QCE'24 Quantum Resource Estimation Educational Challenge Submission - Matrix Inversion by QSVT

Walden Killick

July 17, 2024

Abstract

We produce physical resource estimates for solving systems of linear equations involving banded circulant matrices using the quantum singular value transformation algorithm.

1 Introduction

Since the breakthrough 2009 algorithm of Harrow, Hassidim, and Lloyd (HHL algorithm), it has been well-known that for sparse, well-conditioned matrices, quantum computers are capable of solving the system of linear equations (SLE) problem exponentially faster than the best possible classical algorithms [1]. Due to the complexities of practical implementation, the performance analyses of quantum algorithms for solving SLEs have been primarily restricted to the asymptotic regime. In this project, we aim to answer the question of what physical quantum resources are required to solve SLEs of sizes at which the problem becomes classically intractable.

On the other hand, subsequent improvements to the HHL algorithm have not received the same attention despite achieving super complexity and arguably greater conceptual simplicity. In this project, we consider the use of the quantum singular value transformation (QSVT) [2, 3] in solving systems of linear equations and investigate the physical resource requirements this entails.

All code used to generate the results in this report can be found at https://github.com/Walden-Killick/QCE24-QRE-Challenge.

2 Preliminary

2.1 Problem definition

Given a quantum state $|b\rangle$ and an efficient classical description of a sparse matrix A, the quantum linear system of equations (QSLE) problem is the problem of preparing the state $|x\rangle = A^{-1}|b\rangle$. Broadly speaking, the primary challenges in any QSLE algorithm are (1) accessing A in a quantum circuit, and (2) performing (approximately) the transformation $x \mapsto 1/x$ on A's eigenvalues. The HHL algorithm [1] achieves these goals using a sparse Hamiltonian simulation subroutine [4] and quantum phase estimation [5], respectively. By contrast, QSVT utilises the fundamentally different techniques of block encoding [2] and quantum signal processing [6] and achieves an exponentially improved dependence on the target precision as well as a lesser improvement with respect to the condtion number. We briefly review these techniques in the following subsection.

2.2 Matrix inversion by QSVT

To access A in a quantum circuit, QSVT relies on quantum oracles which efficiently encode both the non-zero structure and non-zero elements of A. These are defined rigorously as follows.

Definition 2.1 (Sparse access oracles [7]). *Let A be a $2^n \times 2^n$ s-sparse matrix. Let $c(j,l)$ be a function which returns the row index of the l-th non-zero matrix element in the j-th column of A. The sparse access oracles O_c and O_A for A are the unitary operators defined as*

$$O_c |l\rangle |j\rangle = |l\rangle |c(j,l)\rangle$$

and

$$O_A |0\rangle |l\rangle |j\rangle = \left(A_{c(j,l),j} |0\rangle + \sqrt{1 - \left|A_{c(j,l),j}\right|^2} |1\rangle \right) |l\rangle |j\rangle .$$

We should note that in this formalism, O_A also implicitly encodes the non-zero structure of A by calling $c(j,l)$; however, for certain sufficiently well-structured matrices such as those considered in this project, this is not prohibitive.

The following informal theorem states that one can construct a unitary matrix which directly contains (a scaled version of) A as a submatrix (block).

Theorem 2.2 (Block-encoding sparse matrices [2, 7]). *Let A, O_c, O_A be as in Definition 2.1. Using one call to each of O_c and O_A, we can construct a larger unitary matrix U_A for which the upper left $2^n \times 2^n$ block is equal to A/s.*

The final ingredient in matrix inversion by QSVT is then to transform the singular values of A/s as $x \mapsto 1/x$. QSVT cannot perform this transformation exactly, but can perform almost-arbitrary polynomial transformations [8] and thus approximate $1/x$ to arbitrary precision.

Theorem 2.3 (Matrix inversion by QSVT [2]). *Let A be a sparse matrix with condition number κ and let U_A block-encode A as in Theorem 2.2. Using $O(\kappa \log(\kappa/\varepsilon))$ calls to U_A and $U_A^\dagger$, we can construct a unitary which block-encodes A^{-1} to within accuracy ε.*

We refer the reader to [3] for a pedagogical introduction to block encodings and QSVT.

2.3 Banded circulant matrices

To construct polynomial-size oracles O_c and O_A, A must be well-structured in some way. For this project, we consider banded circulant matrices due to the known explicit and simple construction of their sparse access oracles [7].

Definition 2.4 (Circulant matrix). *A circulant matrix is a matrix in which each row is equal to the previous row but with each element shifted one place to the right.*

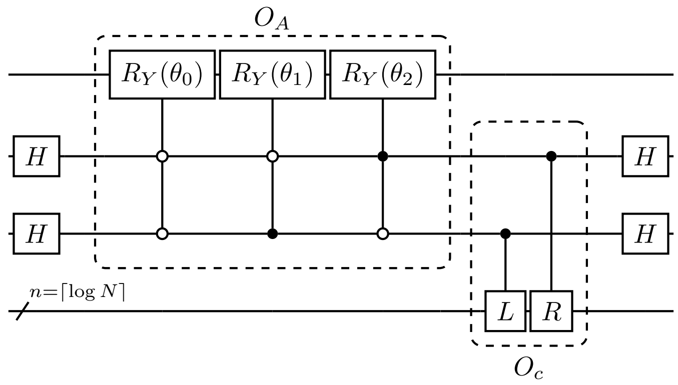

Figure 1: Quantum circuit for block-encoding an $N \times N$ banded circulant matrix, adapted from [7]. For $n = \lceil \log N \rceil$ qubits, the L and R gates each consist of n multi-controlled NOT gates with successively decreasing numbers of controls.

Definition 2.5 (Banded circulant matrix). *A circulant matrix is banded if only the first three entries of the second row are non-zero.*

The following is an example of a banded circulant matrix with diagonal entries α, sub-diagonal entries β, and super-diagonal entries γ.

$$\begin{bmatrix} \alpha & \gamma & 0 & \cdots & \beta \\ \beta & \alpha & \gamma & \cdots & 0 \\ 0 & \beta & \alpha & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \gamma & 0 & 0 & \cdots & \alpha \end{bmatrix}.$$

Reference [7] shows how to construct polynomial-size sparse-access oracles for this class of matrices, for which we show high-level sketches in Figure 1. In particular, the authors show how to construct O_A in $O(1)$ time and O_c in $O(\log(N))$ time[1]. Combining this with Theorems 2.2 and 2.3, we can see that using the aforementioned construction, we can perform matrix inversion by QSVT in time $O(\kappa \log(N) \log(\kappa/\varepsilon))$, retaining the exponential quantum speedup with respect to the matrix size. This furthermore outperforms the HHL algorithm with respect to both precision and the condition number, which has a complexity of $O(\kappa^2 \log(N)/\varepsilon)$ [1].

[1] Their construction of O_c requires multi-controlled NOT gates with $O(\log(N))$ controls; however, this does not prohibitively impact the complexity as such gates can be decomposed into linear-depth circuits with local gates [9].

References

[1] Aram W Harrow, Avinatan Hassidim, and Seth Lloyd. "Quantum algorithm for linear systems of equations". In: *Physical review letters* 103.15 (2009), p. 150502.

[2] András Gilyén et al. "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*. 2019, pp. 193–204.

[3] John M Martyn et al. "Grand unification of quantum algorithms". In: *PRX quantum* 2.4 (2021), p. 040203.

[4] Dominic W Berry et al. "Efficient quantum algorithms for simulating sparse Hamiltonians". In: *Communications in Mathematical Physics* 270 (2007), pp. 359–371.

[5] A Yu Kitaev. "Quantum measurements and the Abelian stabilizer problem". In: *arXiv preprint quant-ph/9511026* (1995).

[6] Guang Hao Low and Isaac L Chuang. "Optimal Hamiltonian simulation by quantum signal processing". In: *Physical review letters* 118.1 (2017), p. 010501.

[7] Daan Camps et al. "Explicit quantum circuits for block encodings of certain sparse matrices (2022)". In: *arXiv preprint arXiv:2203.10236* ().

[8] Christoph Sünderhauf. "Generalized Quantum Singular Value Transformation". In: *arXiv preprint arXiv:2312.00723* (2023).

[9] Adenilton J Da Silva and Daniel K Park. "Linear-depth quantum circuits for multiqubit controlled gates". In: *Physical Review A* 106.4 (2022), p. 042602.