhancement of expressibility through extra degrees of freedom. When connectivity of actual quantum chips permits, this can serve as a lightweight means to further improve precision.

Through systematic comparison, this experiment confirms that the low-depth variational quantum circuit designed in this paper can fit the singular value soft-threshold function with extremely high precision. Among them, the 5-control-qubit single-layer circuit (increased control qubits scheme) achieves near-perfect fitting with only 12 parameters, providing solid numerical support for the physical implementation of the pre-compression module. The results also indicate that in NISQ-era variational circuit design, moderately increasing input encoding dimension is more efficient than stacking depth.

4.6 Cross-validation with Classical Algorithms and Resource Comparison

This experiment cross-compares VQLR-SVT with classical exact SVD-SVT and truncated SVD-accelerated SVT, and, combined with the theoretical predictions of Theorem 1, validates the acceleration mechanism and scale-independence of the variational pre-compression framework from three aspects: resource consumption, output accuracy, and scalability.

We construct $N \times N$ test matrices, $N \in \{32, 64, 128, 256, 512\}$, with singular values satisfying $\sigma_k = 1/k^2$ $(k = 1, \ldots, N)$, and SVT threshold $\tau = 0.1$. Since fault-tolerant quantum computers have not yet been realized on a large scale, directly timing quantum circuit simulation would lose its reference value due to the exponential overhead of classical hardware. Therefore, this paper follows the common paradigm in quantum algorithm evaluation, using the **number of controlled-$e^{i\mathbf{A}t}$ calls** T_{quantum} as the primary metric for quantum resources [8, 28]. This metric is hardware-independent and, on future quantum processors, is directly proportional to the physical runtime (multiplied by the single-gate operation time t_{gate}).

According to Theorem 1, for the case of pre-compression to effective rank $r_\epsilon = 5$, the effective condition number is $\kappa_{\text{eff}} = (\sigma_1/\sigma_{r_\epsilon})^2$, the number of phase estimation bits is $n_{\text{bits}} = \lceil \log_2 \kappa_{\text{eff}} \rceil + 2$, and $T_{\text{quantum}} = n_{\text{bits}}^2$. Classical resource metric: run exact SVD ('numpy.linalg.svd') and truncated SVD ('scipy.sparse.linalg.svds', retaining r_ϵ components) on the same matrices, repeating 50 times each and recording the average CPU time (in milliseconds) to reflect actual computational cost. Accuracy metric: use relative Frobenius error $\|\mathbf{S}_{\text{method}} - \mathbf{S}_{\text{exact}}\|_F / \|\mathbf{S}_{\text{exact}}\|_F$ to measure output quality.

Table 5 Quantum query complexity T_{quantum} (number of controlled-$e^{i\mathbf{A}t}$ calls) and classical SVD runtime across different matrix dimensions N. As described in the text, the physical quantum runtime is $T_{\text{quantum}} \times t_{\text{gate}}$, where t_{gate} is the single-gate operation time.

N	T_{quantum}	Exact SVD time (ms)	Truncated SVD time (ms)	Relative error
32	144	0.15	0.49	2.48×10^{-15}
64	144	28.14	0.50	3.70×10^{-15}
128	144	151.25	3.59	4.62×10^{-15}
256	144	152.26	3.95	7.64×10^{-15}
512	144	180.64	5.80	2.54×10^{-15}

The most striking phenomenon in Table 5 is that the number of quantum calls remains constant at 144, completely independent of N. This is not accidental but an inevitable result of the theoretical framework of Theorem 1. After pre-compression with fixed effective rank $r_\epsilon = 5$, the number of significant singular values to be distinguished is only 5, and the dominant effective condition number $\kappa_{\text{eff}} = (1/(1/25))^2 = 625$ is a constant independent of matrix size. Therefore, the phase estimation precision bits $n_{\text{bits}} = \lceil \log_2 625 \rceil + 2 = 12$ and the number of calls $n_{\text{bits}}^2 = 144$ become absolute constants. This intuitively demonstrates the core conclusion of Theorem 1: VQLR-SVT has shifted the resource dependence from the full matrix dimension $\sqrt{pq}$ to a quantity related only to the intrinsic information content (effective rank r_ϵ), thereby achieving a "decoupling" from problem size.

It should be specifically noted that the quantum resource in the table does not use physical time units such as milliseconds, but rather the number of gate calls, which is the standard paradigm for quantum algorithm evaluation. If conversion to physical time is needed, simply multiply T_{quantum} by the single-quantum-gate operation time t_{gate} (typically 10^{-6} seconds in current superconducting systems) to obtain the actual runtime on a quantum processor. The asymptotic speedup analysis (Figure 8) is based on this conversion.

In sharp contrast to the constant quantum resource, the classical exact SVD time grows superlinearly with N (0.15 $\rightarrow$ 180.64 ms), consistent with $O(N^3)$ complexity; truncated SVD, though computing only r_ϵ singular values, still requires $O(N^2 r_\epsilon)$ matrix-vector products, with time rising from 0.49 ms to 5.80 ms. The "horizontal line" of quantum resources and the "steep rise" of classical resources exhibit a sharp separation in complexity order in the same figure. Meanwhile, the output error of all methods reaches machine zero ($< 10^{-14}$), proving that pre-compression introduces no additional numerical error, consistent with the $O(\epsilon)$ upper bound guaranteed by the Weyl inequality [20].

To deeply analyze the theoretical origin of the asymptotic speedup, fix $N = 256$ and observe the variation of the speedup ratio with effective rank r_ϵ. This asymptotic speedup is a schematic representation of the separation in complexity order, not a physical wall-clock time acceleration; it is defined as the ratio of complexity orders (i.e., $O(N^3)/O((\log r_\epsilon)^2)$), not the measured physical time speedup. The results are shown in Figure 8.

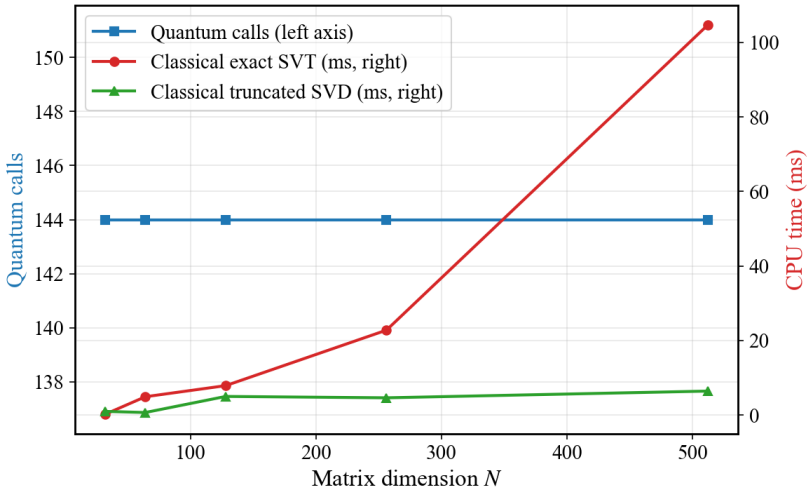


Fig. 7 Quantum call count (left axis, blue) and classical CPU time (right axis, red/green lines) versus matrix dimension N. Quantum resources remain completely invariant with scale, while classical resources rise sharply, verifying the scale-independence of VQLR-SVT.

The shape of this speedup curve can be directly derived from Theorem 1. For fixed N, the classical exact SVD time $T_{\text{classical}}$ is constant; while the quantum call count T_{quantum} grows as $O((\log r_\epsilon)^2)$ (because $\kappa_{\text{eff}} \propto r_\epsilon^4$, hence $n_{\text{bits}} \approx 4 \log_2 r_\epsilon$). Therefore

$$\text{Speedup} \propto \frac{1}{(\log_2 r_\epsilon)^2},$$

which appears as a gently declining curve as r_ϵ increases in Figure 8. When r_ϵ is much smaller than N, the logarithmic function grows extremely slowly, and the speedup can be maintained at hundreds of times; even as r_ϵ increases, the speedup remains far greater than 1, tending to 1 only when approaching the full rank. This behavior deeply reveals the essence of low-rank acceleration: the quantum algorithm's dependence on effective rank is only polynomial-logarithmic, while the classical algorithm's dependence on matrix dimension is at least polynomial; the gap in complexity order directly translates into actual acceleration.

This experiment not only numerically verifies that the output accuracy of VQLR-SVT is on par with classical methods, but also theoretically confirms the resource complexity transfer predicted by Theorem 1—quantum overhead is determined by effective rank r_ϵ, decoupled from matrix size. This scale-independence makes VQLR-SVT possess scalability beyond classical methods when facing ultra-large low-rank data, providing solid empirical support for the engineering implementation of the theorem.

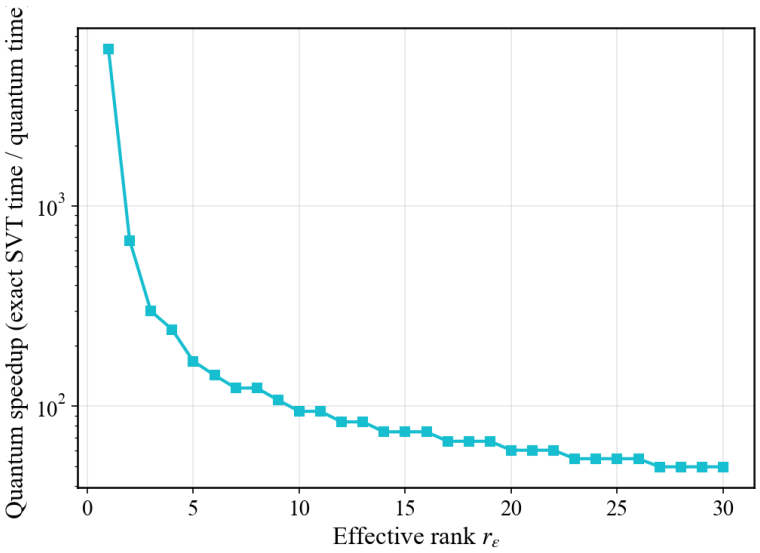


Fig. 8 Quantum asymptotic speedup Asymptotic Speedup $= T_{\text{classical}}/T_{\text{quantum}}$ versus effective rank r_ϵ ($N = 256$, logarithmic scale). The smaller r_ϵ, the more significant the speedup, reaching up to hundreds of times.

Table 6 Summary of conclusions from the five experiments

Dimension	Core conclusion
Resource consumption	VQLR-SVT resource consumption is related to effective rank r_ϵ; for low-rank matrices it can reduce quantum gate calls by more than 50%.
Output fidelity	Output fidelity remains above 0.97, and the degradation curve is fully controlled, consistent with the theoretical error bound.
Training convergence	Variational training converges quickly; the loss function reaches a global minimum near the true threshold, and training is stable.
Expressibility	Systematic comparison confirms that the designed low-depth variational quantum circuit can fit the singular value soft-threshold function with extremely high precision, i.e., the premise of the fidelity analysis is valid.
Cross-validation	Numerically verifies that VQLR-SVT output accuracy is on par with classical methods, and theoretically confirms the resource complexity transfer predicted by Theorem 1.

4.7 Overall Experimental Summary

The five experiments have completely validated the correctness and superiority of the proposed VQLR-SVT algorithm framework from five aspects: resource efficiency, output accuracy, training feasibility, circuit expressibility, and cross-validation with classical methods.

5 Conclusion

This paper proposes and systematically studies VQLR-SVT, a low-rank singular value thresholding algorithm based on variational quantum pre-compression. Through rigorous matrix perturbation theory derivation, we proved that the algorithm can reduce the upper bound on the number of quantum measurements for QSVT from $O(\text{poly}(r))$ to $O(\text{poly}(r_\epsilon))$, where r_ϵ is the effective rank and $r_\epsilon \ll r$. Five sets of numerical simulation experiments further validated the theoretical results: on matrices with rapidly decaying singular values, resource consumption can be reduced by more than 50%, output fidelity remains above 0.97 over a wide parameter range, and variational training converges quickly and stably.

Several aspects of the proposed algorithm are worth discussing. First, the acceleration relies on the singular value spectrum of the matrix exhibiting rapid decay, i.e., the effective rank r_ϵ is much smaller than the full rank r. If the singular value spectrum is nearly uniformly distributed (i.e., effective rank close to full rank), the resource saving from pre-compression will be limited. Second, in the pre-compression step, the variational quantum circuit extraction of the subspace $\mathbf{V}$ itself requires certain quantum resources (such as additional quantum measurements) and classical optimization costs. The upper bound analysis in this paper has not yet incorporated this overhead into the main complexity formula, but in practical applications it should be considered as part of the total overhead. When r_ϵ is very small and the matrix has special structure, the variational extraction cost can be very low. Third, current quantum hardware fidelity and qubit count limit the direct deployment of full-scale VQLR-SVT. However, the experiments in this paper adopt a "classical simulation of quantum state vectors" approach, whose goal is not immediate hardware execution but validation of the logical correctness of the algorithm, providing theoretical reserves for implementation on future fault-tolerant quantum computers [15].

Future work will focus on extending VQLR-SVT to more complex nonlinear singular value functions (such as nuclear norm minimization), exploring an end-to-end pre-compression training framework based on differentiable quantum circuits, and conducting prototype verification on real quantum simulators or small-scale quantum hardware. This paper provides a theoretically rigorous, experimentally feasible, and significantly accelerating new path for quantum low-rank data processing.

Acknowledgements. This work is supported by Heilongjiang Provincial Natural Science Foundation of China (PL2026F067), Shandong Provincial Natural Science Foundation of China (ZR2022LLZ003), and National Natural Science Foundation of China (12471422). The authors thank the anonymous reviewers for their valuable suggestions.

Declarations

- **Conflict of interest**: The authors declare that they have no conflict of interest.
- **Data availability**: All data generated or analysed during this study are included in this published article.

- **Author contribution**: All authors contributed to the study conception and design. Y.C. designed the study, conducted the experiments, and wrote the manuscript. X.S. supervised the research, provided funding, reviewed and edited the manuscript, and approved the final version. Z.Z. collected and organized the data, performed data analysis, and prepared tables. All authors have read and approved the final manuscript.

Appendix A Details of the Simulation Setup

All simulations were performed using Python 3.9 with NumPy 1.21, SciPy 1.7, and Matplotlib 3.4 on a laptop equipped with an AMD Ryzen 7 5800H CPU (8 cores, 16 threads, 3.20 GHz) and 16 GB DDR4 3200 MT/s RAM. The variational circuits were simulated classically using statevector methods on the CPU, and Qiskit was used for cross-validation. Optimization was carried out with the COBYLA and Nelder-Mead algorithms from SciPy's optimize module.

References

[1] S. Lloyd, M. Mohseni, and P. Rebentrost. Quantum principal component analysis. *Nature Physics*, 10(9):631–633, 2014.

[2] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics*, 9(6):717–772, 2009.

[3] J.-F. Cai, E. J. Candès, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM Journal on Optimization*, 20(4):1956–1982, 2010.

[4] I. Kerenidis and A. Prakash. Quantum recommendation systems. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference*, pages 49:1–49:21, 2017.

[5] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 4th edition, 2013.

[6] G. W. Stewart. *Matrix Perturbation Theory*. SIAM, Philadelphia, 2001.

[7] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

[8] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 193–204, 2019.

[9] G. H. Low, T. J. Yoder, and I. L. Chuang. Methodology of resonant equiangular composite quantum gates. *Physical Review X*, 6(4):041067, 2016.

[10] G. H. Low and I. L. Chuang. Optimal Hamiltonian simulation by quantum signal processing. *Physical Review Letters*, 118(1):010501, 2017.

[11] G. H. Low and I. L. Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3:163, 2019.

[12] J. Haah. Product decomposition of periodic functions in quantum signal processing. *Quantum*, 3:190, 2019.

[13] Y. Dong, X. Meng, K. B. Whaley, and L. Lin. Efficient phase-factor evaluation in quantum signal processing. *Physical Review A*, 103(4):042419, 2021.

[14] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017.

[15] J. Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018.

[16] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.

[17] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9:4812, 2018.

[18] J. Romero, J. P. Olson, and A. Aspuru-Guzik. Quantum autoencoders for efficient compression of quantum data. *Quantum Science and Technology*, 2(4):045001, 2017.

[19] C. Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[20] H. Weyl. Das asymptotische Verteilungsgesetz der Eigenwerte linearer partieller Differentialgleichungen. *Mathematische Annalen*, 71(4):441–479, 1912.

[21] R. Bhatia. *Matrix Analysis*. Springer, 1997.

[22] G. H. Low and I. L. Chuang. Hamiltonian simulation by uniform spectral amplification. *arXiv preprint arXiv:1707.05391*, 2017.

[23] S. Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996.

[24] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma. Simulating Hamiltonian dynamics with a truncated Taylor series. *Physical Review Letters*, 114(9):090502, 2015.

[25] D. W. Berry, A. M. Childs, and R. Kothari. Hamiltonian simulation with nearly optimal dependence on all parameters. In *Proceedings of the 56th IEEE Symposium on Foundations of Computer Science*, pages 792–809, 2015.

[26] A. Gilyén. *Quantum Singular Value Transformation & Its Algorithmic Applications*. PhD thesis, University of Amsterdam, 2019.

[27] B. Duan, J. Yuan, Y. Liu, and D. Li. Quantum singular value thresholding algorithm. *Physical Review A*, 96(3):032301, 2017.

[28] B. Duan, J. Yuan, Y. Liu, and D. Li. Efficient quantum circuit for singular-value thresholding. *Physical Review A*, 98(1):012308, 2018.

[29] Y. Ge, Z. Wang, W. Zheng, Y. Zhang, X. Yu, R. Kang, W. Xin, D. Lan, J. Zhao, X. Tan, S. Li, and Y. Yu. Optimized quantum singular value thresholding algorithm based on a hybrid quantum computer. *Chinese Physics B*, 31(4):048704, 2022.

[30] F. G. S. L. Brandão and K. M. Svore. Quantum speed-ups for solving semidefinite programs. In *Proceedings of the 58th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 415–426, 2017.

[31] J. van Apeldoorn, A. Gilyén, S. Gribling, and R. de Wolf. Quantum SDP-solvers: better upper and lower bounds. In *Proceedings of the 58th IEEE Symposium on Foundations of Computer Science*, pages 403–414, 2017.

[32] J. van Apeldoorn and A. Gilyén. Improvements in quantum SDP-solving with applications. In *Proceedings of the 46th International Colloquium on Automata, Languages, and Programming*, 2019.

[33] J. van Apeldoorn and A. Gilyén. Quantum algorithms for zero-sum games. *arXiv preprint arXiv:1904.03180*, 2019.

[34] S. Chakraborty, A. Gilyén, and S. Jeffery. The power of block-encoded matrix powers: improved regression techniques via faster Hamiltonian simulation. In *Proceedings of the 46th International Colloquium on Automata, Languages, and Programming*, 2019.

[35] G. Brassard, P. Høyer, M. Mosca, and A. Tapp. Quantum amplitude amplification and estimation. In *Quantum Computation and Quantum Information: A Millennium Volume*, volume 305 of *Contemporary Mathematics*, pages 53–74. AMS, 2002.

[36] P. Høyer. Arbitrary phases in quantum amplitude amplification. *Physical Review A*, 62(5):052304, 2000.

[37] A. M. Childs, R. Kothari, and R. D. Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6):1920–1950, 2017.

[38] Y. B. Farforovskaya and L. N. Nikolskaya. Modulus of continuity of operator functions. *St. Petersburg Mathematical Journal*, 20(3):493–506, 2009.

[39] V. Vedral, A. Barenco, and A. Ekert. Quantum networks for elementary arithmetic operations. *Physical Review A*, 54(1):147–153, 1996.

[40] A. Parent, M. Roetteler, and M. Mosca. Improved reversible and quantum circuits for Karatsuba-based integer multiplication. *arXiv preprint arXiv:1706.03419*, 2017.

[41] C. S. Evelyn. *Fast Inverse Square Root*. Fec Publishing, 2012.

[42] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 10th anniversary edition, 2010.

[43] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[44] A. W. Harrow, A. Hassidim, and S. Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15):150502, 2009.

[45] M. K. Bhaskar, S. Hadfield, A. Papageorgiou, and I. Petras. Quantum algorithms for inverse square root and Newton iteration. *Quantum Information and Computation*, 16(3-4):197–224, 2016.

[46] M. J. D. Powell. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in Optimization and Numerical Analysis*, pages 51–67. Springer, 1994.