QCQI: CHAPTER 5 SUMMARY

A PREPRINT

Krishna N Agaram

January 21, 2023

ABSTRACT

The Fourier Transform and applications to order-finding/period-finding and factoring.

1 The Fourier Transform

The discrete fourier transform maps a sequence $[\alpha_0, \ldots, \alpha_{n-1}] \in \mathbb{C}^n$ to a sequence $[\beta_0, \ldots, \beta_{n-1}] \in \mathbb{C}^n$ defined by:

$$\beta_j = \frac{1}{n} \sum_{i=0}^{n-1} \alpha_i \exp(2\pi \iota ij/n) = \frac{1}{n} \sum_{i=0}^{n-1} \alpha_i (\omega^j)^i$$

for each $0 \leq j \leq n-1$ where $\omega = e^{2\pi\iota/n}$ is the first nontrivial n^{th} root of unity. Note that the right hand side can be interpreted as a polynomial evaluation at $x = w^j$. In other words, the fourier transform maps polynomial coefficients to the valuations of the polynomial at the n^{th} roots of unity.

The DFT finds use in various domains and is especially useful to 'isolate' things of interest from a 'mixed' form. The key player here is the identity

$$\frac{1}{n} \sum_{i=0}^{n-1} e^{2\pi\iota i(j-k)/n} = \delta_{jk}$$

which incidentally, also allows us to compute the inverse fourier transform:

$$\alpha_k = \frac{1}{n} \sum_{j=0}^{n-1} \beta_j \exp(-2\pi\iota jk/n)$$

Proof.

$$\frac{1}{n} \sum_{j=0}^{n-1} \beta_j \exp(-2\pi\iota jk/n) = \frac{1}{n} \sum_{j=0}^{n-1} \frac{1}{n} \sum_{i=0}^{n-1} \alpha_i e^{2\pi\iota ij/n} e^{-2\pi\iota jk/n} = \frac{1}{n} \sum_{i=0}^{n-1} \alpha_i \delta_{ik} = \alpha_k$$

□

A point that will be useful later on: Notice that the DFT can be viewed as a linear function $\mathbb{C}^n \to \mathbb{C}^n$. And notice that $DFT^{-1} = DFT^\dagger$ from our construction of the inverse. This establishes that the fourier transform is infact a unitary linear transformation on $\mathbb{C}^n$. This motivates the **Quantum Fourier Transform:**

Definition (Quantum Fourier Transform)**.** *The quantum Fourier transform on an orthonormal basis* $|0\rangle, \ldots, |N-1\rangle$ *is defined to be a linear operator with the following action on the basis states:*

$$|i\rangle \to \frac{1}{N} \sum_{j=0}^{N-1} |j\rangle e^{2\pi\iota ji/N}$$

The reader should convince himself/herself that the transformation defined above is the same as the DFT only with the vector space $\mathbb{C}^n$ changed to the N-dimensional state space of the quantum system and the canonical basis to the basis $\{|0\rangle, \ldots, |N-1\rangle\}$. That is,

$$\text{QFT}\left(\sum_{i=0}^{N-1} \alpha_i |i\rangle\right) = \sum_{j=0}^{N-1} \beta_j |j\rangle$$

where $\{\beta_j\}$ is the discrete fourier transform of $\{\alpha_i\}$.

Again, the QFT is unitary, and it can be implemented for an n-qubit system efficiently in $\mathcal{O}(n^3)$ single qubit and CNOT gates. As in the classical case, the QFT has loads of applications. We see a few of them, culminating in Shor's algorithm for factorization. The interested reader should consult Nielsen and Chuang to see the much more general Hidden Subgroup problem that can be tackled using the QFT.

2 The Phase estimation algorithm

Consider a unitary operator U. Say $|u\rangle$ is an eigenvector with eigenvalue $e^{2\pi i\varphi}$. We would like to find (approximately) φ. We assume that we can query a blackbox to apply U^{2^j} for any $j \in \mathbb{Z}_{\geq 0}$. The key idea is to **encode φ into the phase space and then compute the inverse Fourier Transform**.

The key chain of events reads as follows (subscripts on the operator describe the qubit(s) it was applied to):

$$|0\rangle|u\rangle \xrightarrow{H_{[n]}^{\otimes n}} \left(\frac{1}{2^{n/2}}\sum_{j=0}^{2^n-1}|j\rangle\right)|u\rangle \xrightarrow[i\,\in\,[0,n-1]]{U_{n-i}^{2^i},} \frac{1}{2^{n/2}}\sum_{j=0}^{2^n-1}|j\rangle U^j|u\rangle = \left(\frac{1}{2^{n/2}}\sum_{j=0}^{2^n-1}e^{2\pi ij\varphi}|j\rangle\right)|u\rangle \xrightarrow{\text{QFT}_{[n]}^\dagger} \approx |\tilde{\varphi}\rangle|u\rangle$$

whereupon measuring the first register gives us the estimate $0.\tilde{\varphi}$ for φ. The $\approx$ is for two reasons. One, that φ is possibly more than n bits long, in which case φ is an n-bit approximation to φ. The other reason is that the pther statevectors $|j\rangle$ could also be received in the measurement with a small probability - their amplitudes are non-zero if φ is longer than n bits (in which case we say that the algorithm failed). Nevertheless, the algorithm is one of most vital importance and use in what follows.

Finally, another note: Preparing the eigenvector $|u\rangle$ may not be easy. But it may be easy to prepare a superposition of some eigenvectors (for example, in the order-finding algorithm below). In this case, we receive $\tilde{\varphi}$ with high probability for *one of the eigenvectors* in the superposition.

3 Order-finding and Shor's algorithm

Given integers $x, N > 0$ with $\gcd(x, N) = 1$, we would like to find the order r of x modulo N. Classically, this is hard (and cryptosystems exploit this fact to build secure encyption protocols). Here we show that with high probability we can find it in polynomial time and resources!

Let $L \equiv \lceil \log(N) \rceil$ be the number of bits needed to specify N. Here is the meat of the argument:

The quantum algorithm for order-finding is just the phase estimation algorithm applied to the unitary operator

$$U|y\rangle = \begin{cases} |xy \bmod N\rangle & 0 \leq y \leq N-1 \\ |y\rangle & N \leq y \leq 2^L - 1 \end{cases}$$