The HHL Algorithm

Federico Matteoni

A.A. 2021/22

1 Introduction

The HHL algorithm was designed by Aram Harrow, Avinatan Hassidim and Seth Lloyd, and pulished in 2008.
Linear systems of equations are very commonplace throughout all the fields of science and field of research. The efficient manipulation of matrices has become a requirement for many different algorithms and applications, ranging from fluid simulation to machine learning. In all these fields, one of the most common problem is finding a solution to a linear system of equations.
Most classical methods compute an exact solution in polynomial time: this is a different setup from other quantum algorithms, that improve over the exponential time required from their classical counterparts. Linear systems in real context, however, may contain matrices with millions of parameters, and even a polynomial algorithm can take too long to compute.
HHL is an algorithm that approximates a function of the solution vector of a linear system with logarithmic time complexity. This speedup has significant implications regarding applications of machine learning algorithms on quantum computers.

2 Linear Systems

Given a matrix $A \in \mathbb{C}^{N\times N}$ and a vector $b \in \mathbb{C}^N$, the solution of the linear system is $x \in \mathbb{C}^N$ such that $Ax = b$. For example, with $N = 2$:

$$A = \begin{pmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{pmatrix} \quad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \quad b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

The problem can be written as finding $x_1, x_2 \in \mathbb{C}$ such that

$$\begin{cases} x_1 - \dfrac{x_2}{3} = 1 \\ -\dfrac{x_1}{3} + x_2 = 0 \end{cases}$$

The HHL algorithm requires some conditions on the components of the problem:

The matrix A must be

Hermitian, meaning $A = A^H \Leftrightarrow a_{ij} = \overline{a}_{ij}$

s-sparse, meaning at most s non-zero elements per row or column

with conditioning number $\mathbf{k}$, meaning $\mathbf{k} = \frac{\lambda_{\max}}{\lambda_{\min}}$

The vector b must be unitary, meaning $\|b\| = 1$

Given ϵ accuracy, solving this problem on a classical computer requires

$$O\left(Ns\mathbf{k}\left(\frac{1}{\epsilon}\right)\right)$$

while the HHL algorithm can compute **a function of the solution** in

$$O\left(\frac{\log(N)s^2\mathbf{k}^2}{\epsilon}\right)$$

By using the HHL algorithm we obtain an exponential speedup in N, dimension of the system, but we have a very important difference: the classical algorithm returns the exact solution, while HHL can only approximate a function of the solution. The solution can still be obtained, but reading it requires a time linear in the number of elements of the solution, which can cancel out the obtained speedup. So this algorithm is very useful when we're interested in a function $x^T M x$ of the solution x rather than in the solution itself.

3 Encoding the Problem in the Quantum World

We can assume b normalized, meaning $||b|| = 1$, and we want to map it into $|b\rangle$. We will encode b in $|b\rangle$ using the **amplitude encoding**: each b_i will be the amplitude of the ith element of the basis of the quantum state. A is an Hermitian matrix. This means that it is a square matrix that's equal to its transposed conjugate, $A = A^H$. Hence, it has a spectral decomposition: given $\lambda_j \in \mathbb{R}$ eigenvalue of $|u_j\rangle$, the jth eigenvector, we have

$$A = \sum_{j=0}^{N-1} \lambda_j |u_j\rangle \langle u_j|$$

$$A^{-1} = \sum_{j=0}^{N-1} \lambda_j^{-1} |u_j\rangle \langle u_j|$$

The eigenvectors of A form a basis, hence we can rewrite b in that basis

$$|b\rangle = \sum_{j=1}^{N-1} b_j |u_j\rangle$$

with $b_j \in \mathbb{C}$. $|x\rangle$ can also be expressed in this form:

$$|x\rangle = A^{-1} |b\rangle = \sum_{i=0}^{2^{n_b}-1} \lambda_i^{-1} b_i |u_i\rangle$$

The goal of HHL is to find $|x\rangle$ in this form and store it in a quantum register.
This requires some normality conditions:

$$\sum_{j=0}^{2^{n_b}-1} |b_j|^2 = 1 \quad \sum_{i=0}^{2^{n_b}-1} |\lambda_i^{-1} b_i|^2 = 1$$

4 The Algorithm Step-by-Step