Smart Post Processing for Algorithms with QPE

Viraj Dsouza

Quantum Phase Estimation (QPE) has been the key for many quantum algorithms. Suppose a unitary operator U has an eigenvector $|u\rangle$ with eigenvalue $e^{2\pi i\phi}$ where ϕ is unknown, that is, $U|u\rangle = e^{2\pi i\phi}\,|u\rangle$. The goal of the phase estimation algorithm is to estimate ϕ.

To understand the procedure involved in QPE, its useful to review Quantum Fourier Transform (QFT) which is an essential ingredient in enabling QPE.

1 Introduction to Quantum Fourier Transform:

A Discrete Fourier transform (DFT) takes in a input vector of complex numbers and outputs a transformed vector of complex numbers. In the case of QFT, the set of input complex numbers are amplitudes of a quantum state.

Definition 1.1. *DFT of the vector $x = (x_0\ x_1 \dots\ x_{N-1})^T$ is the complex vector $y = (y_0\ y_1 \dots y_{N-1})^T$ where*

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{\frac{2\pi ijk}{N}} x_j. \tag{1}$$

Now suppose that we have an $N = 2^n$-dimensional quantum state vector $|\psi\rangle$, represented by the amplitudes $x = (x_0\ x_1 \dots\ x_{N-1})^T$, that is

$$|\psi\rangle = \sum_{j=0}^{N-1} x_j\,|j\rangle\,.$$

Definition 1.2. *Quantum Fourier Transform QFT of state $|\psi\rangle$ is given by*

$$|\phi\rangle = \sum_{k=0}^{N-1} y_k\,|k\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} e^{\frac{2\pi ijk}{N}} x_j\,|k\rangle\,, \tag{2}$$

where y_k is defined as 1 for $k = 0, \dots, N-1$.

For convenience, we denote $e^{\frac{2\pi i}{N}}$ by ω. With this, the above expression can be expressed as

$$|\phi\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \sum_{j=0}^{N-1} \omega^{jk} x_j |k\rangle .$$

Given the entries of the vector x, the entries of the vector y corresponding to QFT of the vector x is thus given by,

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{\frac{2\pi ijk}{N}} x_j = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{jk} x_j.$$

As an example to illustrate QFT, let's take QFT of the general single qubit quantum state given by $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$.

The quantum state $|\psi\rangle$ is represented by $(\alpha\ \beta)^T$ where $x_0 = \alpha$ and $x_1 = \beta$. Given x_0 and x_1, we calculate y_0 and y_1 from the above expression.

$$y_0 = \frac{1}{\sqrt{2}} \sum_{j=0}^{1} e^{\frac{2\pi ij\cdot 0}{2}} x_j = \frac{\alpha + \beta}{\sqrt{2}}$$

$$y_1 = \frac{1}{\sqrt{2}} \sum_{j=0}^{1} e^{\frac{2\pi i\cdot j\cdot 1}{2}} x_j = \frac{1}{\sqrt{2}} \left(e^{\frac{2\pi i\cdot 1\cdot 0}{2}} x_0 + e^{\frac{2\pi i\cdot 1\cdot 1}{2}} x_1 \right) = \frac{\alpha - \beta}{\sqrt{2}}$$

Hence the new state is $\frac{\alpha+\beta}{\sqrt{2}} |0\rangle + \frac{\alpha-\beta}{\sqrt{2}} |1\rangle$. One can recognize that this is exactly the same state obtained after applying Hadamard $|\psi\rangle$. Hence, the effect of applying QFT to a single qubit is equivalent to applying a Hadamard gate.

1.1 Circuit Implementation of QFT

Now we will look at the effect of applying QFT to a n-qubit system. Let $N = 2^n$.

Let $|j\rangle$ be a basis state where $j = 0, \ldots, N-1$. We will use the binary representation for j, that is $j = j_1 j_2 \cdots j_n$ for $j_i \in \{0, 1\}$. More formally

$$j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0 = 2^n \sum_{l=1}^{n} 2^{-l} j_1 \tag{3}$$

For example if $j = 3$ and we want to represent this using $n = 2$ qubits, then $3 = 1(2^{2-1}) + 1(2^{2-2})$ which implies that $j_1 = 1$ and $j_2 = 1$. Hence the binary representation of 3 using 2 qubits is 11.

We use the representation given by 3 to express the state obtained after applying QFT to $|j\rangle$.

Now $|j\rangle = |j_1 j_2 \cdots j_n\rangle$ and after QFT

$$|j\rangle \rightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{\frac{2\pi ijk}{N}} |k\rangle$$

Replacing k using 3 and using $N = 2^n$ we get,

$$\frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{\frac{2\pi ijk}{N}} |k\rangle = \frac{1}{2^{n/2}} \sum_{k_1=0}^{1} \sum_{k_2=0}^{1} \cdots \sum_{k_n=0}^{1} e^{2\pi ij(\sum_{l=1}^{n} k_l 2^{-l})} |k_1 k_2 \cdots k_n\rangle$$

Writing the quantum state as the tensor product of n qubits, the above expression reduces to,

$$= \frac{1}{2^{n/2}} \sum_{k_1=0}^{1} \sum_{k_2=0}^{1} \cdots \sum_{k_n=0}^{1} \prod_{l=1}^{n} e^{2\pi ijk_l 2^{-l}} |k_1 k_2 \cdots k_n\rangle$$

$$= \frac{1}{2^{n/2}} \otimes_{l=1}^{n} \sum_{k_o=0}^{1} e^{2\pi ijk_l 2^{-l}} |k_o\rangle = \frac{1}{2^{n/2}} \otimes_{l=1}^{n} \Big(|0\rangle + e^{2\pi ij 2^{-l}} |1\rangle \Big)$$

$$= \frac{1}{2^{n/2}} \Big(\big(|0\rangle + e^{2\pi ij2^{-1}} |1\rangle\big) \otimes \big(|0\rangle + e^{2\pi ij2^{-2}} |1\rangle\big) \otimes \cdots \otimes \big(|0\rangle + e^{2\pi ij2^{-n}} |1\rangle\big) \Big) \quad (4)$$

To implement the above operations through a quantum circuit, we define the two qubit CR_k operator which puts a relative phase of $e^{\frac{2\pi i}{2^k}}$ in front of the quantum state if both the controlled and the target qubits are in state $|1\rangle$. The CR_k gate corresponds to a rotation around z-axis and has the following matrix representation:

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{\frac{2\pi i}{2^k}} \end{pmatrix} \quad (5)$$

Let $|j_1 j_2 \cdots j_n\rangle$ be the input state. We'll start witht the first qubit.
Applying Hadamard to the first qubit, we get

$$\rightarrow \frac{1}{2^{1/2}} \big(|0\rangle + e^{2\pi ij_1 2^{-1}} |1\rangle\big) |j_2 \cdots j_n\rangle$$

As we can see, if $j_1 = 0$ the sign before $|1\rangle$ is $+$ and if $j_1 = 1$, the sign before $|1\rangle$ is $-$ as needed.

Now let's apply the operator CR_2 where second qubit is the control and the first qubit is the target. The resulting state is,

$$\rightarrow \frac{1}{2^{1/2}} \big(|0\rangle + e^{2\pi i(j_1 2^{-1} + j_2 2^{-2})} |1\rangle\big) |j_2 \cdots j_n\rangle$$

We then apply CR_i operator where qubit i is the control qubit and the first qubit is the target, consecutively for $i = 3 \ldots n$. We then obtain the state,

$$\rightarrow \frac{1}{2^{1/2}} \left(|0\rangle + e^{2\pi i (j_1 2^{-1} + j_2 2^{-2} + j_3 2^{-3} \ldots j_n 2^{-n})} |1\rangle \right) |j_2 \cdots j_n\rangle \tag{6}$$

$$= \frac{1}{2^{1/2}} \left(|0\rangle + e^{2\pi i j 2^{-n}} |1\rangle \right) |j_2 \cdots j_n\rangle \tag{7}$$

Continuing the above process (applying Hadamard followed by successive CR_i operator) for second qubit, third qubit and so on. We end up with the below state,

$$\frac{1}{2^{n/2}} \left(|0\rangle + e^{2\pi i j 2^{-n}} |1\rangle \right) \otimes \left(|0\rangle + e^{2\pi i j 2^{-n+1}} |1\rangle \right) \otimes \cdots \otimes \left(|0\rangle + e^{2\pi i j 2^{-1}} |1\rangle \right)$$

This looks nearly the same as 4, with only difference being the qubits are in the reverse order. This can be modified by using $SWAP$ gates.

Let us briefly look into the QFT circuit complexity. For the first qubit, we apply a single Hadamard gate followed by $(n-1)$ CR gates, which makes n gates in total. For the second qubit we apply a single Hadamard gate followed by $(n-2)$ CR gates. Continuing in this way, we see that we need n Hadamard gates and $(n-1)+(n-2)+...+1+0$ CR gates (note that only one Hadamard acts on the n^{th} qubit). Thus we require a total of $\frac{(n)(n+1)}{2}$ gate sand furthermore, $\frac{n}{2}$ $SWAP$ gates are required, each of which can be implemented using three $CNOT$ gates. Thus the circuit provides a $\theta(n^2)$ algorithm for applying QFT to an n-qubit system represented by a vector of size $N = 2^n$.

The best known classical algorithm for computing the Discrete Fourier Transform of 2^n entries, such as Fast Fourier Transform (FFT) requires $\Theta(n2^n)$, equivalently $\Theta(N \log N)$ gates, which means that the classical algorithm requires exponentially many more operations to compute DFT.
Nevertheless, this does not mean that we can use QFT directly to accelerate the classical computation process. There are two reasons for this: The first reason is that the amplitudes can not be accessed directly after applying QFT. The second reason is that we may not know how to efficiently prepare the input state to be Fourier Transformed.
As all the gates used in the circuit implementation of QFT are unitary, QFT is a unitary transformation and has an inverse equal to $QFT^\dagger$
Inverse Quantum Fourier Transform ($QFT^\dagger$) is the transformation which satisfies $QFT \cdot QFT^\dagger = I$. Hence to implement $QFT^\dagger$, one should apply all the operations in reverse order to undo the circuit. The $QFT^\dagger$ is defined almost the same as QFT but with the exponents having a negative sign.

$$QFT^{\dagger}|k\rangle = \frac{1}{\sqrt{N}}\sum_{l=0}^{N-1} e^{-\frac{2\pi ijk}{N}}|l\rangle .$$

2 Introduction to Quantum Phase Estimation:

Consider a unitary operator U that has an eigenvector $|u\rangle$ with eigenvalue $e^{2\pi i\phi}$ where ϕ is unknown, that is, $U|u\rangle = e^{2\pi i\phi}|u\rangle$. The goal of the phase estimation algorithm is to estimate ϕ. To perform the estimation, we assume that we have available black boxes (also called *oracles*) capable of preparing the state $|u\rangle$ and performing the controlled-$U^{2^j}(CU^{2^j})$ operation, for suitable non-negative integers j.
The controlled version of the U operator is given by,

$$CU(|0\rangle|u\rangle) \rightarrow |0\rangle|u\rangle \quad \text{and} \quad CU(|1\rangle|u\rangle) \rightarrow e^{2\pi i\phi}|1\rangle|u\rangle$$

Now the matrix representation of the operator CU^{2^j} is same as 5 with the replacement of k by 2^j
Thus for an arbitrary state,

$$\alpha|0\rangle|u\rangle + \beta|1\rangle|u\rangle \xrightarrow{CU} \alpha|0\rangle|u\rangle + e^{2\pi i\phi}\beta|1\rangle|u\rangle = (\alpha|0\rangle + e^{2\pi i\phi}\beta|1\rangle)|u\rangle$$

Algorithm:
QPE requires two registers where first register contains t qubits which are in state $|0\rangle$. t depends on the number of digits of accuracy and the probability of success while estimating ϕ (more on this later). Suppose that the qubits in the first register are numbered from 1 to t. Second register stores $|u\rangle$ and has as many qubits necessary to store $|u\rangle$. We are given controlled U^{2^j} operators as black-box functions.

1. Apply Hadamard to first register which has t qubits, all initialized to $|0\rangle$. The new quantum state is

$$\frac{1}{2^{t/2}}\big(|0\rangle + |1\rangle\big) \otimes \cdots \otimes \big(|0\rangle + |1\rangle\big)|u\rangle .$$

2. Apply CU^{2^j} gate where qubit $t-j$ is the control for $j = 0, \ldots, t-1$ and the qubits representing the state $|u\rangle$ being the target. That is,
For $j = 0$, Apply CU^{2^0} where qubit t is the control qubit.
For $j = 1$, Apply CU^{2^1} where qubit $t-1$ is the control qubit.
Continuing this process until $j = t-1$, where qubit 1 is the control, we get the following state from the first stage of phase estimation,

$$\frac{1}{2^{t/2}}\big(|0\rangle + e^{2\pi i\phi 2^{t-1}}|1\rangle\big) \otimes \cdots \otimes \big(|0\rangle + e^{2\pi i\phi 2^{1}}|1\rangle\big) \otimes \big(|0\rangle + e^{2\pi i\phi 2^{0}}|1\rangle\big)|u\rangle \quad (8)$$

The form of this equation is same as 5, hence back tacking the calculation we did to arrive at 5, we can express 8 as,

$$\frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{2\pi i k\phi}\left|k\right\rangle\left|u\right\rangle$$

To realize why the phase estimation algorithm works, we express ϕ exactly in t bits. That is,

$$\phi = 0.\phi_1\ldots\phi_t = \frac{\phi_1\ldots\phi_t}{N} = \frac{x}{2^t},$$

The resulting state now is,

$$\frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{\frac{2\pi i k x}{N}}\left|k\right\rangle\left|u\right\rangle.$$

The state of the first register is clearly of the form of Fourier transform of state $\left|x\right\rangle$, hence applying $QFT^\dagger$ to the first register, we exactly measure $\left|x\right\rangle\left|u\right\rangle = \left|\phi_1\ldots\phi_t\right\rangle\left|u\right\rangle$.

3. Apply $QFT^\dagger$ on the first register we get,

$$\frac{1}{2^{t/2}}\sum_{k=0}^{2^t-1} e^{\frac{2\pi i k x}{N}}\left|k\right\rangle\left|u\right\rangle \xrightarrow{QFT^\dagger\otimes I}\left|x\right\rangle\left|u\right\rangle = \left|\phi_1\ldots\phi_t\right\rangle\left|u\right\rangle.$$

Finally the required phase is just $\phi = \frac{x}{2^t}$.

Now let's consider the case where $\left|u\right\rangle$ is a linear superposition of states $\left|u_p\right\rangle$, that is $\left|u\right\rangle = \sum_p c_p\left|u_p\right\rangle$ and the unitary operator U is such that $U\left|u_p\right\rangle = e^{\frac{2\pi i\phi_p}{2^n}}\left|u_p\right\rangle$. Now,

$$\left|\psi\right\rangle = QPE(\left|0\right\rangle^{\otimes n}\left|u\right\rangle) = \sum_p c_p QPE(\left|0\right\rangle^{\otimes n}\left|u_p\right\rangle) = \sum_p c_p(\left|\phi_p\right\rangle\left|u_p\right\rangle)$$

Now projecting this state onto $\left|\phi_p\right\rangle\left|u_p\right\rangle$ gives the corresponding phase ϕ_p with probability $|c_p|^2$.

Calculation:

$$\begin{aligned}\left\langle\psi\right|(\left|\phi_p\right\rangle\left\langle\phi_p\right|\otimes I)\left|\psi\right\rangle &= \left\langle\psi\right|(\sum_p c_p w^*\left|\phi_p\right\rangle\left\langle\phi_p|\phi_p\right\rangle\left|u_p\right\rangle)\\ &= (\sum_{p'} c_{p'}^*\left\langle\phi_{p'}u_{p'}\right|)(\sum_p c_p^*\left|\phi_p\right\rangle\left\langle\phi_p|\phi_p\right\rangle\left|u_p\right\rangle)\end{aligned}$$

$$= \sum_{p'} \sum_{p} c_p{}^* c_{p'}{}^* \delta_{pp'} \delta_{pp'} = |c_p|^2$$

We have used the fact that $\langle \phi_{p'} | \phi_p \rangle = \langle \psi_{p'} | \psi_p \rangle = \delta_{pp'}$

It's important to note the particular case of $\phi_p = \phi_p^{'}$ (the case where its not practical to distinguish between ϕ_p and $\phi_p^{'}$), where $p \neq p^{'}$. Doing the projection calculation as done earlier, we will find that we would get the required phase ϕ_p (or $\phi_{p'}$) with probability $|c_p|^2 + |c_{p'}|^2$.
Hence, just as before, phase associated with a given state of the superposition is given by $\phi_p = \frac{x}{2^t}$, where x is a positive integer (representing the bit string encoding the control register of the QPE circuit) and it's obtained when measurement is done on the first register in the QPE algorithm.

It's also useful to note that measuring $|\phi_p\rangle$ not just helps us find the eigenvalues of U but also the measurement causes the projection onto the eigenvector $|u_p\rangle$ of U.

Cost of QPE Algorithm:

The main cost of the algorithm is the implementation of U gates which are $\sum_{j=0}^{n-1} 2^j = 2^n - 1$ in total (note that this is exponential). Apart from this inverse QFT required $O(n^2)$ gates and a total of n Hadamard gates are required (the cost here is not exponential, hence can be ignored for a sufficiently large n). Thus we require $O(2^n)$ queries to know the required phase to n-bits (representing the precision of the phase calculated).
So for now we have an exact measurement of the phase without any errors.

2.1 The case where ϕ is not an integer:

The earlier case applied to the ideal case where ϕ could be written exactly with a t bit binary expression. In this section we show that even if this is not the case (i.e $e^{i2\pi\phi} \neq e^{i2\pi(\frac{x}{2^t})}$ for an integer x) the QPE algorithm would still produces a very good *approximation* of ϕ.

Just as before, after n-Hadamard gates and CU operations will result in the state,

$$\frac{1}{2^{t/2}} \sum_{k=0}^{2^t-1} e^{2\pi i k \phi} |k\rangle |u\rangle$$

Note that this is not exactly a Fourier transformed state and applying an inverse QFT will give,

$$\frac{1}{2^{t/2}} \sum_{kk'} e^{2\pi i k\phi} e^{\frac{-2\pi i k k'}{2^t}} |k'\rangle |u\rangle$$

Now define:

$$\alpha_{k'} = \sum_{k=0}^{2^t-1} \frac{\left[e^{\frac{-2\pi i}{2^t}(k'-\phi 2^t)}\right]^k}{2^t}$$

This is clearly a geometric series and can be easily summed through we get,

$$\alpha_{k'} = \frac{1}{2^t} \left[\frac{1 - \left[e^{\left(\frac{-2\pi i}{2^t}(k'-\phi 2^t)\right)}\right]^{2^t}}{1 - \left[e^{\left(\frac{-2\pi i}{2^t}(k'-\phi 2^t)\right)}\right]} \right]$$