Taylor approximation of the arbitrary rotation in HHL

July 29, 2022

We need to apply a rotation gate that does the following

3. Add an auxiliary qubit and apply a rotation conditioned on $|\lambda_j\rangle$,

$$\sum_{j=0}^{N-1} b_j |\lambda_j\rangle_{n_l} |u_j\rangle_{n_b} \left(\sqrt{1 - \frac{C^2}{\lambda_j^2}} |0\rangle + \frac{C}{\lambda_j} |1\rangle \right)$$

Where C must be smaller than the smallest eigenvalue you can encounter since you need $|C/\lambda_j|^2 \leq 1$ (but it should be also be as large as possible so you are likely to measure $|1\rangle$ and proceed with the algorithm. This rotation can be done via

$$R_y(\theta)$$

where

$$\theta = 2 \arcsin C/\lambda.$$

The tangent line approximation of this function (with variable λ) (first two terms of Taylor series expansion):

$$\theta = 2 \arcsin C/\lambda \approx 2 \arcsin(C/t) - \frac{2}{t^2 \sqrt{1 - (C/t)^2}} (\lambda - t),$$

where t is ideally as close to all the eigenvalues you encounter as possible (t is average of eigenvalues).

For example in the Qiskit example,

$$A = \begin{pmatrix} 1 & -1/3 \\ -1/3 & 1 \end{pmatrix} \quad \text{and} \quad |b\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

We will use $n_b = 1$ qubit to represent $|b\rangle$, and later the solution $|x\rangle$, $n_l = 2$ qubits to store the binary representation of the eigenvalues and 1 auxiliary qubit to store whether the conditioned rotation, hence the algorithm, was successful.

For the purpose of illustrating the algorithm, we will cheat a bit and calculate the eigenvalues of A to be able to choose t to obtain an exact binary representation of the rescaled eigenvalues in the n_l-register. However, keep in mind that for the HHL algorithm implementation one does not need previous knowledge of the eigenvalues. Having said that, a short calculation will give

$$\lambda_1 = 2/3 \quad \text{and} \quad \lambda_2 = 4/3$$

You can choose $C = 2/3$ and $t = \text{average}(\lambda_1, \lambda_2) = 1$. Then the we have

$$\theta \approx 2 \arcsin(2/3) - \frac{2}{\sqrt{1 - (2/3)^2}} \lambda + \frac{2}{\sqrt{1 - (2/3)^2}}$$

This could be implemented through rotation gates

$$R_y(2 \arcsin(2/3)) R_y \left(\frac{2}{\sqrt{1 - (2/3)^2}} \right) \prod_r \text{Controlled } R_y \left(r\text{th bit of } |\lambda\rangle, 2^r \frac{2}{\sqrt{1 - (2/3)^2}} \right)$$

This is a very crude approach a is probably not suited for a quantum computer as written. For one the λ_i values aren't integers and so writing them down in binary isn't an option.