HHL Algorithm

May 28, 2020

1 Introduction

Solving linear system of equations is a common problem that occurs frequently in the field of Machine Learning along with other fields.This algorithm gives us an exponential speed-up in matrix dimension compared to that of the classical analogues.For a given matrix **A** and a vector $\vec{\mathbf{b}}$, it finds a solution vector $\vec{\mathbf{x}}$ such that $A\vec{x} = \vec{b}$.
We consider the case where one doesn't need to know the solution $\vec{x}$ itself, but rather an approximation of the expectation value of some operator associated with $\vec{x}$, e.g., $\vec{x}^{\dagger}M\vec{x}$ for some matrix M. In this case, when A is sparse, $N \times N$ and has condition number κ, classical algorithms can find $\vec{x}$ and estimate $\vec{x}^{\dagger}M\vec{x}$ in $O(N\sqrt{\kappa})$.Here, we exhibit a quantum algorithm for this task that runs in poly(log N, κ) time, an exponential improvement over the best classical algorithm when it comes to matrix dimensions.

2 Approach

Given a Hermitian $N \times N$ matrix A, and a unit vector $\vec{b}$, suppose we would like to find $\vec{x}$ satisfying $A\vec{x} = \vec{b}$.First, the algorithm represents $\vec{b}$ as a quantum state,

$$|b\rangle = \sum_{i=0}^{N} b_i |i\rangle \tag{1}$$

Now we use techniques of Hamiltonian simulation to apply $e^{\iota At}$ to $|b\rangle$.The ability to exponentiate A translates,via the technique of Quantum Phase Estimation, into the ability to decompose $|b\rangle$ into the eigenbasis of A and to find the corresponding eigenvalues λ_i.Informally,after this stage state $|b\rangle$ can be represented as,

$$|b\rangle = \sum_{i=0}^{N} \beta_i |u_i\rangle|\lambda_i\rangle \tag{2}$$

Here, $|u_i\rangle$ represents the eigenbasis of A. Then,we perform a map that takes $|\lambda_i\rangle$ to $C\lambda_i^{-1}|\lambda_i\rangle$, where C is a constant.After this step succeeds, we uncompute the register $|\lambda_i\rangle$ and are left with a state proportional to:

$$\sum_{i=0}^{N} \beta_i \lambda_i^{-1}|u_i\rangle = A^{-1}|b\rangle = |x\rangle \tag{3}$$

3 Algorithm

First, we need to transform the Hermitian A into $e^{\iota At}$, so that we are able to use it at will. If A is s-sparse and efficiently row computable, meaning it has at most s non-zero entries per row and given a row index these entries can be computed in time $O(s)$, then we can compute $e^{\iota At}$ in time $\widetilde{O}(log(N)s^2t)$.If A is non-Hermitian define,

$$\text{C} = \begin{pmatrix} 0 & A \\ A^\dagger & 0 \end{pmatrix}$$

We can then compute $C\vec{y} = \begin{pmatrix} \vec{b} \\ 0 \end{pmatrix}$ to obtain $\vec{y} = \begin{pmatrix} 0 \\ \vec{x} \end{pmatrix}$
But for the rest of the report, we will assume that A is Hermitian.
Next,we have to prepare the state $|b\rangle$.Lets, assume there exists an unitary B that produces this state or there is some sub-routine of a bigger algorithm that does the task.
In the next step,we decompose $|b\rangle$ into the eigenvector basis ($|u_i\rangle$) of the Hermitian A (or equivalently of $e^{\iota At}$) having corresponding eigenvalues λ_i.Let,

$$|\Psi_0\rangle := \sqrt{\frac{2}{T}} \sum_{\tau=0}^{T-1} \sin \frac{\pi(\tau + \frac{1}{2})}{T} |\tau\rangle \tag{4}$$

For some large T.The coefficients of $|\Psi_0\rangle$ is chosen to minimize the error.
Next we apply the Hamiltonian simulation $\sum_{\tau=0}^{T-1} |\tau\rangle\langle\tau|^C \otimes e^{\frac{\iota A\tau t_0}{T}}$ to the state $|\Psi_0\rangle^C |b\rangle$, where $t_0 = O(\kappa/\epsilon)$ *(κ is called the condition number of the matrix*

*A and is defined as the ratio of maximum eigenvalue to minimum eigenvalue. ϵ is the error incurred).*Fourier transforming the first register gives the state

$$\sum_{i=1}^{N}\sum_{k=0}^{T-1}\alpha_{k|i}\beta_i|k\rangle|u_i\rangle \tag{5}$$

where $|k\rangle$ are the Fourier basis state,and $|\alpha_{k|i}|$ is large, iff $\lambda_i \approx \frac{2\pi k}{t_0}$. Defining $\widetilde{\lambda_k} := \frac{2\pi k}{t_0}$ we can relabel $|k\rangle$ register to obtain

$$\sum_{i=1}^{N}\sum_{k=0}^{T-1}\alpha_{k|i}\beta_i\left|\widetilde{\lambda_k}\right\rangle|u_i\rangle \tag{6}$$

Adding an ancilla qubit and rotating conditioned on $\left|\widetilde{\lambda_k}\right\rangle$ gives us

$$\sum_{i=1}^{N}\sum_{k=0}^{T-1}\alpha_{k|i}\beta_i\left|\widetilde{\lambda_k}\right\rangle|u_i\rangle\left(\sqrt{1-\frac{C^2}{\widetilde{\lambda}_i^2}}\,|0\rangle+\frac{C}{\widetilde{\lambda_i}}\,|1\rangle\right) \tag{7}$$

where $C = O(1/\kappa)$. We now undo the phase estimation to uncompute the $\left|\widetilde{\lambda_i}\right\rangle$. If the phase estimation were perfect, we would have $|\alpha_{k|i}| = 1$ if $\widetilde{\lambda_k} = \lambda_i$, and 0 otherwise. Assuming this for now, we obtain

$$\sum_{i=1}^{N}\sum_{k=0}^{T-1}\beta_i|u_i\rangle\left(\sqrt{1-\frac{C^2}{\widetilde{\lambda}_i^2}}\,|0\rangle+\frac{C}{\widetilde{\lambda_i}}\,|1\rangle\right) \tag{8}$$

To finish the inversion we measure the last qubit. Conditioned on measuring $|1\rangle$, we have the state

$$\sqrt{\frac{1}{\sum_{i=1}^{N}C^2|\beta_i|^2/|\lambda_i|^2}}\sum_{i=1}^{N}\beta_i\frac{C}{\lambda_i}|u_i\rangle \tag{9}$$

which corresponds to the state $|x\rangle$ of equation 3 upto normalization. We can determine the normalization factor by determining the probability of obtaining 1.Finally, we make a measurement M whose expectation value $\langle x|\,M\,|x\rangle$ corresponds to the feature of $\vec{x}$ that we wish to evaluate.
Putting everything all together, we obtain the runtime of $\widetilde{O}(s^2\kappa^2 log(N)/\epsilon)$.

4 Note

The report is based on the seminal work: `https://arxiv.org/abs/0811.3171`
A much easier overview of the algorithm can be found in: `https://qiskit.org/textbook/ch-applications/hhl_tutorial.html`

5 Conclusion

To conclude, using the techniques of Block Encoding,Linear Combination of Unitaries,Quantum Signal Processing,etc.much more optimal complexities have been obtained theoretically leading to better fidelity,query complexity,time complexity,better dependence on condition number(κ),error bound,etc.