Quantum Fourier transform beyond Shor's algorithm

András Gilyén

Alfréd Rényi Institute of Mathematics
Budapest, Hungary

Quantum Computing Graduate Summer School, Park City Mathematics Institute, 2023 July 17

Day 1 – The Basics:
Discrete and Quantum Fourier Transform

Review of Chapter 4 of Ronald de Wolf's Quantum Computing lecture notes

`https://arxiv.org/abs/1907.09415v5`

Motivation and Applications of Fourier Transform

The Fourier transform is a widely used theoretical and practical tool to isolate different periodic parts of a function, signal, etc.

Some applications of the continuous Fourier Transform

- Solving differential equations
- Uncertainty principle in quantum mechanics
- ...

The discrete Fourier Transform can be viewed as its discretization (more about this tomorrow).

Some applications of the discrete Fourier Transform

- Signal processing (music)
- Image compression (jpeg)
- Fast multiplication of polynomials
- ...

And of course quantum computing!

The Discrete Fourier Transform

The Discrete Fourier transform is a unitary map over $\mathbb{C}^N$, whose matrix elements have the same absolute value in the computational basis. More precisely let $\omega_N := e^{-2\pi i/N}$, then

$$F_N := \frac{1}{\sqrt{N}} \begin{pmatrix} & \vdots & \\ \cdots & \omega_N^{jk} & \cdots \\ & \vdots & \end{pmatrix},$$

where $j, k \in \{0, 1, \ldots, N-1\}$ are row and column indices. In particular

$$H = F_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

(Note that here we are using mathematics convention for the phases $e^{-2\pi i/N}$, which might differ form the convention elsewhere including several quantum computing papers.)

Properties of the Discrete Fourier Transform

Unitarity

Calculating the sum of geometric sequences we can see that the columns are orthonormal

$$\sum_{j=0}^{N-1} \frac{1}{\sqrt{N}} (\omega_N^{jk})^* \frac{1}{\sqrt{N}} \omega_N^{jk'} = \frac{1}{N} \sum_{j=0}^{N-1} \omega_N^{j(k'-k)} = \begin{cases} 1 & \text{if } k = k' \\ 0 & \text{otherwise.} \end{cases}$$

- $F_N^{-1} = F_N^*$ (since F_N is symmetric)
- $\hat{v} := F_N v$ (standard notation for Fourier transform)
- The Fast Fourier transform (FFT) algorithm can compute $\hat{v}$ in $\mathcal{O}(N \log(N))$ steps instead of the naïve matrix-vector multiplication algorithm which makes $\approx N^2$ steps.
- One of the most important algorithms ever, in signal processing, etc.

Efficient Quantum Fourier Transform for $N = 2^n$

$$F_N|k\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega_N^{jk} |j\rangle$$

Efficient implementation using $\mathcal{O}\left(n^2\right)$ one- and two-qubit quantum gates

"Exponetially" faster than FFT (but access to output is limited).
Key property: $F_N|k\rangle$ is a product state. Let $j = j_1 \ldots j_n$ and $k = k_1 \ldots k_n$ in binary, then

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega_N^{jk} |j\rangle = \frac{1}{\sqrt{2^n}} \sum_{j \in \{0,1\}^n} \prod_{\ell=1}^{n} e^{-2\pi i j_\ell k/2^\ell} |j_1 \ldots j_n\rangle = \bigotimes_{\ell=1}^{n} \frac{1}{\sqrt{2}} \Big(|0\rangle + e^{-2\pi i k/2^\ell} |1\rangle \Big)$$

$$= \bigotimes_{\ell=1}^{n} \frac{1}{\sqrt{2}} \Big(|0\rangle + e^{-2\pi i\, 0.k_{n-\ell+1} \ldots k_n} |1\rangle \Big).$$

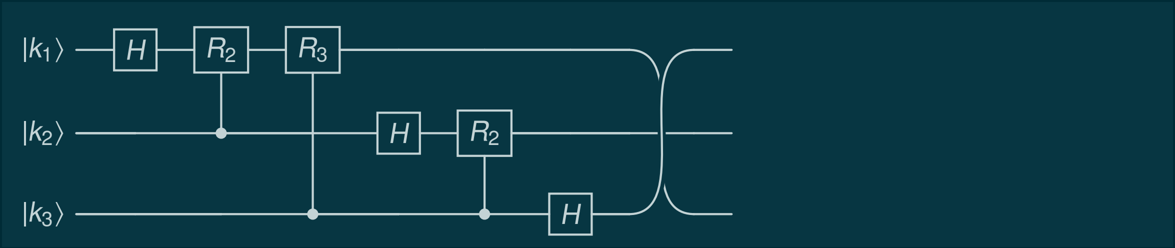
Efficient Quantum Fourier Transform for $N = 2^3$

$$F_8|k_1k_2k_3\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{-2\pi i 0.k_3}|1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{-2\pi i 0.k_2k_3}|1\rangle) \otimes \frac{1}{\sqrt{2}}(|0\rangle + e^{-2\pi i 0.k_1k_2k_3}|1\rangle)$$

We will use the following rotation gates

$$R_s = \begin{pmatrix} 1 & 0 \\ 0 & e^{-2\pi i/2^s} \end{pmatrix},$$

noting that R_1 and preparing the uniform superposition $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ can be performed jointly using a Hadamard gate H. Arranging these Hadamard gates and controlled versions of the above rotations so that we only Hadamard transform a bit after all its corresponding controlled rotations are done, we get the following circuit for QFT:



The general case $n > 3$ is analogous.

The Hidden Subgroup Problem for Abelian Groups

Review of Chapter 6 of Ronald de Wolf's Quantum Computing lecture notes

https://arxiv.org/abs/1907.09415v5

Fourier transform on (finite) groups

Representation theory basics

Representation theory uses *linear algebra* to study groups.

- Given a (finite) group G we call a *homomorphism* $\varphi\colon G \mapsto \mathbb{C}^{d\times d}$ into the multiplicative group of $d \times d$ complex matrices a d-dimensional *representation*.
- A representation φ is *irreducible* iff no non-trivial subspace is invariant under all linear maps (matrices) in the image of φ.
- A 1-dimensional representation χ is called a *character*. Note that $\chi(e) = \chi(e^2) = \chi(e)^2$, therefore $1 = \chi(e) = \chi(g^{|G|}) = \chi(g)^{|G|}$ implying that $\chi(g)$ is a $|G|$-th root of unity $\forall g \in G$.
- For an Abelian group G, all irreducible representations are 1-dimensional, and there are $|G|$ different such representations (characters).

Character group of Abelian groups

The 1-dimensional representations of G form a group $\hat{G}$ under *point-wise multiplication*, called the *character group*.

- Let $\varphi, \chi\colon G \mapsto \mathbb{C}$ be 1-dimensional representations, then the point-wise multiplication yields $(\varphi \cdot \chi)(g) = \varphi(g) \cdot \chi(g)$.

Fourier transform on finite Abelian groups

Cyclic groups

- The k-th column of F_N is essentially a character χ_k such that $\chi_k(j) := \sqrt{N}(F_N)_{jk} = \omega_N^{jk}$. Then $\chi_k(j+j') = \omega_N^{(j+j')k} = \chi_k(j)\chi_k(j')$ is indeed a 1-dimensional representation.
- Thus we can consider $F_N \colon |k\rangle \to \frac{1}{\sqrt{N}}|\chi_k\rangle$ a map $G \to \hat{G}$ (which is a homomorphism).

Finite Abelian groups in general

- Any Abelian group G has $|G|$ characters that are also orthogonal to each other.
- The "Basis Theorem" from group theory states that every finite Abelian groups is in fact isomorphic to a product (or direct sum in additive notation) of cyclic groups

$$G \simeq \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_t}.$$

- The characters of G are then simply the (tensor) products of their cyclic components

$$\hat{G} \simeq \hat{\mathbb{Z}}_{N_1} \times \hat{\mathbb{Z}}_{N_2} \times \cdots \times \hat{\mathbb{Z}}_{N_t} \quad \text{and} \quad F_G \simeq F_{N_1} \otimes F_{N_2} \otimes \cdots \otimes F_{N_t}.$$

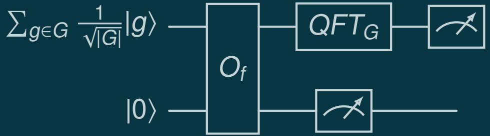
- For example $F_{\mathbb{Z}_2^n}$ is $H^{\otimes n}$.

The (Abelian) Hidden Subgroups Problem

The Hidden Subgroup Problem (HSP)

- Given a function $f\colon G \mapsto X$ that *hides* the subgroup $H \leq G$, i.e., $f(g_1) = f(g_2)$ iff $g_1 H = g_2 H$ find H with a few queries to f.
- Equivalently, f is an injective function on cosets.

An efficient quantum algorithm for Abelian HSP

- 

-

$$\sum_{g\in G} \frac{1}{\sqrt{|G|}}|g\rangle|0\rangle \xrightarrow{O_f} \sum_{g\in G} \frac{1}{\sqrt{|G|}}|g\rangle\big|f(g)\big\rangle \xrightarrow{\text{meas.}} \sum_{h\in H} \frac{1}{\sqrt{|H|}}|\underbrace{f^{-1}(x)}_{s:=} + h\rangle|x\rangle \xrightarrow{QFT_G} \sum_{h\in H} \frac{1}{\sqrt{|H||G|}}\big|\chi_{s+h}\big\rangle|x\rangle$$

"Decoding" the Abelian HSP

How to use the measurement outcome of the first register?

- What is the outcome of the measurement on the final state?

$$\frac{1}{\sqrt{|H||G|}}\sum_{h\in H}\left|\chi_{s+h}\right\rangle = \frac{1}{\sqrt{|H||G|}}\sum_{h\in H}\sum_{g\in G}\chi_{s+h}(g)|g\rangle$$
$$= \frac{1}{\sqrt{|H||G|}}\sum_{g\in G}\chi_s(g)\sum_{h\in H}\chi_h(g)|g\rangle = \sqrt{\frac{|H|}{|G|}}\sum_{g:\chi_g\in H^\perp}\chi_s(g)|g\rangle,$$

- For the last equality note that χ_g restricted to H is a character of H, and let $H^\perp \leq \hat{G}$ be the subgroup of characters that are constant-1 on H:

$$\sum_{h\in H}\chi_h(g) = \sum_{h\in H}\chi_g(h) = \begin{cases} |H| & \text{if } \chi_g \in H^\perp \\ 0 & \text{if } \chi_g \notin H^\perp. \end{cases}$$

- Thus we obtain a uniformly random g such that $\chi_g \in H^\perp$.
- Each such g gives a linear constraint on H (since $\chi_g(h) = 1$ for all $h \in H$). Collecting a few such g uniquely determines H.

The non-Abelian HSP

What works and what does not

- QFT_G is somewhat harder to define and implement
- Unclear how to efficiently recover the subgroup
- However, the same algorithm is actually query efficient (Barnum & Knill 2002)
- Some cases can be solved efficiently, e.g., normal subgroups (Hallgren, Russell, Ta-Shma 2000), solvable groups (Watrous 2001), nil-2 groups (Ivanyos, Sanselme, Sántha 2007), and certain semidirect product p-groups of constant nilpotency class (Ivanyos, Sántha 2015)
- Kuperberg's algorithm (2003) solves HSP in the dihedral group in time

$$2^{O\left(\sqrt{\log(|G|)}\right)}$$

Important example: Graph isomorphism (i.e., deciding whether $G \simeq G'$)

- **Group:** S_{2n}, **Function:** permute the vertices of $G \cup G'$
- **Subgroup:** Automorphisms of $G \cup G'$
- **Output:** whether there is a generator interchanging vertices of G and G'