HHL Quantum Algorithm Advancements

Brian Goldsmith

Computer Science, Vanderbilt University
Nashville, United States of America
brian.d.goldsmith@vanderbilt.edu

***Abstract*—Quantum algorithms have the potential for significant advantages over classical algorithms. However, there are a number of obstacles that need to be overcome to realize the potential of quantum algorithms. Most of these obstacles are the result of limitations of quantum devices and therefore, the algorithms need to be optimized in order to use fewer quantum resources and to use those resources more efficiently.**

The HHL quantum algorithm proposed an exponential speedup in processing linear systems of equations. However, implementing the HHL algorithm is not feasible with current quantum devices. Therefore, work is being done to apply optimizations to the original HHL algorithm to realize the exponential speedups. An example is the work being done by the Future Lab for Applied Research and Engineering of JPMorgan Chase Bank, N.A. who have introduced optimizations that allow for a version of HHL to run on current Noisy Intermediate Scale Quantum hardware.

I. Introduction

I would like to enter the quantum computing field to help advance the use of quantum computing. In order to best understand what advancements are possible, it is useful to understand the advancements currently being made by the scientific and corporate communities. Presently, a number of companies are working on software aspects of quantum computing[1] with the intention of using algorithms on current quantum machines, as well as implementing the algorithms when more advanced quantum hardware is available. This paper will analyze a recent, published advancement with the goal of having a better understanding of how to develop and optimize algorithms for quantum computing.

A. Linear Systems of Equations

This paper will discuss linear systems of equations. It is helpful to show a small example so that it can be clear why computing is needed and how quantum computing compares to classical computing. A system of linear equations is represented as $A\vec{x} = \vec{b}$ where A is a matrix, x is a vector of unknowns, and b is the vector of the solutions.

A simple example of a system of linear equations is:

$$x_1 + 2x_2 = 6$$

$$-2x_1 + x_2 = 8$$

This can be re-written in the Ax=b format as:

$$\begin{bmatrix} 1 & 2 \\ -2 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 8 \end{bmatrix}$$

where A= $\begin{bmatrix} 1 & 2 \\ -2 & 1 \end{bmatrix}$ $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ $\vec{b} = \begin{bmatrix} 6 \\ 8 \end{bmatrix}$

In the example, calculating the values of x1 and x2 is not difficult. However, as the number of equations and variables grows, the calculations become more complex.

II. HHL

In 2009, published their paper Quantum Algorithm for Linear Systems of Equations.[2] The quantum algorithm put forth in the paper has since been known as HHL and is the foundation for solving systems of linear equations. The authors used Shor's factoring algorithm[3] as an example of how for certain problems, quantum algorithms could provide exponential speedups over classical algorithms. Since linear equations are heavily utilized in science and engineering, they saw this as a prime area to apply a quantum algorithm, as the resulting algorithm could be widely applied.[2]

Interestingly, the quantum algorithm does not find an exact solution for a given linear system of equations, but instead the algorithm attempts to find an approximation of an expectation value of an operator related to the linear system of equations.[2] While this may sound confusing, another way to think of it is that the HHL algorithm looks to estimate features of the solution to a set of linear equations. By knowing the estimates of the features, one is able to have enough information to make useful conclusions.

Again, HHL is not attempting to find $\vec{x}$, but instead find an approximation of the expectation value $\vec{x}^{\dagger}\mathrm{M}\vec{x}$ for some matrix, operator, M. It is assumed that the matrix A is Hermitian and also sparse[2], meaning most of the elements are zero. The matrix A has a condition number κ which informs whether the matrix is ill formed or well formed. (A well conditioned matrix has a κ that is approximately 1, while an ill-conditioned matrix has a conditioned matrix greater than 1.[4]) Classical algorithms can find $\vec{x}$ and $\vec{x}^{\dagger}\mathrm{M}\vec{x}$ in time scaling roughly as $N\sqrt{\kappa}$. However, the runtime of the HHL algorithm to estimate $\vec{x}^{\dagger}\mathrm{M}\vec{x}$ is polynomial of log(N) and κ.[2] Therefore, for small values of κ or rather very

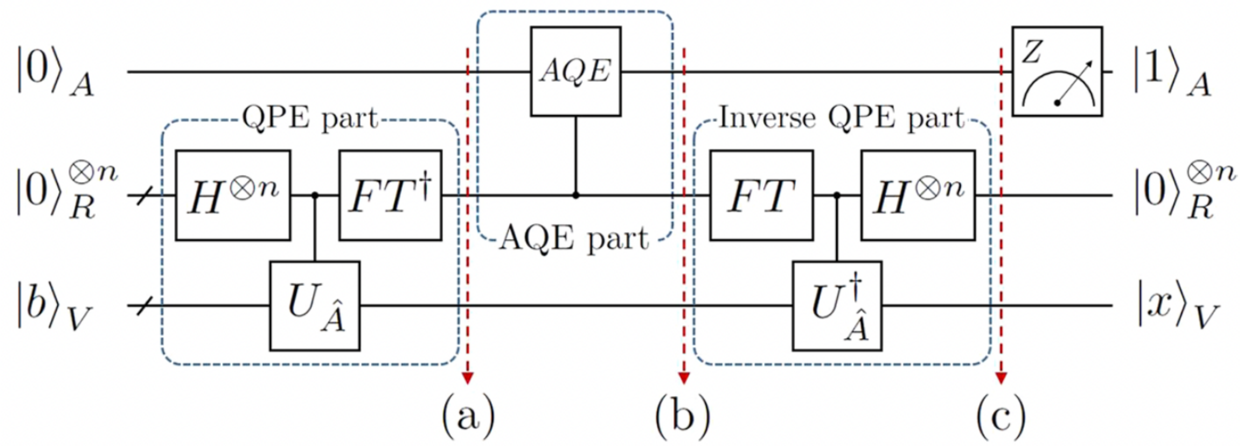

Fig. 1. The circuit diagram for the HHL algorithm[2] : the circuit consists of the QPE part, the AQE part, and the inverse QPE part. The unitary gate UÂ=eiAt is used for controlled-unitary gates between the register R and the input qubit V while the controlled AQE indicates a set of controlled gates between the register R and the ancillary qubit A.[8]

well-conditioned matrices, the quantum algorithm is exponentially faster than the classical algorithms.

A. Background

Since a system of linear equations is expressed as $A\vec{x} = \vec{b}$, it can also be written as $\vec{x} = A^{-1}\vec{b}$ where A^{-1} is the inverse matrix of A. For the quantum algorithm using qubit states, the equation would be written as $|x\rangle = A^{-1}|b\rangle$.

For the HHL algorithm, it is assumed that A is an $N \times N$ Hermitian matrix.[2] This is important because Hermitian matrices can be expressed as unitary matrices, even if the original matrix is not unitary, by using the form $e^{iA2\pi}$.[5]–[7] Continuing with the math to define A,

$$e^{iA2\pi} = \sum_{j=1}^{N} e^{i\lambda_j 2\pi} |u_j\rangle \langle u_j|$$

where λ_j is an eigenvalue of matrix A and u_j is an eigenvector.[7] This can be rewritten so that $A = \sum_{j=1}^{N} \lambda_j |u_j\rangle \langle u_j|$ and $A^{-1} = \sum_{j=1}^{N} \frac{1}{\lambda_j} |u_j\rangle \langle u_j|$ [6], [7]

For vector b, it can be expressed as a linear combination of eigenvectors $|u_j\rangle$ as $|b\rangle = \sum_{j=1}^{N} \alpha_j |u_j\rangle$ where again u_j is an eigenvector and α_j is an element of the set of complex numbers such that $\sum_{j=1}^{N} |\alpha_j|^2 = 1$.[7], [8]

B. Algorithm

The HHL algorithm is made up of three parts, the QPE part, the AQE part, and finally the inverse QPE part.

The first part uses the quantum phase estimation[9]–[12] (QPE) algorithm to find the eigenvalues of A.[7] The circuit for the QPE part consists of a Hadamard gate to ensure superposition. Then a controlled unitary gate is applied to get the eigenvalue in Fourier basis.[7] Since the eigenvalue needs to be in measurement basis (zeros or ones), an inverse Fourier transform is then applied. The state at step (a) in Fig. 1 is

$$|0\rangle_A \otimes \sum_{j=1}^{N} \sum_{x=0}^{2^n-1} \alpha_j \beta_{x|j} |\lambda_x\rangle_R \otimes |u_j\rangle_V$$

where

$$\beta_{x|j} = \frac{1}{2^n} \sum_{y=0}^{2^n-1} e^{2\pi i y (\lambda_j - \frac{x}{2^n})} .[8]$$

The second part of the HHL circuit is the ancillary quantum encoding[8] (AQE). In this piece, the inverse eigenvalue is generated by rotating conditioned on $|\lambda_x\rangle$. Mathematically, this is represented by:

$$|0\rangle_A \otimes |\lambda_x\rangle_R \mapsto$$
$$\left(\sqrt{1 - \frac{c^2}{\lambda_j^2}} |0\rangle_A + \frac{c}{\lambda_j} |1\rangle_A\right) \otimes |\lambda_x\rangle_R .[8]$$

This results in the following state at step (b) in Fig. 1,

$$\sum_{j=1}^{N} \sum_{x=0}^{2^n-1} \left(\sqrt{1 - \frac{c^2}{\lambda_x^2}} |0\rangle_A + \frac{c}{\lambda_x} |1\rangle_A\right) \otimes \alpha_j \beta_{x|j} |\lambda_x\rangle_R \otimes |u_j\rangle_V$$

.[8] If the eigenvalues λ_j are perfectly n-estimated then the equation can be reduced to

$$\sum_{j=1}^{N} \left(\sqrt{1 - \frac{c^2}{\lambda_j^2}} |0\rangle_A + \frac{c}{\lambda_j} |1\rangle_A\right) \otimes \alpha_j |\lambda_x\rangle_R \otimes |u_j\rangle_V .[8]$$

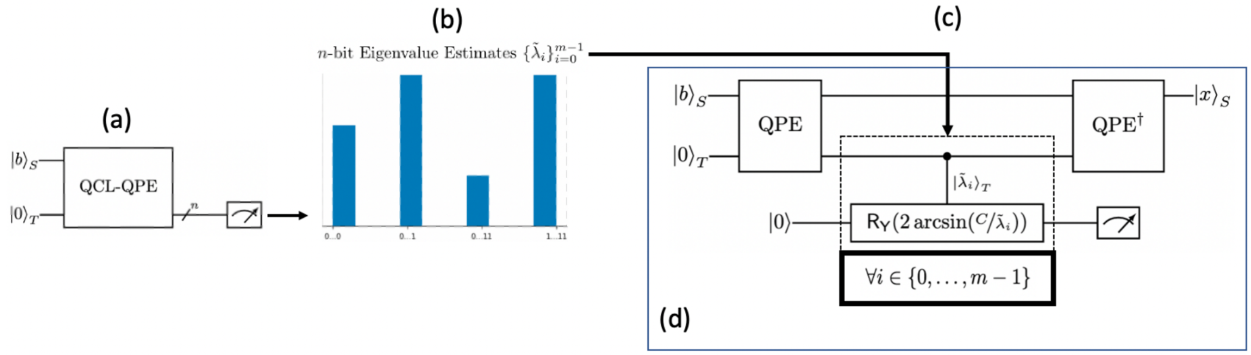

Fig. 2. NISQ-HHL end-to-end flow. (a) The QCL-QPE procedure to construct a distribution over the estimates of relevant eigenvalues. (b) Classical post-processing to obtain n-bit estimates of relevant eigenvalues. (c) Estimates from (b) are used to implement the eigenvalue inversion. (d) The standard HHL algorithm is used with the modified AQE.[14]

The final, main piece of the HHL circuit is the inverse QPE which not only uncomputes the R registers to set it back to the zero state and de-entangles the qubits, but also does a measurement of the ancillary qubit.[7], [13] After doing the inverse QPE, the state at step (c) in Fig 1. is $\sum_{j=1}^{N}\left(\sqrt{1-\frac{c^2}{\lambda_j^2}}\,|0\rangle_A + \frac{c}{\lambda_j}|1\rangle_A\right) \otimes |0\rangle_R \otimes \alpha|u_j\rangle_V$.[8], [13] At this point, the ancillary qubit is measured. If the state is zero, everything is discarded and the process is repeated. However, if the ancillary qubit is 1, the output is $|x\rangle_V$ where $|x\rangle_V = \sum_{j=1}^{N}\frac{\alpha_j}{\lambda_j}|u_j\rangle$ and again α_j is an element of the set of complex numbers such that $\sum_{j=1}^{N}|\alpha_j|^2 = 1$ [7], [8], λ_j is an eigenvalue of matrix A, and u_j is an eigenvector.[7]

As noted, the HHL algorithm does not return the full vector $\vec{x}$, but instead the quantum state $|x\rangle$. In order to get the full vector $\vec{x}$, the algorithm would need to be run N times.[2], [8] As previously mentioned, with an expectation value of $\vec{x}^{\dagger}M\vec{x} = \langle x|M|x\rangle$, where M is a linear operator, a number of features of the full vector $\vec{x}$ can be extracted. Some of the important features include normalization, weights in different parts of the space state, and moments.[2]

III. NISQ-HHL

In January 2022, the Future Lab for Applied Research and Engineering (FLARE) which is a research branch of JPMorgan Chase Bank, N.A. published NISQ-HHL: Portfolio Optimization for Near-Term Quantum Hardware.[14] In the paper, the team describes how they not only optimized HHL for speed, but also made it possible for it to run on currently available quantum computers. Rather than starting with the original HHL algorithm, the FLARE team used a classical/quantum hybrid of the HHL algorithm introduced by .[8] The hybrid implementation introduced classical computing for preparing the AQE portion of the original HHL algorithm as it was determined that quantum computing was not necessary or currently feasible for that portion.[8] Similarly, the FLARE team looked to optimize the AQE portion along with only using a subset of the eigenvalues to reduce the number of qubits needed to run the algorithm on noisy intermediate-scale quantum devices (NISQ.)[14]

To better understand the optimization to the AQE portion of the original HHL algorithm, it is helpful to look closer at what is happening during the rotation in order to get the eigenvalue inverse. Again, after the AQE portion, it is at a state $\sum_{j=1}^{N}\left(\sqrt{1-\frac{c^2}{\lambda_j^2}}\,|0\rangle_A + \frac{c}{\lambda_j}|1\rangle_A\right) \otimes |0\rangle_R \otimes \alpha|u_j\rangle_V$. The c in the equation is a normalization constant and it is used in calculating the angle of rotation.[8] The equation for the rotation is $\theta_j = 2\,arcsin\left(\frac{c}{\lambda_j}\right)$ for the j[th] rotation.[14] In order to complete the calculation, either information about the eigenvalues must be known or a coherent computation of the arcsine function must be done using quantum arithmetic. Unfortunately, doing the quantum arithmetic for the arcsine function is not feasible on current NISQ devices due to the number of CNOT gates required.[15] Therefore, to make the original HHL algorithm usable, prior information about the eigenvalues is necessary. This is the specific piece that the classical/quantum hybrid algorithm by Lee et al.[8] and the NISQ-HHL[14] algorithm change from the original HHL algorithm. In the classical/quantum hybrid algorithm, first the QPE algorithm is run to find the eigenvalues and then classical computing is used to create a reduced AQE. This replaces the original AQE that relies on quantum arithmetic.[8] The NISQ-HHL algorithm builds on this approach, as described below and seen in Fig. 2.

A. QCL-QPE

In the NISQ-HHL algorithm, when getting the estimates of the eigenvalues that will be used in the AQE rotation, the FLARE team replaced the standard QPE with a version that uses Quantum Conditional Logic (QCL)[14] which they call QCL-QPE. An interesting feature of QCL-QPE is that it does not attempt to generate the set of all eigenvalues of matrix A. Instead, it looks to create a set of *relevant eigenvalues* of the matrix A. This is described as $\Lambda_b := \{\lambda_i : |\frac{\beta_i}{\lambda_i}| > \epsilon$ where $\epsilon \geq 0$ is a configurable threshold.[14] Therefore, rather than returning the estimates of all eigenvalues, QCL-QPE returns a subset of eigenvalues where the absolute values of the amplitudes in the solution are above a predefined threshold.[14] By limiting the set of eigenvalues deemed relevant, the algorithm is able to even further reduce the number of controlled rotations in the AQE portion of the HHL algorithm.

B. Scaling Parameter

A standard QPE outputs the normalized phase angle of the eigenvalues of the unitary form of A (which in unitary form is $e^{iA2\pi}$.) Although the normalized phase angles are in (0,1][14], matrix A may have eigenvalues not contained in this interval. NISQ-HHL therefore introduces a scaler, γ, that ensures $||\gamma A||_2 \leq 1$.[14] To get the value of the γ, the algorithm estimates the λ_{max} and sets $\gamma = \lambda_{max}^{-1}$. However, since the full set of eigenvalues is not used, it is important that λ_{max} be in the subset of *relevant eigenvalues*. Therefore, the scaler's optimal value is noted as $\lambda_{max,b}^{-1}$ since the subset of eigenvalues is Λ_b.[14] The NISQ-HHL algorithm utilizes two algorithms to efficiently find the optimal γ value, so that QCL-QPE can run efficiently with minimal qubits.[14]

IV. Conclusion

The HHL algorithm is a foundational quantum algorithm for linear systems of equations. Since linear systems of equations can be found in many different disciplines, it is logical that others would be looking to optimize and build on the foundation. Analyzing the improvements for this particular example reveals how young the field of quantum algorithms is and how much improvement and growth is still attainable. This is especially true when pairing the quantum algorithms with quantum devices which are also at a very early stage of development. The possibilities and undiscovered advancements make the science of quantum algorithms an exciting field that will have a large impact on the future.

References

[1] Y. Naveh, "Council Post: Quantum Is Strategic To Companies And Countries, Upside And Downside Alike," *Forbes*. https://www.forbes.com/sites/forbestechcouncil/2022/06/29/quantum-is-strategic-to-companies-and-countries-upside-and-downside-alike/ (accessed Oct. 20, 2022).

[2] A. W. Harrow, A. Hassidim, and S. Lloyd, "Quantum Algorithm for Linear Systems of Equations," *Phys. Rev. Lett.*, vol. 103, no. 15, p. 150502, Oct. 2009, doi: 10.1103/PhysRevLett.103.150502.

[3] P. Shor, "Algorithms for Quantum Computation - Discrete Logarithms and Factoring," in *35th Annual Symposium on Foundations of Computer Science, Proceedings*, Los Alamitos, 1994, pp. 124–134. Accessed: Nov. 03, 2022. [Online]. Available: http://www.webofscience.com/wos/woscc/full-record/WOS:A1994BC14D00011

[4] *Matrix Condition Number and Matrix Norms - Well conditioned or Ill conditioned (Numerical Methods)*, (Mar. 03, 2017). Accessed: Nov. 03, 2022. [Online Video]. Available: https://www.youtube.com/watch?v=GDYmWtWWtNQ

[5] "lect3.26.pdf." Accessed: Nov. 04, 2022. [Online]. Available: https://www.math.purdue.edu/~eremenko/dvi/lect3.26.pdf

[6] "Hermitian matrix," *Wikipedia*. Oct. 30, 2022. Accessed: Nov. 04, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Hermitian_matrix&oldid=1118969984

[7] *Solving Linear Equations/ HHL algorithm*, (Mar. 21, 2021). Accessed: Nov. 04, 2022. [Online Video]. Available: https://www.youtube.com/watch?v=VIPURRU0WDI

[8] Y. Lee, J. Joo, and S. Lee, "Hybrid quantum linear equation algorithm and its experimental test on IBM Quantum Experience," *Sci. Rep.*, vol. 9, no. 1, Art. no. 1, Mar. 2019, doi: 10.1038/s41598-019-41324-9.

[9] A. Y. Kitaev, "Quantum measurements and the Abelian Stabilizer Problem." arXiv, Nov. 20, 1995. Accessed: Nov. 03, 2022. [Online]. Available: http://arxiv.org/abs/quant-ph/9511026

[10] S. Massar and S. Popescu, "Optimal Extraction of Information from Finite Quantum Ensembles," *Phys. Rev. Lett.*, vol. 74, no. 8, pp. 1259–1263, Feb. 1995, doi: 10.1103/PhysRevLett.74.1259.

[11] R. Derka, V. Buzek, and A. K. Ekert, "Universal algorithm for optimal estimation of quantum states from finite ensembles via realizable generalized measurement," *Phys. Rev. Lett.*, vol. 80, no. 8, pp. 1571–1575, Feb. 1998, doi: 10.1103/PhysRevLett.80.1571.

[12] S. F. Huelga, C. Macchiavello, T. Pellizzari, A. K. Ekert, M. B. Plenio, and J. I. Cirac, "Improvement of frequency standards with quantum entanglement," *Phys. Rev. Lett.*, vol. 79, no. 20, pp. 3865–3868, Nov. 1997, doi:

10.1103/PhysRevLett.79.3865.
[13] *Quantum Machine Learning - 37 - Overview of the HHL Algorithm*, (Dec. 28, 2019). Accessed: Nov. 04, 2022. [Online Video]. Available: https://www.youtube.com/watch?v=hQpdPM-6wtU
[14] R. Yalovetzky, P. Minssen, D. Herman, and M. Pistoia, “NISQ-HHL: Portfolio Optimization for Near-Term Quantum Hardware.” arXiv, Jan. 21, 2022. Accessed: Nov. 02, 2022. [Online]. Available: http://arxiv.org/abs/2110.15958
[15] T. Häner, M. Roetteler, and K. M. Svore, “Optimizing Quantum Circuits for Arithmetic.” arXiv, May 31, 2018. doi: 10.48550/arXiv.1805.12445.